A Novel Uncertainty Sampling Algorithm for Cost-Sensitive Multiclass Active Learning

Kuan-Hao Huang[1] and Hsuan-Tien Lin[1,2]

[1]Department of Computer Science & Information Engineering
National Taiwan University

[2]Appier Inc.

ICDM, December 15, 2016

Active Learning

Active learning for multiclass classification

- labeled pool $\mathcal{D}_l = \{\text{feature} : \mathbf{x}^{(n)}, \text{label} : y^{(n)}\}_{n=1}^{N_l}$.
- unlabeled pool $\mathcal{D}_u = \{\text{feature} : \mathbf{x}^{(n)}\}_{n=1}^{N_u}$
- for round $t = 1, 2, ..., T$
  - select instance $\mathbf{x}_s \in \mathcal{D}_u$ by a **querying strategy** to get label y_s
  - move $(\mathbf{x}_s, y_s)$ from unlabeled pool $\mathcal{D}_u$ to labeled pool $\mathcal{D}_l$
  - learn a classifier $f^{(t)}$ from the current labeled pool $\mathcal{D}_l$
- improve the performance of $f^{(t)}$ with respect to #queries

Active Learning

Active learning for multiclass classification

- labeled pool $\mathcal{D}_l = \{\text{feature} : \mathbf{x}^{(n)}, \text{label} : y^{(n)}\}_{n=1}^{N_l}$.
- unlabeled pool $\mathcal{D}_u = \{\text{feature} : \mathbf{x}^{(n)}\}_{n=1}^{N_u}$
- for round $t = 1, 2, ..., T$
  - select instance $\mathbf{x}_s \in \mathcal{D}_u$ by a **querying strategy** to get label y_s
  - move $(\mathbf{x}_s, y_s)$ from unlabeled pool $\mathcal{D}_u$ to labeled pool $\mathcal{D}_l$
  - learn a classifier $f^{(t)}$ from the current labeled pool $\mathcal{D}_l$
- improve the performance of $f^{(t)}$ with respect to #queries

Querying strategies

- **uncertainty sampling** [Lewis et al., 2010; Tong et al. 2001; Jing et al., 2004]
- representative sampling [Settles et al., 2008; Huang et al., 2014; Dasgupta et al., 2008]
- error reduction [Roy et al., 2001]

Evaluation Criteria

Regular (Error rate)

	healthy	cold	Zika
healthy	0	1	1
cold	1	0	1
Zika	1	1	0

- **same** costs of errors
- most common criterion

Evaluation Criteria

Regular (Error rate)

	healthy	cold	Zika
healthy	0	1	1
cold	1	0	1
Zika	1	1	0

- **same** costs of errors
- most common criterion

Cost matrix

	healthy	cold	Zika
healthy	0	10	50
cold	200	0	100
Zika	**1000**	800	0

- **different** costs of errors
- cost matrix $\mathbf{C}_{i,j}$: predict c_i as c_j

Evaluation Criteria

Regular (Error rate)

	healthy	cold	Zika
healthy	0	1	1
cold	1	0	1
Zika	1	1	0

- **same** costs of errors
- most common criterion

Cost matrix

	healthy	cold	Zika
healthy	0	10	50
cold	200	0	100
Zika	**1000**	800	0

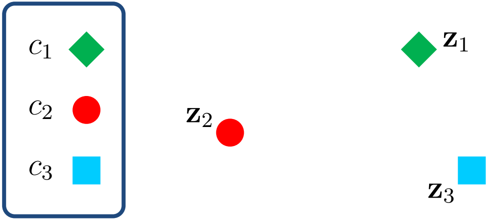
- **different** costs of errors
- cost matrix $\mathbf{C}_{i,j}$: predict c_i as c_j

Cost-sensitive active learning algorithms

- cost-sensitive multiclass classifcation takes cost matrix $\mathbf{C}$ into account
- our goal: **active learning for cost-sensitive multiclass classification**

	querying strategy	classifier f
regular algorithms	by f, $\mathcal{D}_l$, and $\mathcal{D}_u$	learned from $\mathcal{D}_l$
cost-sensitive algorithms	by f, $\mathcal{D}_l$, $\mathcal{D}_u$, and $\mathbf{C}$	learned from $\mathcal{D}_l$ and $\mathbf{C}$

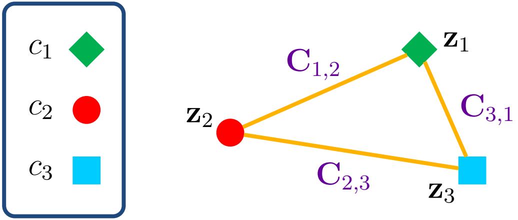
Cost Embedding (Training)



Training stage

- for classes $c_1, c_2, ..., c_K$, find K **hidden points** $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_K$

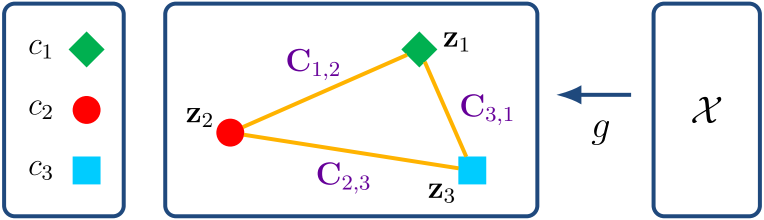
Cost Embedding (Training)



Training stage

- for classes $c_1, c_2, ..., c_K$, find K **hidden points** $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_K$
- **higher (lower) cost** $\mathrm{C}_{i,j} \Leftrightarrow$ **larger (smaller) distance** $d(\mathbf{z}_i, \mathbf{z}_j)$
- preserve the **order** of the costs in **distance**
- by **non-metric multidimensional scaling**

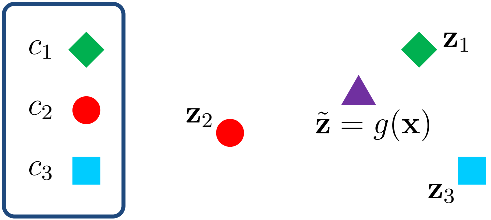
Cost Embedding (Training)



Training stage

- for classes $c_1, c_2, ..., c_K$, find K **hidden points** $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_K$
- **higher (lower) cost** $\mathrm{C}_{i,j} \Leftrightarrow$ **larger (smaller) distance** $d(\mathbf{z}_i, \mathbf{z}_j)$
- preserve the **order** of the costs in **distance**
- by **non-metric multidimensional scaling**
- learn a **regressor** g from $\{\mathbf{x}^{(n)}, \mathbf{z}^{(n)}\}_{n=1}^{N_l}$

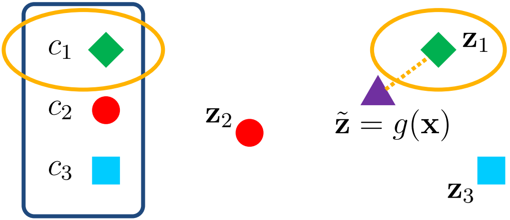
Cost Embedding (Predicting)



Predicting stage

- for a testing instance $\mathbf{x}$, get the **predicted hidden point** $\tilde{\mathbf{z}} = g(\mathbf{x})$

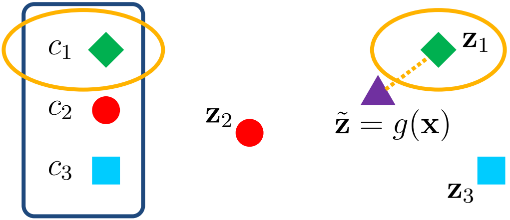
Cost Embedding (Predicting)



Predicting stage

- for a testing instance $\mathbf{x}$, get the **predicted hidden point** $\tilde{\mathbf{z}} = g(\mathbf{x})$
- find the **nearest hidden point** of $\tilde{\mathbf{z}}$ from $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_K$
- take the corresponding class as the **cost-sensitive prediction**

Cost Embedding (Predicting)

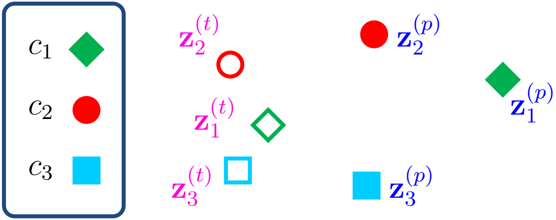


Predicting stage

- for a testing instance $\mathbf{x}$, get the **predicted hidden point** $\tilde{\mathbf{z}} = g(\mathbf{x})$
- find the **nearest hidden point** of $\tilde{\mathbf{z}}$ from $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_K$
- take the corresponding class as the **cost-sensitive prediction**

asymmetric cost ($\mathrm{C}_{i,j} \neq \mathrm{C}_{j,i}$) vs. **symmetric distance**?

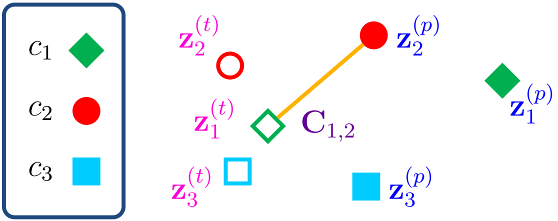
Mirroring Trick



Two roles of class

- two roles of class c_i: ground truth role $\mathbf{z}_i^{(t)}$ and prediction role $\mathbf{z}_i^{(p)}$

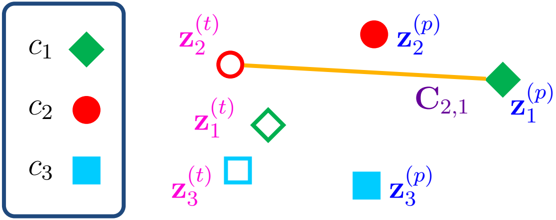
Mirroring Trick



Two roles of class

- two roles of class c_i: ground truth role $\mathbf{z}_i^{(t)}$ and prediction role $\mathbf{z}_i^{(p)}$
- $\mathbf{C}_{i,j} \Rightarrow c_i$ is ground truth and c_j is prediction $\Rightarrow$ for $\mathbf{z}_i^{(t)}$ and $\mathbf{z}_j^{(p)}$

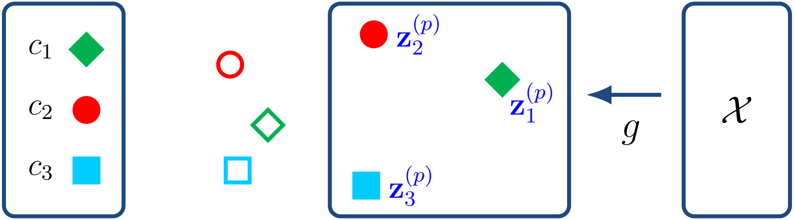
Mirroring Trick



Two roles of class

- two roles of class c_i: ground truth role $\mathbf{z}_i^{(t)}$ and prediction role $\mathbf{z}_i^{(p)}$
- $\mathbf{C}_{i,j} \Rightarrow c_i$ is ground truth and c_j is prediction $\Rightarrow$ for $\mathbf{z}_i^{(t)}$ and $\mathbf{z}_j^{(p)}$
- $\mathbf{C}_{j,i} \Rightarrow c_i$ is prediction and c_j is ground truth $\Rightarrow$ for $\mathbf{z}_i^{(p)}$ and $\mathbf{z}_j^{(t)}$

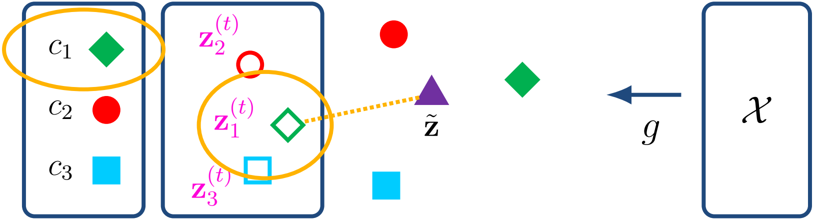
Mirroring Trick



Two roles of class

- two roles of class c_i: ground truth role $\mathbf{z}_i^{(t)}$ and prediction role $\mathbf{z}_i^{(p)}$
- $\mathbf{C}_{i,j} \Rightarrow c_i$ is ground truth and c_j is prediction $\Rightarrow$ for $\mathbf{z}_i^{(t)}$ and $\mathbf{z}_j^{(p)}$
- $\mathbf{C}_{j,i} \Rightarrow c_i$ is prediction and c_j is ground truth $\Rightarrow$ for $\mathbf{z}_i^{(p)}$ and $\mathbf{z}_j^{(t)}$
- learn a **regressor** g from $\mathbf{z}_1^{(p)}, \mathbf{z}_2^{(p)}, \ldots, \mathbf{z}_K^{(p)}$

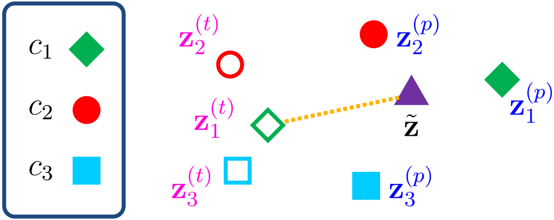
Mirroring Trick



Two roles of class

- two roles of class c_i: ground truth role $\mathbf{z}_i^{(t)}$ and prediction role $\mathbf{z}_i^{(p)}$
- $\mathbf{C}_{i,j} \Rightarrow c_i$ is ground truth and c_j is prediction $\Rightarrow$ for $\mathbf{z}_i^{(t)}$ and $\mathbf{z}_j^{(p)}$
- $\mathbf{C}_{j,i} \Rightarrow c_i$ is prediction and c_j is ground truth $\Rightarrow$ for $\mathbf{z}_i^{(p)}$ and $\mathbf{z}_j^{(t)}$
- learn a **regressor** g from $\mathbf{z}_1^{(p)}, \mathbf{z}_2^{(p)}, \ldots, \mathbf{z}_K^{(p)}$
- find the **nearest hidden point** of $\tilde{\mathbf{z}}$ from $\mathbf{z}_1^{(t)}, \mathbf{z}_2^{(t)}, \ldots, \mathbf{z}_K^{(t)}$

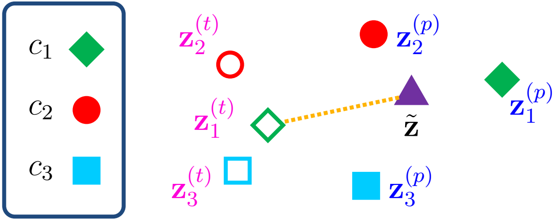
Active Learning with Cost Embedding



Cost-sensitive Uncertainty

- nearest hidden point with large distance $\Rightarrow$ uncertain prediction
- **cost-sensitive uncertainty**: distance between nearest hidden point and predicted hidden point $\tilde{\mathbf{z}}$

Active Learning with Cost Embedding



Cost-sensitive Uncertainty

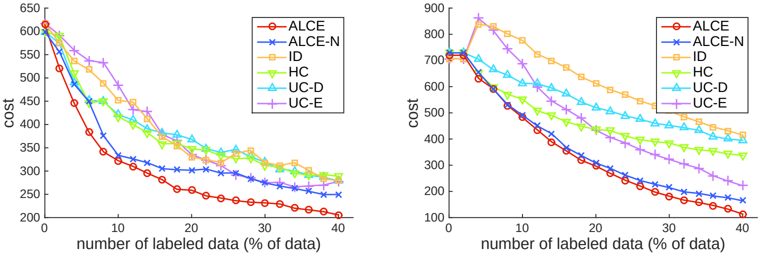
- nearest hidden point with large distance $\Rightarrow$ uncertain prediction
- **cost-sensitive uncertainty**: distance between nearest hidden point and predicted hidden point $\tilde{\mathbf{z}}$

Active learning with cost embedding (ALCE)

- for round $t = 1, 2, ..., T$
  - select $\mathbf{x}_s \in \mathcal{D}_u$ with highest cost-sensitive uncertainty to query the label y_s
  - update $\mathcal{D}_l$ and $\mathcal{D}_u$, and learn a classifier $f^{(t)}$ by cost embedding

Comparison with Cost-Insensitive Algorithms

- ID, HC, UC-D, UC-E: their querying strategies + RBF kernel SVM
- ALCE-N (blue line): proposed querying strategy + RBF kernel SVM
- ALCE (red line): proposed querying strategy + cost embedding

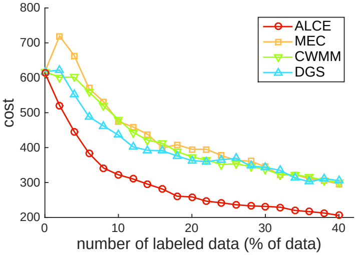


(a) vehicle (b) vowel

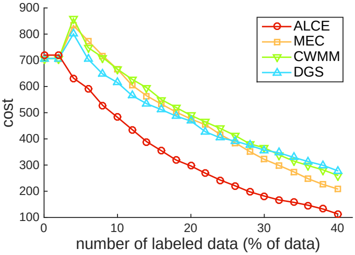
- ALCE-N outperforms ID, HC, UC-D, UC-E ⇒ querying strategy is useful
- ALCE outperforms ALCE-N ⇒ cost embedding is useful

Comparison with Cost-Sensitive Algorithms

- MEC, CWMM, DGS: probabilistic uncertainty + RBF kernel SVM
- ALCE (red line): non-probabilistic uncertainty + cost embedding



(a) vehicle



(b) vowel

- ALCE outperforms MEC, CWMM, DGS

Conclusion

- propose **active learning with cost embedding (ALCE)**
  - embedding view for cost-sensitive multiclass classification
  - embed cost information in distance by non-metric multidimensional scaling
  - mirroring trick for asymmetric cost matrix
  - define cost-sensitive uncertainty by distance
- **promising performance** of ALCE compared with state-of-the-art cost-sensitive active learning algorithms

Thank you! Any question?