Brain-Inspired Hyperdimensional Computing for Outlier Detection

Ruixuan Wang, Xun Jiao

{rwang8, xjiao}@villanova.edu

VILLANOVA UNIVERSITY

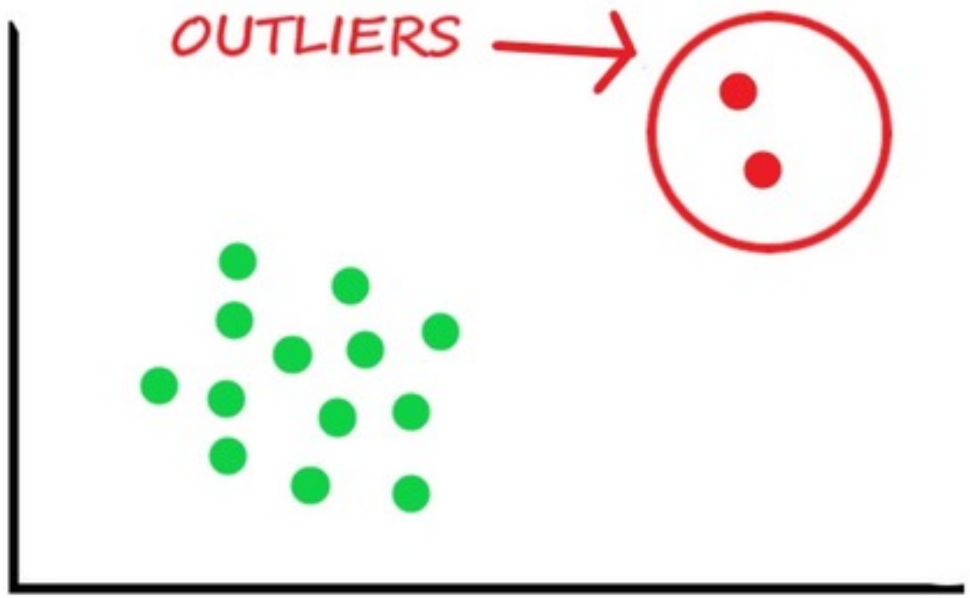
ASIA SOUTH PACIFIC DESIGN AUTOMATION CONFERENCE

Agenda

- Motivation
- HDC Background
- HDC-based Outlier Detection
  - HDAD
  - ODHD
- Future work & Conclusions

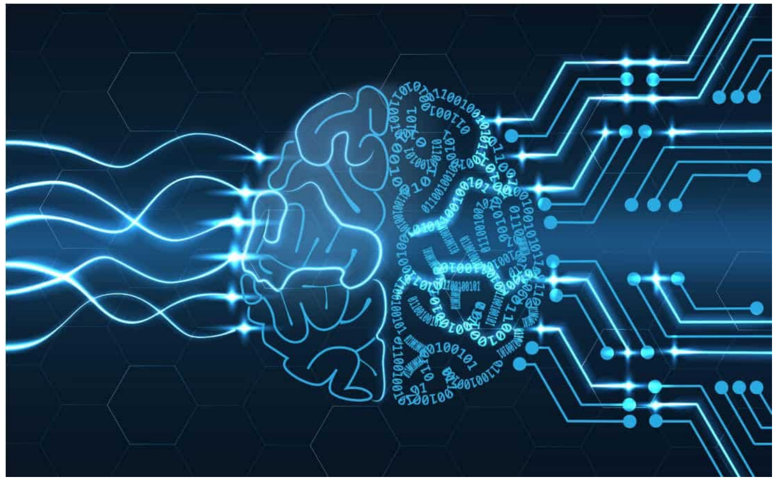
Motivation – Outlier detection

- Outlier detection or anomaly detection task is an essential and everlasting problem which have been widely researched across different application domains.
- In recent years, the development of machine learning and representation learning leads to a surge of interest in the outlier detection field.



Motivation - HDC

- Meanwhile, as an emerging machine learning method and a promising computing paradigm, Hyperdimensional computing (HDC) has become noticeable according to its smaller model size, less computational cost and acceptable accuracy compared with deep neural networks (DNNs).



Agenda

- Motivation
- HDC Background
- HDC-based Anomaly Detection
  - HDAD
  - ODHD
- Future work & Conclusions

HDC Background - Preliminary

- In HDC domain, hyper-vectors (HVs) are the general computing symbols.

$$H = \langle h_1, h_2, \ldots, h_d \rangle$$

- In general, HVs have bipolar elements in {-1, 1} and dimension D = 10000, and arbitrary two random generated HVs are close to orthogonal.

HDC Background - Operations

- HDC utilizes three critical operations for representing and integrating information, named bundling, binding and perturbation.

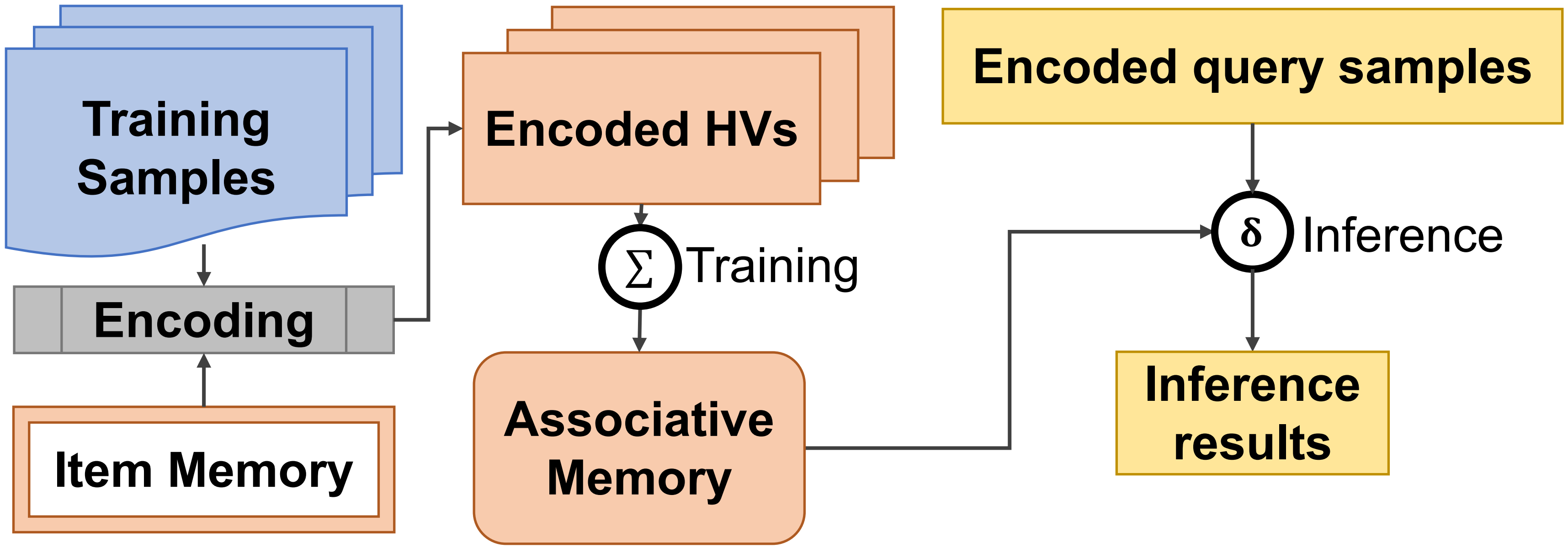
$$\overrightarrow{H_x} + \overrightarrow{H_y} = \langle h_{x1} + h_{y1}, h_{x2} + h_{y2}, \ldots, h_{xd} + h_{yd} \rangle$$
$$\overrightarrow{H_x} * \overrightarrow{H_y} = \langle h_{x1} * h_{y1}, h_{x2} * h_{y2}, \ldots, h_{xd} * h_{yd} \rangle$$
$$\rho^1(\overrightarrow{H}) = \langle h_d, h_1, h_2, \ldots, h_{d-1} \rangle$$

- In the meantime, HDC measures the distance between HVs for revealing the correlation based on distance metric **δ**. To be specific, here we use cosine distance as an example:

$$\delta(\overrightarrow{H_x}, \overrightarrow{H_y}) = \frac{\overrightarrow{H_x} \cdot \overrightarrow{H_y}}{||\overrightarrow{H_x}|| \times ||\overrightarrow{H_y}||} = \frac{\sum_{i=1}^{d} h_{xi} \cdot h_{yi}}{\sqrt{\sum_{i=1}^{d} {h_{xi}}^2} \cdot \sqrt{\sum_{i=1}^{d} {h_{yi}}^2}}$$

HDC Background – Model Implementation

- Three main processes are required to establish a general HDC model, named the encoding, the training, and the inference process.
- A general pipeline of a HDC model shows as following:



Agenda

- Motivation
- HDC Background
- <u>HDC-based Outlier Detection</u>
  - HDAD
  - ODHD
- Future work & Conclusions

HDC-based Outlier Detection

- Targeting at the outlier detection task, we develop and propose two HDC based approaches for outlier detection, which are named HDAD and ODHD respectively.
- To be specific, HDAD [1] detects anomalies based on reconstruction errors, while ODHD [2] utilizes a one-class classification strategy to filter out outliers.

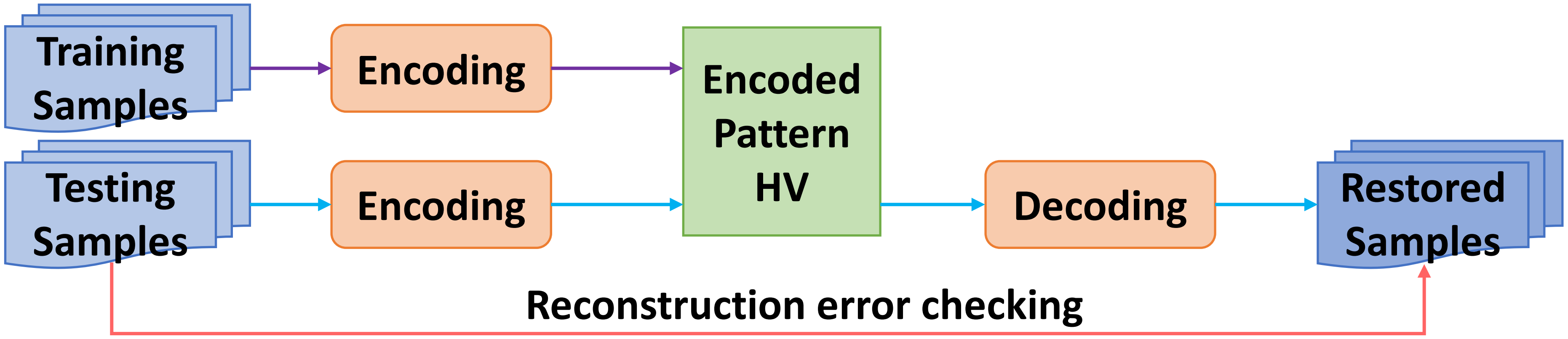
[1] Ruixuan Wang, Fanxin Kong, Hasshi Sudler, and Xun Jiao. "Brief Industry Paper: HDAD: Hyperdimensional Computing-based Anomaly Detection for Automotive Sensor Attacks." *2021 IEEE 27th RTAS.* IEEE, 2021.
[2] Ruixuan Wang, Xun Jiao, and Sharon Hu. "ODHD: one-class brain-inspired hyperdimensional computing for outlier detection." Proceedings of the 59th ACM/IEEE Design Automation Conference. 2022.

Agenda

- Motivation
- HDC Background
- <u>HDC-based Outlier Detection</u>
  - <u>HDAD</u>
  - ODHD
- Future work & Conclusions

HDAD - Methodology

- HDAD contains three key phases:
  1) Pattern encoding, where we encode training samples into HVs for the pattern learning purpose;
  2) Pattern decoding, where we decode the HVs and reconstruct them to the original samples;
  3) Reconstruction error check, where we check the reconstruction error between original and reconstructed sample.



HDAD – Methodology

- For HDAD training, we use a set of random generated base HVs (item memory) to encode all the training samples into HVs and aggregate them to one reference HV: RV. This RV represents all the normal samples (like normal pattern of data samples) in the training set.

$$RV = \sum_{i=0}^{N} HV_i$$

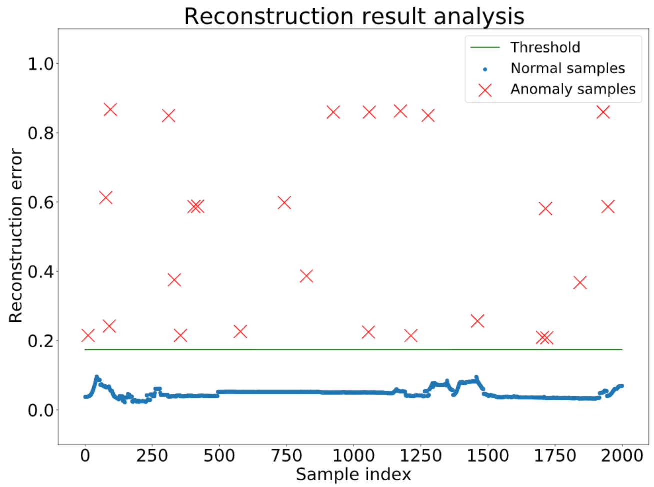
- For HDAD inference, we first encode the query sample T into HV TV and bundle it with RV as TV', where TV' = (TV + RV). And we decode sample T' from this TV' and we

$$T' = \{\mathrm{TV}' * \frac{b_1}{D}, \mathrm{TV}' * b_2/D, \dots, \mathrm{TV}' * b_n/D\}$$

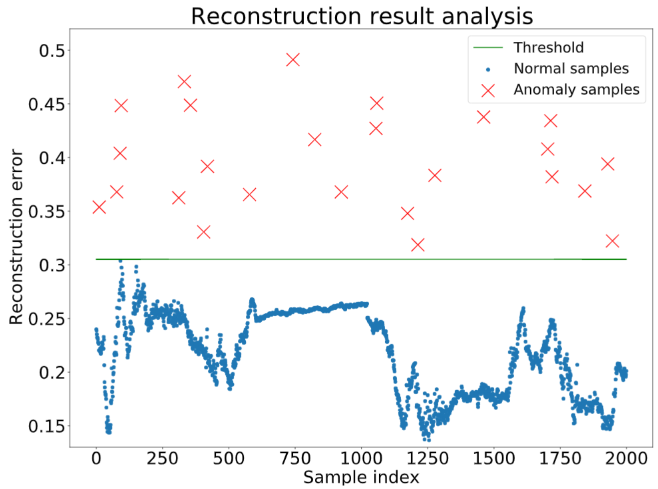
- For the purpose of anomaly detection, HDAD computes the reconstruction error between the original data sample and reconstructed data and utilizes a predefined threshold to filter out the anomalies.

HDAD - Experiment Result

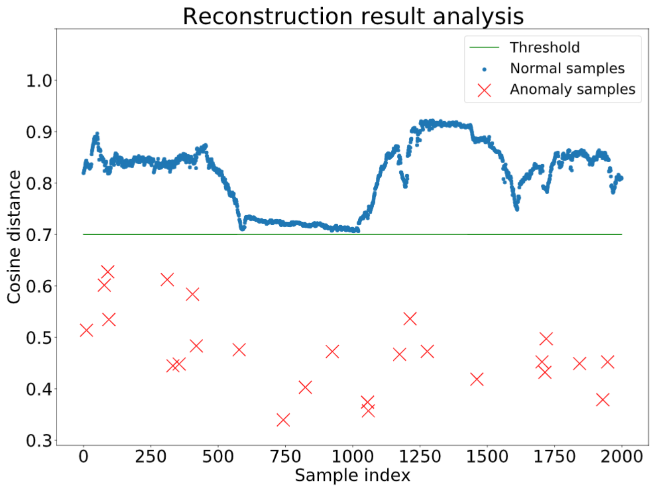
- For the metrics of reconstruction error, we use MSE (mean squared error), MAE (mean absolute error) between the T and T' and the SIM (cosine similarity) between the HVs encoded T and T' .
- We evaluate the performance of HDAD through an autonomous driving dataset from the AEGIS Project [1], which contains variant continuous sensor readings during the driving trip.



MSE



MAE



SIM

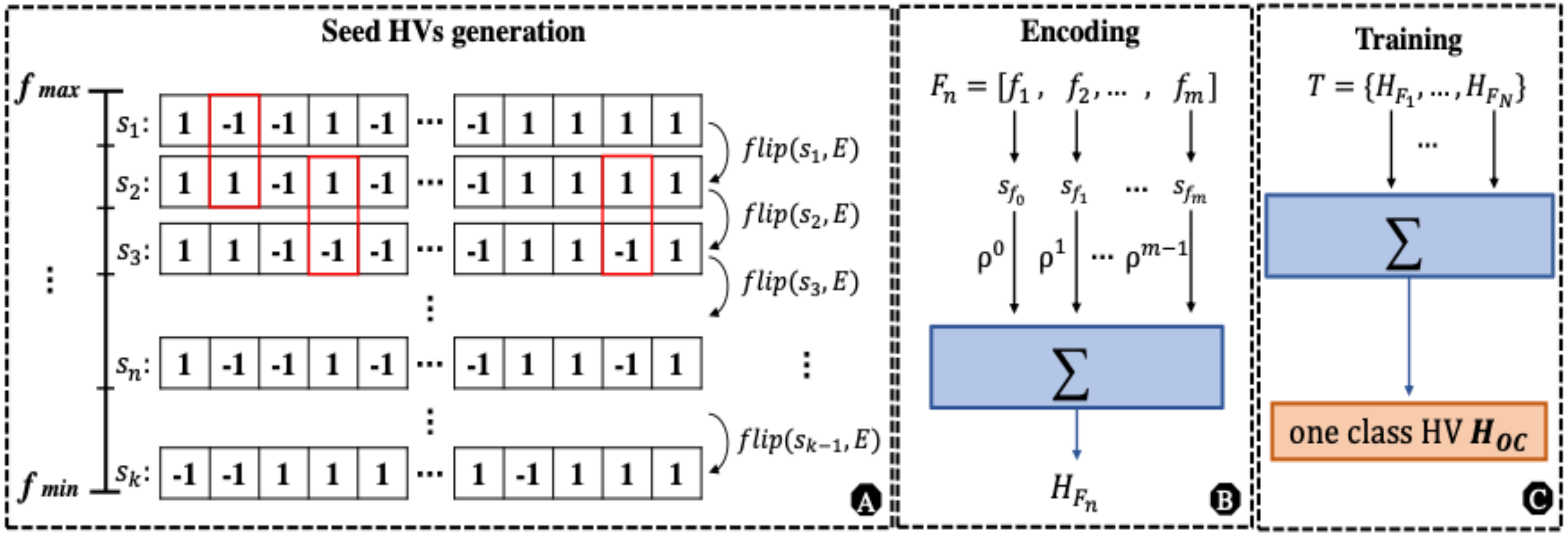
[1] Kaiser, Christian, Alexander Stocker, and Andreas Festl. "Automotive CAN bus data: An example dataset from the AEGIS Big Data Project." *Zenodo* (2019).

Agenda

- Motivation
- HDC Background
- <u>HDC-based Outlier Detection</u>
  - HDAD
  - <u>ODHD</u>
- Future work & Conclusions

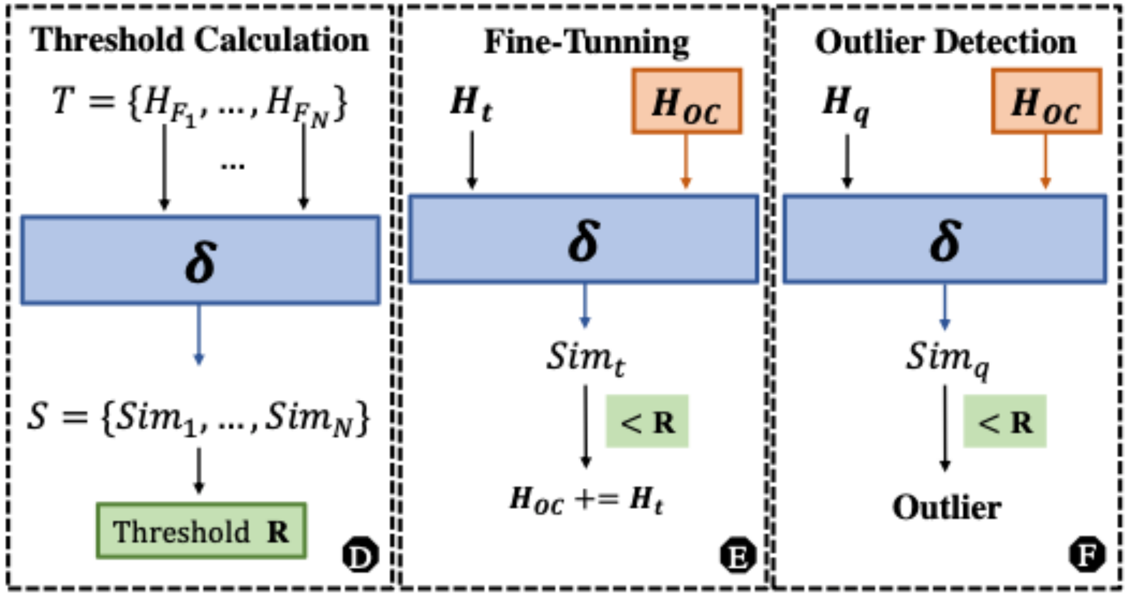
ODHD - Methodology

- ODHD based on a semi-supervised learning approach, which first encodes all the training samples into HVs and establishes a one-class HV only using samples from the positive class. This one-class HV represents the abstracted pattern of the positive class samples.



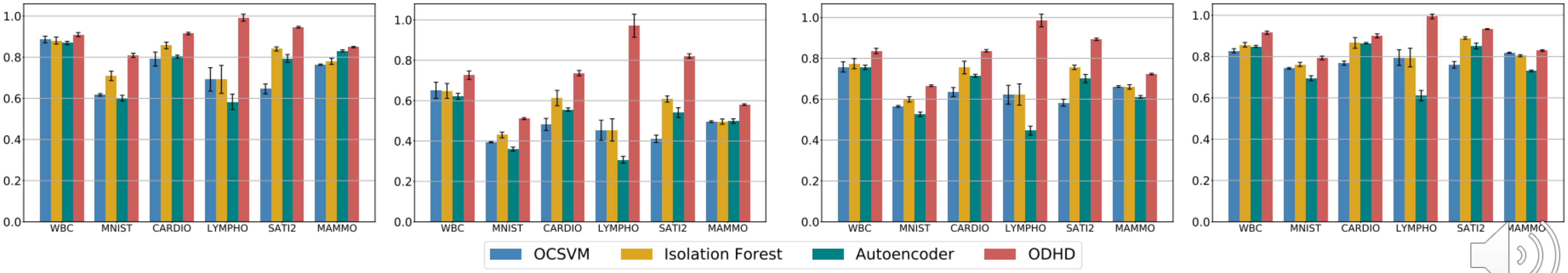
ODHD - Methodology

- Then ODHD measures the similarity between all the unlabeled encoded HVs in the inference set with the one-class HV, if the similarity lower than a predefined threshold, the sample is detected as an outlier.



ODHD - Experiment Setup

- We compare the performance of ODHD with other widely employed machine learning algorithms including one-class SVM (OCSVM), isolation forest and auto-encoder across 6 real datasets under the accuracy, average precision, f1-score and ROC-AUC metrics.
- According to our experimental result, ODHD can outperform all the baseline methods on every dataset for every metric.



Agenda

- Motivation
- HDC Background
- HDC-based Outlier Detection
  - HDAD
  - ODHD
- <u>Future work & Conclusions</u>

Future work & Conclusions

- Outlier detection aims to detect abnormal values that deviate from other observations in a dataset, or an observation that diverges from an overall pattern on a sample.
- Based on the proposed HDC based outlier detection method, the promising results we showed open a new viable alternative to traditional learning algorithms for outlier detection.
- Our future work will consider the implementation of HDC in not only accuracy but also reliability perspectives, which will lead to a HDC model with both effectiveness and robustness in the outlier detection task.

Thank you!