Enhancing ensemble learning and transfer learning in multimodal data analysis by adaptive dimensionality reduction

Presented by:
Chandana

ABSTRACT....

- Traditional ensemble and transfer learning techniques doesn’t give good results on mutli model data which is contrast.
- This paper proposes a method to improve data analysis on multimodal datasets—by addressing the challenges of inconsistent data quality, sparsity, and class imbalance
- The authors propose a graph-based, adaptive dimensionality reduction method that identifies the most important features in different data subsets

BACKGROUND & MOTIVATION SUMMARY

- Ensemble Learning: Uses multiple models (learners) to improve data analysis.
- Model level: Combines predictions from different models.
- Data level: Splits data into diverse parts to enhance learning.
- Transfer Learning: Useful when training and test data come from different distributions.
- Focus on transductive transfer learning, which transfers knowledge from labeled to unlabeled data in a related domain.
- Dimensionality Reduction: Commonly used in both ensemble and transfer learning to improve performance and reduce complexity.
- Motivation: Traditional methods struggle with multimodal datasets that are noisy, uneven, and complex.
- This work proposes an adaptive, graph-based dimensionality reduction method to enhance ensemble and transfer learning in such contexts.

MAIN APPROACHES IN MODEL-LEVEL ENSEMBLE LEARNING

1.Bagging:

- Bootstrap Sampling: Multiple subsets of training data are created by sampling with replacement.
- Weak Classifiers: Simple models (e.g., decision trees) are trained on these subsets, then combined by majority voting.
- Limitation: Sensitive to training data and may not work well with noisy datasets.

2.Boosting:

- Classifiers are trained sequentially, each correcting errors from the previous one.
- AdaBoost: A popular boosting algorithm that adjusts weights to focus on harder-to-classify examples.
- Advantage: Robust to overfitting and improves generalization accuracy.

3.Pruning

- After training, rather than using all classifiers, pruning selects a subset of the most relevant ones to improve accuracy and reduce computational cost.

Methods:

- **Ordering-based Pruning**: Classifiers are ranked by their relevance using metrics like error distance or diversity.
- **Optimization-based Pruning**: Uses optimization algorithms (e.g., genetic algorithms) to select the best subset of classifiers.
- **Clustering-based Pruning**: Classifiers are grouped into clusters, and those closest to the cluster centroids are selected to maximize diversity and efficiency.

Challenges:

- Balancing diversity and accuracy is crucial.
- Pruning adds complexity but can significantly improve efficiency and generalization.

MAIN APPROACHES TRANSFER LEARNING

1. Importance Sampling Methods

- Minimize expected risk over target domain.

Examples:

- **Kernel Mean Matching (KMM)**: Aligns mean of source and target data in kernel space.
- **Importance Weighted Twin Gaussian Processes**: Learns importance weights via density estimation.

2. Density Ratio Estimation

- Estimate how likely a sample is from source vs. target.

Techniques:

- **KLIEP**: Uses Kullback-Leibler divergence to estimate density ratio.
- **Domain Classifier**: Trains a model to distinguish source and target data.

3.Advanced Domain Adaptation Techniques

1. Domain Adversarial Methods
2. Statistical & Geometrical Alignment

4.Deep Learning-Based Approaches

1. Dynamic Domain Adaptation (DDA)
2. Deep Residual Correction Network (DRCN)

Main Contributions......

1.Novel Adaptive Architecture:

- Uses double graph Laplacian for adaptive dimensionality reduction.
- Combines local (Gaussian) and global (mutual information) metrics.

2.Adaptive Feature Clustering:

- Tailors feature selection per sample to increase relevance and informativeness.

3.Hybrid Learning Strategy:

- Applies ensemble learning and transductive transfer learning on graph-reduced data.

4.Non-Euclidean Metrics:

- Enables robust classification in complex, non-linear, high-variability datasets.

Improved Generalization and Lower Complexity for operational datasets.

Method

The architecture processes complex, multimodal datasets using graph-based representation and spectral clustering for adaptive dimensionality reduction

- The system Represents data samples as nodes in a graph.
- Selects a subset of features per sample using graph clustering, preserving structure and reducing redundancy.
- Leverages local (Gaussian kernel) and global metrics for similarity.
- Constructs a graph Laplacian matrix to capture the structure and variability in the data.
- Uses eigenvalue decomposition of Laplacian matrices to guide feature selection and clustering.

Challenges in Classical Spectral Clustering

Difficulty identifying:

- The number of clusters automatically.
- Effective clustering with sparse data.
- Similarity metrics capturing data variability.
- Top eigenvectors often fail in noisy, high-dimensional, or multimodal datasets due to blurred values and weak separability

Solution?

Supersample-Based Graph Construction

- Partition data into supersamples Sm of similar statistical/spatial characteristics.
- Each supersample forms a feature graph $G_m^{\mathcal{K}} = (\mathcal{V}, \mathcal{E}_m^{\mathcal{K}}, \mathcal{E}_m^{\mathcal{M}})$.

1. Gaussian Kernel Similarit:Used for local feature similarity

$$w^{\mathcal{K}}_{x_{m_1} x_{m_2}} = \exp\left(-\frac{\|x_{m_1} - x_{m_2}\|^2}{2\sigma^2}\right)$$

- σ: controls width of the kernel.
- High $w^{\mathcal{K}}_{x_{m_1} x_{m_2}}$: features are similar in local context.

2.Mutual Information-Based Similarity

Used to evaluate global feature redundancy

$$w^{\mathcal{M}}_{n_1 n_2} = \sum_{i=1}^{P} \sum_{j=1}^{P} P(x_{in_1}, x_{jn_2}) \log \frac{P(x_{in_1}, x_{jn_2})}{P(x_{in_1})P(x_{jn_2})}$$

- Mutual information $w^{\mathcal{M}}_{n_1 n_2}$: shared information between features.
- High value → redundancy; low value → novelty.

Benefits

- Local adaptation to preserve geometry/statistics.
- Reduces redundancy while retaining relevant, non-overlapping feature sets.
- Handles heterogeneity in large multimodal datasets efficiently.

Deep.....

Joint Graph-Based Feature Selection via Dual Laplacians

Adjacency and Laplacian Matrices

Degree matrices:

- $D_m^{\mathcal{K}} = \mathrm{diag}(d_{m,n}^{\mathcal{K}}), \quad d_{m,n}^{\mathcal{K}} = \sum_j w_{n,j}^{\mathcal{K}}$
- $D_m^{\mathcal{M}} = \mathrm{diag}(d_{m,n}^{\mathcal{M}}), \quad d_{m,n}^{\mathcal{M}} = \sum_j w_{n,j}^{\mathcal{M}}$

Normalized Laplacians:

$$\bar{L}_m^{\mathcal{K}} = I - (D_m^{\mathcal{K}})^{-1/2} W_m^{\mathcal{K}} (D_m^{\mathcal{K}})^{-1/2} \quad , \quad \bar{L}_m^{\mathcal{M}} = I - (D_m^{\mathcal{M}})^{-1/2} W_m^{\mathcal{M}} (D_m^{\mathcal{M}})^{-1/2}$$

Joint Diagonalization for Feature Selection

Identify features such that the subgraph vertices are strongly connected by both similarity graphs.

Optimization problem:

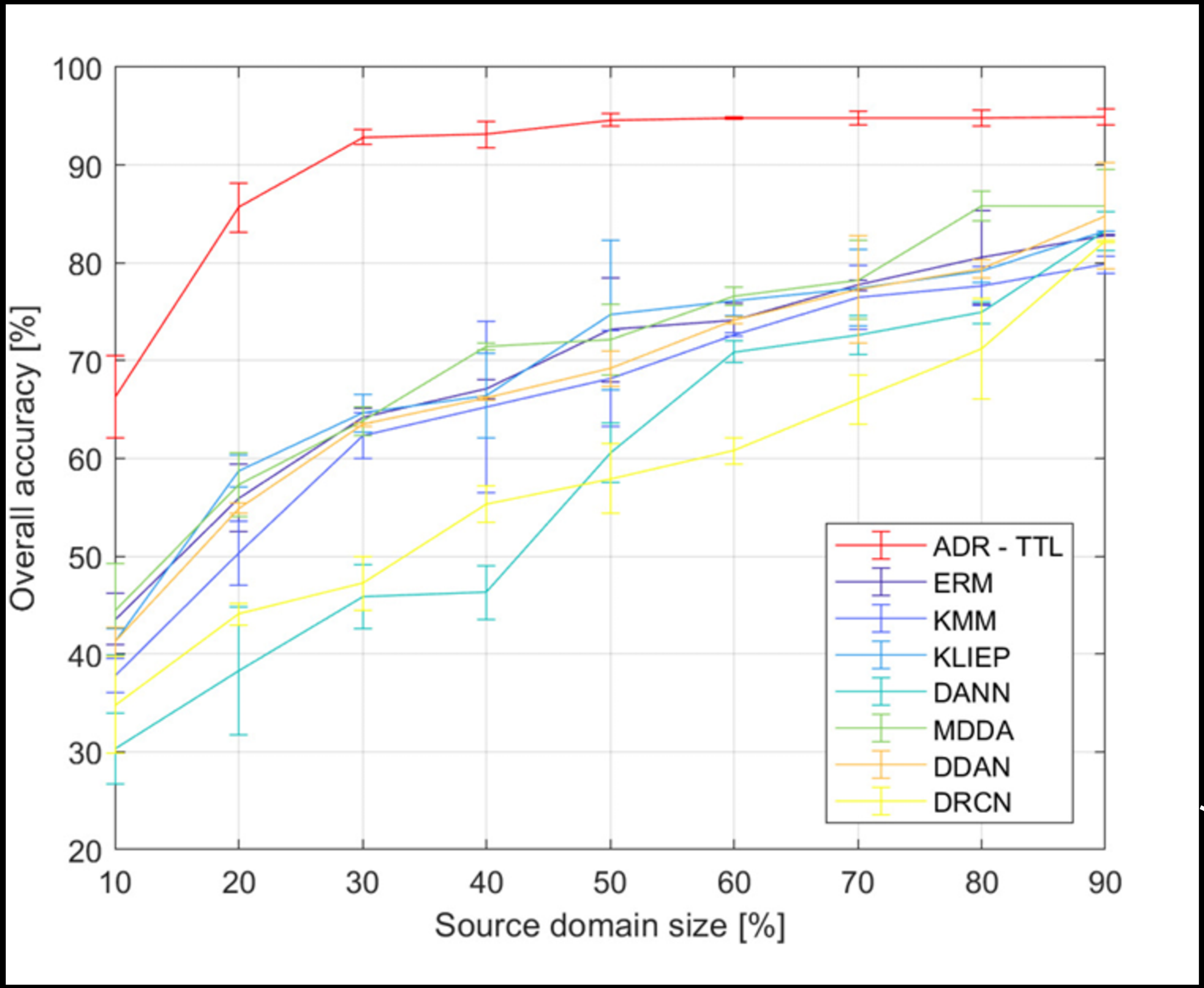
$$\min_{H^T H = I} \mathrm{Tr}(H^T \bar{L}_m^{\mathcal{K}} H) \cdot \mathrm{Tr}(H^T \bar{L}_m^{\mathcal{M}} H) \tag{7}$$

Alternatively, using joint diagonalization (Pham's criterion):

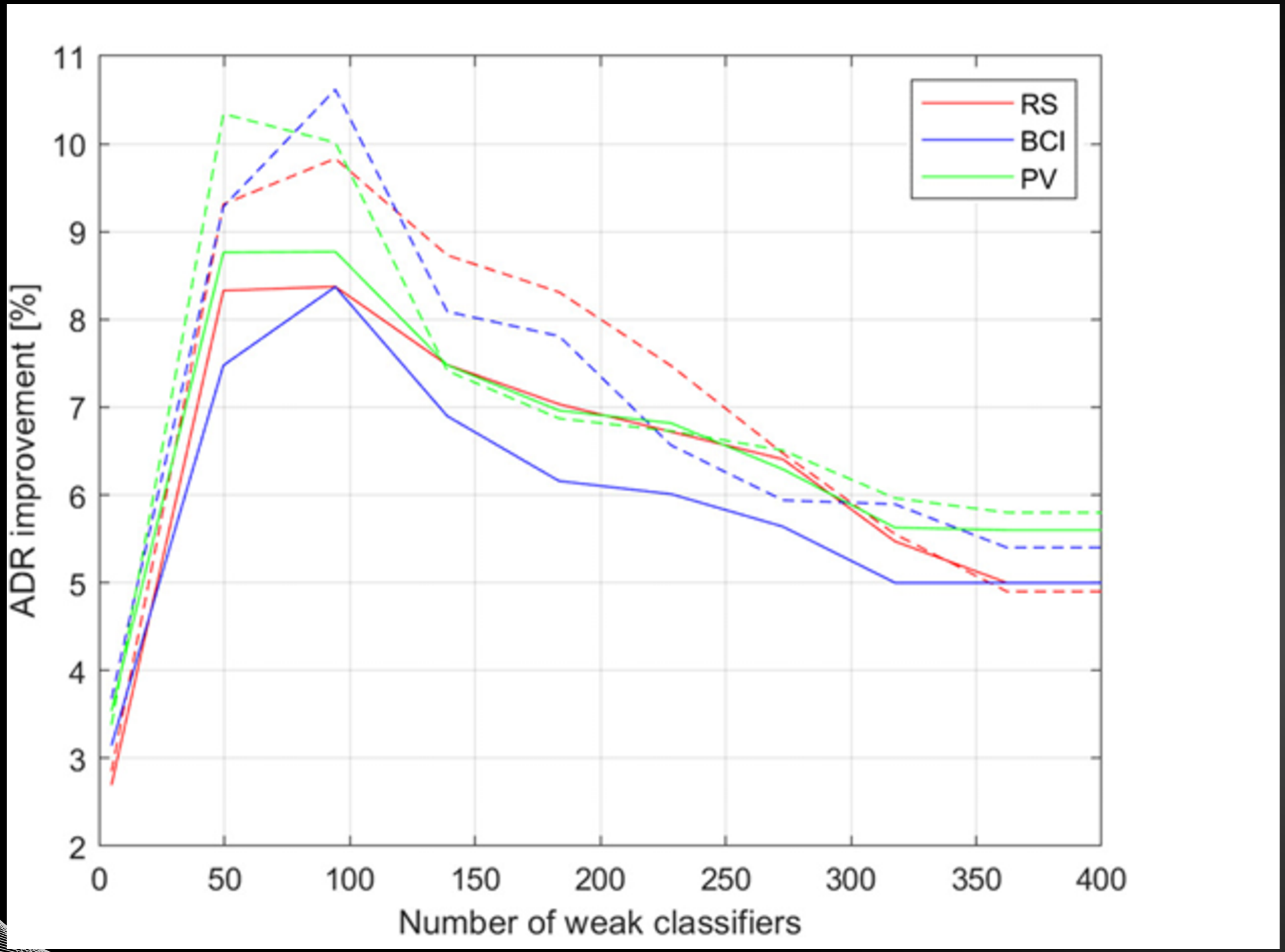
$$\min_{V_m} \left[\mathrm{diag}(V_m^T \bar{L}_m^{\mathcal{K}} V_m) + \log\left(\frac{V_m^T \bar{L}_m^{\mathcal{K}} V_m}{\mathrm{diag}(V_m^T \bar{L}_m^{\mathcal{K}} V_m)} \right) + \mathrm{diag}(V_m^T \bar{L}_m^{\mathcal{M}} V_m) + \log\left(\frac{V_m^T \bar{L}_m^{\mathcal{M}} V_m}{\mathrm{diag}(V_m^T \bar{L}_m^{\mathcal{M}} V_m)} \right) \right] \tag{8}$$

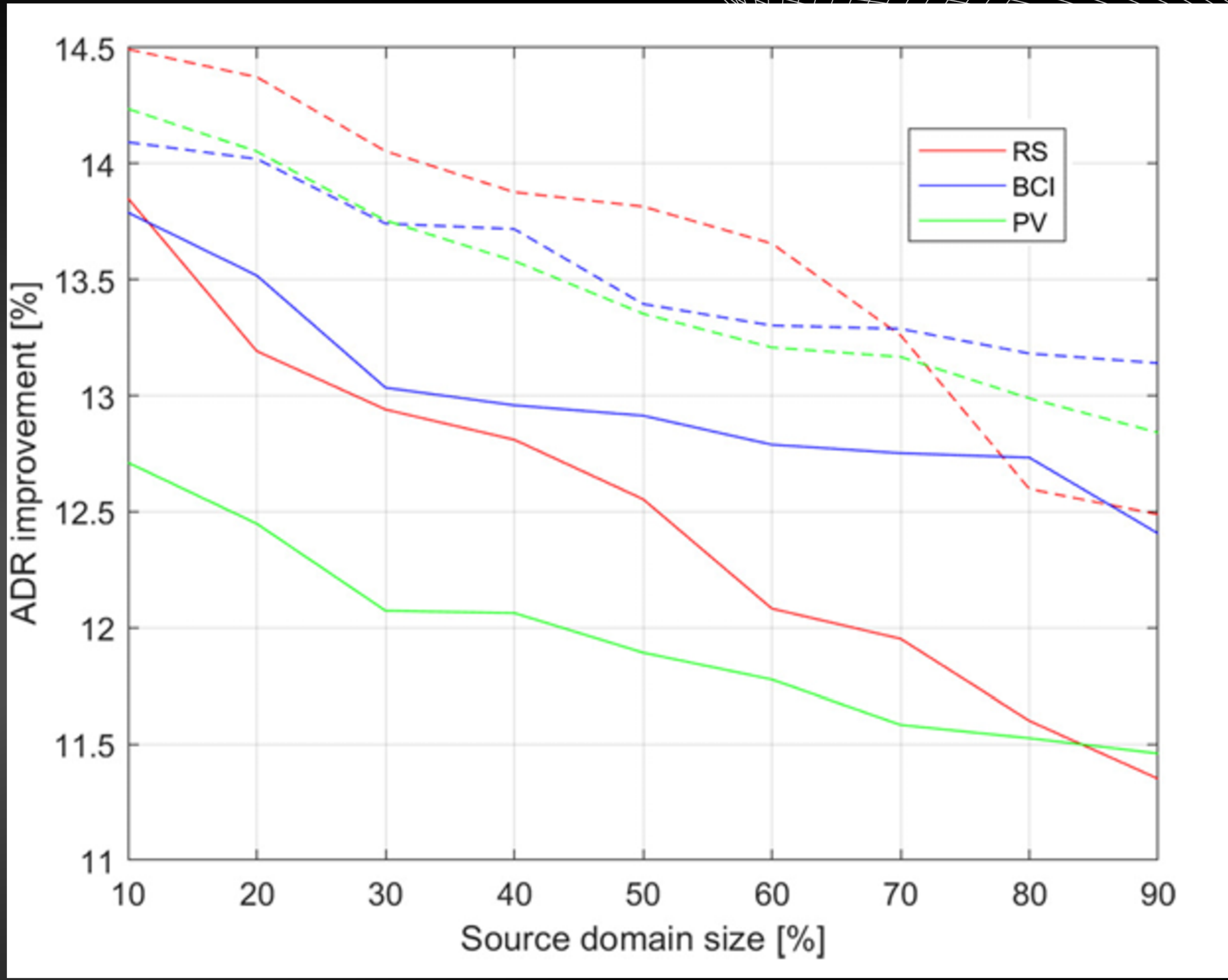
- V_m: matrix of joint eigenvectors.
- Low-dimensional manifold constructed via matrix H (eigenvectors of smallest K_m non-zero eigenvalues).

Results



Overall accuracy [%]
100
90
80
70
60
50
40
30
20
10
20
30
40
50
60
70
80
90
Source domain size [%]
ADR - TTL
ERM
KMM
KLIEP
DANN
MDDA
DDAN
DRCN


RS
BCI
PV
ADR improvement [%]
Number of weak classifiers


RS
BCI
PV
ADR improvement [%]
Source domain size [%]