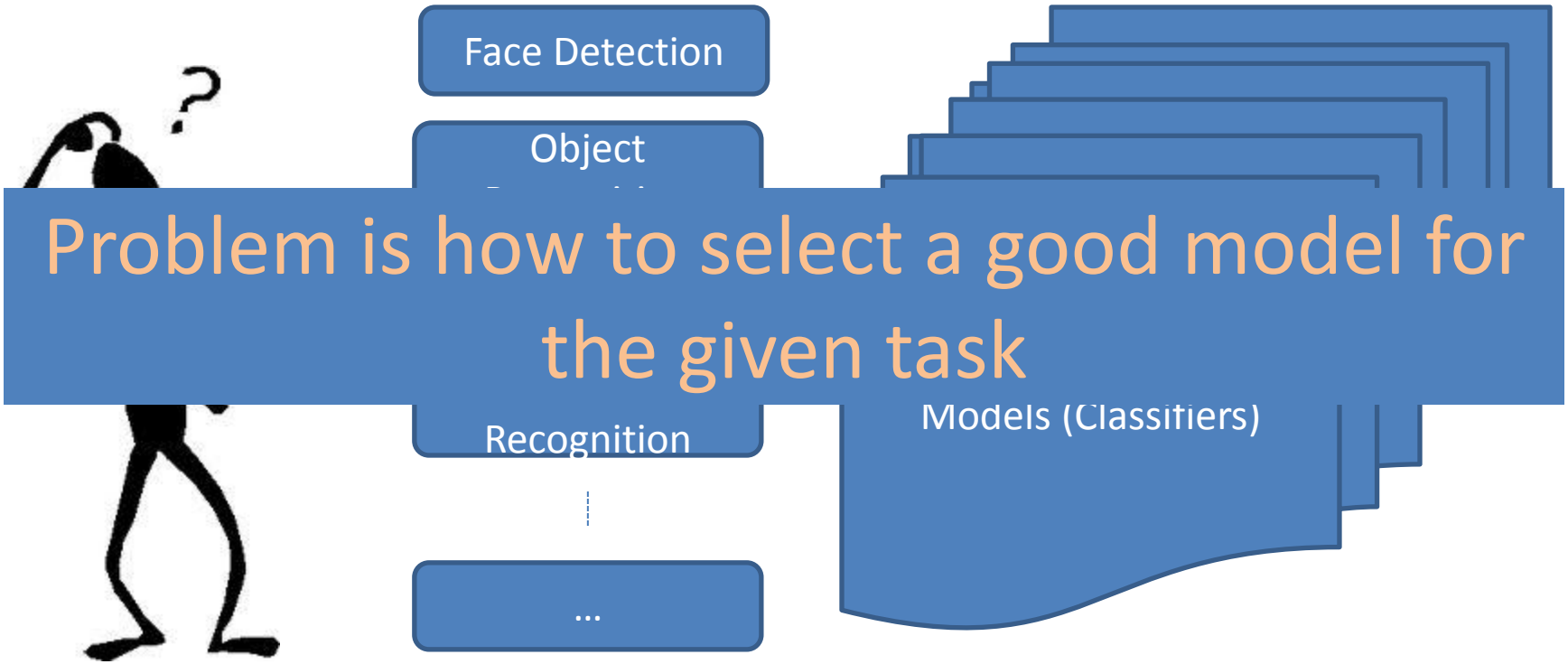
# Paper

**Model Recommendation for Action Recognition**, P. Matikainen, R. Sukthankar and M. Hebert, CVPR 2012

**Ahmet BUĞDAY**

- Introduction
- Overview and Terminology
- Related Work
- Method
- Evaluation
- Conclusion

# Introduction



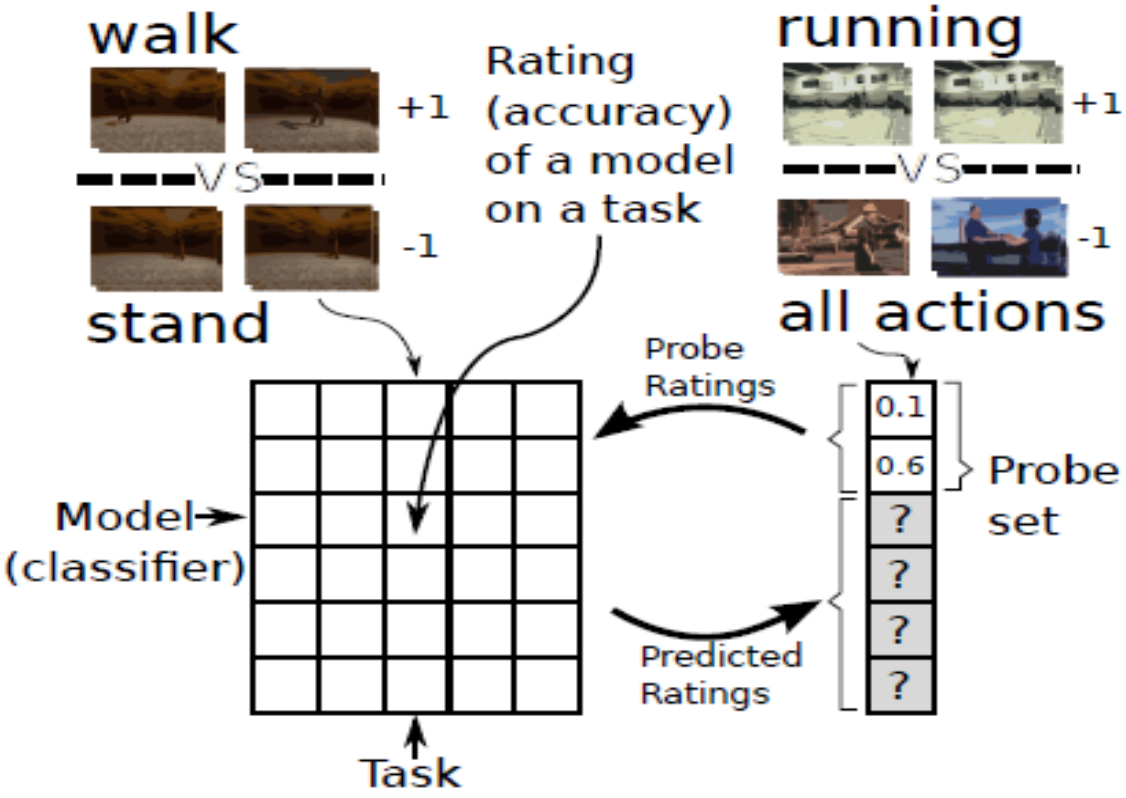
# Introduction

- Given
  - Collection of models (classifier)
    - Large, Unorganized, Heterogeneous
  - A task
  - Training data
    - Very limited number, at  most ten samples
- Goal
  - Select good model for a given task

# Introduction

- Naive Solution: evaluate every model against training data
  - Impractical to evaluate for a large library
  - Evaluating a large number of models against a very small dataset will be very noisy
- Key of this paper
  - Selection problem is reinterpreted as type of recommendation problem adressed by collaborative filtering methods
    - Netflix use for movie recommandations

# Overview and Terminology



# Overview and Terminology

- Problem redefinition
  - Find a model performs well on a given task with limited data
  - User rates only small subset of the models
    - Probe set
  - Goal
    - Predict rating of remainig models using probe set
    - Return model with highest predicted rating

# Overview and Terminology

- Task
  - $T_j = \{(x_{j1}, y_{j1}), (x_{j2}, y_{j2}), \ldots\}$
  - $x_{jz}$ ; zth data sample
    - A video clip
  - $y_{jz}$ ; label corresponding to that sample
    - Binary value indicates class of data sample
    - Different interpretations across tasks

# Overview and Terminology

- Model
  - A classifier whose accuracy can be measured on a task
  - Libraries of pre-trained classifiers used
    - Share same metodology like STIP+HOG3D+SVM
    - Differ in training data
- Rating $R(m_i, T_j) = r_{i,j}$
  - Describes how good that model on a task
  - Restricted to [0,1] range
  - Classification accuracy is used

# Overview and Terminology

- A Thought Experiment

Models trained to recognize walking from different viewing angle

Task with unknown viewing angle

- $\theta_m$ model trained for, $\theta_t$ target viewing angle
- Rating is inversly correlated with the difference in angles
- $f_m = \langle sin(\theta_m), cos(\theta_m) \rangle$ and $f_t = \langle sin(\theta_t), cos(\theta_t) \rangle$

# Overview and Terminology

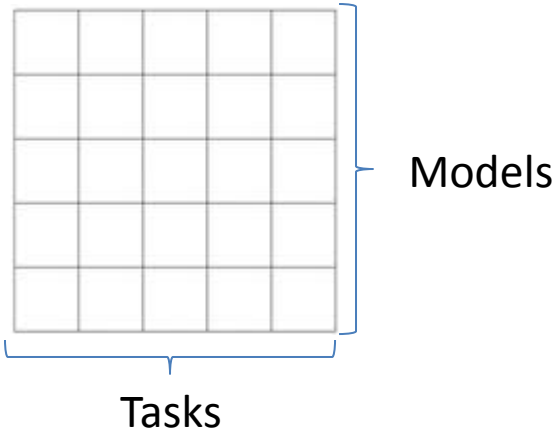
- A Thought Experiment
  - Rating proportional to to the dot product between respective factors vectors
    - $r_{m,t} \propto f_m \cdot f_t = f_m^T f_t$
  - By evaluating only two model the performance of every other model could be predicted
  - More than two model improve prediction quality

# Related Work

- Gopalan et al.[10], Saenko et al.[22] map samples from source to target
  - Model samples are not available
- Blitzer et al. pivot features, Miersa and Wurst base features, Lee et al. meta-features
- Neural network [1,6] force all the task to share intermediate represantations
- Rückert and Kramer  use meta-kernel to compare how similar datasets are.
- This methos does not use these
  - Colloborative filtering approaches used by the BellKor team
  - Focussed on basic factorization technique for this paper
  - Offline and online factorization techniques

# Method

- Rating Store




- Assume matrix is complete
- Then given a new target task and a subset of rated models on that task, colloborative filtering techniques to predict ratings of all the models

# Method

- Low-level input
  - Consider models based on two low-level inputs
    - STIP+HOG3D combination, gridded histogram of optical flow (HOF)
  - HOF divides optical flow into 10x10x5 pixel spatio temporal cells, 9 dimensional HOF descriptor for each cell
    - 320x240x100 video represented by 32x24x20x9 cells
    - Trained model might be applied to a 12x12x10x9 scanning window of the full grid
    - FlowLib GPU-accelerated oprical flow library is used

# Method

- Rating Store Generation
  - Generate rating for both synthetic and real data
  - Mind's eye and UCF-YT-50 datasets are used
  - Models are generated by training SVM classifier on the low-level input
    - Each classifier is trained 1 vs. N classifier of actions from dataset
  - Tasks are generated 1 vs. all tasks
    - Each task is binary classification problem videos of one action vs. videos of all other actions
  - Every model is rated on every task using accuracy on that task

# Method

- Collaborative Filtering
  - Used to predict ratings of entire library based on the probe set ratings and rating store
  - Baseline estimation and  prediction by using factorization and sparse coding

# Method

- Baseline Estimation
  - Mean model ratings
    - Avarage accuracy for each classifier across all actions
  - Mean task ratings
    - Avarage accuracy of all classifiers on an action
  - Formally $r_{i,j} = \mu + \phi_i + \psi_j$
    - $\mu$ global mean rating, $\phi_i$ model-spesific factor, $\psi_j$ task-spesific factor
  - m number of models, n number of tasks, number of rating m.n
  - Koren[13] formulation used for estimating these factors

# Method

- Baseline Estimation
  - Koren formulation

**1.** Estimate global mean: $\mu = \frac{\sum_i \sum_j r_{i,j}}{mn}$;

**2.** Initial factors: $\phi_i = \frac{\sum_j (r_{i,j} - \mu)}{n}$, $\psi_j = \frac{\sum_i (r_{i,j} - \mu)}{m}$;

**3.** Model factors: $\phi_i = \frac{\sum_j (r_{i,j} - \mu - \psi_j)}{n}$;

**4.** Task factors: $\psi_j = \frac{\sum_i (r_{i,j} - \mu - \phi_i)}{m}$.

- Estimation of target task's factor formulation

$$\psi_t = \frac{\sum_{i \in P} (r_{i,t} - \mu - \phi_i)}{|P|}$$

- $\psi_t$, Target task
- P probe set feature
- $r_{i,t}$ evaluated rating of feature ion task t
- |P| number of probe set

# Method

- Baseline Estimation
  - This will not fit data; $\left| r_{i,j} - \mu - \phi_i - \psi_j \right| > 0$
  - Ratings predicted by baseline differ from observed rating. Difference is called residual $\bar{r}_{i,j}$
  - $\bar{r}_{i,j} = r_{i,j} - (\mu + \phi_i + \psi_j)$ remains after baseline estimation
  - $\bar{R}$ entire mxn residual matrix for the source tasks
  - $\bar{r}_{i,t} = r_{i,t} - (\mu + \phi_i + \psi_t)$ residual for target task

# Method

- Factorization methods
  - The goal of factorization methods is represent the residual rating of a model on a task
  - $\bar{R} = F^T D$
    - $F^T$ mxk matrix of model factors
    - D kxn matrix of task factors
  - Singular Value Decomposition (SVD) used as factorization schema
  - $\bar{R} = USV^T$
    - k latent factors are sought. $S_k$ is kxk upper left sub-matrix of S. $U_k$ firts k columns of U.
    - $F^T = U_k S_k$ model factors matrix , $\mathrm{D} = V_k$ task factors constructed

# Method

- Factorization methods
  - Estimate the target task's factor vector by solving the linear least-squares problem for $x$
    - $\left(\hat{F}^T\right)x = \bar{r}_p$
      - $x$ is kx1 vector of target task's factors
      - $\hat{F}^T$ is pxk matrix of the first p rows of $F^T$
  - Predict target task's residual ratings for all models
    - $\bar{r}' = F^T x$
  - Final predicted ratings
    - $r_i' = \bar{r}_i' + \mu + \phi_i + \psi_t$

# Method

- Sparse Coding
  - Represent the column of residual probe ratings as a sparse linear combination of columns (tasks) from the residual ratings matrix
  - Optimizes the problem:

$$\arg\min_{\alpha} ||\bar{r}_p - \bar{R}_p\alpha||_2^2 + \tau||\alpha||_1$$

    - $\bar{r}_p$ residual of probe ratings, $\bar{R}_p$ rows of the residual rating matrix corresponding to probe models, α weight vector , $\tau$ controls sparsity
  - Once α has been computed, the predicted residual ratings $\bar{r}'$ for all models can be computed simply as the weighted combination of columns of $\bar{R}$, or the matrix product $\bar{r}' = \bar{R}\alpha$

# Evaluation

- First validation of thought experiment
  - Synthetic data of walking used
- Next recommending models across different actions
  - Synthetic tasks and real tasks (UCF-YT and Mind's Eye (ME) datasets used)

(a) UCF-YT-50

(b) Mind's Eye

(c) Semi-synthetic

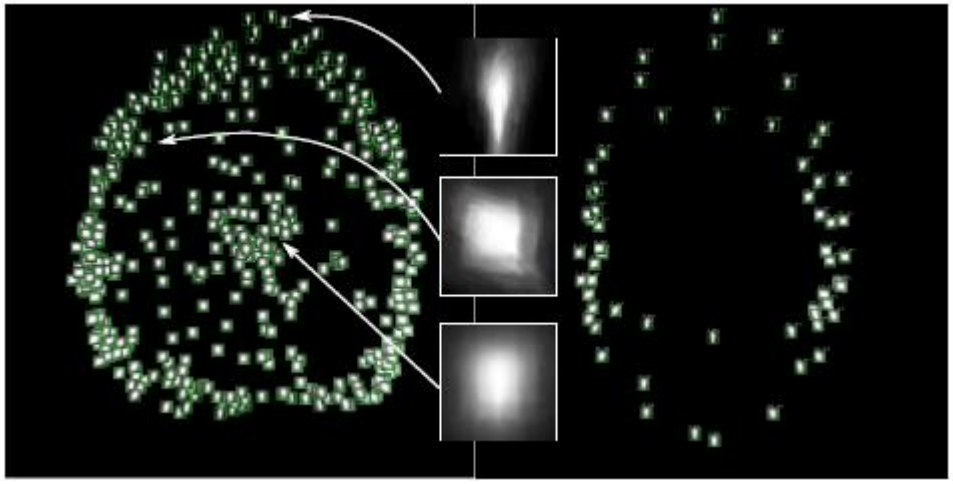
# Evaluation

- Validating thought experiment
  - Goal is to recognize walking from a task-specific range of viewing angles

(a) Models (b) Tasks

For one action (“walk”), models (a) are trained to recognize the action from different viewing angles. Each task (b) is likewise to recognize walking from a specific range of angles

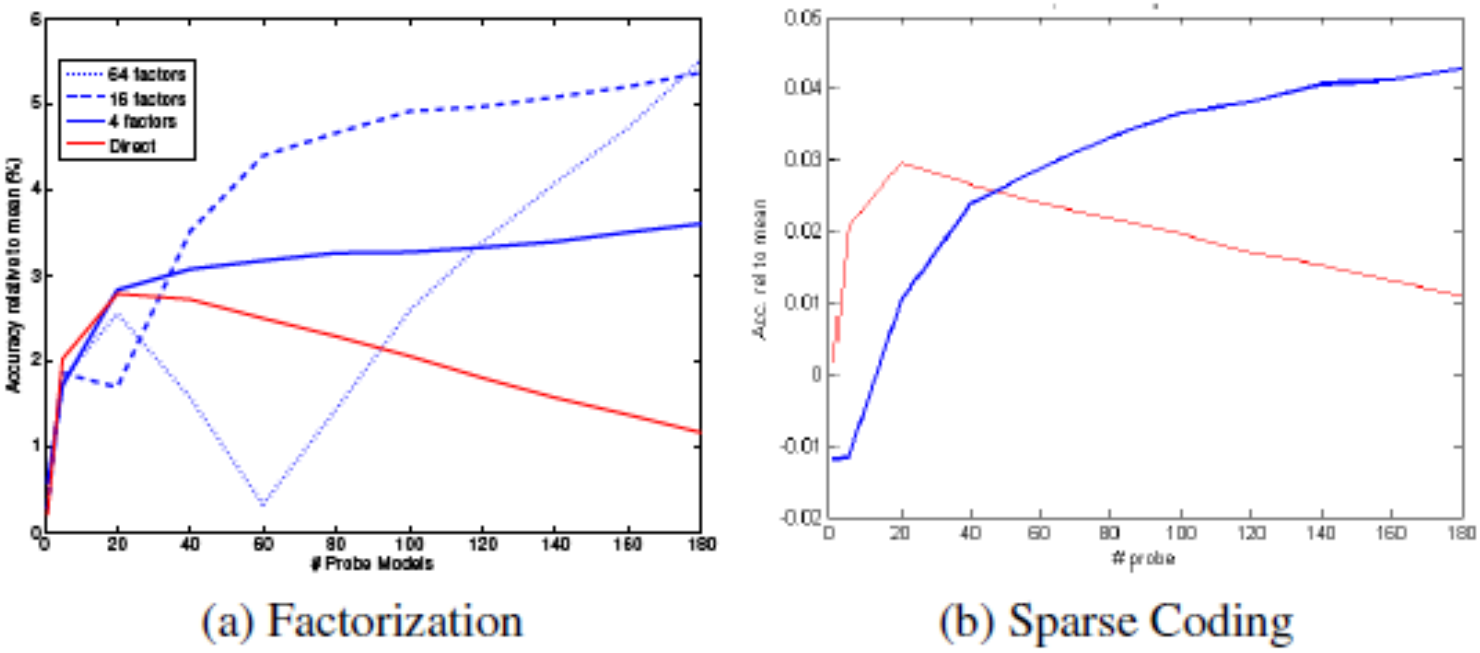
# Evaluation



Scatter plot of tasks (left) and models (right), according to their first two factors. Each “point” is the average silhouette of the positive videos for that task

First two factor encodes the angle for tasks and models

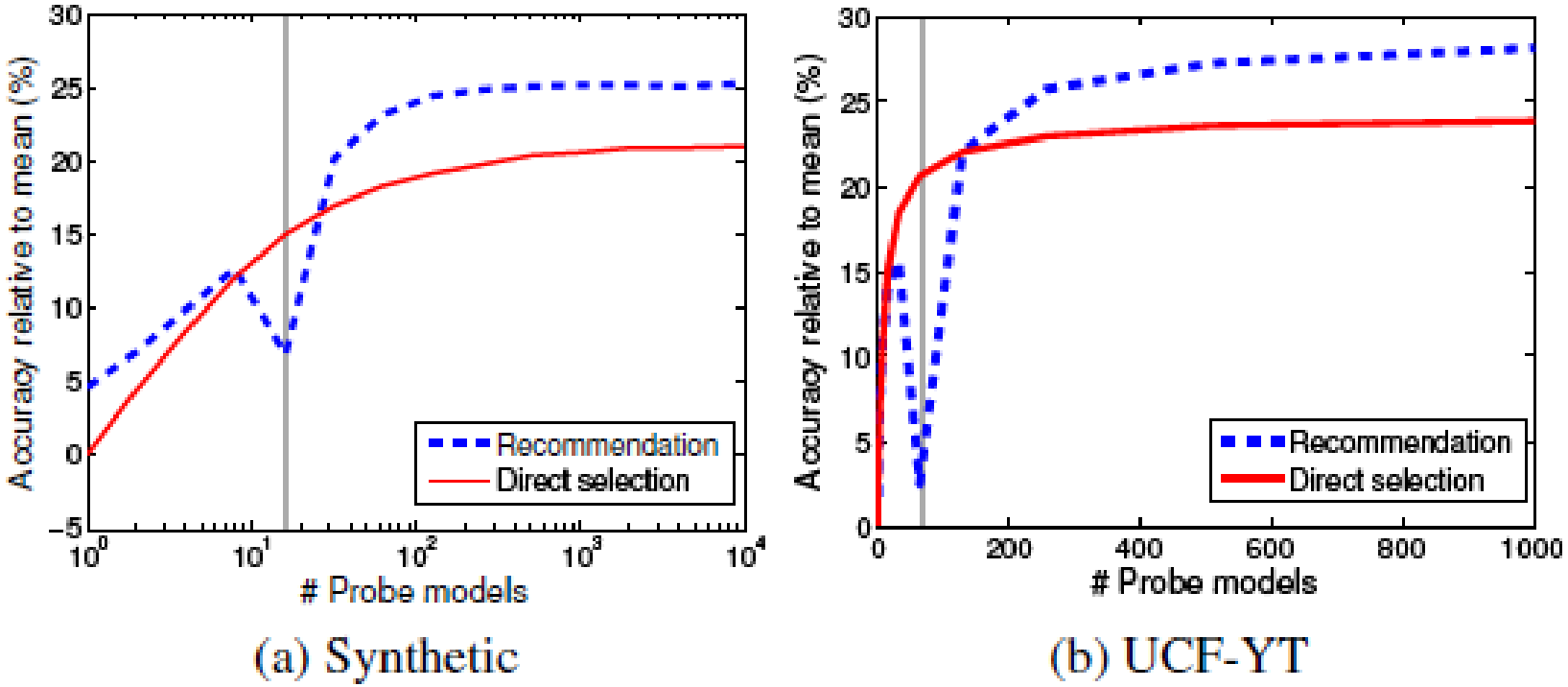
# Evaluation




(a) Factorization (b) Sparse Coding

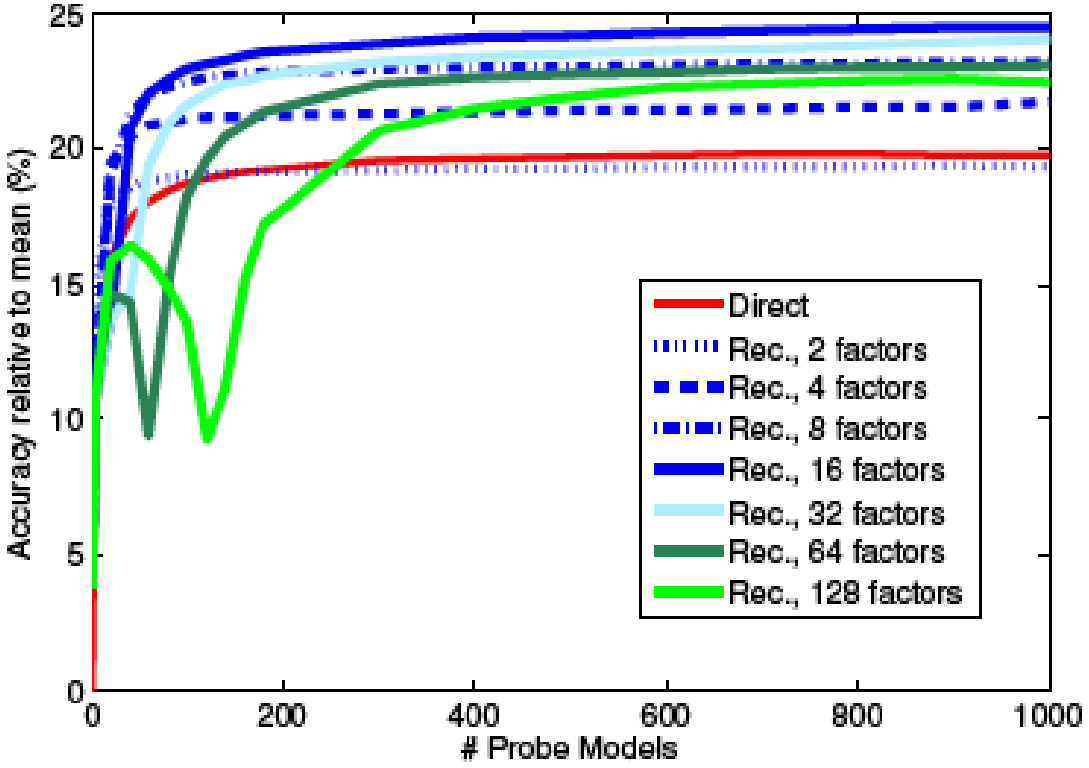
Effect of the number of factors used for the factorization model on accuracy vs. probe set size for the ME dataset. A larger number of factors results in a higher asymptotic accuracy, at the cost of lower performance when few probe models are evaluated. Sparse coding exhibits the same effect, but with a more graceful degradation. ‘Direct’ is the direct selection baseline.

# Evaluation




Mean relative accuracy vs. number of probe models for synthetic (a) and UCF-YT (b) datasets. The same trend is observed in both datasets, although the magnitude of the effect is larger in the synthetic data. At the gray line the number of probes is equal to the number of factors (16 and 64 respectively); for fewer probes, the factor estimation is underconstrained.

# Evaluation




Effect of the number of factors in the factorization model for the synthetic tasks

# Conclusion

- Recommendation systems can select better classifiers than directly evaluating every model on the new training set

# Thanks...