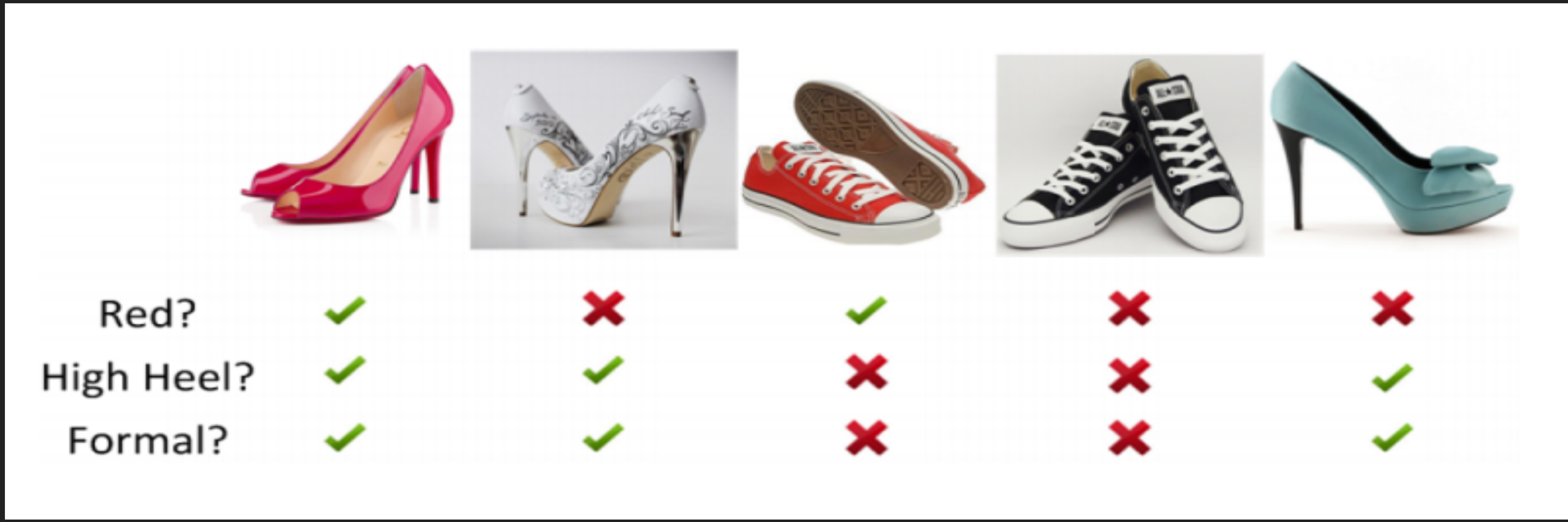
Clustering-Based Joint Feature Selection for semantic attribute prediction



PRESENTED BY: TEAM GENESIS

[AVANI RAWAT] - 20172029

[SURBHI GOYAL] - 20172023

[SAMEER SINGH] - 20172134

[SHUBHAM VERMA] - 20172035

MOTIVATION

- Need to handle High - Dimensional Data
- Representation of semantic features effectively
- Effective utilisation of relatedness among multiple attributes
- Not all low-level features have equal contribution to all the attributes
- Attributes are related in clustering structures

INTRODUCTION

- Features: Metric for property or characteristic of an entity
- Semantic Features: Give more understanding about the entity that can be used for its recognition/classification
- AIM: To learn semantic features
- However, for this learning if we use raw high-dimensional data, it will suffer from the curse of dimensionality
- Besides, not all low-level features have equal contribution to all the attributes

KEY IDEA

- Same types of entities share some common set of features
- Thus, attributes occur in clustering structures
- Methodology:
  - F = {f1,f2, . . . ,fd} - Features
  - X = {x1,x2,. . . ,xn} - Data
  - C = {c1,c2,. . .,cm} - Classes
  - Y = {y1,y2,. . .,yn} - Predicted Labels
  - s = f(0,. . .,0,1,. . .,1) - Feature Selection Vector
  - W = [w1,w2,. . .,wm] - Linear Projection Matrix for mapping X to Y

$$\min_{W,\mathbf{s}} \quad L(W^\top \mathrm{diag}(\mathbf{s})X, Y)$$
$$s.t., \ \ \mathbf{s} \in \{0,1\}^n, \ \mathbf{s}^T \mathbf{1}_n = K$$

FEATURE SELECTION VECTOR

$$s = f(0,...,0,1,...,1)$$

- 'f' is a permutation function that contains 0 and 1s
- S is the feature selection vector that is a binary vector
- It contains 1 for the kth cluster as selected
- And, 0 for the kth clustered not selected

LINEAR PROJECTION MATRIX

$$W_i = [\boldsymbol{w}_1^{(i)}, \boldsymbol{w}_2^{(i)}, \ldots, \boldsymbol{w}_{n_i}^{(i)}]$$

- Linear Projection Matrix contains the Linear Projection Vectors Wi
- Where, Wi = [wi(1),wi(2),. . .,wi(ni)]
- where, wi(1) is the weight vector for the ith cluster

STEP TOWARDS LABEL CORRELATION

- The linear projection matrix that was described, doesn't consider any correlation among their vectors
- Correlation is captured using the mean vector for ith cluster and the difference is tracked with each weight vector
- K-means is utilized to capture the correlation

MODELING LABEL CORRELATION

- Idea: Correlated attributes share the same features
- Model: Learning the clustering structures through k-means
- WE = [W1,W2,. . .,Wk]

  where, W = $[\boldsymbol{w}_1^{(i)}, \boldsymbol{w}_2^{(i)}, \ldots, \boldsymbol{w}_{n_i}^{(i)}]$

  and, E = Permutation Partition Matrix

$$\sum_{i=1}^{k} \sum_{j=1}^{n_i} \|\boldsymbol{w}_j^{(i)} - \boldsymbol{m}_i\|^2, \boldsymbol{m}_i = \sum_{j=1}^{n_i} \boldsymbol{w}_j^{(i)} / n_i \qquad (1)$$

DEDUCTION OF COST FUNCTION IN VECTORIZED FORM

$$\sum_{i=1}^{k}\sum_{j=1}^{n_i}\|\boldsymbol{w}_j^{(i)}-\boldsymbol{m}_i\|^2=\sum_{i=1}^{k}\|W_i(I_{n_i}-\frac{e_i{e_i}^\top}{n_i})\|_F^2$$

$$=\sum_{i=1}^{k}\mathbf{Tr}(W_i^\top W_i)-(\frac{{e_i}^\top}{\sqrt{n_i}})W_i^\top W_i(\frac{e_i}{\sqrt{n_i}}) \qquad (2)$$

Let $F = \mathrm{diag}(\frac{e_1}{\sqrt{n_1}}, \frac{e_2}{\sqrt{n_2}}, \ldots, \frac{e_k}{\sqrt{n_k}}) \in \mathbb{R}^{m\times k}$ be an orthonormal matrix, then Eq. (2) can be rewritten as

$$\mathbf{Tr}(W^\top W)-\mathbf{Tr}(F^\top W^\top W F)$$

mi = Mean Vector of ith-cluster

ei = transpose([1,1,. . .,1]) (ni x 1) vector

FROBENIUS NORM

- The Frobenius norm is also called the Euclidean norm
- Frobenius norm is matrix norm of an m×n matrix A defined as the square root of the sum of the absolute squares of its elements

$$\|A\|_F \equiv \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

REDUCED OPTIMIZATION PROBLEM

$$\min_{F^\top F = I_k} \mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F) + \gamma \mathbf{Tr}(W^\top W) \quad (3)$$

- Given above the optimization problem, that captures the label correlation
- The above problem needs to be solved with the feature selection model to give the overall functionality for predicting the semantic features out of the dataset

FEATURE SELECTION

$$\min_{W,F,\mathbf{s}} L(W^\top \mathrm{diag}(\mathbf{s})X, Y) + \gamma \mathbf{Tr}(W^\top W) + \beta(\mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F))$$

$$s.t. \quad F^\top F = I_k,\ \mathbf{s} \in \{0,1\}^n,\ \mathbf{s}^\top \mathbf{1}_n = K \qquad (4)$$

$$\min_{W,F;F^\top F = I_k} L(W^\top X, Y) + \alpha \sum_{i=1}^{k} \|W_i\|_{2,1} + \gamma \mathbf{Tr}(W^\top W) + \beta(\mathbf{Tr}(W^\top W) - \mathbf{Tr}(F^\top W^\top W F)) \qquad (5)$$

- As we have captured the attribute correlation from eqn 3, the above optimization problem combines it with the methodology

where,

- α = parameter to control the sparsity of W
- β = control contribution from label correlation
- γ = control generalization performance

OPTIMIZATION

- The optimization problem that was just deducted is non-convex, non-smooth
- Thus, the problem becomes untractable
- Relaxations need to be done to transfer it to convex one

$$\beta \mathbf{Tr}(W((1+\eta)I - FF^\top)W^\top)$$

$$\beta\eta(1+\eta)\mathbf{Tr}(W(\eta I + M)^{-1}W^\top)$$

$$s.t. \quad tr(M) = k,\ M \preceq I,\ M \in \mathbb{S}_+^m$$

$\eta = \gamma/\beta > 0$. Let $M = FF^\top$

ASO (ALTERNATING STRUCTURE OPTIMIZATION)

- Multi-task learning (MTL) learns multiple related tasks simultaneously to improve generalization performance
- It's the approach that is based on some common structure among various different problems
- For every problem there are 2 parts:
  - <u>Common Property</u>
  - <u>Associated Property</u>

OPTIMIZING M WHEN FIXING W

- To optimize M and W simultaneously is difficult, however using ASO it can be done easily as follows:

$$\min_{M} \mathbf{Tr}(W(\eta I + M)^{-1} W^{\top})$$
$$s.t.\ \ tr(M) = k,\ M \preceq I,\ M \in \mathbb{S}_{+}^{m}$$

- W is decomposed using SVD while M using Eigen decomposition, and putting $Q^* = V$

$$\Lambda^* = \arg\min_{\Lambda} \sum_{i=1}^{q} \frac{\sigma_i^2}{\eta + \lambda_i}$$
$$s.t.\ \ \sum_{i=1}^{q} \lambda_i = k, 0 \leq \lambda_i \leq 1$$

- [By: Zhou et. al , 2011]

OPTIMIZING W WHEN FIXING M

- L 2,1 norm is difficult to calculate
- Using dummy variables, L 2,1 is opted out by introducing dummy variables

$$\sum_{i=1}^{k}(\|W_i\|_{2,1})^2 = (\sum_{i=1}^{k}\sum_{j=1}^{d}\|\boldsymbol{w}_{i,j}\|_2)^2 \leq \sum_{i=1}^{k}\sum_{i=1}^{d}\frac{(\|\boldsymbol{w}_{i,j}\|_2)^2}{\delta_{ij}}$$

$$\sum_i \sum_j \delta_{ij} = 1$$
$$\delta_{ij} \in \mathbb{R}^{+}$$

where $\boldsymbol{w}_{i,j} \in \mathbb{R}^{1\times m}$ is the row vector of W_i. Thus δ_{ij} can be updated by holding the equality:

- Where,

$$\delta_{ij} = \|\boldsymbol{w}_{i,j}\|_2 / \sum_{j=1}^{d}\|\boldsymbol{w}_{i,j}\|_2.$$

$$\arg\min_W \|W^T X - Y\|_F^2 + \alpha \sum_{i=1}^{k}\sum_{i=1}^{d}\frac{(\|\boldsymbol{w}_{i,j}\|_2)^2}{\delta_{ij}} - \beta\eta(1+\eta)\mathbf{Tr}(W(\eta I + M)^{-1}W^{\top})$$

FEATURE SELECTION OPTIMIZATION

Algorithm 1 Feature Selection Optimization

Input:

1. Multiple attribute data $\{X, Y\}$;
2. Parameters α, β, k(optional) and the number of selected features K;
3. The initial projection matrix W_0;

Procedure:

1: Set $W = W_0$;
2: **repeat**
3: Update M according to Eq. (8);
4: Update $\boldsymbol{r}$ according to Alg. 2;
5: Update $\boldsymbol{\delta}$ according to Eq. (10);
6: Update W according to Eq. (11);
7: **until** Converges
8: Sort each feature according to $\|\mathbf{w}^i\|_2$ in descending order of each group;
9: **return** The group-wise top-K ranked features;

CLUSTER ASSIGNMENT ESTIMATION

Algorithm 2 Cluster Assignment Estimation

Input: M;

Procedure:

1: Approximate F by top-ranked eigenvector of Q;
2: Calculate R_{11}, R_{12} by applying QR decomposition with column pivoting on F by Eq. (12);
3: Calculate $\hat{R}$ by Eq. (13);
4: calculate r by Eq. (14) for each attribute;
5: **return** Cluster assignment vector $\boldsymbol{r}$;

ESTIMATING ATTRIBUTE ASSIGNMENT

- The group-wise feature selection is conducted by the clustering structure (F) of the attribute
- For reconstructing F, eigen decomposition is being used

$$F^T = [\underbrace{t_{11}\boldsymbol{v}_1, \cdots, t_{1s_1}\boldsymbol{v}_1}_{cluster1}, \cdots, \underbrace{t_{k1}\boldsymbol{v}_k, \cdots, t_{ks_1}\boldsymbol{v}_k}_{clusterk}]$$

where $V^T = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_k] \in \mathbb{R}^{k \times k}$ is an orthogonal matrix.

KEY IDEA FOR FEATURE SELECTION

- Using the clustering regularizer to partition the tasks into groups where strong correlation exists among tasks in the same group
- So, feature selection based on such group structures would make sure appropriate feature subsets are selected to represent the respective semantic attributes

Thank You