Semi-Supervised Time Series Classification

Self-Supervised Learning Approach

*Shayan Jawed, Josif Grabocka & Lars Schmidt-Thieme.
University of Hildesheim, Germany.

*presenting author

Introduction

Learning Generalizable Representations

Semi-supervision

Semi-supervised learning is a central problem in machine learning.

- Learn with labeled and unlabeled data.

- Extensive applicability to time series especially.

Self-supervision

Self-supervised learning enables representation learning on unlabeled data.

- Defines an annotation-free pretext task in data itself.

- Task provides a surrogate supervision signal for feature learning.

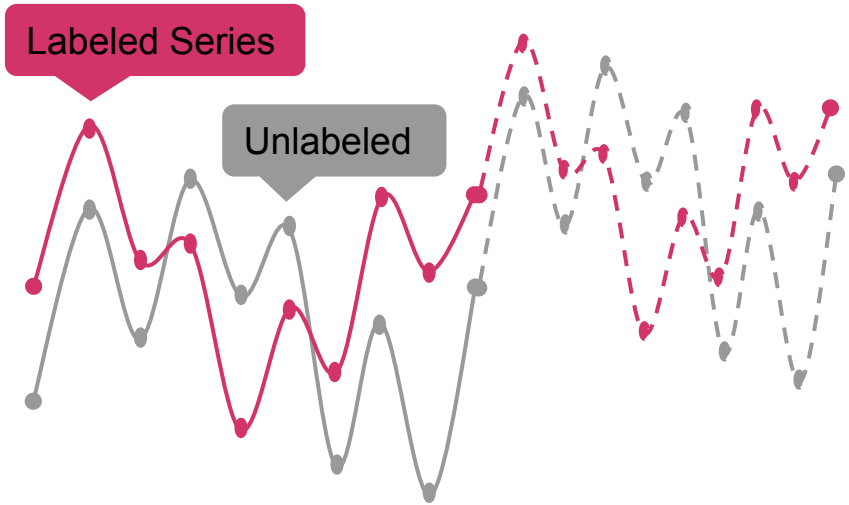
Multi-tasking

A set of tasks is learned in parallel to share knowledge in between.

- Aim is to do better than learning 1 task in isolation.

- Several domains have benefitted from the MTL paradigm.

Problem Formulation



A ConvNet model is to jointly classify the labeled samples and forecast future series values.

We propose an auxiliary forecasting task that is inherent in labeled and unlabeled time series data both.

Method

Notation for problem formulation

Self-supervised learning

Network architecture

Multi-task learning

Methodology

The problem is formulated as a multi-task process jointly estimating $f(.)$ and $g(.)$ for forecasting and classification resp.

Univariate time series: $X = \{X_1, X_2, \ldots, X_n\}$

Labeled and Unlabeled Sets: $X^U = \{X_1^U, X_2^U, \ldots, X_k^U\}$ & $X^L = \{X_1^L, X_2^L, \ldots, X_l^L\}$

Sliding window function $w(.)$ parametrized by horizon h and stride s

Methodology

The loss function w.r.t $f(.)$

$$L_f(X^F, \theta_f) = \frac{1}{n \times m \times h} \sum_i^n \sum_j^m \sum_t^h (y_{jt}^i - \hat{y}_{jt}^i)^2$$

With respect to $g(.)$ minimizing cross entropy for C classes:

$$L_c(X^L, \theta_c) = -\frac{1}{l} \sum_i^l log \left(\frac{e^{\hat{y}_{i=c}}}{\sum_j^C e^{\hat{y}_i}} \right)$$

Forecasting as a Self-Supervised task

Forecasting as an auxiliary task forces the ConvNet to learn a set of rich hidden state representations from unlabeled but structured data.

If an unlabeled sample belongs to the same class as a labeled sample:

- The latent features that were activated for the unlabeled sample could be also leveraged to classify.

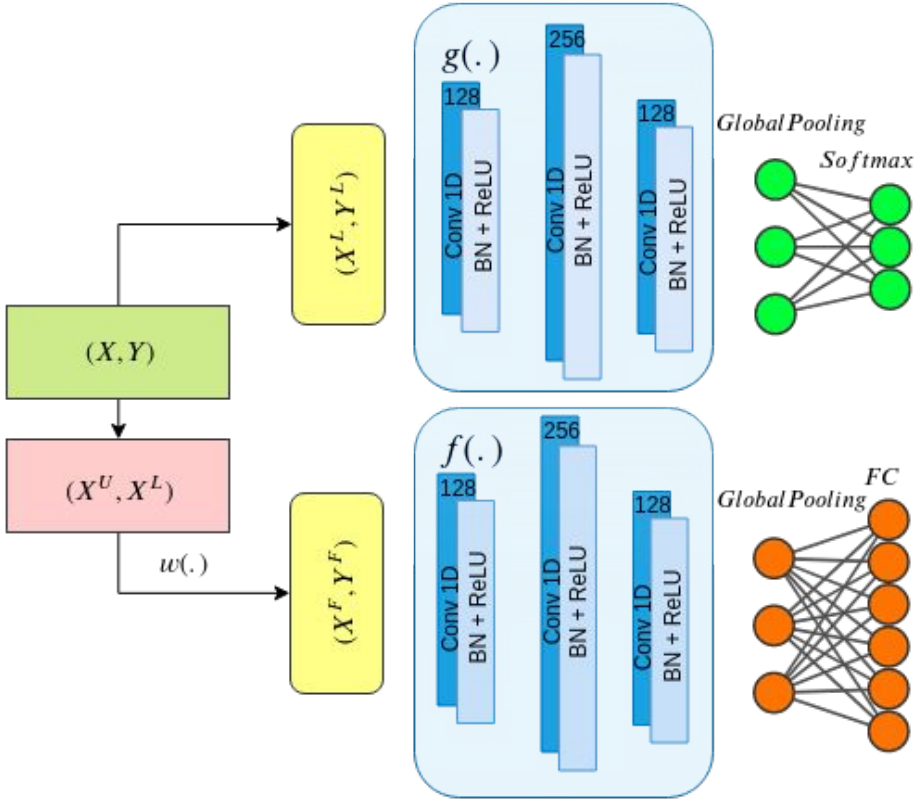
Multi-task Learning Approach

Forecasting and Classification tasks mutually benefit each other by sharing parameters in between.

Input distribution is the same.

Representations are to be correlated in turn as well.

Network Architecture



Multi-task Learning Approach

We specifically cast forecasting as an auxiliary task by setting the loss component weight to be marginal compared to main classification loss.

The approach is cast as an optimization process over the weighted sum of the 2 loss functions:

$$L_{MTL}(X^F, \theta_f, X^L, \theta_c) = L_c(X^L, \theta_c) + \lambda L_f(X^F, \theta_f)$$

Experiments

Datasets

Baselines

Results

Baselines

We compare against several baselines from over the years. Additionally as ablations:

- Base model
  - Single task network for classification task only.
- Transfer Learning
  - Common in the regime of self-supervised learning based methods.
  - Training a vastly different network for forecasting task only.
  - Evaluating self-supervised learned features by measuring classification accuracy without fine-tuning.

Proposed Methods vs. Baselines

Datasets	*Results verbatim from table in [17]*						*Proposed*			
	Wei.	**DTW-D**	**SUC.**	**Xu.**	**BoW**	**SSSL**	**Base**	**II**	**Tr.**	**MTL**
Coffee	0.571	0.601	0.632	0.588	0.620	0.792	**1.0**	**1.0**	**1.0**	**1.0**
CBF	0.995	0.833	0.997	0.921	0.873	**1.0**	**1.0**	**1.0**	0.784	**1.0**
ECG	0.763	0.953	0.775	0.819	0.955	0.793	0.9	0.875	0.9	**0.975**
FaceFour	0.818	0.782	0.800	0.833	0.744	0.851	0.913	0.913	0.739	**0.957**
OSULf.	0.468	0.701	0.534	0.642	0.685	0.835	0.977	0.977	0.460	**0.978**
ItalyPower	0.934	0.664	0.924	0.772	0.813	0.941	0.986	0.986	0.959	**0.991**
Light.2	0.658	0.641	0.683	0.698	0.721	0.813	0.92	0.84	0.88	**0.92**
Light.7	0.464	0.503	0.471	0.511	0.677	0.796	0.758	0.689	0.482	**0.828**
GunPoint	0.925	0.711	0.955	0.729	0.925	0.824	**1.0**	**1.0**	0.825	**1.0**
Trace	0.950	0.801	**1.0**	0.788	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
WordSyn.	0.590	0.863	0.618	0.639	0.795	**0.875**	0.497	0.491	0.342	0.519
OliveOil	0.633	0.732	0.617	0.639	0.766	0.776	0.916	**1.0**	0.833	**1.0**
StarLight	0.860	0.743	0.800	0.755	0.851	0.872	0.982	0.983	**1.0**	0.991

Main findings

Transfer learning results serve to quantify the usefulness of features learned purely for the self-supervised forecasting task.

Without any fine-tuning on these features we are still learn a useful classifier.

Feature spaces thus correlate heavily among the forecasting and classification tasks.

Main findings

Best results were achieved where the stride and horizon lead to the most samples.

Network was found to be robust against the different values of λ

Network performance was biased to the initial labeled set.

- Averaging 10 times helped fixed this.

Recap of contributions

Our Contributions

Novel Self-Supervised Task:	• Provides a strong surrogate supervisory signal for main task of classification.
End-to-End Multi-tasking:	• Sharing latent representations and high-order interactions automatically.
Experimental Analysis:	• Ablation study covering transfer learning and forecasting components.

References

[1] Wang, Zhiguang, Weizhong Yan, and Tim Oates. "Time series classification from scratch with deep neural networks: A strong baseline." 2017 International joint conference on neural networks (IJCNN). IEEE, 2017.

[17] Wang, Haishuai, et al. "Time series feature learning with labeled and unlabeled data." Pattern Recognition 89 (2019): 55-66.