Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Bonn-Aachen
International Center for
Information Technology

Feature Extraction for Motion Data

R&D Defense

March 8, 2021

Ganesamanian Kolappan

Prof. Dr. Paul G. Plöger
Dr. Anastassia Küstenmacher

Introduction

What is the project about?

- **Motion data** - collection of readings from motion detection sensors

Introduction

What is the project about?

- **Motion data** - collection of readings from motion detection sensors
- **Domain** - Human Activity Recognition/Human Activity Classification (HAR/HAC)

Introduction

What is the project about?

- **Motion data** - collection of readings from motion detection sensors
- **Domain** - Human Activity Recognition/Human Activity Classification (HAR/HAC)

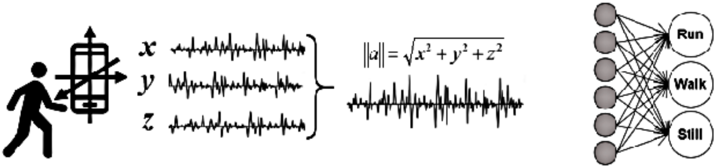

Figure 1: *Abstract view of this research work. Reproduced from [16]*

Introduction

What is motion data?

- Time-series data

$$T_d = \{X_i\}_{i=0}^{N-1} \tag{1}$$

  - X_i is variable (sensor readings)
  - N is number of observations
  - $N \times 1$ data size

Introduction

What is motion data?

- Time-series data

$$T_d = \{X_i\}_{i=0}^{N-1} \tag{1}$$

  - X_i is variable (sensor readings)
  - N is number of observations
  - $N \times 1$ data size

- Motion data

$$T_{md} = \{X_{1i}, X_{2i}, ..., X_{Mi}\}_{i=0}^{N-1} \tag{2}$$

  - M is number of dimensions
  - $N \times M$ data size

Introduction

What is the problem?

Motion data consists of more samples, complex patterns, high dimensions, and noise [7]

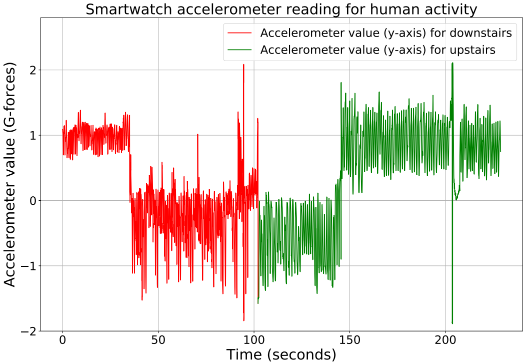

Figure 2: *Activity upstair's readings with noise near to 200^{th} second. Data from [18]*

Motivation

Why is it relevant?

- Healthcare [22][13][24]
  - Symptoms tracking
  - Evaluating exercise

Motivation

Why is it relevant?

- Healthcare [22][13][24]
  - Symptoms tracking
  - Evaluating exercise
- Smartwatch manufacturers [8]
  - Apple
  - Samsung
  - Fitbit
  - Fossil

Motivation

Why is it relevant?

- Healthcare [22][13][24]
  - Symptoms tracking
  - Evaluating exercise
- Smartwatch manufacturers [8]
  - Apple
  - Samsung
  - Fitbit
  - Fossil
- Other areas [17][6]
  - Robotics
  - Manufacturing sectors

Related work

What other people have done?

Methods varies according to number of observations and task

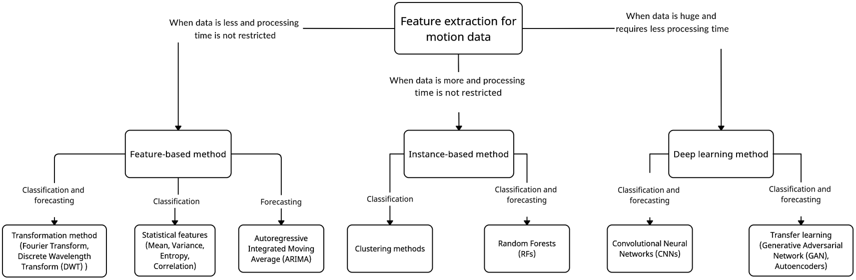

Figure 3: *Related work organization chart*

Related work

Why is it not sufficient?

Method	Limitations
Feature-based method	1. Requires domain knowledge [15] 2. Works in trail and error [3] 3. Time consuming [3]

Table 1: *Limitations of related work*

Related work

Why is it not sufficient?

Method	Limitations
Feature-based method	1. Requires domain knowledge [15] 2. Works in trail and error [3] 3. Time consuming [3]
Instance-based method	1. Consumes more memory [2] 2. Consumes more time [9] 3. High computational load [2]

Table 1: *Limitations of related work*

Related work

Why is it not sufficient?

Method	Limitations
Feature-based method	1. Requires domain knowledge [15] 2. Works in trail and error [3] 3. Time consuming [3]
Instance-based method	1. Consumes more memory [2] 2. Consumes more time [9] 3. High computational load [2]
Deep learning method	1. Conversion to images [11] 2. Requires smaller dataset [19] 3. Hyperparameter tuning [10]

Table 1: *Limitations of related work*

Methodology

What is the proposed approach?

- Autoencoder $\rightarrow$ feature extraction
- Support Vector Machine (SVM) $\rightarrow$ classification

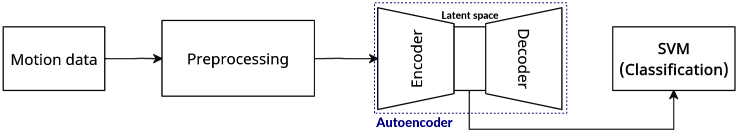

Figure 4: *Proposed method pipeline*

Methodology

What are the preprocessing steps?

- Checking for null values and missing values

Methodology

What are the preprocessing steps?

- Checking for null values and missing values
- Sliding window $\rightarrow$ disintegrate continuous data to discrete

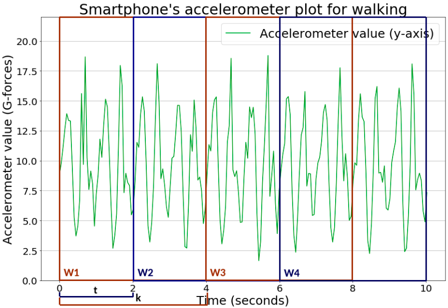

Figure 5: *Sliding window with window size (k), step size (t) and window number W_i. Data from [12]*

Methodology

Working of sliding window

- Sampling frequency of 20 Hz for 10 s $\implies$ N = 200

$$T_d = \{X_i\}_{i=0}^{199} \tag{3}$$

- **Parameters** - k = 4 s (80 values) and t = 2 s (40 values)

$$T_d = \{\{X_i\}_i^{i+k}\}_{i=0}^{N-k} \tag{4}$$

where, $i = \{0, t, 2 * t, ...\}, \forall i \leq N - k$

Therefore, substituting $i = \{0, 40, 80, 120\}$ in Equation 4

$$\begin{aligned} T_d = \{\{X_0, ..., X_{79}\}, \{X_{40}, ..., X_{119}\}, \\ \{X_{80}, ..., X_{159}\}, \{X_{120}, ..., X_{199}\}\} \end{aligned} \tag{5}$$

- Data size is $K \times k \times 1$. Where, K is total number of windows

Methodology

What is autoencoder?

- Autoencoder is an unsupervised deep learning approach
- Features are available at latent space (low dimension)

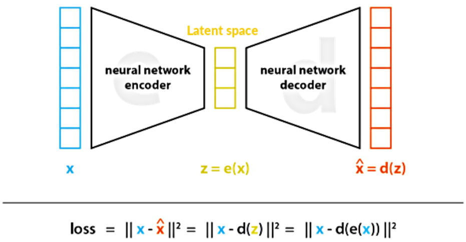

Figure 6: *Autoencoder and its loss. Reproduced from [12]*

Methodology

How is it different from autoencoder used in NLP?

Embedding layer is **ignored** since data is by default numeric

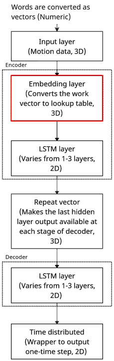

Figure 7: *Autoencoder for machine translation*

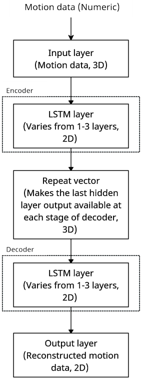

Figure 8: *Autoencoder for motion data feature extraction*

Methodology

How to validate the proposed method?

- Performance comparison with state of the art

Methodology

How to validate the proposed method?

- Performance comparison with state of the art
- Fast Fourier Transform (FFT) and statistical features [21]

Methodology

How to validate the proposed method?

- Performance comparison with state of the art
- Fast Fourier Transform (FFT) and statistical features [21]

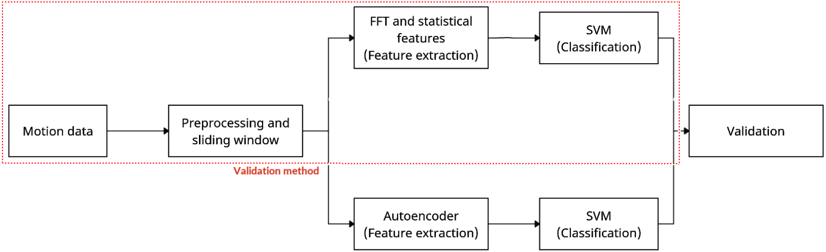

Figure 9: *Proposed method pipeline with validation*

Experiments

Dataset

- Motionsense dataset from kaggle repository [20]
- Data recorded in Queen Mary University of London Mille End campus

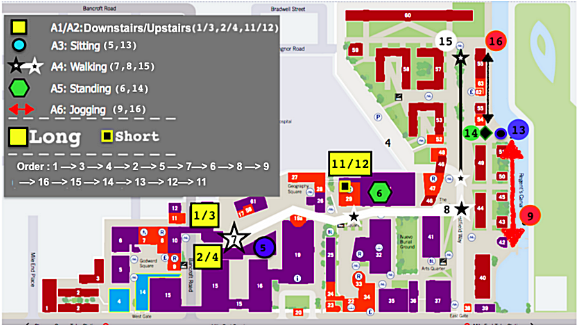

Figure 10: *Motionsense trails. Reproduced from [18]*

Experiments

What are the attributes?

Attributes	**Value / type**
Number of participants	24
Number of activities/labels	6
Annotated	Yes
Sampling frequency of sensors	50 Hz
Types of sensors	Triaxial accelerometer and triaxial gyroscope
Number of device/sensor placements	2 (wrist, waist)
Type of devices	Smartphone (iphone 6s) and smartwatch (Apple)
Continous activities	Yes
Recorded environment	Outdoor condition
Number of dimensions	12
Total discrete values	1,412,865

Table 2: *Attributes of Motionsense dataset [18]*

Experiments

Do all the statistical features are useful?

Entropy is **ignored** as it under performed comparatively

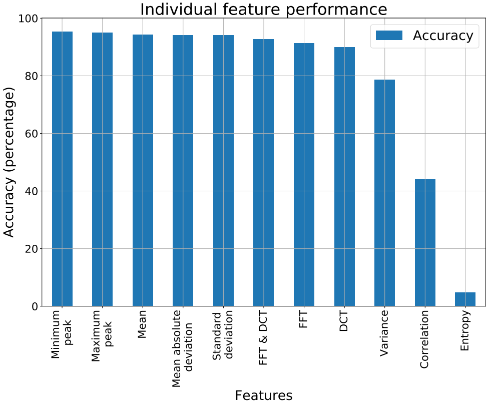

Figure 11: *Performance of each features for feature-based method (bars are arranged in descending order)*

Experiments

How is the performance of feature-based method?

- Jogging, sitting, and standing is classified aptly
- Upstairs and downstairs is misclassified with walking or vice versa

Accuracy score: 99.38

	precision	recall	f1-score	support
Walking	1.00	0.99	0.99	34648
Jogging	1.00	1.00	1.00	13512
Sitting	1.00	1.00	1.00	33649
Standing	1.00	1.00	1.00	30428
Upstairs	0.98	0.99	0.98	15863
Downstairs	0.99	0.99	0.99	13162
accuracy			0.99	141262
macro avg	0.99	0.99	0.99	141262
weighted avg	0.99	0.99	0.99	141262

Figure 12: *Classification report for Motionsense dataset for feature-based method. Correctly classified activities are highlighted (green box)*

Experiments

How much does each features contribute?

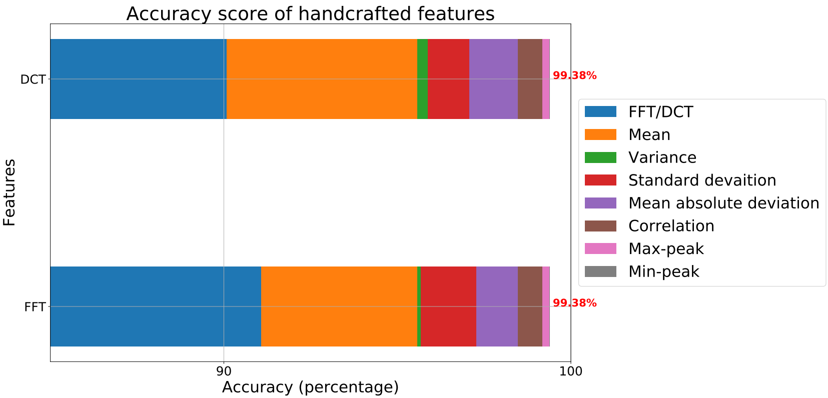

Figure 13: *Graph depicts importance of statistical features*

Experiments

What is the inference?

Requires triaxial altimeter additionally

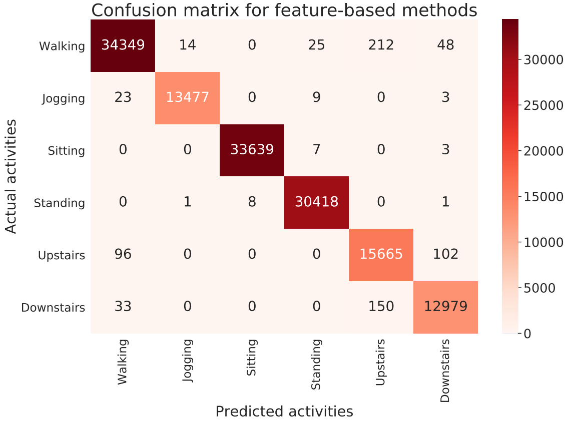

Figure 14: *Confusion matrix for Motionsense dataset (feature-based method)*

Experiments

How is the performance of autoencoder?

- Sitting is classified aptly
- Other activities are misclassified

Accuracy score: 98.64

	precision	recall	f1-score	support
Walking	0.98	0.99	0.99	34648
Jogging	0.99	0.99	0.99	13512
Sitting	1.00	1.00	1.00	33649
Standing	0.99	1.00	0.99	30428
Upstairs	0.96	0.96	0.96	15863
Downstairs	0.97	0.96	0.96	13162
accuracy			0.99	141262
macro avg	0.98	0.98	0.99	141262
weighted avg	0.99	0.99	0.99	141262

Figure 15: *Classification report for Motionsense dataset for autoencoders. Correctly classified activity is highlighted (green box)*

Experiments

How much does each features of autoencoder contribute?

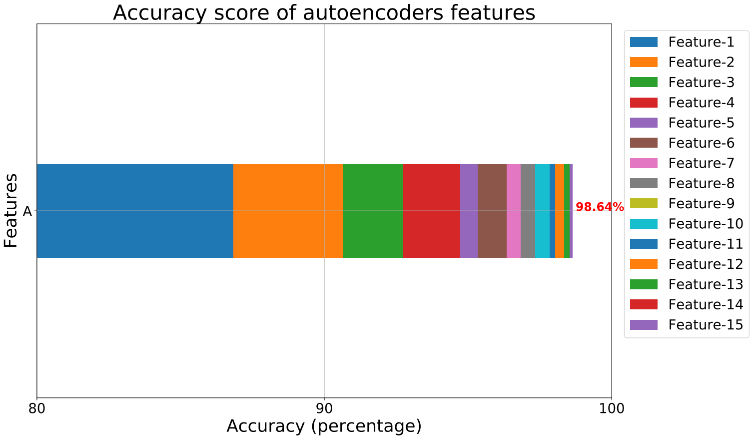

Figure 16: *Contribution of each features from autoencoder. Contribution of Feature-14 is not visible yet when removed accuracy drops by 0.5%*

Experiments

What is the inference?

Requires additional sensor placement → leg or chest or both [5]

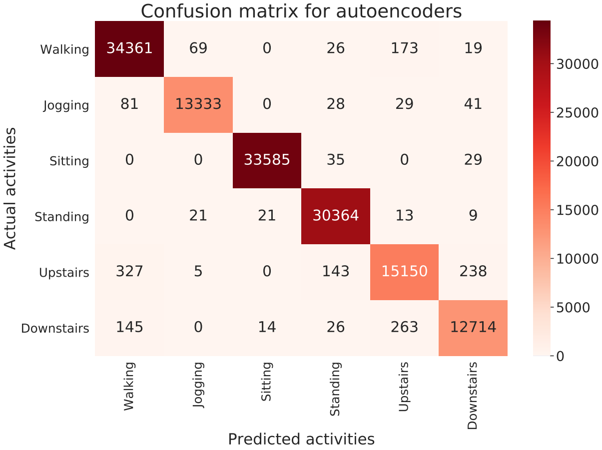

Figure 17: *Confusion matrix for Motionsense dataset (autoencoder)*

Experiments

Visualization of Motionsense data

Activities are tightly bonded

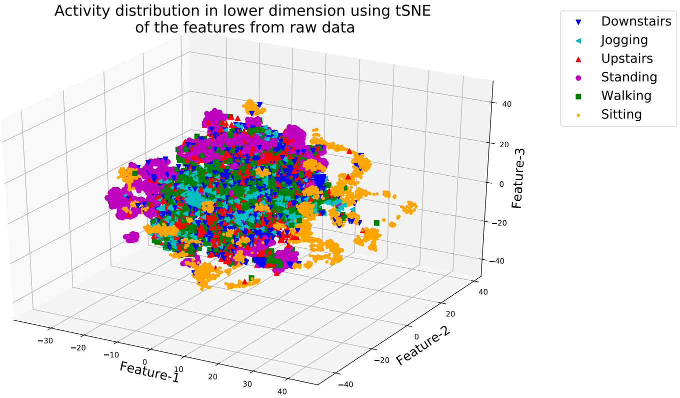

Figure 18: *Graph depicts that activities are continuous*

Experiments

Visualization of features from feature-based method

Activities are visibly separated.

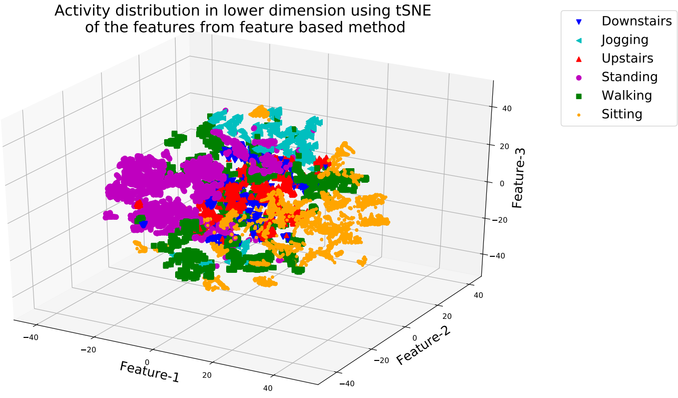

Figure 19: *Graph depicts diversity in activities*

Experiments

Visualization of features from autoencoders

Activities are clustered yet diversity is not visible

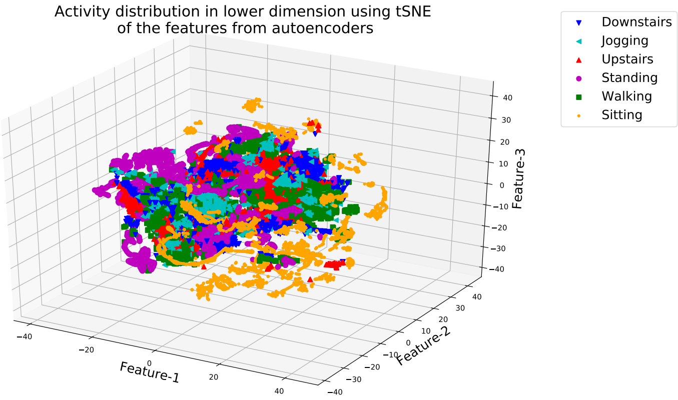

Figure 20: *Graph depicts under performance of autoencoder compared to feature-based method*

Experiments

Dataset-2

- Wireless Sensor Data Mining version 1.1 (WISDM_v_1.1) available in UCI repository
- From WISDM Lab, Department of Computer and Information Science, Fordham University [12]

Attributes	**Value / type**
Number of participants	36
Number of activities/labels	6
Types of sensors	Triaxial accelerometer and triaxial gyroscope (smart-phone)
Continous activities	Yes (lab environment)
Total discrete values	1,098,207 (labelled)

Table 3: *Attributes of WISDM_v_1.1 dataset [12]*

Experiments

How is the performance of feature-based method?

All activities are misclassified

Accuracy score: 85.50

	precision	recall	f1-score	support
Walking	0.87	0.88	0.87	85022
Jogging	0.89	0.89	0.89	68489
Sitting	0.89	0.88	0.89	11911
Standing	0.87	0.89	0.88	9677
Upstairs	0.77	0.75	0.76	24626
Downstairs	0.75	0.75	0.75	19876
accuracy			0.86	219601
macro avg	0.84	0.84	0.84	219601
weighted avg	0.85	0.86	0.85	219601

Figure 21: *Classification report for WISDM_v_1.1 for feature-based methods. Support refers to number of test samples*

Experiments

How is the performance of autoencoder?

Standing, upstairs and downstairs are classified poorly

Accuracy score: 70.46

	precision	recall	f1-score	support
Walking	0.61	0.97	0.75	85022
Jogging	0.85	0.88	0.87	68489
Sitting	0.99	0.90	0.94	11911
Standing	1.00	0.00	0.00	9677
Upstairs	0.38	0.05	0.09	24626
Downstairs	0.37	0.02	0.04	19876
accuracy			0.70	219601
macro avg	0.70	0.47	0.45	219601
weighted avg	0.68	0.70	0.62	219601

Figure 22: *Classification report for WISDM_v_1.1 for autoencoders. Misclassified activities are highlighted (red box). Support refers to number of test samples)*

Experiments

Dataset-3

- Wireless Sensor Data Mining (WISDM) from UCI repository
- From WISDM Lab, Department of Computer and Information Science, Fordham University [12]

Attributes	**Value / type**
Number of participants	51
Number of activities/labels	18
Types of sensors	Triaxial accelerometer and triaxial gyroscope (smartphone & smartwatch)
Continous activities	Yes (conditioned environment)
Total discrete values	15,630,426 (labelled)

Table 4: *Attributes of WISDM dataset [12]*

Experiments

How is the performance of feature-based method?

Drinking from cup is misclassified with eating sandwich

Accuracy score: 99.65

	precision	recall	f1-score	support
Walking	1.00	1.00	1.00	7153
Jogging	1.00	1.00	1.00	7140
Climbing Stairs	1.00	1.00	1.00	7149
Sitting	1.00	1.00	1.00	7134
Standing	1.00	1.00	1.00	7080
Typing	1.00	1.00	1.00	7148
Brushing Teeth	1.00	1.00	1.00	7086
Eating Soup	1.00	1.00	1.00	7278
Eating Chips	1.00	1.00	1.00	7144
Eating Pasta	1.00	1.00	1.00	7185
Drinking from cup	1.00	0.89	0.94	7134
Eating Sandwich	0.91	1.00	0.95	7129
Kicking	1.00	1.00	1.00	6933
Playing Tennis	1.00	1.00	1.00	7104
Basketball	1.00	1.00	1.00	7137
Writing	1.00	1.00	1.00	7289
Clapping	1.00	1.00	1.00	7070
Folding Clothes	1.00	1.00	1.00	7227
accuracy			0.99	128520
macro avg	1.00	0.99	0.99	128520
weighted avg	0.99	0.99	0.99	128520

Figure 23: *Classification report for WISDM dataset for feature-based methods. Misclassified activities are highlighted (red box). Support refers to number of test samples*

Experiments

How is the performance of autoencoder?

Typing is misclassified with writing

Accuracy score: 99.99

	precision	recall	f1-score	support
Walking	1.00	1.00	1.00	7153
Jogging	1.00	1.00	1.00	7140
Climbing Stairs	1.00	1.00	1.00	7149
Sitting	1.00	1.00	1.00	7134
Standing	1.00	1.00	1.00	7080
Typing	1.00	0.98	0.99	7148
Brushing Teeth	1.00	1.00	1.00	7086
Eating Soup	1.00	1.00	1.00	7278
Eating Chips	1.00	1.00	1.00	7144
Eating Pasta	1.00	1.00	1.00	7185
Drinking from cup	1.00	1.00	1.00	7134
Eating Sandwich	1.00	1.00	1.00	7129
Kicking	1.00	1.00	1.00	6933
Playing Tennis	1.00	1.00	1.00	7104
Basketball	1.00	1.00	1.00	7137
Writing	0.98	1.00	0.99	7289
Clapping	1.00	1.00	1.00	7070
Folding Clothes	1.00	1.00	1.00	7227
accuracy			0.99	128520
macro avg	1.00	0.99	0.99	128520
weighted avg	0.99	0.99	0.99	128520

Figure 24: *Classification report for WISDM dataset for autoencoders. Misclassified activities are highlighted (red box). Support refers to number of test samples*

Conclusion

What is the take away?

- Autoencoders are capable of extracting features from motion data

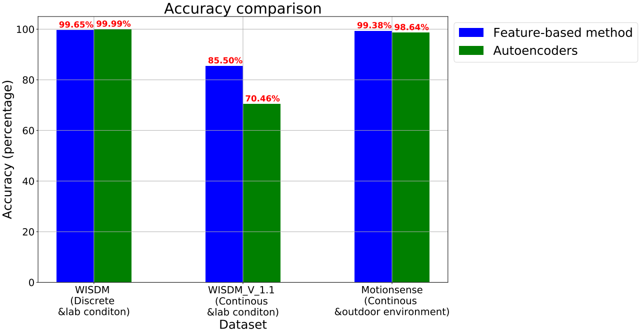

Figure 25: *Results for all datasets*

Conclusion

What is the take away?

- Autoencoders are capable of extracting features from motion data
- Dataset should consist of atleast 3 or 4 sensor placements [5]

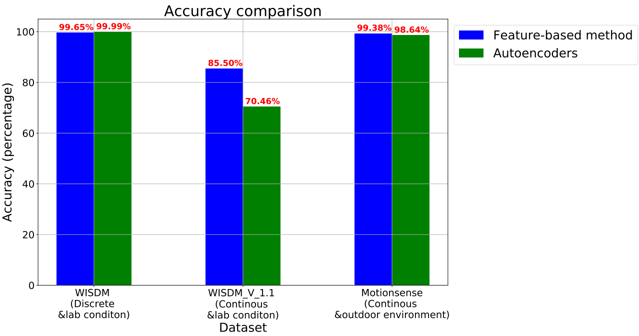

Figure 25: *Results for all datasets*

Conclusion

What is the take away?

- Autoencoders are capable of extracting features from motion data
- Dataset should consist of atleast 3 or 4 sensor placements [5]
- Accurate detection of transition between activities

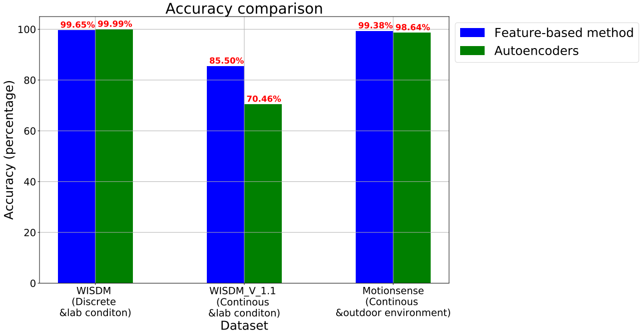

Figure 25: *Results for all datasets*

Contributions

Why is it better?

- Automatic feature extraction

Contributions

Why is it better?

- Automatic feature extraction
- Model for transfer learning

Contributions

Why is it better?

- Automatic feature extraction
- Model for transfer learning
- Comparative analysis

Contributions

Why is it better?

- Automatic feature extraction
- Model for transfer learning
- Comparative analysis
- Importance of statistical features is proved

Future Work

What is left to be done?

- Dataset creation
  - WISDM like activities
  - Motionsense like environment
  - More sensor placement [5]

Future Work

What is left to be done?

- Dataset creation
  - WISDM like activities
  - Motionsense like environment
  - More sensor placement [5]
- Multiple sliding windows to detect transition between activities [1]

Future Work

What is left to be done?

- Dataset creation
  - WISDM like activities
  - Motionsense like environment
  - More sensor placement [5]
- Multiple sliding windows to detect transition between activities [1]
- Explainable Artificial Intelligence (XAI) [23]

Future Work

What is left to be done?

- Dataset creation
  - WISDM like activities
  - Motionsense like environment
  - More sensor placement [5]
- Multiple sliding windows to detect transition between activities [1]
- Explainable Artificial Intelligence (XAI) [23]
- Automatic hyperparameter optimization [4][10][14]

Thank you

Reference (1/2)

[1] S. Baek, D. Lee, G. Kim, and H. Bae. Summarization Method for Multiple Sliding Window Aggregate Queries. In *2009 Software Technologies for Future Dependable Distributed Systems*, pages 205–209, 2009.

[2] Kasun Bandara, Christoph Bergmeir, and Slawek Smyl. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Systems with Applications*, 140:112896, 2020.

[3] I. Batal and M. Hauskrecht. A Supervised Time Series Feature Extraction Technique Using DCT and DWT. In *2009 International Conference on Machine Learning and Applications*, pages 735–739, 2009.

[4] R. J. Borgli, H. Kvale Stensland, M. A. Riegler, and P. Halvorsen. Automatic Hyperparameter Optimization for Transfer Learning on Medical Image Datasets Using Bayesian Optimization. In *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*, pages 1–6, 2019.

[5] Barbara Bruno, Fulvio Mastrogiovanni, and Antonio Sgorbissa. HOOD: a Real Environment Human Odometry Dataset for Wearable Sensor Placement Analysis. 10 2015.

[6] M. Christ, A. Kempa-Liehr, and M. Feindt. Distributed and parallel time series feature extraction for industrial big data applications. *ArXiv*, abs/1610.07717, 2016.

[7] Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing*, 307:72 – 77, 2018.

[8] Jiska Classen, Daniel Wegemer, Paul Patras, Tom Spink, and Matthias Hollick. Anatomy of a Vulnerable Fitness Tracking System: Dissecting the Fitbit Cloud, App, and Firmware. 2(1), 2018.

[9] Houtao Deng, George Runger, Eugene Tuv, and Martyanov Vladimir. A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153, 2013.

[10] F. Dernoncourt and J. Y. Lee. Optimizing neural network hyperparameters with Gaussian processes for dialog act classification. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 406–413, 2016.

[11] A. H. Fakhrulddin, X. Fei, and H. Li. Convolutional neural networks (CNN) based human fall detection on Body Sensor Networks (BSN) sensor data. In *2017 4th International Conference on Systems and Informatics (ICSAI)*, pages 1461–1465, 2017.

[12] UCI Machine Learning Repository Center for Machine Learning and Intelligent Systems. Time series dataset, 2020. Accessed on: 2020-11-26. [Online].

Reference (2/2)

[13] Mark S Goldberg, Richard T Burnett, and David Stieb. A review of time-series studies used to evaluate the short-term effects of air pollution on human health. *Reviews on environmental health*, 18(4):269–303, 2003.

[14] J. Han, D. Choi, S. Park, and S. Hong. Hyperparameter Optimization for Multi-Layer Data Input Using Genetic Algorithm. In *2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)*, pages 701–704, 2020.

[15] Martin Längkvist, Lars Karlsson, and Amy Loutfi. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters*, 42:11–24, 2014.

[16] Song-Mi Lee, Sang Min Yoon, and Heeryon Cho. Human activity recognition from accelerometer data using convolutional neural network. In *2017 ieee international conference on big data and smart computing (bigcomp)*, pages 131–134. IEEE, 2017.

[17] E. Lughofer. Self-adaptive forecast models in predictive maintenance systems. In *2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 8–8, 2018.

[18] Mohammad Malekzadeh, Richard G. Clegg, Andrea Cavallaro, and Hamed Haddadi. Mobile Sensor Data Anonymization. In *Proceedings of the International Conference on Internet of Things Design and Implementation*, IoTDI '19, pages 49–58, 2019.

[19] A. Meiseles and L. Rokach. Source Model Selection for Deep Learning in the Time Series Domain. *IEEE Access*, 8:6190–6200, 2020.

[20] Kaggle Repository. Datasets, 2020. Accessed on: 2020-11-26. [Online].

[21] H. B. Sandya, P. H. Kumar, and S. B. Patil. Feature extraction, classification and forecasting of time series signal using fuzzy and garch techniques. In *National Conference on Challenges in Research Technology in the Coming Decades (CRT 2013)*, pages 1–7, 2013.

[22] S. Shukla, P. Singh, N. Neopane, and Rishabh. Health Care Management System Using Time Series Analysis. In *2019 4th International Conference on Information Systems and Computer Networks (ISCON)*, pages 63–69, 2019.

[23] Duy Khoi Tran. Feature extraction for time series classification. Master's thesis, Hochschule Bonn-Rhein-Sieg, Germany, 2020.

[24] Anjana Wijekoon, N. Wiratunga, and K. Cooper. MEx: Multi-modal Exercises Dataset for Human Activity Recognition. *ArXiv*, abs/1908.08992, 2019.