NeuralWarp: Time-Series Similarity with Warping Networks

한양대학교 산업공학과
IDSL 석사과정 고경준

HANYANG UNIVERSITY

Contents

1. Introduction

Introduction

- DTW, TWED(TimeWarpEditDistance) 등의 방식은 static. (non-parametric)
- 일반적으로 image는 CNN, text는 RNN model으로 similarity를 측정.
- 하지만 Siamese architecture는 time-series instance의 elastic alignment를 반영하지 않음.
- 따라서 deep neural representation에서 time-series의 alignment를 반영하는 방법을 제시.
- Optimal alignment paths는 warping indicator function을 포함하는 all-pairs distance로 재 표현될 수 있다는 사실에서 motivated.

Contributions

- Deep representation space에서 time-series indices의 alignment를 modeling.
- Similarity model을 활용한 time-series classification 적용.

2. Related Work

Similarity Learning

- Similarity learning은 image, music, video, text context 등 다양한 분야에 필요함.
- Deep similarity measure는 Siamese network에 의해 학습됨.

Time-series Similarity

- Time-series similarity는 다른 분야와 다르게 static measures가 주로 연구되었음.
- DTW와 관련하여 run-time을 줄이거나 loss function에 대한 적용, soft-min DTW, all-pairs similarity join에 관한 연구 등이 진행되었음.

Deep Learning Similarity for Time Series

- 주로 RNN과 CNN을 통해 modeling됨. (LSTM, LSTM fully convolutional networks etc.)
- Time series의 latent representation을 추출하는 Siamese network가 연구됨.

2. Related Work

Novelty and Research Hypothesis

Deep learning이 아직 time-series similarity measure를 완전히 optimizing하지 못 함.
Warping aspect는 deep network에 직접적으로 modeling되지 못 함.

- two sequences 사이의 alignment(warping path)를 deep representation에서의 neural network function으로 modeling
- Latent space에서 value들을 align하는 warping neural network과 함께 time series의 deep embedding을 함께 학습.
- 이후에 Non-parametric warping measure, Un-warped deep variants와 성능 비교.

3. Alignment Paths

Alignment Paths of DTW

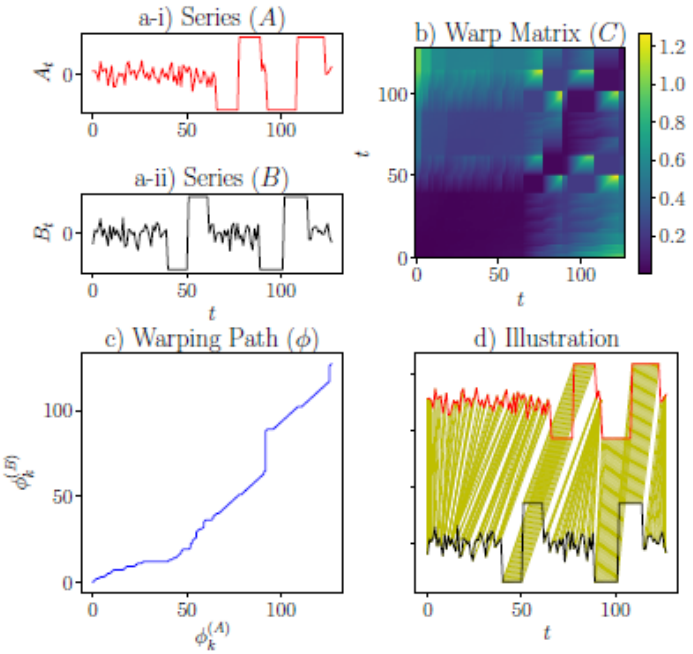
$$\mathcal{D}(A,B) := \min_{\phi} \sum_{k=1}^{|\phi|} ||A_{\phi_k^{(A)}} - B_{\phi_k^{(B)}}||_2^2 \quad (1)$$

$$\begin{pmatrix} \phi_k^{(A)} - \phi_{k-1}^{(A)} \\ \phi_k^{(B)} - \phi_{k-1}^{(B)} \end{pmatrix} := \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\} \quad (2)$$

$$D(A,B) = C_{T,T}, \quad \text{where:} \quad (3)$$

$$C_{i,j} = ||A_i - B_j||_2^2 + \min\{C_{i-1,j}, C_{i,j-1}, C_{i-1,j-1}\} \quad (4)$$

- $A \in R^{TXD}$, $B \in R^{TXD}$ with T measurements of D channels
- $(\phi_k^{(A)}, \phi_k^{(B)})$, $\phi \in \{1,\ldots,T\}$

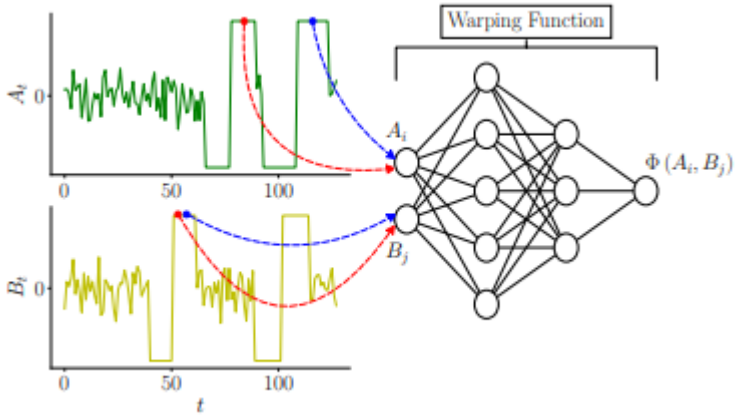


4. Parametric Warping Similarity

Indicator function of warping path

$$\mathcal{D}(A,B) = \sum_{i=1}^{T}\sum_{j=1}^{T} \|A_i - B_j\|_2^2 \ \Phi(A_i, B_j;\ \phi) \quad (5)$$

$$\Phi(A_i, B_j;\ \phi) = \begin{cases} 1 & (i,j) \in \phi \\ 0 & (i,j) \notin \phi \end{cases} \quad (6)$$



- (1)을 (5)로 재표현하면, (6)의 indicator function을 supervised manner로 정의해야 함.
- 단순히 두 time-series를 input, binary indicator를 output으로 하는 NN모형은 한계가 존재.
- Time-series value 자체를 그대로 input으로 사용하는 대신, index의 context를 포함하는 version의 input variable이 필요.

(DTW는 warping matrix의 계산 과정에서 index context 정보가 포함)

4. Parametric Warping Similarity

NeuralWarp

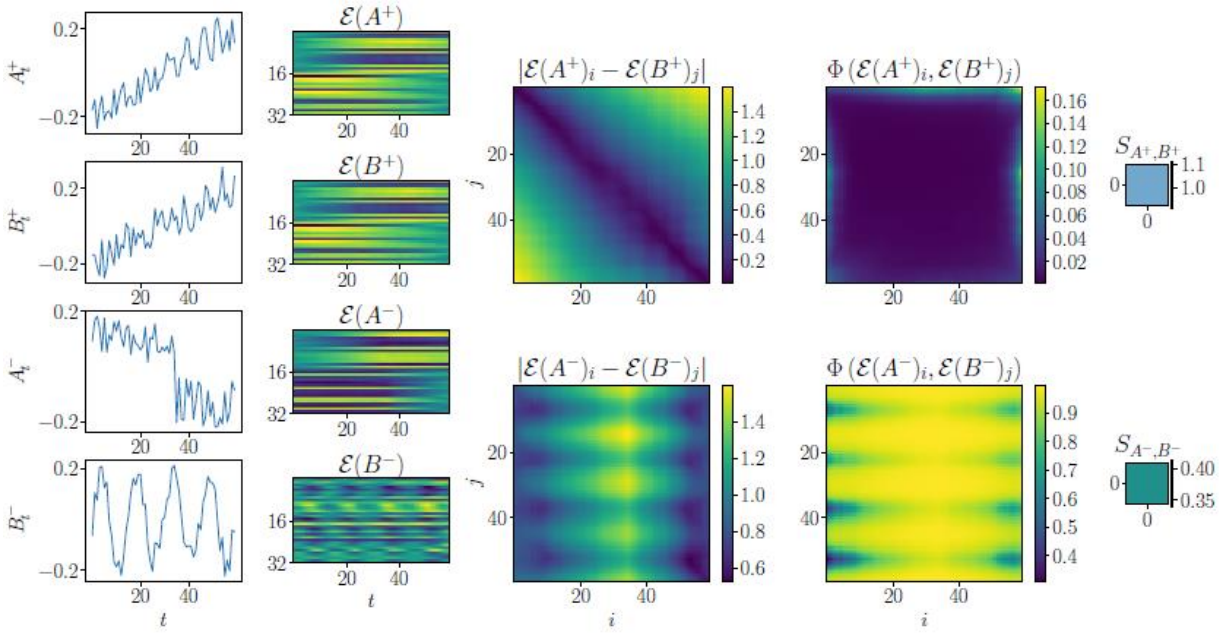
- Encoder function으로 time series index의 context 정보를 포함하는 vector를 추출
- Encoder function : $\mathcal{E} : \mathbb{R}^{T \times D} \rightarrow \mathbb{R}^{T \times K}$ (CNN or RNN)
- Warper function : $\Phi : \mathbb{R}^{2K} \rightarrow [0, 1]$ (fully-connected deep forward network)

$$\mathcal{S}_{A,B} = \exp\left(-\frac{1}{T^2}\sum_{i=1}^{T}\sum_{j=1}^{T}|\mathcal{E}(A)_i - \mathcal{E}(B)_j|\,\Phi\left(\mathcal{E}(A)_i, \mathcal{E}(B)_j\right)\right) \quad (7)$$

- K-dimensional context vector들을 warper function의 alignment에 대해 거리 계산.
- Context difference는 L-1 norm이 가장 좋은 성능을 보임.
- Warped all-pairs distance을 negative exponential function으로 probability로 표현.

4. Parametric Warping Similarity

Encoder, Warper 예시



- 1-layer RNN encoder (16 cells) + 4-layers fully-connected warper (16,8,4,1 cells)
- (A^+, B^+)와 (A^-, B^-)는 similar/dissimilar time series pairs를 의미.
- Warper Φ 가 similar input에는 작고 dissimilar input에는 커지도록 encoder $\mathcal{E}$ 학습.

4. Parametric Warping Similarity

Optimization Objective

$$\mathcal{L}(\mathcal{E},\Phi) =: \arg\min_{\mathcal{E},\Phi} \frac{1}{|\mathcal{P}|}\sum_{(A^+,B^+)\in\mathcal{P}} \log\left(\mathcal{S}_{A^+,B^+}(\mathcal{E},\Phi)\right) + \frac{1}{|\mathcal{N}|}\sum_{(A^-,B^-)\in\mathcal{N}} \log\left(1-\mathcal{S}_{A^-,B^-}(\mathcal{E},\Phi)\right) \quad (8)$$

$$\mathcal{P} = \{(T_n,T_m) \mid Y_n = Y_m\},\ \ \mathcal{N} = \{(T_p,T_q) \mid Y_p \neq Y_q\}$$

- Similar/dissimilar pairs는 classification을 위한 data set에서 같은 class면 similar로 가정.
- Similar pairs의 거리는 줄이고 dissimilar pairs의 거리는 늘리는 의미.

4. Parametric Warping Similarity

Optimization Objective

Algorithm 1: Learning the NeuralWarp similarity

Input : List of similar series pairs $\mathcal{P}$, List of dissimilar series pairs $\mathcal{N}$, Number of iterations I, Batch size K, Learning rate η.

for $i = 1, \ldots, I$ **do**

$(A_k^+, B_k^+) \sim \mathcal{P}; \quad (A_k^-, B_k^-) \sim \mathcal{N}; \quad k = 1, \ldots, K;$

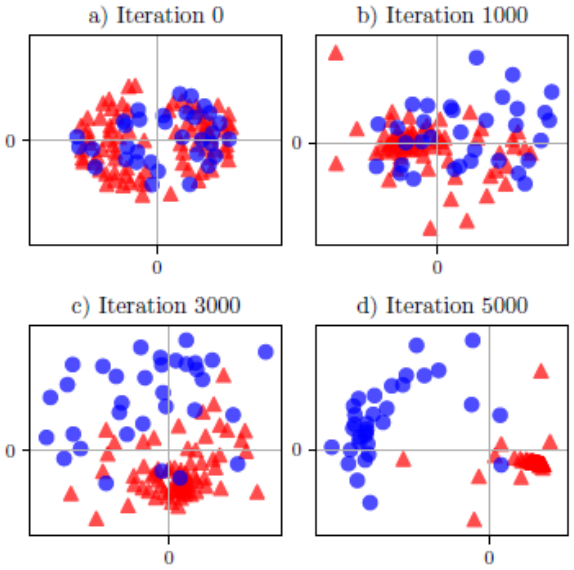
$\mathcal{L}^{(i)} = \frac{1}{K} \sum_{k=1}^{K} \log\left(S_{A_k^+, B_k^+}\right) + \log\left(1 - S_{A_k^-, B_k^-}\right);$

$\theta_{\mathcal{E}} \leftarrow \theta_{\mathcal{E}} - \eta_{\theta_{\mathcal{E}}}^{(i)} \frac{\partial \mathcal{L}^{(i)}}{\partial \theta_{\mathcal{E}}};$

$\theta_{\Phi} \leftarrow \theta_{\Phi} - \eta_{\theta_{\Phi}}^{(i)} \frac{\partial \mathcal{L}^{(i)}}{\partial \theta_{\Phi}};$

end

Return: Model parameters $\theta_{\mathcal{E}}, \theta_{\Phi}$



- 일반적인 방식. Adam optimizing strategy로 learning rate을 iteration마다 updating.
- Iteration이 커짐에 따라 다른 class의 time series의 거리를 멀게 측정하는 모습.

Language translation의 Attention mechanism과 some intuition을 공유.
따라서 저자는 NeuralWarp을 form of generalized attention으로 구분한다.

5. Experiments

Baselines

- Non-parametric : DTW, TWED
- Siamese Deep Similarity : RNN, CNN
- NeuralWarp : RNN-W, CNN-W

Experimental Protocol

- Hyper-parameter들은 참고 논문들의 사례를 참고.
- Loss가 발산하는 경우 learning rate를 작은 값으로 수정 등.

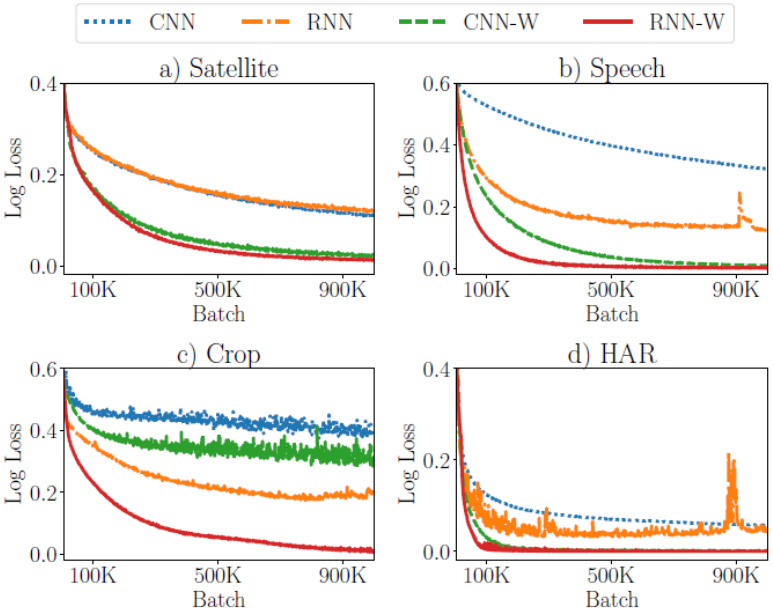
Datasets

- Satellite, Google Speech, Crop, HAR 4가지 datasets
- Large set of uni-variate, small and simplified sets 등 over-fit 위험이 큰 데이터 사용.

5. Experiments

Results

Dataset	Dataset Description				Non-parametric		Siamese		NeuralWarp	
	Train-Test	Classes	Length	Channels	DTW	TWED	CNN	RNN	CNN-W	RNN-W
Satellite	89720-9967	9	23	10	0.8781	0.7630	0.9166	0.8867	**0.9295**	0.9206
Speech	59987-6471	30	71	13	0.6525	0.0355	0.2896	0.7894	0.8605	**0.8969**
Crop	21600-2400	24	46	1	0.7553	**0.7690**	0.6433	0.7029	0.6117	0.7604
HAR	3531-392	17	57	3	0.7500	0.7678	0.9566	0.9642	0.9617	**0.9719**
		Wins			0	1	0	0	1	2
		Ranks			4.5 ± 1.2	4.5 ± 2.38	4.3 ± 0.9	3.2 ± 0.9	3.0 ± 2.1	1.5 ± 0.6

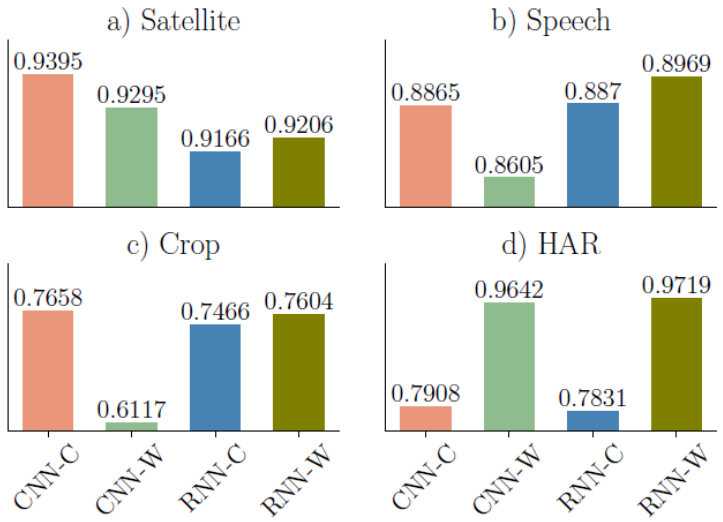


5. Experiments

Results

Dataset	Siamese		NeuralWarp	
	CNN	RNN	CNN-W	RNN-W
Satellite	16/30	25/80	32/69	47/113
Speech	26/28	78/39	47/32	95/92
Crop	24/23	56/22	38/23	65/24
SHAR	21/0.5	41/0.6	36/0.7	78/0.8
Number of weights	1.41M	1.04M	1.42M	1.05M

$$\frac{\partial \mathcal{S}_{A,B}}{\partial \mathcal{E}(A)_i} = -\frac{\mathcal{S}_{A,B}}{T^2} \sum_{j=1}^{T} \frac{\mathcal{E}(A)_i - \mathcal{E}(B)_j}{|\mathcal{E}(A)_i - \mathcal{E}(B)_j|} \frac{\partial \Phi\left(\mathcal{E}(A)_i, \mathcal{E}(B)_j\right)}{\partial \mathcal{E}(A)_i} \quad (9)$$



- Similarity learning은 instance-wise classification과 전혀 다른 task이고,
- Classification은 training에 포함되지 않은 경우는 operate할 수 없음.
- $Z_A = \frac{1}{T}\sum_{i=1}^{T} \mathcal{E}(A)_i$ 으로 설정 후, Z_A의 soft-max로 classification
- 당연히 classification model 성능이 우수해야 하는데, RNN의 경우 NeuralWarp이 더 좋음.

감사합니다.