Deep Learning for Time Series Classification

Application to Surgical Data Science

Germain Forestier

UNIVERSITÉ HAUTE-ALSACE

3rd ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data

Invited Talk

September 14th 2018

Deep Learning

- Revolutionized the field of computer vision [1]
- Reached human level performance in image recognition tasks [2]
- Adopted by the Natural Language Processing (NLP) community for:
  - machine translation
  - learning word embeddings
  - document classification
- Improved state of the art speech recognition systems [3]

[1] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems

[2] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition

[3] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., Jaitly, N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine

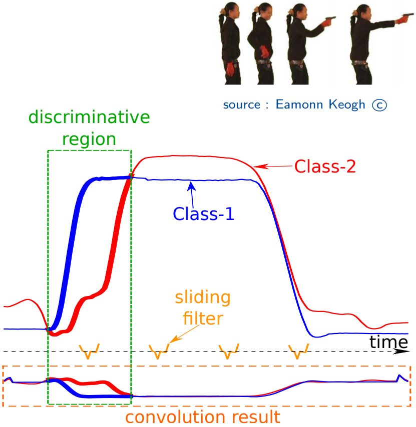
Why should it work on time series data ?

- Recent success in *sequential* data analysis (text, audio, etc.) [1]
- Ability to detect time invariant characteristics
  - Similar to spatially invariant filters in 2D images
  - Should require less data to find patterns in 1D data (time series) ?
- Ability to handle high dimentional Multivariate Time Series (MTS)

[1] Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate, In Proceedings of International Conference on Learning Representations

Convolution on images vs. time series

(a) The result of a applying an edge detection convolution on an image



(b) The result of applying a learned discriminative convolution on the GunPoint dataset

Generative vs. Discriminative deep learning approaches [1]

1. Generative approaches
   - Use an unsupervised training step to learn generating the time series
   - Learn a latent representation of the data
   - Later fed to a discriminative off-the-shelf classifier
2. Discriminative approaches
   - Do not include an unsupervised step
   - Learn the mapping to a class probability distribution from:
     - extracted features
     - raw input time series

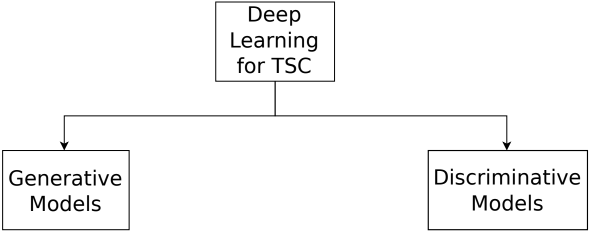
[1] Längkvist, M., Karlsson, L., & Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognition Letters

Generative vs. Discriminative deep learning approaches [1]

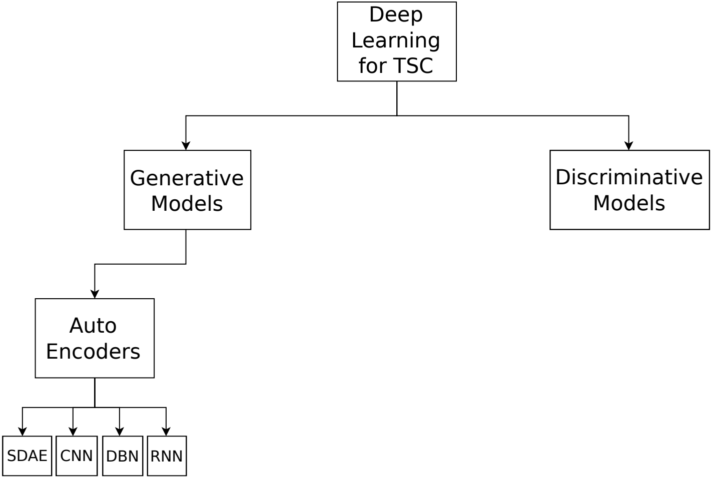
1. Generative approaches
   - Use an unsupervised training step to learn generating the time series
   - Learn a latent representation of the data
   - Later fed to a discriminative off-the-shelf classifier
2. Discriminative approaches
   - Do not include an unsupervised step
   - Learn the mapping to a class probability distribution from:
     - extracted features
     - raw input time series

[1] Längkvist, M., Karlsson, L., & Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognition Letters

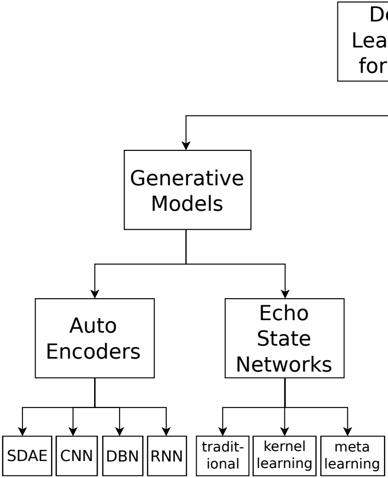
Generative vs Discriminative



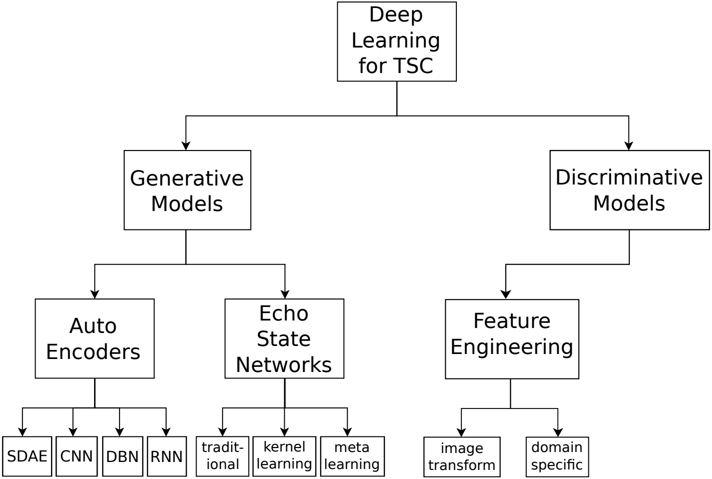
Generative vs Discriminative



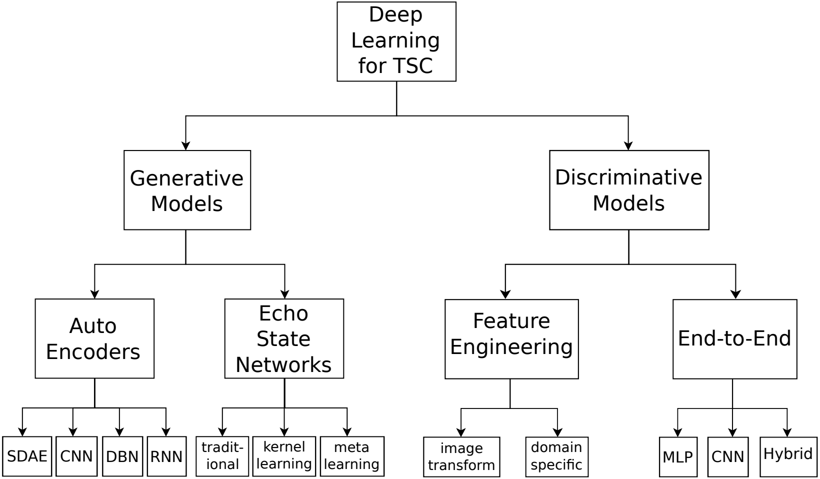
Generative vs Discriminative



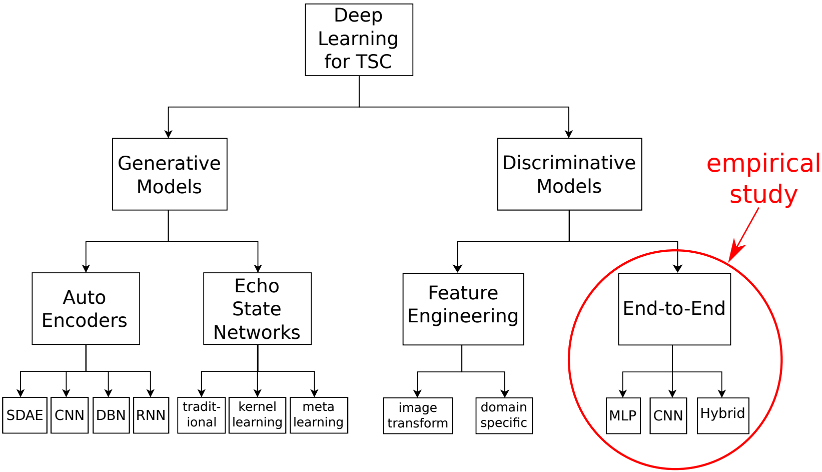
Generative vs Discriminative



Generative vs Discriminative



Generative vs Discriminative



Why focusing on End-to-End approaches ?

- Discriminative Models (DM) instead of Generative Models (GM)
  - GM are usually proposed for tasks other than classification
  - Informal consensus: DM are more accurate than GM [1, 2]
  - GM implementation is much more complicated than DM
  - GM accuracy depends highly on the chosen off-the-shelf classifier
- Why focusing only on End-to-End discriminative approaches ?
  - End-to-End remove any bias due to manually designed features [3]
  - End-to-End learn features directly from raw input data
  - End-to-End are domain agnostic (applicable to any type of dataset)

[1] Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 31(3), 606-660.

[2] Le Nguyen, T., Gsponer, S., & Ifrim, G. (2017). Time Series Classification by Sequence Learning in All-Subsequence Space. In Proceedings of IEEE International Conference on Data Engineering (pp. 947-958)

[3] Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors, 16(1), 115.

Why focusing on End-to-End approaches ?

- Discriminative Models (DM) instead of Generative Models (GM)
  - GM are usually proposed for tasks other than classification
  - Informal consensus: DM are more accurate than GM [1, 2]
  - GM implementation is much more complicated than DM
  - GM accuracy depends highly on the chosen off-the-shelf classifier
- Why focusing only on End-to-End discriminative approaches ?
  - End-to-End remove any bias due to manually designed features [3]
  - End-to-End learn features directly from raw input data
  - End-to-End are domain agnostic (applicable to any type of dataset)

[1] Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 31(3), 606-660.

[2] Le Nguyen, T., Gsponer, S., & Ifrim, G. (2017). Time Series Classification by Sequence Learning in All-Subsequence Space. In Proceedings of IEEE International Conference on Data Engineering (pp. 947-958)

[3] Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors, 16(1), 115.

Improving deep learning's generalization

1. Data augmentation
    - Generating synthetic data to augment the training set
    - Especially useful for small datasets
    - Can be used for imbalanced classification
    - Will be discussed this afternoon [1]
2. Transfer learning [2]
    - Fine-tuning a pre-trained model instead of learning from scratch
    - Potentially useful when transferring between well selected datasets
    - Could negatively impact the model's accuracy if applied naively

[1] Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018). Data augmentation using synthetic data for time series classification with deep residual networks. International Workshop on Advanced Analytics and Learning on Temporal Data, ECML PKDD.

[2] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. In Advances in neural information processing systems (pp. 3320-3328).

Improving deep learning's generalization

1. Data augmentation
   - Generating synthetic data to augment the training set
   - Especially useful for small datasets
   - Can be used for imbalanced classification
   - Will be discussed this afternoon [1]
2. Transfer learning [2]
   - Fine-tuning a pre-trained model instead of learning from scratch
   - Potentially useful when transferring between well selected datasets
   - Could negatively impact the model's accuracy if applied naively

[1] Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018). Data augmentation using synthetic data for time series classification with deep residual networks. International Workshop on Advanced Analytics and Learning on Temporal Data, ECML PKDD.

[2] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. In Advances in neural information processing systems (pp. 3320-3328).

Criteria to implement the approaches

- Cover different types of architectures
- Focus on approaches validated on the whole (or a subset of) the UCR archive [1] and/or Baydogan's MTS archive [2]
- Choose approaches not focusing on specific TSC tasks, e.g.
  - Early Time Series Classification [3]
  - Imbalanced Time Series Classification [4]

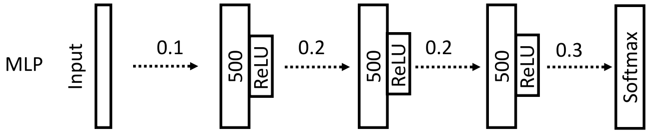
[1] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., & Batista, G. (2015). The UCR time series classification archive. www.cs.ucr.edu/~eamonn/time_series_data/

[2] Baydogan, M. G. (2015). Multivariate Time Series Classification Dataset. http://www.mustafabaydogan.com

[3] Santos, T. and Kern, R. (2017). A Literature Survey of Early Time Series Classification and Deep Learning. In Proceedings of International Conference on Knowledge Technologies and Data-driven Business

[4] Geng, T. and Luo, X. (2018). Cost-Sensitive Convolution based Neural Networks for Imbalanced Time-Series Classification. In ArXiv

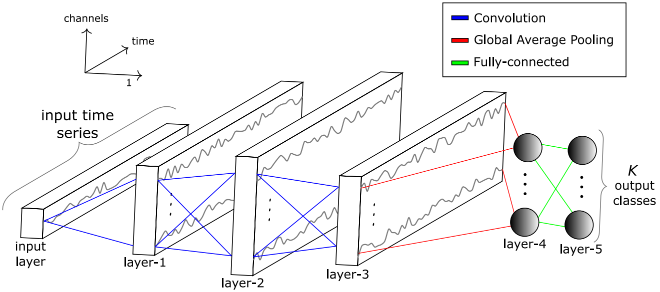
Multi Layer Perceptron (MLP)



- The simplest form of a “deep” learning architecture
- Contains neurons that are fully-connected to each other
- Uses a *Dropout* regularization technique to prevent overfitting
- Loses the temporal information when learning the features
- Depends on the length of the input time series
- Does not contain any transferable layer (across different datasets)

Wang, Z., Yan, W., & Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of International Joint Conference on Neural Networks

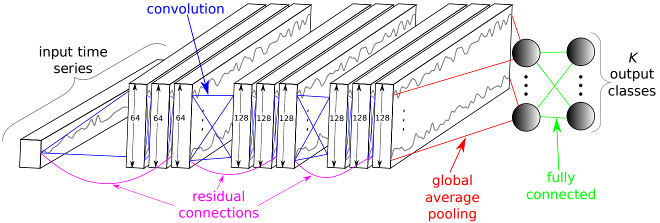
Fully Convolutional Neural Network (FCN)



- ▷ Detects discriminative features that are temporally invariant
- ▷ Contains 4 transferable layers (out of 5 in total)
- ▷ Provides interpretability with the Class Activation Map (CAM)

Wang, Z., Yan, W., & Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of International Joint Conference on Neural Networks

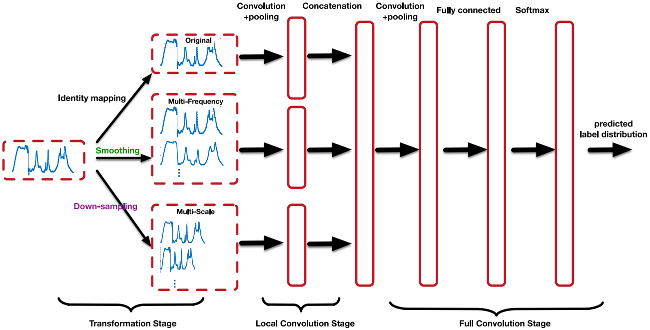
Residual Network (ResNet)



- ▷ Is basically a deeper FCN with residual connections
- ▷ Has the ability to learn to skip unnecessary convolutions
- ▷ Inherits the FCN characteristics such as transferability and CAM

Wang, Z., Yan, W., & Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of International Joint Conference on Neural Networks

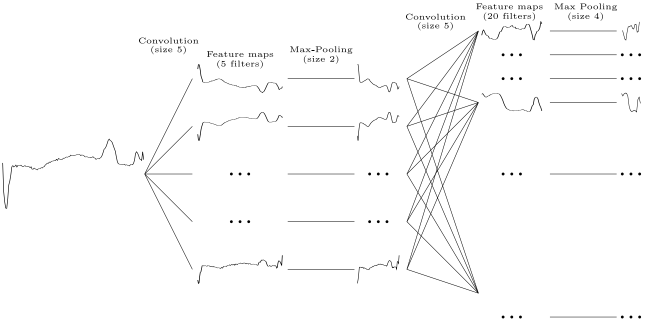
Multi-scale Convolutional Neural Network (MCNN)



- Is one of the earliest DNN approaches validated on the UCR archive
- Transforms an univariate time series into an input MTS for the DNN
- Introduces a cropping data augmentation technique

Cui, Z., Chen, W., & Chen, Y. (2016). Multi-scale convolutional neural networks for time series classification. In ArXiv

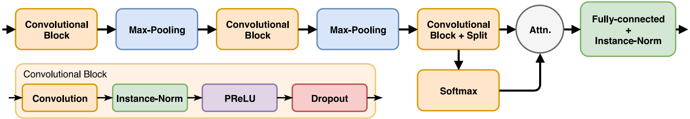
Time Le-Net (t-LeNet)



- ▷ Inspired by the great performance of LeNet's architecture
- ▷ Considered as a traditional CNN architecture
- ▷ Extends MCNN's data augmentation technique with window warping

Le Guennec, A., Malinowski, S., & Tavenard, R. (2016). Data augmentation for time series classification using convolutional neural networks. In Proceedings of ECML/PKDD workshop on advanced analytics and learning on temporal data.

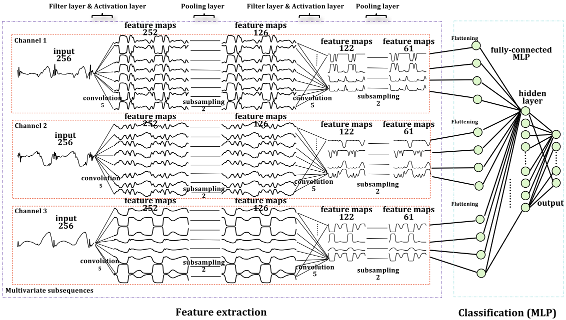
Encoder



- Is a hybrid deep CNN with an *attention* mechanism
- Designed for transferability like FCN and ResNet
- Uses a Dropout regularization technique to prevent overfitting
- Uses relatively long filters (i.e. length of 21)

Serrà, J., Pascual, S., & Karatzoglou, A. (2018). Towards a universal neural network encoder for time series. In ArXiv

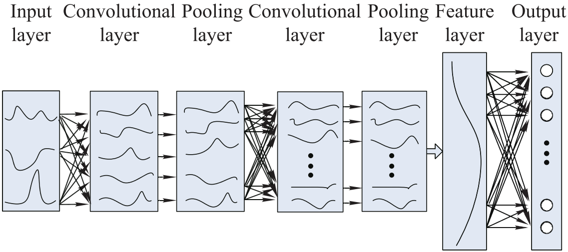
Multi Channel Deep Convolutional Neural Network (MCDCNN)



- ▷ Proposed originally for multivariate time series classification
- ▷ Applies an independent convolution on each channel
- ▷ Uses a second convolution over the concatenated convoluted channels

Zheng, Y., Liu, Q., Chen, E., Ge, Y., & Zhao, J. L. (2016). Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. Frontiers of Computer Science, 10(1), 96-112.

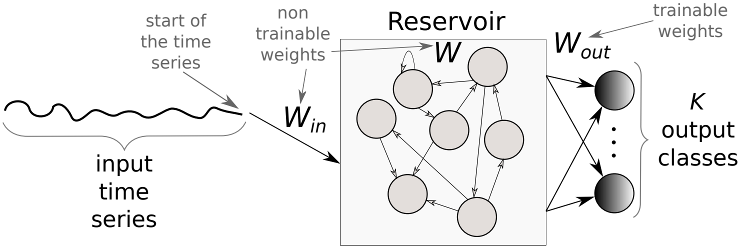
Time Convolutional Neural Network (Time-CNN)



- Proposed originally for both univariate and multivariate time series
- Applies the same convolution over all channels
- Uses a mean-squared error as an objective function
- Contains the lowest number of parameters to learn
- Removes completely the last Global Average Pooling layer

Zhao, B., Lu, H., Chen, S., Liu, J., & Wu, D. (2017). Convolutional neural networks for time series classification. Journal of Systems Engineering and Electronics, 28(1), 162-169.

Time Warping Invariant Echo State Network (TWIESN)



- ▷ Is the only non-convolutional *recurrent* approach considered in this study
- ▷ Trains a Ridge Classifier to predict the class of each time series element
- ▷ Averages the a posteriori probability for each class over all data points

Tanisaro, P., & Heidemann, G. (2016). Time series classification using time warping invariant echo state networks. In Proceedings of IEEE International Conference on Machine Learning and Applications (pp. 831-836).

Hyperparameters

Methods	Architecture							
	#layers	#conv	#invar	normalize	pooling	feature	activate	regularize
MLP	4	0	0	none	none	FC	ReLU	dropout
FCN	5	3	4	batch	none	GAP	ReLU	none
ResNet	11	9	10	batch	none	GAP	ReLU	none
Encoder	5	3	4	instance	max	Att	PReLU	dropout
MCNN	4	2	2	none	max	FC	sigmoid	none
t-LeNet	4	2	2	none	max	FC	ReLU	none
MCDCNN	4	2	2	none	max	FC	ReLU	none
Time-CNN	3	2	2	none	avg	Conv	sigmoid	none

Methods	Optimization						
	algorithm	valid	loss	epochs	batch	learning rate	decay
MLP	AdaDelta	train	entropy	5000	16	1.0	0.0
FCN	Adam	train	entropy	2000	16	0.001	0.0
ResNet	Adam	train	entropy	1500	16	0.001	0.0
Encoder	Adam	train	entropy	100	12	0.00001	0.0
MCNN	Adam	$\text{split}_{20\%}$	entropy	200	256	0.1	0.0
t-LeNet	Adam	train	entropy	1000	256	0.01	0.005
MCDCNN	SGD	$\text{split}_{33\%}$	entropy	120	16	0.01	0.0005
Time-CNN	Adam	train	mse	2000	16	0.001	0.0

Experimental setup

- For univariate TSC: validation on the UCR archive [1]
- For multivariate TSC: validation on the Baydogan's MTS archive [2]

- Each model was evaluated with the accuracy metric at test time
- We used the original train/test split for comparison with other papers
- We tested 9 models on 97 datasets with 10 different random initializations
- Resulting in 8730 deep learning experiments ran on a cluster of 60+ GPUs

- For comparing classifiers, we performed a Friedman test followed by a Wilcoxon Signed Rank Test with Holm's alpha correction [3]
- Critical difference diagrams were used to visualize the classifiers' rank [4]

[1] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., & Batista, G. (2015). The UCR time series classification archive.

[2] Baydogan, M. G. (2015). Multivariate Time Series Classification Datasetq. `http://www.mustafabaydogan.com`.

[3] Garcia, S., & Herrera, F. (2008). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research.

[4] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research

Experimental setup

- ▷ For univariate TSC: validation on the UCR archive [1]
- ▷ For multivariate TSC: validation on the Baydogan's MTS archive [2]

- ▷ Each model was evaluated with the accuracy metric at test time
- ▷ We used the original train/test split for comparison with other papers
- ▷ We tested 9 models on 97 datasets with 10 different random initializations
- ▷ Resulting in 8730 deep learning experiments ran on a cluster of 60+ GPUs

- ▷ For comparing classifiers, we performed a Friedman test followed by a Wilcoxon Signed Rank Test with Holm's alpha correction [3]
- ▷ Critical difference diagrams were used to visualize the classifiers' rank [4]

[1] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., & Batista, G. (2015). The UCR time series classification archive.

[2] Baydogan, M. G. (2015). Multivariate Time Series Classification Datasetq. `http://www.mustafabaydogan.com`.

[3] Garcia, S., & Herrera, F. (2008). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research.

[4] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research

Experimental setup

- For univariate TSC: validation on the UCR archive [1]
- For multivariate TSC: validation on the Baydogan's MTS archive [2]

- Each model was evaluated with the accuracy metric at test time
- We used the original train/test split for comparison with other papers
- We tested 9 models on 97 datasets with 10 different random initializations
- Resulting in 8730 deep learning experiments ran on a cluster of 60+ GPUs

- For comparing classifiers, we performed a Friedman test followed by a Wilcoxon Signed Rank Test with Holm's alpha correction [3]
- Critical difference diagrams were used to visualize the classifiers' rank [4]

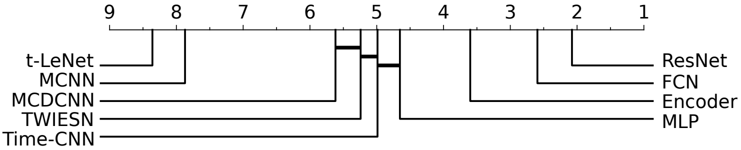
[1] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., & Batista, G. (2015). The UCR time series classification archive.

[2] Baydogan, M. G. (2015). Multivariate Time Series Classification Datasetq. `http://www.mustafabaydogan.com`.

[3] Garcia, S., & Herrera, F. (2008). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research.

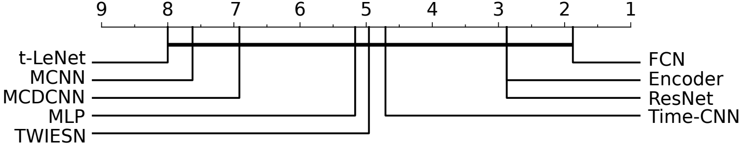
[4] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research

Results on the univariate UCR archive



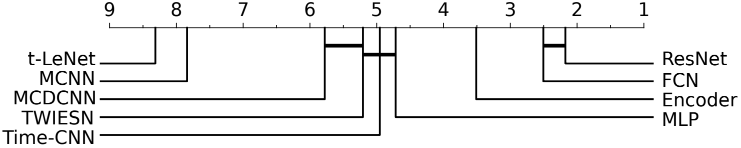
- ResNet won on 50 problems while FCN won on 36 problems showing the benefit of using residual connections and deeper models
- Encoder won only on 17 problems showing the potential superiority of GAP over the attention mechanism
- MLP, Time-CNN and MCDCNN showed a similar average performance probably due to using a final fully-connected layer
- TWIESN (recurrent model) did not manage to out-perform the best CNNs but still showed some promising results
- MCNN and t-LeNet who use the cropping data augmentation technique, obtained the worst performance

Results on Baydogan's MTS archive



CD diagram showing statistical comparison of classifiers for the MTS archive

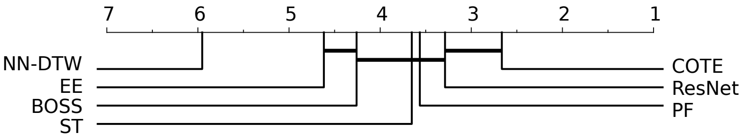
We did not find any significant results when analyzing only the MTS archive



CD diagram showing statistical comparison of classifiers for MTS & UCR archive

Results suggest a careful design of an architecture may be needed for MTS data

Comparing with state of the art classifiers



- ResNet is the only classifier with similar performance to COTE [1]
- BOSS [2], ST [3], PF [4] & ResNet did not show any significant difference
- NN-DTW [5] and EE [6] were far behind ResNet

[1] Bagnall, A., Lines, J., Hills, J., & Bostrom, A. (2015). Time-series classification with COTE: the collective of transformation-based ensembles. IEEE Transactions on Knowledge and Data Engineering, 27(9), 2522-2535.

[2] Schäfer, P. (2015). The BOSS is concerned with time series classification in the presence of noise. Data Mining and Knowledge Discovery, 29(6), 1505-1530.

[3] Hills, J., Lines, J., Baranauskas, E., Mapp, J., & Bagnall, A. (2014). Classification of time series by shapelet transformation. Data Mining and Knowledge Discovery, 28(4), 851-881.

[4] Lucas, B., Shifaz, A., Pelletier, C., O'Neill, L., Zaidi, N., Goethals, B., ... & Webb, G. I. (2018). Proximity Forest: An effective and scalable distance-based classifier for time series. ArXiv

[5] Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 31(3), 606-660.

[6] Lines, J., & Bagnall, A. (2015). Time series classification with ensembles of elastic distance measures. Data Mining and Knowledge Discovery, 29(3), 565-592.

Performance with respect to the dataset's theme

Themes (#)	MLP	FCN	ResNet	Encoder	MCNN	t-LeNet	MCDCNN	Time-CNN	TWIESN
DEVICE (6)	0.0	**66.7**	**66.7**	0.0	0.0	0.0	0.0	0.0	0.0
ECG (7)	0.0	**71.4**	14.3	42.9	0.0	0.0	14.3	0.0	0.0
IMAGE (29)	3.4	41.4	**44.8**	17.2	0.0	0.0	0.0	6.9	3.4
MOTION (14)	14.3	42.9	**57.1**	21.4	0.0	0.0	0.0	0.0	0.0
SENSOR (16)	6.2	37.5	**81.2**	25.0	6.2	6.2	0.0	0.0	6.2
SIMULATED (6)	0.0	33.3	**100.0**	33.3	0.0	0.0	0.0	0.0	0.0
SPECTRO (7)	28.6	14.3	**71.4**	0.0	0.0	0.0	0.0	28.6	14.3

Percentage of dataset themes an algorithm is most accurate for

- ▷ ResNet still dominates for most themes
- ▷ FCN performs exceptionally well for ECG datasets
- ▷ Small sample size suggests no conclusive evidence [1]

[1] Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 31(3), 606-660.

Performance with respect to the time series' length

Length	MLP	FCN	ResNet	Encoder	MCNN	t-LeNet	MCDCNN	Time-CNN	TWIESN
<81	4.38	2.75	**2.56**	3.12	7.38	7.81	4.88	4.62	4.88
81-250	4.21	2.42	**1.46**	3.33	7.79	8.46	5.88	4.79	5.17
251-450	4.48	**2.13**	2.3	2.87	7.57	8.22	5.43	4.74	5.13
451-700	4.86	1.71	**1.29**	3.86	7.64	7.29	6.29	4.64	5.07
701-1000	4.08	**2.08**	2.42	3.25	7.67	7.83	5.42	4.92	5.08
>1000	4.75	2.12	**1.75**	3.38	7.88	8.75	5.12	4.75	5.12

Average ranks grouped by the datasets' length

- ▷ Time series length does not give information on the classifiers' performance
- ▷ Short filter lengths did not affect FCN and ResNet's performance
- ▷ TWIESN maintained its performance for long time series thus mitigating the vanishing gradient effect of RNNs

Performance with respect the dataset's train size

Train size	MLP	FCN	ResNet	Encoder	MCNN	t-LeNet	MCDCNN	Time-CNN	TWIESN
<100	4.18	2.15	**1.61**	3.39	7.79	8.15	5.94	4.76	4.94
100-399	4.6	2.33	**2.28**	3.22	7.52	8.1	5.12	4.55	5.08
400-799	4.0	2.12	**1.38**	3.0	8.12	8.12	6.62	5.0	5.62
>799	4.69	2.25	**2.19**	3.19	7.38	7.81	5.25	5.06	5.12

Average ranks grouped by the training sizes

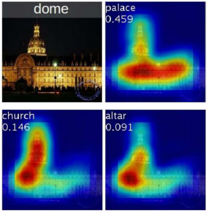
- ▷ ResNet and FCN still dominates for most training set sizes

Focus on DiatomSizeReduction:

- ▷ The smallest train size in the archive
- ▷ ResNet and FCN have the worst performance ($\approx$ 30%)
- ▷ Time-CNN, the smallest deep model, reached the highest accuracy (97%)
- ▷ $\rightsquigarrow$ Overfitting small datasets seems limited with relatively small models

The need of interpretability

- ▷ Important and active area of research [1]
- ▷ Explaining the classifier's decision [1]
- ▷ Critical in domains such as medicine [1] and finance [2]
- ▷ Deep learning is sometimes avoided for its black-box effect
- ▷ Class Activation Map (CAM) was proposed for images [3]
  - ▷ Highlights a region's contribution in a classification
  - ▷ Requires a Global Average Pooling (GAP)



source : [3]

[1] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). Why should i trust you?: Explaining the predictions of any classifier. In Proceedings ACM SIGKDD International Conference on Knowledge Discovery and Data mining (pp. 1135-1144).

[2] Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). Predicting mortgage default using convolutional neural networks. Expert Systems with Applications, 102, 207-217.

[3] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (pp. 2921-2929).

Class Activation Map for time series

- For TSC, it highlights subsequences' contribution to a certain classification
- Input sent to neuron's class c is the contribution to classifying as c
- GAP enables projecting back the weights for class c to the last convolution

The CAM can be be derived from the input of class c:

input class c

weight class c filter m

$$z_c = \sum_m w_m^c \sum_t A_m(t) \iff z_c = \sum_t \underbrace{\sum_m w_m^c A_m(t)}_{CAM_c(t)}$$

filter m

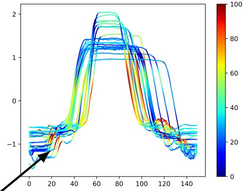
output of last convolution

for class c at time t

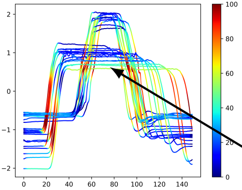
Wang, Z., Yan, W., & Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of International Joint Conference on Neural Networks

Example of CAM on GunPoint: (FCN & ResNet have respectively 100% & 99% accuracy)

source : Eamonn Keogh ©



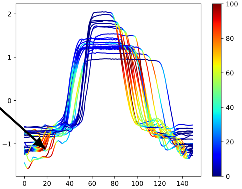
(a) FCN on GunPoint: Class-1



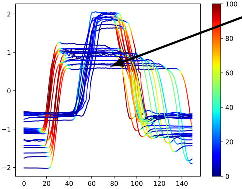
(b) FCN on GunPoint: Class-2

non-discriminative region (plateau) detected

discriminative region (bump) detected

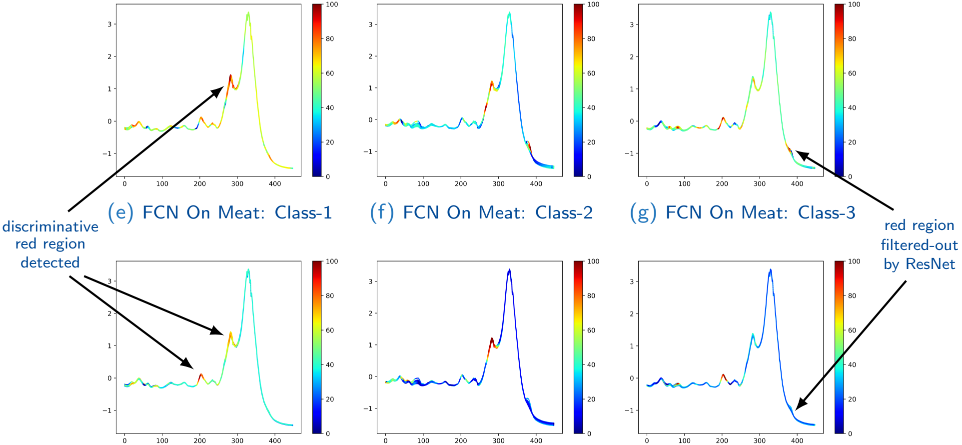


(c) ResNet on GunPoint: Class-1



(d) ResNet on GunPoint: Class-2

Example of CAM on Meat: (FCN & ResNet have respectively 83% & 97% accuracy)

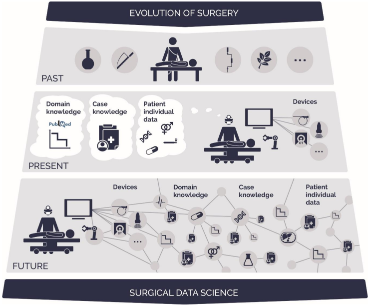


(e) FCN On Meat: Class-1 (f) FCN On Meat: Class-2 (g) FCN On Meat: Class-3

(h) ResNet On Meat: Class-1 (i) ResNet On Meat: Class-2 (j) ResNet On Meat: Class-3

Surgical Data Science: Enabling Next-Generation Surgery

- Data Science for Surgical Data
- Emerging field
- Surgical process generates data (before, during and after surgery)
- **How to use data science to support patients and surgeons ?**



Maier-Hein, L. et al (2017). Surgical data science for next-generation interventions. Nature Biomedical Engineering, 1(9), 691.

Surgical Data Science

- ▷ Evolution of surgery $\Longrightarrow$ adapting training [1]
- ▷ Problem: Subjective/un-explainable evaluation [2]
- ▷ Need: Improved feedback for surgical training [3]
  1. Objective
  2. Accurate
  3. Interpretable

source : Mark Clifford ©

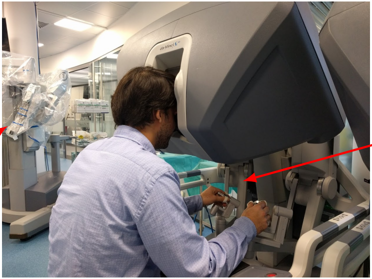
[1] Polavarapu, H. V., Kulaylat, A. N., Sun, S., & Hamed, O. H. (2013). 100 years of surgical education: the past, present, and future. Bull Am Coll Surg, 98(7), 22-27.

[2] Kassahun, Y., Yu, B., Tibebu, A. T., Stoyanov, D., Giannarou, S., Metzen, J. H., & Vander Poorten, E. (2016). Surgical robotics beyond enhanced dexterity instrumentation: a survey of machine learning techniques and their role in intelligent and autonomous surgical actions. International journal of computer assisted radiology and surgery, 11(4), 553-568.

[3] Islam, G., Kahol, K., Li, B., Smith, M., & Patel, V. L. (2016). Affordable, web-based surgical skill training and evaluation tool. Journal of biomedical informatics, 59, 102-114.

Da Vinci Surgical System

slave robot end effectors controlled by the master manipulators



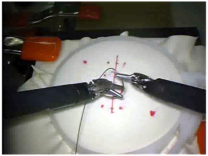
master manipulators controlled by the surgeon

Photo taken during a visit to IRCAD-Strasbourg (France)

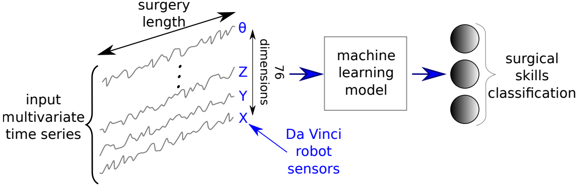
Challenges

- ▷ Multiple sensors $\Longrightarrow$ multivariate time series
- ▷ High sampling rates $\Longrightarrow$ big data
- ▷ Heterogeneous data $\Longrightarrow$ hybrid models

JIGSAWS



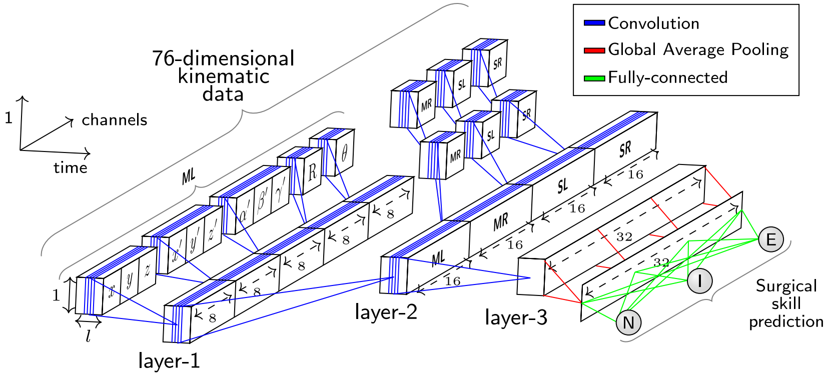
The three surgical tasks (from left to right): suturing, knot-tying, needle-passing



Pipeline for automatic surgical skills identification

Gao, Y., Vedula, S. S., Reiley, C. E., Ahmidi, N., Varadarajan, B., Lin, H. C., & Chen, C. C. G. (2014). JHU-ISI gesture and skill assessment working set (JIGSAWS): A surgical activity dataset for human motion modeling. In MICCAI Workshop: M2CAI (Vol. 3, p. 3).

Architecture



Fully Convolutional Neural Network architecture for surgical skills identification

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018). Evaluating surgical skills from kinematic data using convolutional neural networks. International Conference On Medical Image Computing and Computer Assisted Intervention (MICCAI)

Experiments & Results

- ▷ No hyper-parameters optimization $\Longrightarrow$ same architecture for all tasks
- ▷ For each task: a model was trained on an Nvidia GTX 1080 Ti
- ▷ Leave One Super trial Out (LOSO) cross-validation scheme

Method	Suturing		Needle Passing		Knot Tying	
	Micro	Macro	Micro	Macro	Micro	Macro
S-HMM [1]	97.4	n/a	96.2	n/a	94.4	n/a
ApEn [2]	**100**	n/a	**100**	n/a	**99.9**	n/a
Sax-Vsm [3]	89.7	86.7	96.3	95.8	61.1	53.3
CNN (proposed)	**100**	**100**	**100**	**100**	92.1	**93.2**

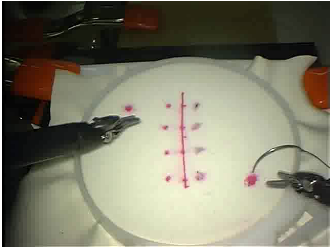
Surgical skill classification results (%)

[1] Tao, L., Elhamifar, E., Khudanpur, S., Hager, G. D., & Vidal, R. (2012). Sparse hidden markov models for surgical gesture classification and skill evaluation. In International conference on information processing in computer-assisted interventions (pp. 167-177).

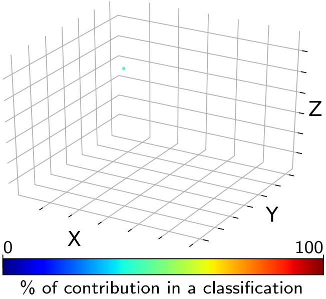
[2] Zia, A., & Essa, I. (2018). Automated surgical skill assessment in RMIS training. International journal of computer assisted radiology and surgery, 13(5), 731-739.

[3] Forestier, G., Petitjean, F., Senin, P., Despinoy, F., & Jannin, P. (2017). Discovering discriminative and interpretable patterns for surgical motion analysis. In Proceedings of Artificial Intelligence in Medicine in Europe (pp. 136-145).

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task

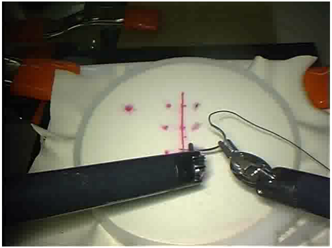


(b) Trial's corresponding trajectory for the left master manipulator

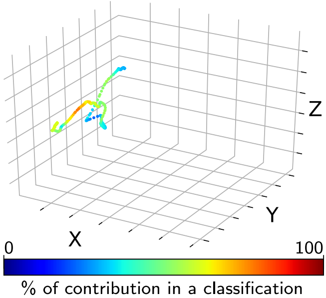
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task

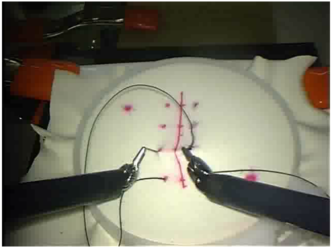


(b) Trial's corresponding trajectory for the left master manipulator

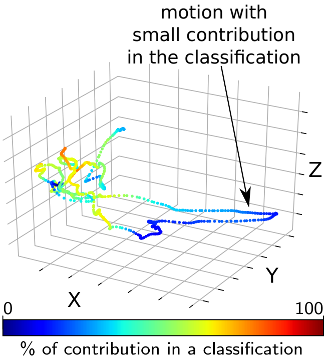
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task

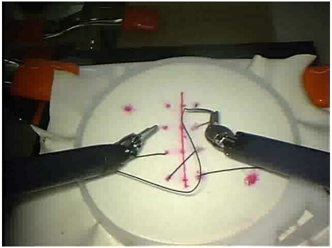


(b) Trial's corresponding trajectory for the left master manipulator

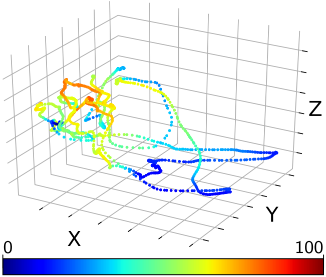
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task

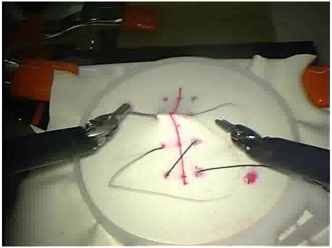


(b) Trial's corresponding trajectory for the left master manipulator

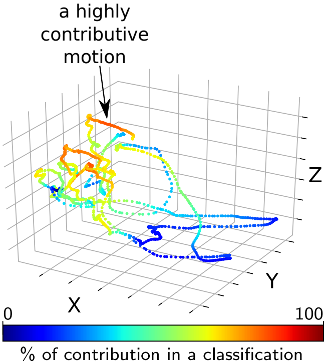
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task



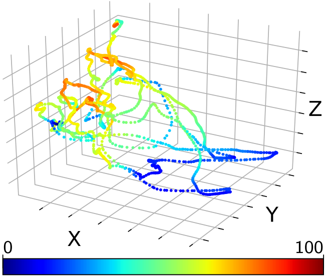
(b) Trial's corresponding trajectory for the left master manipulator

Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification

(a) The last frame of subject (Novice) H's fourth trial of the suturing task

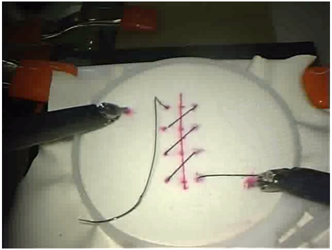


(b) Trial's corresponding trajectory for the left master manipulator

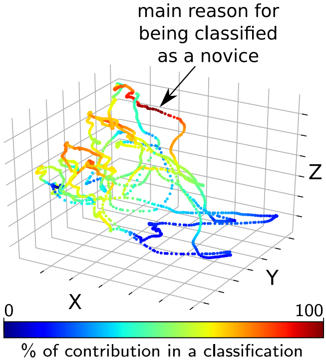
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task

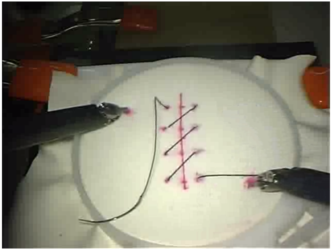


(b) Trial's corresponding trajectory for the left master manipulator

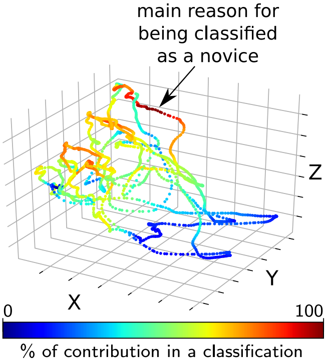
Example of feedback using the Class Activation Map

A video illustrating this feedback is available on https://germain-forestier.info/src/miccai2018/

Interpretable skill identification



(a) The last frame of subject (Novice) H's fourth trial of the suturing task



(b) Trial's corresponding trajectory for the left master manipulator

Example of feedback using the Class Activation Map

A video illustrating this feedback is available on
`https://germain-forestier.info/src/miccai2018/`

Future directions

- Cross-validate the network's hyper-parameters for each dataset
- Evaluate on a larger archive especially for MTS data
- Use a meta-learning approach when building a deep TSC model
- Create some guidelines for designing data-dependent models for TSC
- Investigate new data augmentation techniques specific for time series data
- Provide time series practitioners with pre-trained models
- Adapt current transfer learning methods for time series data
- Extend the current empirical study to generative deep learning models
- Study the accuracy of deep Recurrent Neural Networks for TSC
- Create a community arround deep learning for TSC

Thanks!

More details : https://arxiv.org/abs/1809.04356

Source code available : https://github.com/hfawaz/dl-4-tsc

People involved in this work:
Hassan Ismail Fawaz, Jonathan Weber,
Lhassane Idoumghar, Pierre-Alain Muller

More info on our projects: http://germain-forestier.info

Acknowledgements : NVIDIA Corp. for the GPU Grant, Mésocentre for providing access to their cluster, François Petitjean and Charlotte Pelletier for their feedback.