# Matrix Completion in the Unit Hypercube via Structured Matrix Factorization

Emanuele Bugliarello[1], Swayambhoo Jain[2] and Vineeth Rakesh[2]

IJCAI 2019

1 Tokyo Institute of Technology

2 technicolor AI Lab

# Motivation

# Motivation

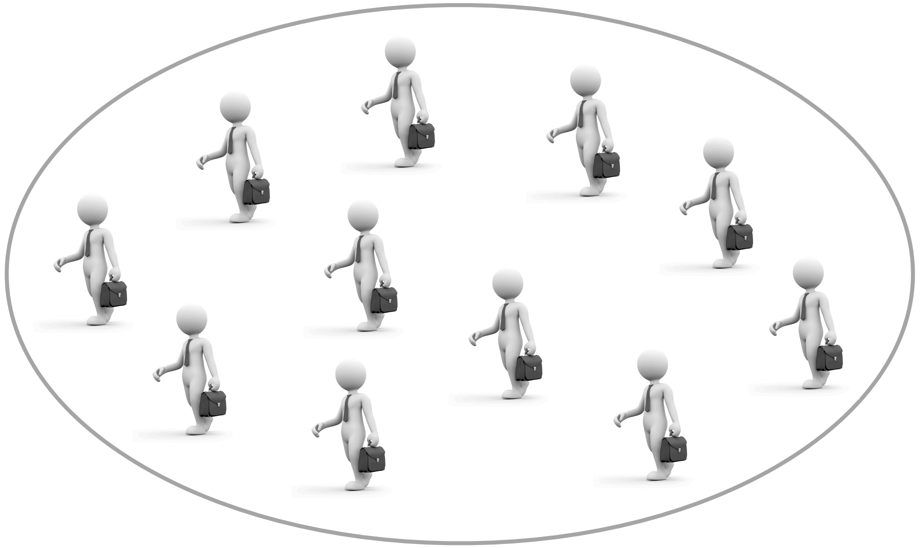
- Challenge: For a given job, how to $\boxed{\text{select}}$ assets among available ones?

# Motivation

- Challenge: For a given job, how to $\boxed{\text{select}}$ assets among available ones?

**Job A**

- Task 1
- Task 2

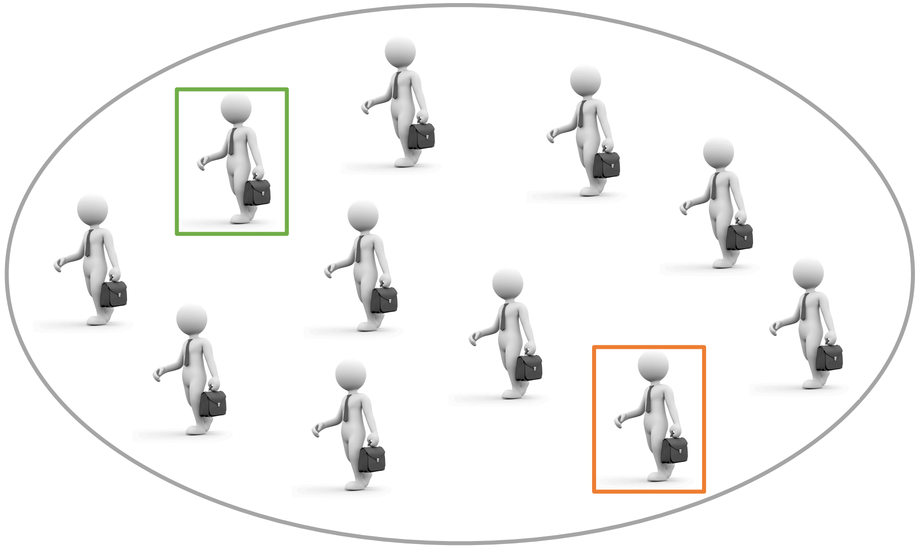


# Motivation

- Challenge: For a given job, how to select assets among available ones?

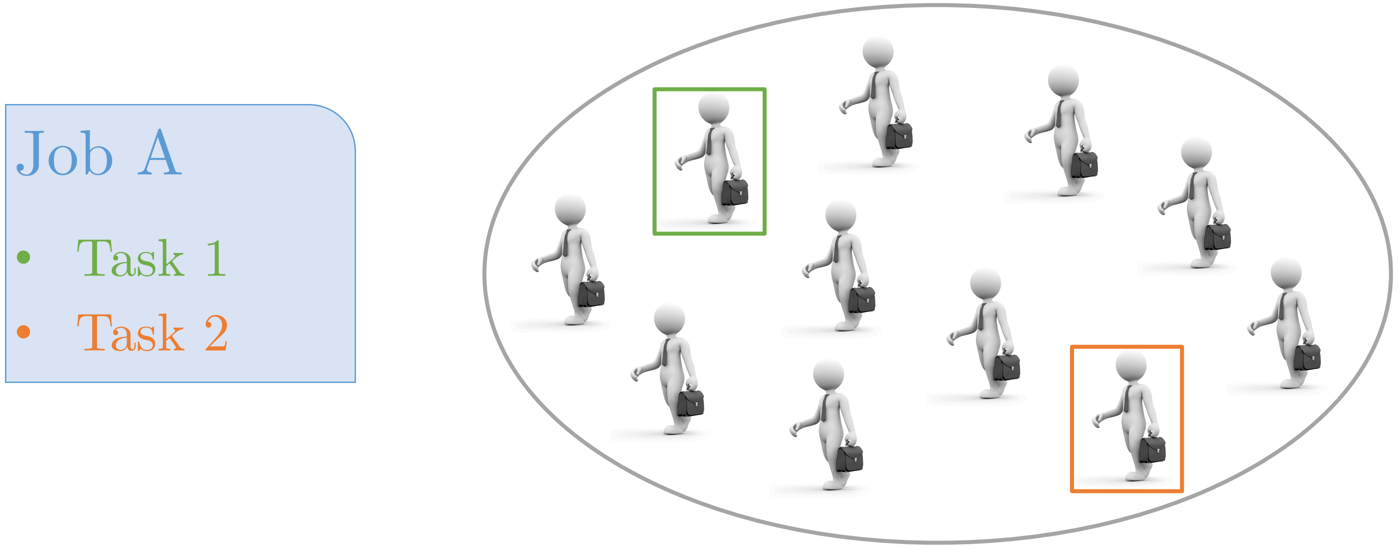
**Job A**

- Task 1
- Task 2



# Motivation

- Challenge: For a given job, how to select assets among available ones?




- Approach: Predict the efficiency of employees in each task!

# Key Contributions

# Key Contributions

- Transform industrial setups into a sparse real-valued matrix with entries lying in the [0, 1] interval (e.g., efficiencies of employees)

# Key Contributions

- Transform industrial setups into a sparse real-valued matrix with entries lying in the [0, 1] interval (e.g., efficiencies of employees)

- Propose 2 novel structured matrix factorization models that leverage our knowledge of the environment
    - *expertise matrix factorization* (EMF): exploits the [0, 1] boundary constraint
    - *survival matrix factorization* (SMF): probabilistic model of employee efficiency

# Key Contributions

- Transform industrial setups into a sparse real-valued matrix with entries lying in the [0, 1] interval (e.g., efficiencies of employees)

- Propose 2 novel structured matrix factorization models that leverage our knowledge of the environment
  - *expertise matrix factorization* (EMF): exploits the [0, 1] boundary constraint
  - *survival matrix factorization* (SMF): probabilistic model of employee efficiency

- Validate the effectiveness of our models on 3 real-world datasets with values bounded in the [0, 1] interval

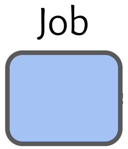
# Problem Formulation (1/2)

# Problem Formulation (1/2)

- Work allocation framework (based on Technicolor's, but very general)

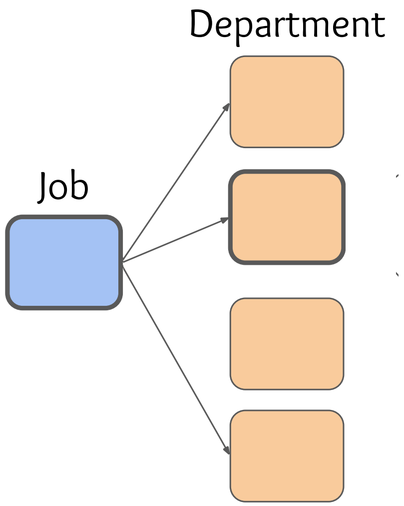
# Problem Formulation (1/2)

- Work allocation framework (based on Technicolor's, but very general)



# Problem Formulation (1/2)

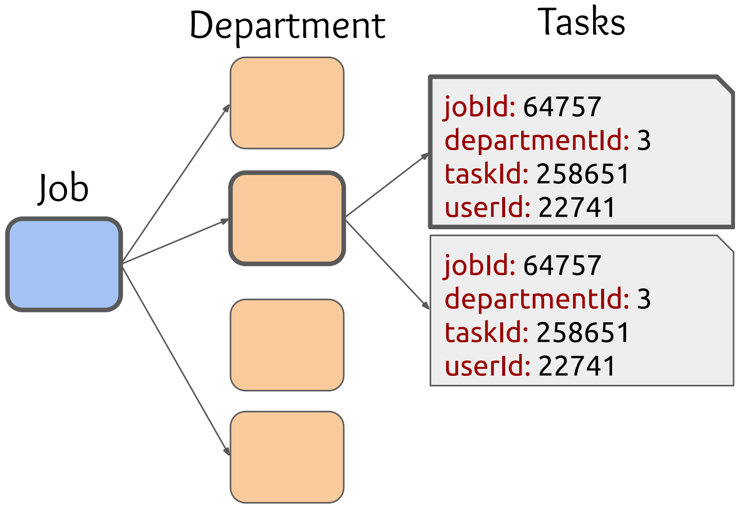
- Work allocation framework (based on Technicolor's, but very general)




  - *A job* for the organization requires contributions from different *departments*

# Problem Formulation (1/2)

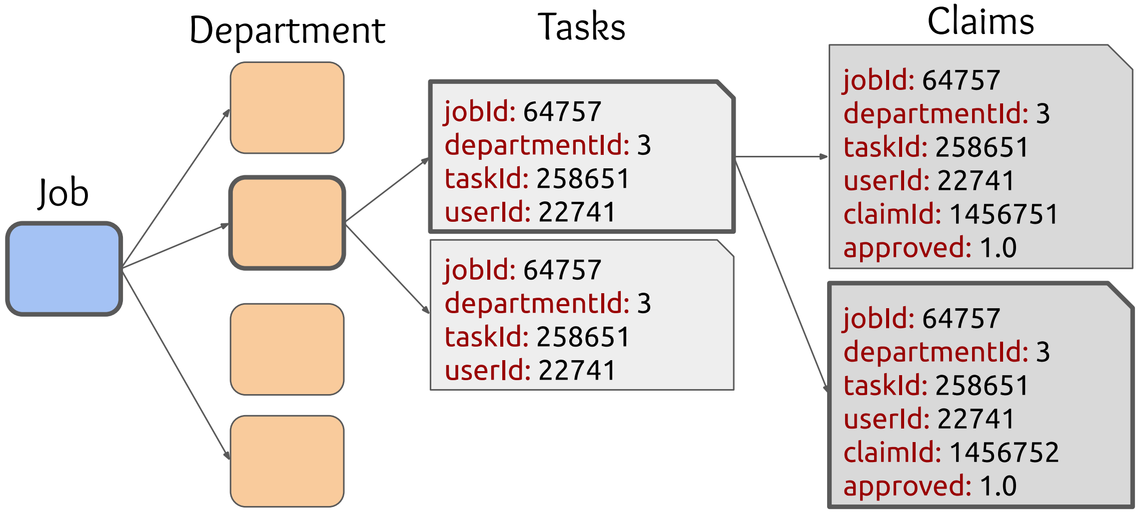
- Work allocation framework (based on Technicolor's, but very general)




  - *A job* for the organization requires contributions from different *departments*
  - Each department is led by a *manager* who divides the work in their department into *tasks*

# Problem Formulation (1/2)

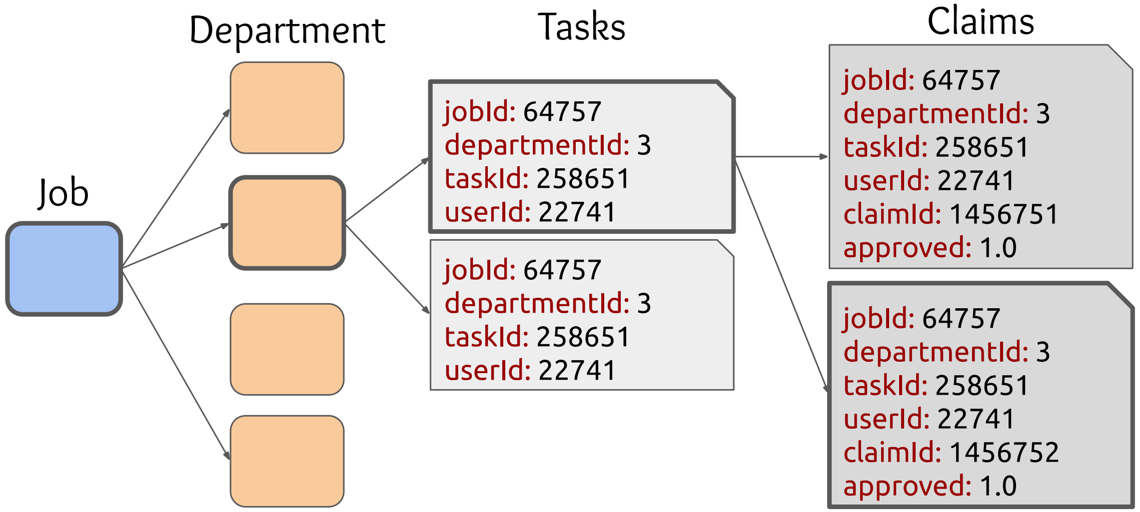
- Work allocation framework (based on Technicolor's, but very general)




- *A job* for the organization requires contributions from different *departments*
- Each department is led by a *manager* who divides the work in their department into *tasks*
- Each *task* is assigned to a single *employee*, who submits partial results called *claims*

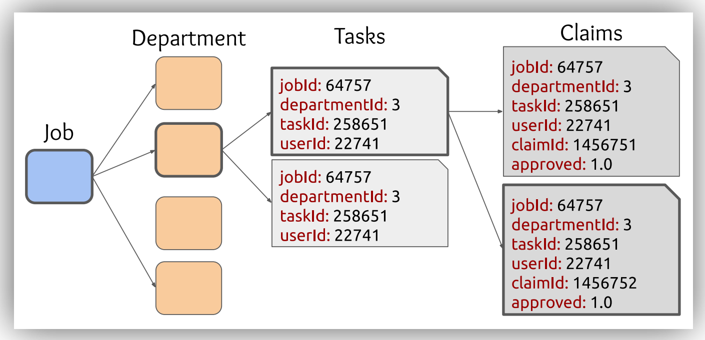
# Problem Formulation (1/2)

- Work allocation framework (based on Technicolor's, but very general)



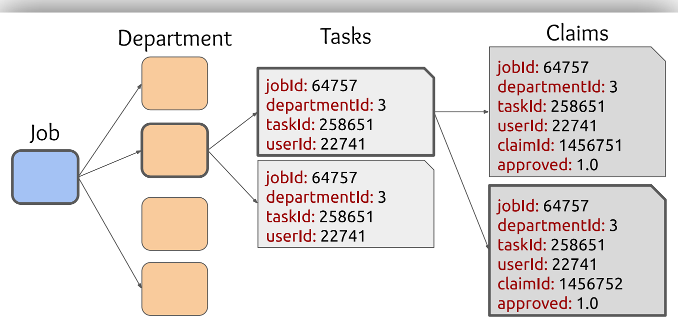

  - *A job* for the organization requires contributions from different *departments*
  - Each department is led by a *manager* who divides the work in their department into *tasks*
  - Each *task* is assigned to a single *employee*, who submits partial results called *claims*
  - Managers assess the quality of claims and decide to approve them or not

# Problem Formulation (2/2)

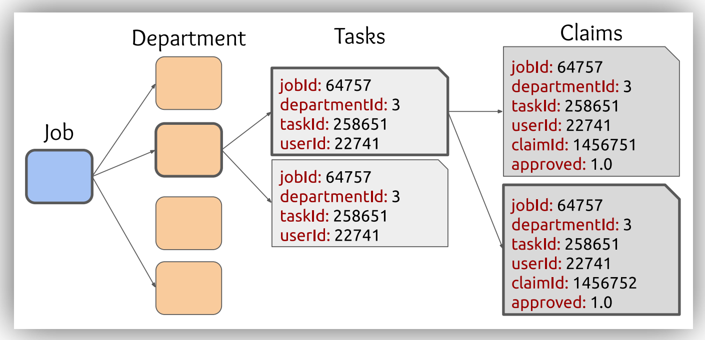


# Problem Formulation (2/2)

- Employee's competency: ratio of accepted claims
  - Rejected claim ⇒ performance loss for the organization



# Problem Formulation (2/2)



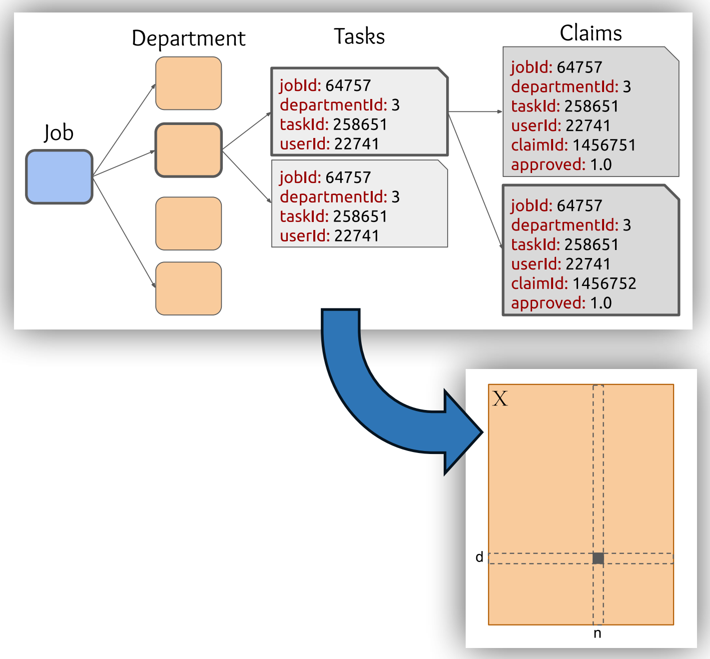

- Employee's competency: ratio of accepted claims
  - Rejected claim $\Rightarrow$ performance loss for the organization

- We define the efficiency of employee $d$ in department $n$ as:

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \in [0,1]$$

  - $N_{dn}$ : total number of claims by employee $d$ in department $n$
  - $a_{dn}^{(i)}$ : $i$-th claim by employee $d$ in department $n$ was accepted (1) or rejected (0)

# Problem Formulation (2/2)




- Employee's competency: ratio of accepted claims
  - Rejected claim ⇒ performance loss for the organization

- We define the efficiency of employee $d$ in department $n$ as:

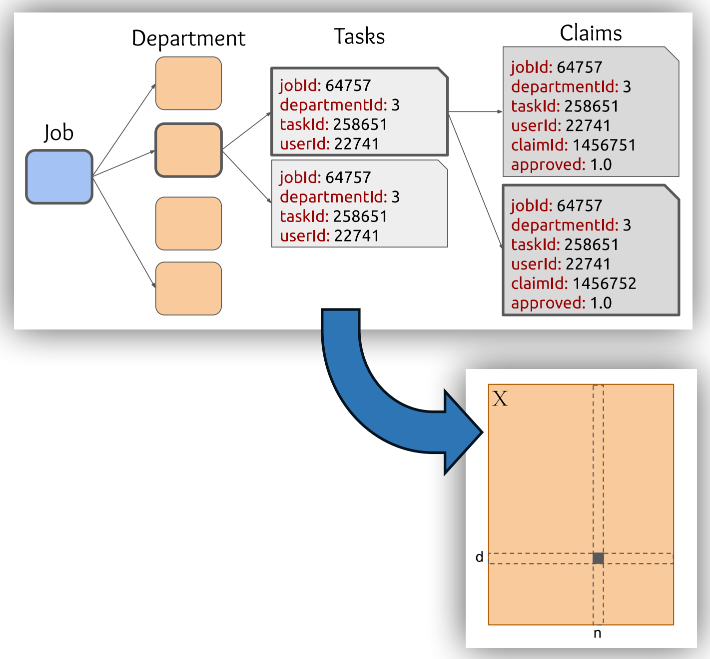
$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \in [0,1]$$

  - $N_{dn}$ : total number of claims by employee $d$ in department $n$
  - $a_{dn}^{(i)}$ : $i$-th claim by employee $d$ in department $n$ was accepted (1) or rejected (0)

- Goal: Predicting employee efficiency ⇒ predicting the missing entries of $\mathbf{X}$

# Problem Formulation (2/2)

- Employee's competency: ratio of accepted claims
  - Rejected claim ⇒ performance loss for the organization




- We define the efficiency of employee $d$ in department $n$ as:

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \in [0,1]$$

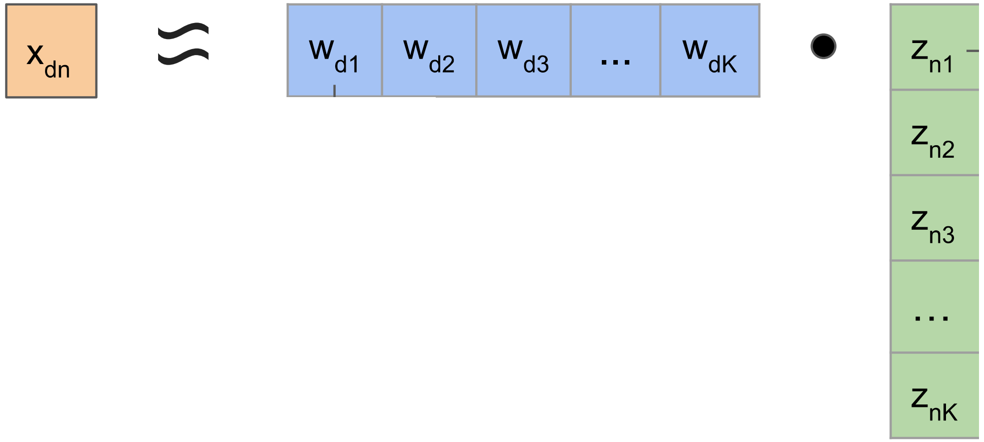
  - $N_{dn}$ : total number of claims by employee $d$ in department $n$
  - $a_{dn}^{(i)}$ : $i$-th claim by employee $d$ in department $n$ was accepted (1) or rejected (0)

- Goal: Predicting employee efficiency ⇒ predicting the missing entries of $\mathbf{X}$
- Most employees work in a few departments ⇒ efficiency matrix $\mathbf{X}$ is sparse

# Proposed Models: Expertise MF

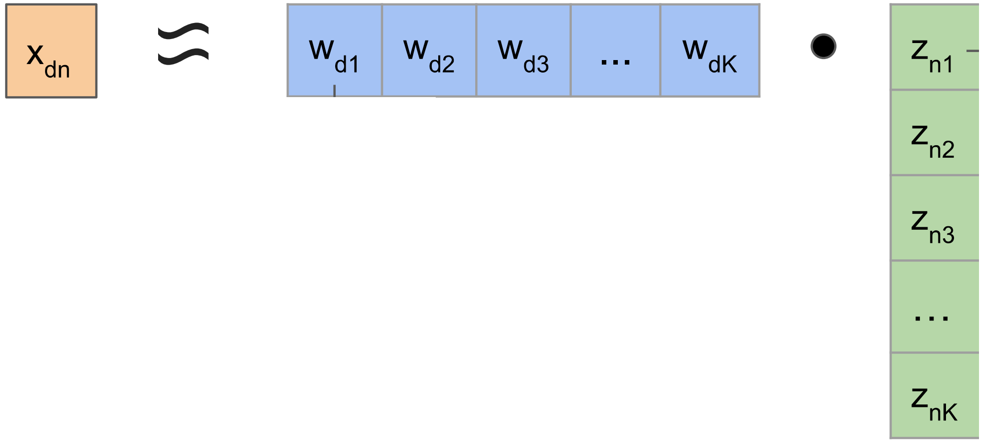
# Proposed Models: Expertise MF

- Model the efficiency matrix as a low-rank matrix $\mathbf{X} \approx \mathbf{W} \cdot \mathbf{Z}^T$



# Proposed Models: Expertise MF

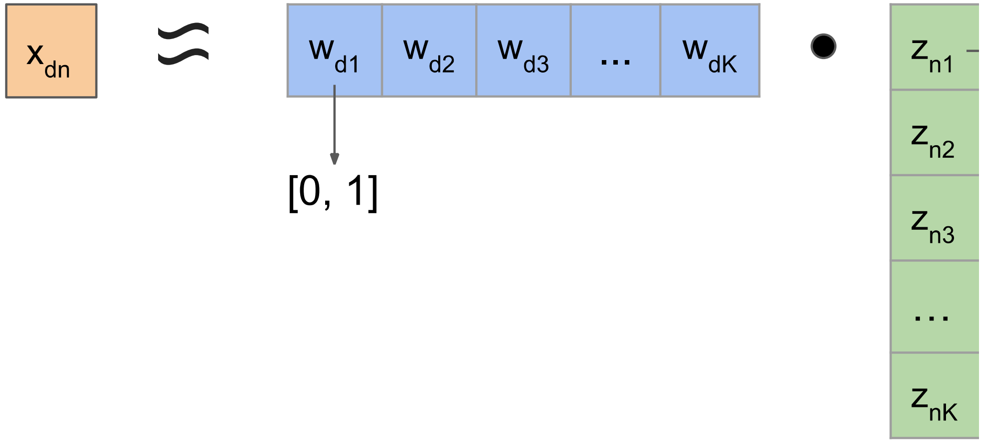
- Model the efficiency matrix as a low-rank matrix $\mathbf{X} \approx \mathbf{W} \cdot \mathbf{Z}^T$




- Latent factors: set of *skills* or *expertise* required for a department.

# Proposed Models: Expertise MF

- Model the efficiency matrix as a low-rank matrix $\mathbf{X} \approx \mathbf{W} \cdot \mathbf{Z}^T$




- Latent factors: set of *skills* or *expertise* required for a department.
  - Employees' latent factors (skills): from 0 (no ability) to 1 (proficiency)

# Proposed Models: Expertise MF

- Model the efficiency matrix as a low-rank matrix $\mathbf{X} \approx \mathbf{W} \cdot \mathbf{Z}^T$


- Latent factors: set of *skills* or *expertise* required for a department.
  - Employees' latent factors (skills): from 0 (no ability) to 1 (proficiency)
  - Departments' latent factors: non-negative and sum to 1
    - each department has a distribution of skills required to complete tasks in it

# Proposed Models: Survival MF

# Proposed Models: Survival MF

- Probabilistic approach

# Proposed Models: Survival MF

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$

# Proposed Models: Survival MF

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right]$$

# Proposed Models: Survival MF

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$
  - Each submitted claim has a random quality $Q_{dn}$

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right]$$

# Proposed Models: Survival MF

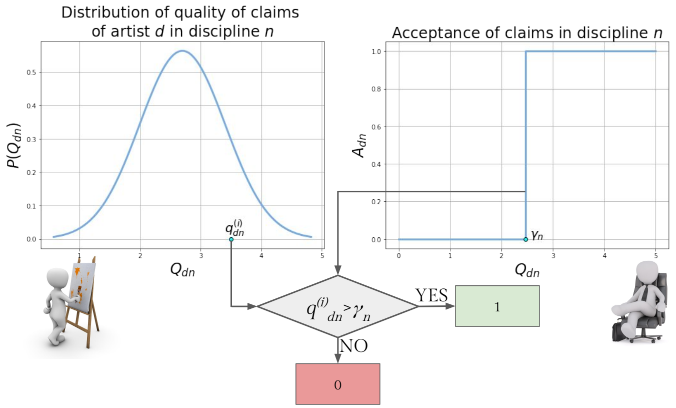
- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$
  - Each submitted claim has a random quality $Q_{dn}$
  - The manager of department $n$ has quality threshold $\gamma_n$

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right]$$

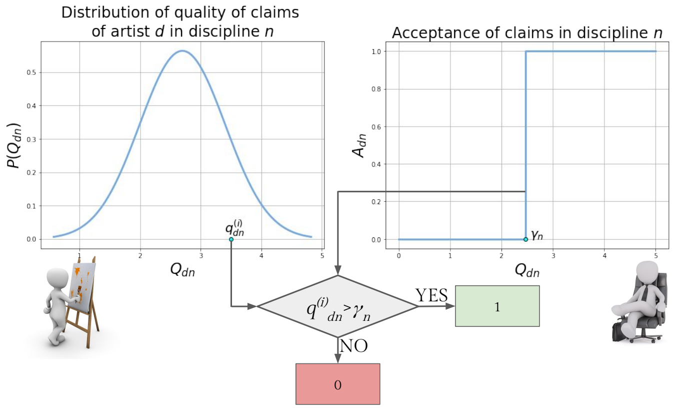
# Proposed Models: Survival MF

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$
  - Each submitted claim has a random quality $Q_{dn}$
  - The manager of department $n$ has quality threshold $\gamma_n$

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right]$$



# Proposed Models: Survival MF



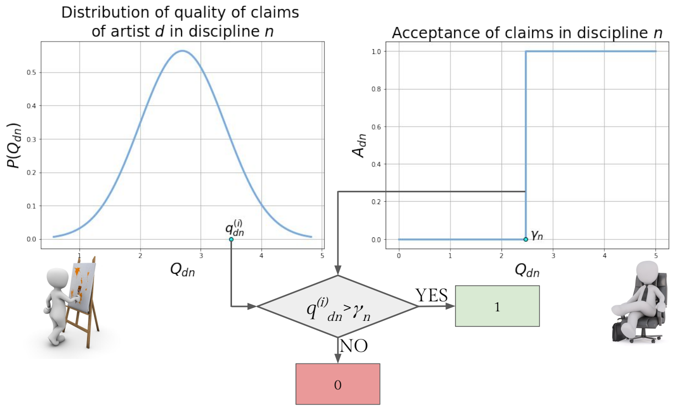

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$
  - Each submitted claim has a random quality $Q_{dn}$
  - The manager of department $n$ has quality threshold $\gamma_n$

$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right] = \mathbb{P}\left[Q_{dn} > \gamma_n\right] = S_{Q_{dn}}\left(\gamma_n\right) = \int_{\gamma_n}^{+\infty} f_{Q_{dn}}(u)\, du$$

  - $S_{Q_{dn}}\left(\gamma_n\right)$ is the survival function of $Q_{dn}$ at $\gamma_n$

# Proposed Models: Survival MF

- Probabilistic approach
  - Each claim acceptance is an i.i.d. Bernoulli variable $A_{dn}$
  - Each submitted claim has a random quality $Q_{dn}$
  - The manager of department $n$ has quality threshold $\gamma_n$




$$x_{dn} = \frac{\sum_{i=1}^{N_{dn}} a_{dn}^{(i)}}{N_{dn}} \approx \mathbb{E}\left[A_{dn}\right] = \mathbb{P}\left[A_{dn} = 1\right] = \mathbb{P}\left[Q_{dn} > \gamma_n\right] = S_{Q_{dn}}\left(\gamma_n\right) = \int_{\gamma_n}^{+\infty} f_{Q_{dn}}(u)\, du$$

  - $S_{Q_{dn}}\left(\gamma_n\right)$ is the survival function of $Q_{dn}$ at $\gamma_n$
  - Assume: Gaussian quality distribution $f_{Q_{dn}}$ with variance $\sigma^2$ and mean $\mu_{dn} \approx \mathbf{w}_d^T \cdot \mathbf{z}_n$

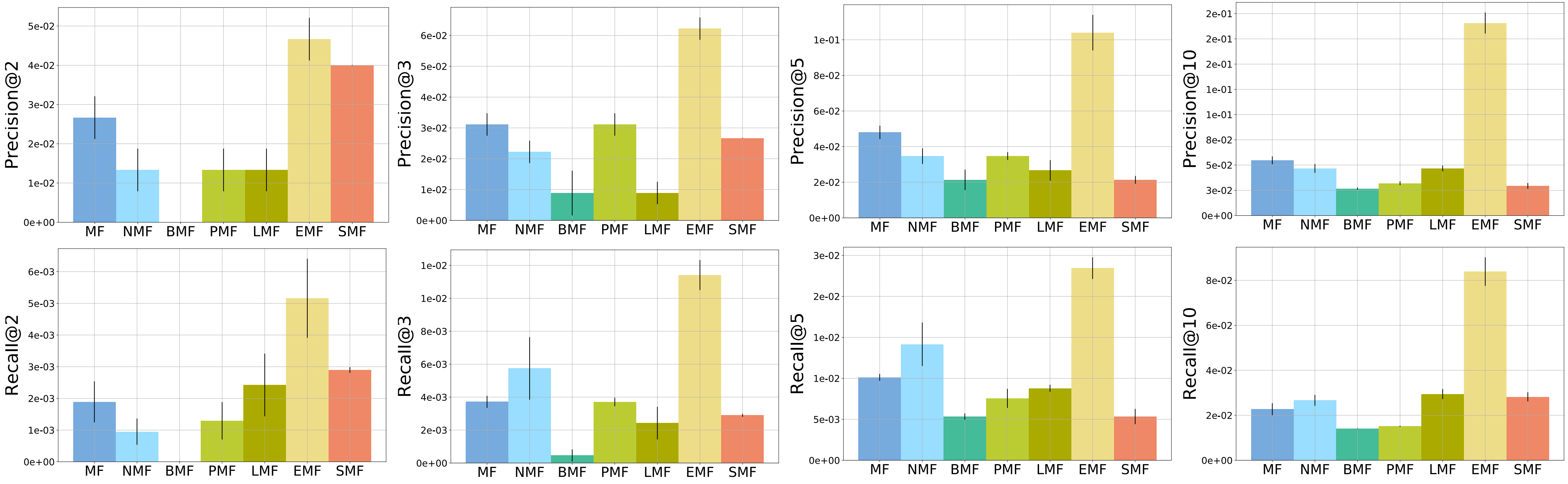
$$x_{dn} \approx \int_{\gamma_n}^{+\infty} \frac{1}{\sqrt{2\pi\sigma^2}} exp\left[-\frac{\left(u - \mathbf{w}_d^T \cdot \mathbf{z}_n\right)^2}{2\sigma^2}\right] du$$

# Experiments: Methodology

- Datasets
  - Movie Production [In-house]
  - Over-The-Top [In-house]
  - Click-Through Rate [Public – Outbrain Click Prediction competition]
- Evaluation metrics
  - Precision@$N$
  - Recall@$N$
- Baselines
  - MF: Matrix Factorization
  - NMF: Nonnegative Matrix Factorization
  - BMF: Bounded Matrix Factorization [Kannan et al., 2014]
  - PMF: Probabilistic Matrix Factorization [Mnih and Salakhutdinov, 2008]
  - LMF: Logistic Matrix Factorization [Johnson, 2014]

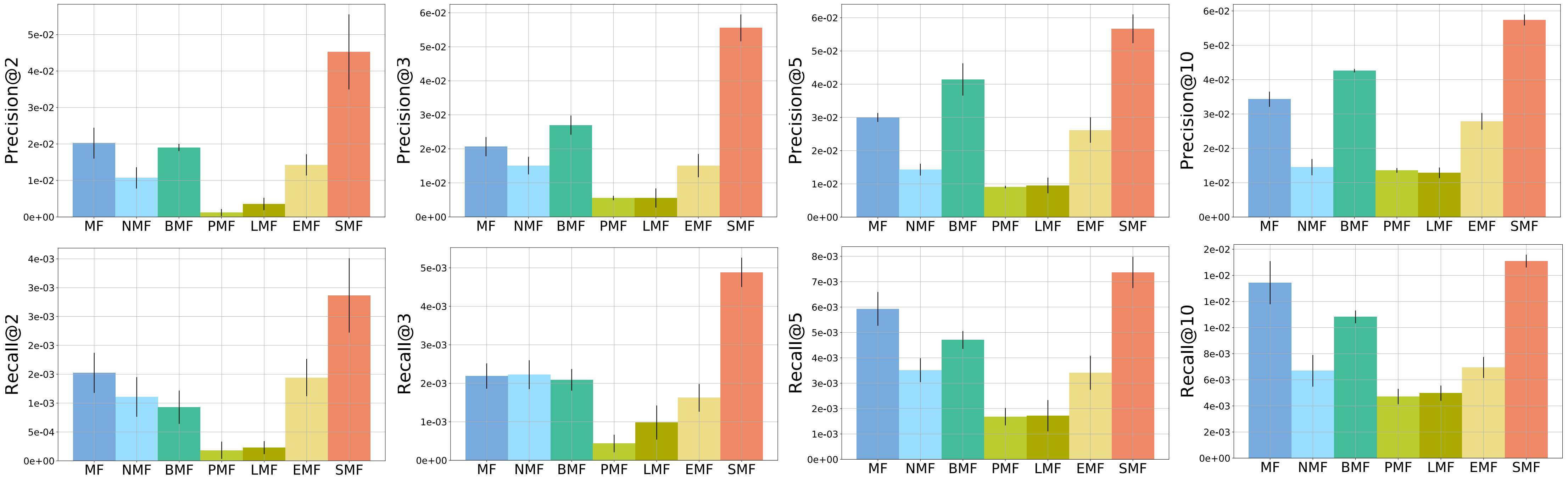
# Experiments: Movie Production Data

- Recommend employees to departments according to their efficiencies
- $\mathbf{X}$ : $312 \times 25$ – $86.85\,\%$ sparse – *very few ratings per user*



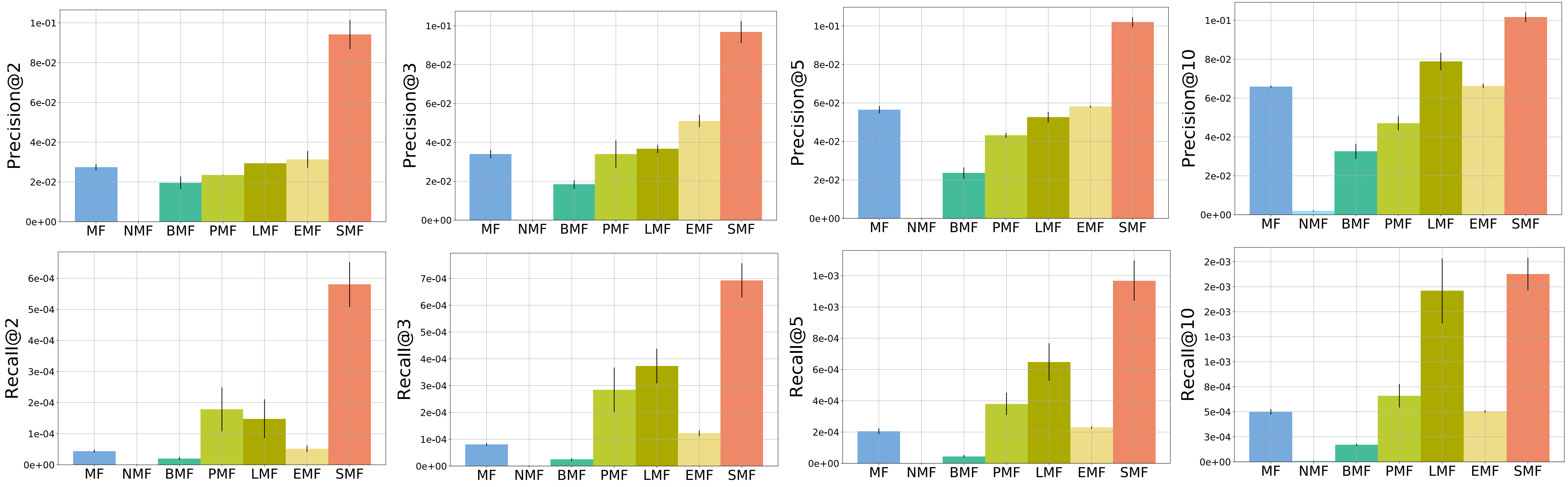
# Experiments: Over-The-Top Data

- Recommend apps to users according to their watching rates
- $\mathbf{X}$ : $934 \times 140$ – $99.91\ \%$ sparse



# Experiments: Click-Through Rate Data

- Recommend website categories to ad placer to maximize click-through rate
- $\mathbf{X} : 15647 \times 85 - 79.99\,\%$ sparse



# Experiments: Discussion

# Experiments: Discussion

- EMF outperforms every method in the Movie Production dataset ($> 3\times$)
  - The hypotheses of EMF match the Movie Production framework, not the others
  - EMF outperforms PMF in a dataset where users have very few ratings (max 7)

# Experiments: Discussion

- EMF outperforms every method in the Movie Production dataset ($> 3\times$)
  - The hypotheses of EMF match the Movie Production framework, not the others
  - EMF outperforms PMF in a dataset where users have very few ratings (max 7)

- SMF is a general model, outperforming every method in the other datasets
  - SMF underperforms in the Movie Production dataset due to very few available entries rather than due to sparsity (Over-The-Top data is much sparser)

# Conclusion

- EMF and SMF: Two novel structured MF techniques for entries in [0, 1]
  - They give better recommendations than popular MF techniques
  - It is beneficial to explicitly model entries in [0, 1]

- SMF is the simplest in a class of probabilistic models
  - Here, we assumed
    - Normally-distributed quality of work
    - Managers modeled by a single threshold

Poster #663

Experimental Results: https://github.com/e-bug/unit-mf

# Conclusion

Thank you!

- EMF and SMF: Two novel structured MF techniques for entries in [0, 1]
  - They give better recommendations than popular MF techniques
  - It is beneficial to explicitly model entries in [0, 1]

- SMF is the simplest in a class of probabilistic models
  - Here, we assumed
    - Normally-distributed quality of work
    - Managers modeled by a single threshold

Poster #663

Experimental Results: https://github.com/e-bug/unit-mf