Learning Graph Quantization

Jain B.J., Srinivasan S.D., Tissen A., Obermayer K.

18 Aug 2010

FG Neuronale Informationsverarbeitung, Technische Universitaet Berlin

Setting the stage

- Attributed Graphs- commonly used in Structural Pattern Recognition to represent data
  - Potentially very powerful means to represent data with high descriptive power
  - Constrained in applications by lack of methods for ML tasks like classification and clustering
- Classification/Clustering of patterns represented by Attributed Graphs is a fundamental problem in Structural Pattern Recognition
- Embed the graphs into a *feature*(vector) space and to carry out the Classification/Clustering taks there
- Loss of structural information when embedding

Structure Spaces- A quick review

- Isomorphic and isometric embedding of combinatorial structures into a vector space
- An *Attributed Graph* $X = (V, E, \alpha)$ given by its *adjacency matrix* $\mathbf{X}$
  - V *vertices*, $E \subseteq V \times V$ *edges, attribute function* $\alpha : V \to \mathcal{D}^V; E \to \mathcal{D}^E$
- $\mathcal{X} = \mathbb{E}^{n \times n}$ be the space of all $(n \times n)$-matrices, $\mathcal{T}$ denote a subgroup of all $(n \times n)$-permutation matrices
- Two adjacency matrices are equivalent if $\mathbf{P}^{\top}\mathbf{X}\mathbf{P} = \mathbf{X}'$
- The equivalent class of structures partition the underlying space into quotient sets, projecting every structure to its abstract representation

Structure Spaces- A quick review

- Consider the graph distance measure,
  $d(X, Y) = \min\left\{\|\mathbf{x} - \mathbf{y}\|^2, \mathbf{x} \in X, \mathbf{y} \in Y\right\}$
    - This is indeed a metric
- Consider the generalized inner product,
  $< X, Y >= \max\{< \mathbf{x}, \mathbf{y} >, \mathbf{x} \in X, \mathbf{y} \in Y\}$
    - This behaves much like inner product, measures structural similarity between graphs
- Compute these quantities by fixing representation of one graph and maximizing/minimizing over all possible representations of the other
- Price to pay: computational complexity

Learning Graph Quantization

- An algorithm to construct classifier for set of Attributed Graphs
- A set of prototypes are chosen such that the patterns are classified by means of Nearest prototype (NPC)
- Advantages
  - Easy extension to Multi-Class problem
  - Prototypes act as typical class representatives that could be used for further applications

Learning Graph Quantization

- Classifier $c : \mathcal{X}_\mathcal{T} \rightarrow \mathcal{C}$ that maps graphs from $\mathcal{X}_\mathcal{T}$ to class labels from a finite set $\mathcal{C}$
- k prototypes $W_1, \ldots, W_k \in \mathcal{X}_\mathcal{T}$ with class labels $c_1, \ldots, c_k \in \mathcal{C}$
- Objective- The prototypes should best predict the class labels of graphs from $\mathcal{X}_\mathcal{T}$
- Prototypes should be far away from the decision surface
- This is done by pulling the prototypes from the incorrect labelled data and pushing them towards the correct labelled data

Learning Graph Quantization

Algorithm 1 (LGQ1)

Given:

training set $\mathcal{S} = \{(X_1, l(X_1)), \dots, (X_n, l(X_n))\} \subseteq \mathcal{X}_\mathcal{T} \times \mathcal{C}$

Procedure:

1. choose initial prototypes $\mathcal{W} = \{(W_1, l(W_1)), \dots, (W_k, l(W_k))\} \subseteq \mathcal{X}_\mathcal{T} \times \mathcal{C}$
2. do until convergence or learning rate vanishes
   - 2.1. randomly select a training example $(X, y) \in \mathcal{S}$
   - 2.2. let $W_X = \arg\min_{W \in \mathcal{W}} d(X, W)$
   - 2.3 Align X with W_X
   - 2.4. With learning rate $\eta > 0$
   - 2.5. update according to the rule

$$\mathbf{w_x} \leftarrow \begin{cases} \mathbf{w_x} + \eta\,(\mathbf{x} - \mathbf{w_x}) & \text{if } l(X) = l(W_X) \\ \mathbf{w_x} - \eta\,(\mathbf{x} - \mathbf{w_x}) & \text{if } l(X) \neq l(W_X) \end{cases}$$

Return: set $\mathcal{W}, l(.)$ of prototypes along with class labels

IAM graph dataset repository- Letter (HIGH), GREC, Fingerprint

data set	#(graphs)	#(classes)	avg(nodes)	max(nodes)	avg(edge
letter	750, 750	15	4.7	8	3.1
grec	286, 528	22	11.5	24	11.9
fingerprint	500, 2000	3	5.4	26	4.4

Table: Summary of main characteristics of the data sets

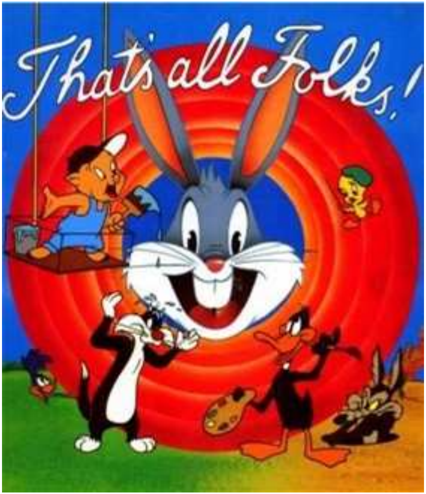
Method	Letter	Fingerprint	GREC
LGQ + NPC	80.9%	79.2%	94.7%
LGQ2 + NPC	83.7%	82.2%	97.3%
k-NN (Ref.)	90%	76.7%	95.5%
SK + SVM (SoA)	79.1%	41%	94.9%
LE + SVM (SoA)	92.5%	82.8%	96.8%
Top.Emb. (SoA)	-	-	95.8%

Distinctiveness of suggested approach

- A structure preserving representation of attributed graphs is proposed. The quotient space is the simplest example of an *orbifold*
- This leads to an inherent definition of graph metric and update rule
- Principled extension of standard machine learning algorithms to the domain of structural pattern recognition
- Prototypes could act as typical class representatives; for expert interpretation or to devise better classification strategies

Open questions and way ahead

- Computational costs
  - Any shortcuts ??- Explore the geometry of *structure spaces*
  - Faster, better graph matching algorithms ??
- Extend other LVQ algorithms to attributed graphs
- A general theory of *learning over orbifolds*
- Theory must be developed and tuned in conjunction with applications


That's all Folks!