Temporal Feature Selection on Networked Time Series

Haishuai Wang, Jia Wu, Peng Zhang, Chengqi Zhang
Centre for Quantum Computing & Intelligent Systems, University of Technology Sydney
haishuai.wang@student.uts.edu.au; {jia.wu, peng.zhang, chengqi.zhang}@uts.edu.au

Abstract

This paper formulates the problem of learning discriminative features (i.e., segments) from networked time series data considering the linked information among time series. For example, social network users are considered to be social sensors that continuously generate social signals (tweets) represented as a time series. The discriminative segments are often referred to as shapelets in a time series. Extracting shapelets for time series classification has been widely studied. However, existing works on shapelet selection assume that the time series are independent and identically distributed (i.i.d.). This assumption restricts their applications to social networked time series analysis, since a user’s actions can be correlated to his/her social affiliations. In this paper we propose a new Network Regularized Least Squares (NetRLS) feature selection model that combines typical time series data and user network data for analysis. Experiments on real-world networked time series Twitter and DBLP data demonstrate the performance of the proposed method. NetRLS performs better than LTS, the state-of-the-art time series feature selection approach, on real-world data.

Introduction

Shapelets are discriminative segments of time series that best predict class labels (Ye and Keogh 2011). We have also observed a line of work on extracting accurate and interpretable shapelets for time series classification. Examples include: the decision tree-based shapelet extraction (Ye and Keogh 2011; Rakthanmanon and Keogh 2013), regression-based shapelet learning (Grabocka et al. 2014a), and time series transformation methods (Lines et al. 2012). On the other hand, a recent work (Grabocka et al. 2014b) proposes a new approach to time series shapelet learning. Instead of searching for shapelets in a candidate pool, they used regression learning with the aim of learning shapelets from time series. This way, shapelets are detached from candidate segments and the learnt shapelets may differ from all the candidate segments. More importantly, shapelet learning has very fast runtimes, is scalable to large datasets, and is robust to noise.

However, state-of-the-art methods for shapelet discovery assume the time series to be independent and identically distributed (i.i.d.) and hence traditional classifiers are applicable. In emerging applications such as social networks, time series data generated by social users no longer follow the i.i.d. assumption. For example, each node in Fig. 1 is a social sensor (Sakaki, Okazaki, and Matsuo 2013) that generates social signals (tweets). Because the number of tweets generated by social nodes in the same community are highly correlated, the i.i.d. assumption is violated and new models are required to incorporate network information for time series analysis.

In this paper, we propose a Network Regularized Least Square feature selection method (NetRLS in short) to incorporate network information for shapelet selection. We test the model on a real-world Twitter data set to discover shapelets of social robots engaging in viral marking in Twitter. We also test the model on another real-world DBLP data set to classify influential authors in the data mining area. The results demonstrate the advantages of the model.

Related Work

In this section, we give a brief review of the state-of-the-art classification methods for feature learning in time series, followed by a review of feature selection approaches in networks.

Feature based Time Series Classification

Discriminative features for temporal data classification have been studied extensively (Wang et al. 2016a; 2016b), such
as bursts (Kleinberg 2003), periods (ElFeeky, Aref, and Elmagarmid 2005), anomalies (Wei et al. 2005), motifs (Lonardi and Patel 2002), shapelets (Ye and Keogh 2009; Grabocka et al. 2014b) and discords (Yankov et al. 2007). Recently, time series shapelets attract increasing interest in data mining (McGovern et al. 2011), as shapelets are usually much shorter than the original time series which allows us only need one shapelet to classify other than the entire data set. Shapelets are first proposed by (Ye and Keogh 2009) as time-series segments that maximally predict the target variable. However, the runtime of brute-force shapelet discovery is not feasible due to the large number of candidates. Therefore, a series of speed-up techniques such as early abandoning of distance computations and entropy pruning of the information gain metric have been proposed (Ye and Keogh 2009). The work (Mueen, Keogh, and Young 2011) relies on the reuse of computations and pruning of the search space to speed up the shapelets discovery. The work (Grabocka et al. 2014b) proposed a novel method that learns near-to-optimal shapelets directly, without the need to search exhaustively among a pool of candidates extracted from time-series segments. However, all the existing time series issues fall into either univariate or multivariate problems. They ignore the structural information behind time series.

Feature Selection in Networked Data

Many supervised feature selection algorithms have been proposed to select informative features from labeled data. A commonly used criterion in feature selection is to score the features. State-of-the-art methods to score the features are filter-based, wrapper-based, and embedded approach (Wang et al. 2015). The embedded methods combine feature selection with the classifier, and are often considered as more effective than the first two methods (Guyon and Elisseeff 2003). However, traditional feature selection approaches assume that the data are independent and identically distributed, which are not suited for networked data. Based on the embedded method and graph regularization, Laplacian Regularized Least Squares (LapRLS) (Belkin, Niyogi, and Sindhwani 2006) is proposed for networked data, (Gu and Han 2011) combines linear regression and graph regularization to select features in networked data and outperforms traditional feature selection methods from networked method.

However, None of the aforementioned works can directly address the networked time series classification problem studied in this work.

Preliminaries and Problem Definition

In our problem setting, there are two types of data: a network $G = (V, E)$ where nodes $|V| = n$ and edges $|E| = l$, and a set of time series data denoted by a matrix $X \in \mathbb{R}^{q \times n}$ where the $j$-th column vector $x_j = [x_j^{(1)}, x_j^{(2)}, \cdots, x_j^{(q)}] \in \mathbb{R}^q$ represents time series generated by node $v_j \in V$. There are a total of $c$ class labels denoted by a label matrix $Y \in \{0, 1\}^{c \times n}$ where each row $y_j \in \mathbb{R}^c$ is a unit vector denoting the label of node $v_j$ and $x_j \in X$.

Time series segments. Consider a sliding window of length $t$, when the window slides along a time series, a set of segments can be obtained. For time series $x_j \in X$, we can generate a total of $q - t + 1$ segments by sliding the window from $x_j^{(1)}$ to $x_j^{(q-t+1)}$. Thus, for the entire time series $X$, there are totally $(q - t + 1) \times n$ segments, i.e., $\Omega = [\varphi_1, \cdots, \varphi_{(q-t+1) \times n}]$ where each $\varphi_i \in \Omega$ denotes a segment. Each element $s_{i,k}$ is the distance between time series $x_j$ and segment $\varphi_k$. It can be defined as the differential minimum distance that approximately denotes the minimum distance between the time series and the segment. Note that the segment length $t \ll q$, the number of segments $(q - t + 1) \times n$ is very large.

Shapelets. Shapelets are defined as the most discriminative time series segments (Ye and Keogh 2011). Therefore, time series segments are shapelet candidates, and we can use $\Omega$ as the feature space for shapelets selection. To represent each time series $x_j \in X$ in the space $\Omega$, we use a column vector $s_j = [s_{1,j}, \cdots, s_{(q-t+1) \times n,j}]$ to record $x_j$’s feature values, where each element $s_{i,j}$ depends on a distance function between $x_j$ and segment $\varphi_i \in \Omega$, i.e., $s_{i,j} = d(x_j, \varphi_i)$ (We will discuss this distance function later). This way, the time series data set $X$ can be represented by a data matrix $S = [s_1, s_2, \cdots, s_n] \in \mathbb{R}^{(q-t+1) \times n}$, where each column vector $s_j$ represents a time series $x_j$ in space $\Omega$. Note that each $s_j$ is a ultra-high dimensional vector.

Goal. The purpose is to select the most discriminative segments as shapelets. Consider a multi-class problem with $c$ class labels, denote a mapping matrix $W \in \mathbb{R}^{q \times c}$ where the $j$-th column stores the class label $w_j$ that identifies the $j$-th class from the remaining $c - 1$ classes. We expect to obtain a sparse matrix $W$ with only a few non-zero row vectors by minimizing the $L_{2,0}$-norm $\|W\|_{2,0}$. The $L_{2,0}$-norm of $W$ is defined as $\|W\|_{2,0} = \text{card}(\|w_1\|_2, \cdots, \|w_c\|_2)$. $w_j$ shrinks to zero if the $j$-th feature is not discriminative. Therefore, the features corresponding to zero column of $W$ will be discarded when performing feature selection. This way, a few segments (row vectors in $W$) are selected as the shapelets for building the classifiers.

Network Regularized Least Squares Shapelets Learning

Network regularization. Network information can help identify the classifiers $W$. The idea behind is to use network regularization under the rule that, if two nodes are linked together, then they are likely to share the same class label. Technically, consider an undirected network with the adjacent matrix $A \in \mathbb{R}^{n \times n}$ derived from the edge set $E$, the network regularization term $R_G(W)$ can be formulated as:

$$R_G(W) = \frac{1}{2} \sum_{k=1}^{c} \sum_{i,j} (w_k^T s_i - w_k^T s_j)^2 A_{ij}$$

$$= \sum_{k=1}^{c} \sum_{i} w_k^T s_i D_i s_i^T w_k - \sum_{k=1}^{c} \sum_{i,j} w_k^T s_i A_{ij} s_j^T w_k$$

$$= \sum_{k=1}^{c} w_k^T S(D - W) S^T w_k = tr(W^T S L S L^T W),$$

where $L = D - A$ is an undirected graph Laplacian (Boyd 2006) and $D$ is a diagonal matrix called degree matrix with
\( D_{kl}^0 = \sum_j A_{ij}, \) Eq. (1) can be easily extended to a directed network by replacing the undirected graph laplacian with a directed graph laplacian \( L = \Pi - \frac{1}{2}(\Pi P + B^T \Pi), \) where \( \Pi \) is a diagonal matrix and \( P \) is the transition matrix of random walk on the directed network (Costello 2005).

Shapetlets selection. We use the embed-based feature selection (Gu and Han 2011). Specifically, we propose to use a network regularized least squares learning (NetRLS for short) for shapetlets selection. The NetRLS aims to learn \( c \) linear classifiers and select the top-\( k \) most discriminative shapetlets as shown in Eq. (2).

\[
\min_W \|Y - W^T S\|_F^2 + \alpha \|W\|_F^2 + \beta \text{tr}(W^T SLS^T W)
\]

s.t. \( \|W\|_{2,1} \leq \lambda, \alpha, \beta > 0. \) (2)

The first two terms in the objective function are the regularized least squares and the third term is the network regularization. The constraint \( \|W\|_{2,1} \leq \lambda \) is a relaxation of the \( L_{2,0} \) norm \( \|W\|_{2,0} \leq \lambda \) and \( \|W\|_{2,1} \) is defined as the sum of the \( l_2 \) norm of all the column vectors \( w_j \in W, \) i.e., \( \|W\|_{2,1} = \sum_j \|w_j\|_{2,1} \). \( \|W\|_{2,1} \leq \lambda \) guarantees that at most \( \lambda \) rows in \( W \) are selected.

Challenges. The matrix \( S \) is intimidatingly large because the segment space \( \Omega \) is ultra-high. Therefore, Eq. (2) cannot be solved directly on \( S. \) In the sequel, we propose to trim matrix \( S \) by using the correlation of segments.

Keogh et al. (Ding et al. 2008) conducted an experimental comparison of time series representations and distance measures, where they compared eight representation methods and nine similarity measures and their variants and testing their performance on 38 time series data sets. They claimed that the Euclidean distance is surprising competitive with other more complex approaches, although it is very sensitive to misalignments. Because the focus of this paper is not introducing new representation/distance methods, we simply use the Euclidean distance to measure the similarity between segments. Note our method can be extended to other representation/distance methods discussed in their work (Ding et al. 2008).

Define a diagonal matrix \( M = \text{diag}(0, 1, \cdots, 1, 0), \) where \( \text{rank}(M) = \rho \ll (q - t + 1) \times n \) which means only \( \rho \) element one in \( M, \) and the problem turns to calculating \( M. \)

Based on the Euclidean distance, we define a distance matrix \( \bar{U} \) as in Eq. (3). Note that the matrix is symmetric and non-negative.

\[
\bar{U} = \begin{pmatrix}
  d(\varphi_1, \varphi_1) & \cdots & d(\varphi_1, \varphi_{(q-t+1)n}) \\
  d(\varphi_2, \varphi_1) & \cdots & d(\varphi_2, \varphi_{(q-t+1)n}) \\
  \vdots & \ddots & \vdots \\
  d(\varphi_{(q-t+1)n}, \varphi_1) & \cdots & d(\varphi_{(q-t+1)n}, \varphi_{(q-t+1)n})
\end{pmatrix}
\]

(3)

Based on the matrix, we define a diagonal matrix \( \bar{P} = \bar{D} - \bar{U}, \) where \( D_h = \sum_j U_{ij}. \) Then, selecting the optimal \( S \) is equivalent to selecting the maximum triangle elements, i.e.,

\[
\bar{M} = \arg \max_M \text{tr}(\bar{P} M)
\]

s.t. \( \text{rank}(M) = \rho. \) (4)

Then, we can obtain a lower space segment space \( \bar{S} \) based on \( S, \) and \( \bar{W} \) based on \( W, \) as shown in Eq. (5).

\[
\bar{S} = \bar{SM} \in \mathbb{R}^{\rho \times n}, \quad \bar{W} = \bar{WM}.
\]

(5)

Eq. (4) aims to finding the top-\( \rho \) segments \( \bar{S} \) from all the segment candidates \( S. \) The constraints denotes only \( \rho \) segments from \( \bar{P} \) are selected. The objective function denotes we want to obtain which segments have the maximal distances to all the other segments. Eq. (4) is easily to solve, because it is equivalent to select the maximal values on the diagonal matrix \( \bar{P}. \)

Convexity. In Eq. (5), we have reduced the high dimension \( W \) and \( S \) into low dimension space \( \bar{W} \) and \( \bar{S}. \) Once we replace \( S \) with \( \bar{S} \) in Eq. (2), we want to show if the problem is convex. If so, then we can use gradient-based algorithms as the solution.

Theorem 1. The problem in Eq. (2) is convex w.r.t. \( \bar{W} \) and gradient-based algorithms can achieve a global optimum.

Proof. Due to \( \|\bar{W}\|_F^2 = \text{tr}(\bar{W}^T \bar{W}), \) Eq. (2) can be converted to the following optimization problem with parameters \( \alpha, \beta > 0, \)

\[
\min_{\|W\|_{2,1} \leq \lambda} \text{tr}(\bar{W}^T [XX^T + \alpha I + \beta \bar{S} \bar{S}^T] \bar{W} - 2 \text{tr}(\bar{S}^T Y \bar{W}^T))
\]

(6)

Let \( A = \bar{S} \bar{S}^T + \alpha I + \beta \bar{S} \bar{S}^T. \) Obviously, \( A \) is always non-negative and thus it is a positive semi-definite matrix. The constraint \( \|\bar{W}\|_{2,1} < \lambda \) is also convex. Therefore, the optimization problem is convex. \( \square \)

Algorithm. We use a recently proposed gradient-based algorithm, the Accelerated Proximal Gradient Descent (APG) algorithm (Gu and Han 2011), as the solution. The convergence rate of APG is very fast of \( O(\frac{1}{k^2}). \) Because Eq. (2) is convex, APG can achieve a global optimum.

Recall that the purpose of APG is to find a sequence of variables \( \{\cdots, \bar{W}_{k+1}, \cdots\} \) such that the objective function converges to a global minimum. Eq. (2) can be relaxed to \( F(\bar{W}) = f(\bar{W}) + \lambda \|\bar{W}\|_{2,1}, \) where \( f(\bar{W}) \) is the objective function in Eq. (2) and \( \lambda \|\bar{W}\|_{2,1} \) is a relaxation of the constraint. According to the Taylor series expansion, \( F(\bar{W}) \) approximately equals to \( G_{\theta_k}(\bar{W}, \bar{W}_k) \) as follows,

\[
G_{\theta_k}(\bar{W}, \bar{W}_k) = f(\bar{W}_k) + \langle \nabla f(\bar{W}_k), \bar{W} - \bar{W}_k \rangle + \frac{\theta_k}{2} \|\bar{W} - \bar{W}_k\|^2 + \lambda \|\bar{W}\|_{2,1}
\]

(7)

where \( \nabla f(\bar{W}_k) \) is the first order derivative of \( f(\bar{W}) \) at \( \bar{W}_k. \)

Now, the iterative step \( \bar{W}_{k+1} \) can be obtained by minimizing \( G_{\theta_k}(\bar{W}, \bar{W}_k), \) i.e.,

\[
\bar{W}_{k+1} = \arg \min_{\bar{W}} G_{\theta_k}(\bar{W}, \bar{W}_k)
\]

\[
= \arg \min_{\bar{W}} \left\{ \frac{1}{2} \|\bar{W} - \bar{V}_k\|^2 + \lambda \|\bar{W}\|_{2,1} \right\}
\]

(8)
Algorithm 1: NetRLS for Shapelets Selection on Networked Time Series.

Input: Time series $X \in \mathbb{R}^{n \times n}$, Network $G = (V, E)$, window length $t$, # of classes $c$, # of segments $\eta$, # of shapelets $p$

Output: Shapelets $S^\ast$

Initialize $\alpha, \beta, \theta_1, \tilde{W}_1, \gamma_1 = 1$

Generate a segment space $\Omega = \{q_1, \cdots, q_{(q-1)\times n}\}$

Generate an Euclidean distance matrix $\tilde{U}$ based on $\Omega$.

Generate a diagonal matrix $\tilde{P}$ based on $\tilde{U}$.

Generate a selection matrix $M = diag(0, \ldots, 1), \text{rank}(M) = \rho$.

Solve $\tilde{M} = \arg\max_M \text{tr}(\tilde{P}M) \ s.t.: \text{rank}(M) = \rho$.

Generate the candidate shapelet matrix $\tilde{S} = S\tilde{M}$.

Prune matrices in Eq. (2), $L = LM, \tilde{W} = W\tilde{M}$:

repeat

while $F(\tilde{W}_k) > G_{\theta_{k-1}}(\tilde{W}_{k+1}, \tilde{W}_k)$ do

Set $\theta_k = \gamma_{k-1}$.

$\tilde{W}_{k+1} = \arg\min_{\tilde{W}} G_{\theta_k}(\tilde{W}, \tilde{J}_k)$;

$\gamma_{k+1} = \frac{1 + \sqrt{1 + 4\gamma_k^2}}{2}$;

$J_{k+1} = W_k + \gamma_k\tilde{W}_{k+1} - \tilde{W}_k$;

until Convergence;

Score(i) = $\sqrt{\sum_i W_{i,j}^2}$;

Output: Segments $\tilde{S}_k$ with the largest Scores;

\[ w_{k+1}^i = \begin{cases} (1 - \lambda \eta_i\|v_i\|)v_k^i, & \text{if } \|v_k^i\| > \frac{1}{\eta_i} \\ 0, & \text{otherwise.} \end{cases} \quad (9) \]

Moreover, we construct a linear combination of $\tilde{W}_k$ and $\tilde{W}_{k+1}$ to update $J_{k+1}$ as follows,

$\tilde{J}_{k+1} = \tilde{W}_k + (\gamma_k - 1)(\tilde{W}_{k+1} - \tilde{W}_k)(\gamma_{k+1})$, \quad (10)

where the sequence of $\gamma_k$ is conventionally set to be $\gamma_{k+1} = \frac{1 + \sqrt{1 + 4\gamma_k^2}}{2}$. The algorithm is summarized in Algorithm 1.

Experiments

These experiments are designed to validate whether the proposed NetRLS model, which combines both time series data and network data, can obtain better performance than using only time series data. All the experiments were conducted on a Linux Ubuntu server with 16*2.9GHZ CPU and 64G memory. The experiments were implemented in Matlab.

Data.

We collected a Twitter data set and a DBLP data set for testing to show the performance of the proposed model on real-world data.

Twitter: The task is to detect social robots that auto-distribute advertisements for viral marketing on social networks. We located and collected 200 social time series in the last 30 days from three types of nodes: 1) Social robots which are zombies controlled by a master node, that occasionally distribute spam over the network. Because we had already located the master nodes, we could infer the network link among these zombies nodes. 2) Active social users who are famous/VIP users. These users are very active and work link among these zombies nodes. 3) Ordinary social users who barely post messages, their links are sparest.

DBLP: The task is to classify if an author is influential in a given research field. We retrieved around 700 authors from DBLP \footnote{http://dblp.uni-trier.de/db/} in the data mining area, i.e., authors in the ICDM.
PKDD/ECML and KDD conferences. There were 300 influential authors (e.g., Jiawei Han, Philip S. Yu, etc.) in the data set, with the remaining authors considered ‘normal’ in the data mining area. Note that authors considered to be normal do not necessarily represent authors that only publish a small number of papers each year or a small number of papers in particular data mining conferences. We labelled the influential and normal authors based on prior knowledge. Some influential authors are already well-known researchers, and their frequent co-authors also publish in the top-five data mining conferences. Some authors share surnames, denoted as noise authors. We manually removed the noise and mislabelled authors. We crawled the number of papers published each year to form time series data for each author between 1996 and 2015. Then, based on co-authorship, we constructed network information. For example, an edge exists if two authors have a co-authorship relationship and the weight is set to 1, otherwise it is set to 0.

Baseline Methods. To show the power of network information in building a classifier, we compare the proposed NetRLS model with a regularized least squares (RLS) model which only uses time series data for classification. We also compare NetRLS with the state-of-the-art shapelet learning method, LTS (Grabocka et al. 2014a). LTS’ performance has been demonstrated as better than other state-of-the-art methods (Grabocka et al. 2014a), e.g., the Fast Shapelets (FSH) which is a fast random projection technique on SAX representation (Ding et al. 2008), and the Dynamic Time Warping (DTW) (Rakthanmanon and Keogh 2013).

Measures. We measure the performance by classification accuracy and AUC. We use 70% for training and the remaining 30% for testing.

Parameter testing. Fig. 3(a) shows the parameter tests with respect to α, β, λ on the Twitter data set. The parameters are α = 1, β = 1, λ = 1 by default. We can observe that when the parameter β is set to a small value of 0.1, the model obtains the worst result, this is because the network information is nearly dropped in the analysis. The sparse terms ||W||^2 and ||W||^2,1 obtain the best results when the parameters λ and β equal to 10. Thus, we will set the parameters α = 10, β = 10, λ = 10.

Comparisons. Fig. 3(b) shows the improvement of NetRLS over RLS on Twitter data set. We can observe that NetRLS steadily lays above RLS. The accuracy gap is 3% on average. The accuracy gap reflects the power of the network data. This is because social time series contain more noise than traditional time series data and network data is useful in improving performance.

Table 1: Accuracy comparison on Twitter data set w.r.t. various window length and parameter ρ.

| Methods | ρ  | t=3 | t=4 | t=5 | t=6 |
|---------|----|-----|-----|-----|-----|
| NetRLS  | 10 | 0.725 \pm 0.007 | 0.745 \pm 0.009 | 0.705 \pm 0.007 | | |
| RLS     | 10 | 0.706 \pm 0.007 | 0.725 \pm 0.007 | 0.705 \pm 0.007 | | |
| LTS     | 10 | 0.714 \pm 0.007 | 0.725 \pm 0.007 | 0.705 \pm 0.007 | | |

Table 2: Accuracy comparison on DBLP data set w.r.t. various window length and parameter ρ.

| Methods | ρ  | t=2 | t=3 | t=4 | t=5 |
|---------|----|-----|-----|-----|-----|
| NetRLS  | 10 | 0.710 \pm 0.004 | 0.749 \pm 0.011 | 0.791 \pm 0.003 | 0.780 \pm 0.007 | |
| RLS     | 10 | 0.712 \pm 0.015 | 0.749 \pm 0.017 | 0.712 \pm 0.027 | 0.702 \pm 0.025 | |
| LTS     | 10 | 0.714 \pm 0.015 | 0.749 \pm 0.011 | 0.714 \pm 0.027 | 0.702 \pm 0.025 | |

From the results, we can conclude that: 1) given a shapelet space ρ, increasing the window length t will not guarantee better prediction results. When the length is 5, both NetRLS and RLS obtain relatively better results. 2) given a window length t, increasing ρ will generally improve the performance, but the improvement is insignificant when ρ > 40. For example, on the Twitter data set, NetRLS is 0.768 when ρ = 30 and t = 4, which is the same as ρ = 50. Considering that increasing ρ will generate more candidate segments and increase memory and computation costs, ρ can be set to between 30 and 50. 3) given the same ρ and t, NetRLS obtains higher accuracy and AUC value than RLS because NetRLS can model both time series data and network data and obtain more accurate and robust results. 4) NetRLS performs better than LTS. LTS shows high accuracy on the UCR time series data sets but produces a different result in social network data sets (Twitter and DBLP). This is because there are undistinguishable users and noise when based solely on the time series data. For example, in the Twitter data set, a famous user may have only posted a few tweets during the date ranges fetched in our data set. This could easily be categorized to normal user if the number of tweets was the only consideration. In the DBLP data set, an author may only fo-
on the top conferences in data mining, and contribute a lot to them, but the total number of papers published per year may limited. Without additional data, this might indicate classification as a non-influential author.

Fig. 6 lists the shapelets learnt by NetRLS on the Twitter and DBLP data set. Fig. 6 (a) shows that the social robots demonstrate sharp fluctuant shapelets, since they usually post a mass of tweets in a short period of time, while normal users tend to post tweets represented by regular fluctuant shapelets. VIP users post tweets with gently fluctuant shapelet because they regularly post many tweets. Fig. 6 (b) demonstrates normal authors have subtle changing shapelets because the papers published by normal authors do not significantly change each year, while influential authors show great changes due to co-authors or new topics.

Discussions. In this part, we discuss two problems regarding the selection of shapelets from networked time series.

First, one may ask if the network information can alter the shapelets? In Eq. (2), the optimization problem has only one variable, the classifier weight $W$, so the network information will change only the classifier boundaries but not shapelets themselves. In fact, our solution can be taken a two-step hierarchical way that first solve Eq. (4) to get a trimmed concise segment space and then solve Eq. (2) to obtain shapelets. In our experiments, the network links are relatively sparse². This is because social time series data often contain heavy noise and social nodes label are usually incorrect, network link information can somewhat alleviate the noise and mislabel problems, and thus improve the classification accuracy.

Second, in what case the link information can alter the shapelets themselves? We can relax Eq. (2) to be a more flexible problem that optimize the objective function between variables $W$ and $S$. This way, the third item of network regularization $L$ will impact the optimal value of $S$. However, similar to the work of learning shapelets (Grabocka et al. 2014a), if we allow to optimize w.r.t. $S$, we will arrive higher accuracy at cost of obtaining shapelets that slightly compare all the segments derived from time series data.

Figure 4: AUC comparison on Twitter data set w.r.t. various window length and parameter $\rho$.

Figure 5: AUC comparison on DBLP data set w.r.t. various window length and parameter $\rho$.

Figure 6: An illustration of the shapelets learned by NetRLS on the Twitter and DBLP datasets.

Conclusion

In this paper, we have explored a new problem of networked time series classification, where the data contain both the typical time series data and network structure data. A network regularized least square feature selection method (NetRLS) is proposed to incorporate the network structure information for shapelet selection, accordingly. Our work drops the independent and identically distributed (i.i.d.) assumption and enables to use rich network structure information to improve the performance. The experiments and comparisons on real-world Twitter and DBLP data sets demonstrate that NetRLS outperforms state-of-the-art time series shapelet learning algorithms and is suitable for a wide range of learning tasks. Due to space, we only used the Euclidean distance as the measure. In the future, we will report more comparisons with other shapelets selection models based on advanced time series representation and distance measures.

²Fig. 2 shows the 200 nodes and 210 edges. Because we have located all the social robots, so all the links between social robots are captured. On the other hand, the edges between real users, including VIP users and ordinary users, are relatively sparse. Even though the network is sparse, we can still observe performance improvement as shown in Fig. 3(b).
References
Belkin, M.; Niyogi, P.; and Sindhwani, V. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. The Journal of Machine Learning Research 7:2399–2434.
Boyd, S. 2006. Convex optimization of graph laplacian eigenvalues. In Proceedings of the International Congress of Mathematicians, volume 3, 1311–1319.
Costello, K. 2005. Random walks on directed graphs.
Ding, H.; Trajcevski, G.; Scheuermann, P.; Wang, X.; and Keogh, E. 2008. Querying and mining of time series data: experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment 1(2):1542–1552.
Elfeky, M. G.; Aref, W. G.; and Elmagarmid, A. K. 2005. Periodicity detection in time series databases. Knowledge and Data Engineering, IEEE Transactions on 17(7):875–887.
Grabocka, J.; Schilling, N.; Wistuba, M.; and Schmidt-Thieme, L. 2014a. Learning time-series shapelets. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 392–401. ACM.
Grabocka, J.; Schilling, N.; Wistuba, M.; and Schmidt-Thieme, L. 2014b. Learning time-series shapelets. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 392–401.
Gu, Q., and Han, J. 2011. Towards feature selection in network. In Proceedings of the 20th ACM international conference on Information and knowledge management, 1175–1184. ACM.
Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. The Journal of Machine Learning Research 3:1157–1182.
Kleinberg, J. 2003. Bursty and hierarchical structure in streams. Data Mining and Knowledge Discovery 7(4):373–397.
Lines, J.; Davis, L. M.; Hills, J.; and Bagnall, A. 2012. A shapelet transform for time series classification. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 289–297. ACM.
Lonardi, J. L. E. K. S., and Patel, P. 2002. Finding motifs in time series. In Proc. of the 2nd Workshop on Temporal Data Mining, 53–68.
McGovern, A.; Rosendahl, D. H.; Brown, R. A.; and Droege, M. 2011. Identifying predictive multidimensional time series motifs: an application to severe weather prediction. Data Mining and Knowledge Discovery 22(1-2):232–258.
Mueen, A.; Keogh, E.; and Young, N. 2011. Logical-shapelets: an expressive primitive for time series classification. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 1154–1162. ACM.
Rakthanmanon, T., and Keogh, E. 2013. Fast shapelets: A scalable algorithm for discovering time series shapelets.
In Proceedings of the thirteenth SIAM conference on data mining (SDM), 668–676. SIAM.
Sakaki, T.; Okazaki, M.; and Matsuo, Y. 2013. Tweet analysis for real-time event detection and earthquake reporting system development. Knowledge and Data Engineering, IEEE Transactions on 25(4):919–931.
Wang, H.; Zhang, P.; Tsang, I.; Chen, L.; and Zhang, C. 2015. Defragging subgraph features for graph classification. In Proceedings of the 24th ACM international on conference on information and knowledge management, 1687–1690. ACM.
Wang, H.; Wu, J.; Zhu, X.; and Zhang, C. 2016a. Time-variant graph classification. arXiv preprint arXiv:1609.04350.
Wang, H.; Zhang, P.; Zhu, X.; Tsang, I. W.-H.; Chen, L.; Zhang, C., and Wu, X. 2016b. Incremental subgraph feature selection for graph classification. IEEE Transactions on Knowledge and Data Engineering 29(1):128–142.
Wei, L.; Kumar, N.; Lolla, V. N.; Keogh, E. J.; Lonardi, S.; and Ratanamahatana, C. A. 2005. Assumption-free anomaly detection in time series. In SSDBM, volume 5, 237–242.
Yankov, D.; Keogh, E.; Medina, J.; Chiu, B.; and Zordan, V. 2007. Detecting time series motifs under uniform scaling. In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, 844–853. ACM.
Ye, L., and Keogh, E. 2009. Time series shapelets: A new primitive for data mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 947–956.
Ye, L., and Keogh, E. 2011. Time series shapelets: a novel technique that allows accurate, interpretable and fast classification. Data mining and knowledge discovery 22(1-2):149–182.