Leverage **Item Popularity** and **Recommendation Quality** via Cost-sensitive Factorization Machines

Chih-Ming Chen[*,†], Hsin-Ping Chen[*], Ming-Feng Tsai[*] and Yi-Hsuan Yang[†]
[*] Department of Computer Science, National Chengchi University, Taiwan
[†] Research Center for Information Technology Innovation, Academia Sinica, Taiwan
Email: cmchen@citi.sinica.edu.tw, s10019@cs.nccu.edu.tw, mftsai@cs.nccu.edu.tw, yang@citi.sinica.edu.tw

Highlights

- A Cost-sensitive Factorization Machine (FM)
- Two cost functions
- Experiments are conducted on a real-world dataset.

Recommendation Problem

Music Dataset Given a user's listening history

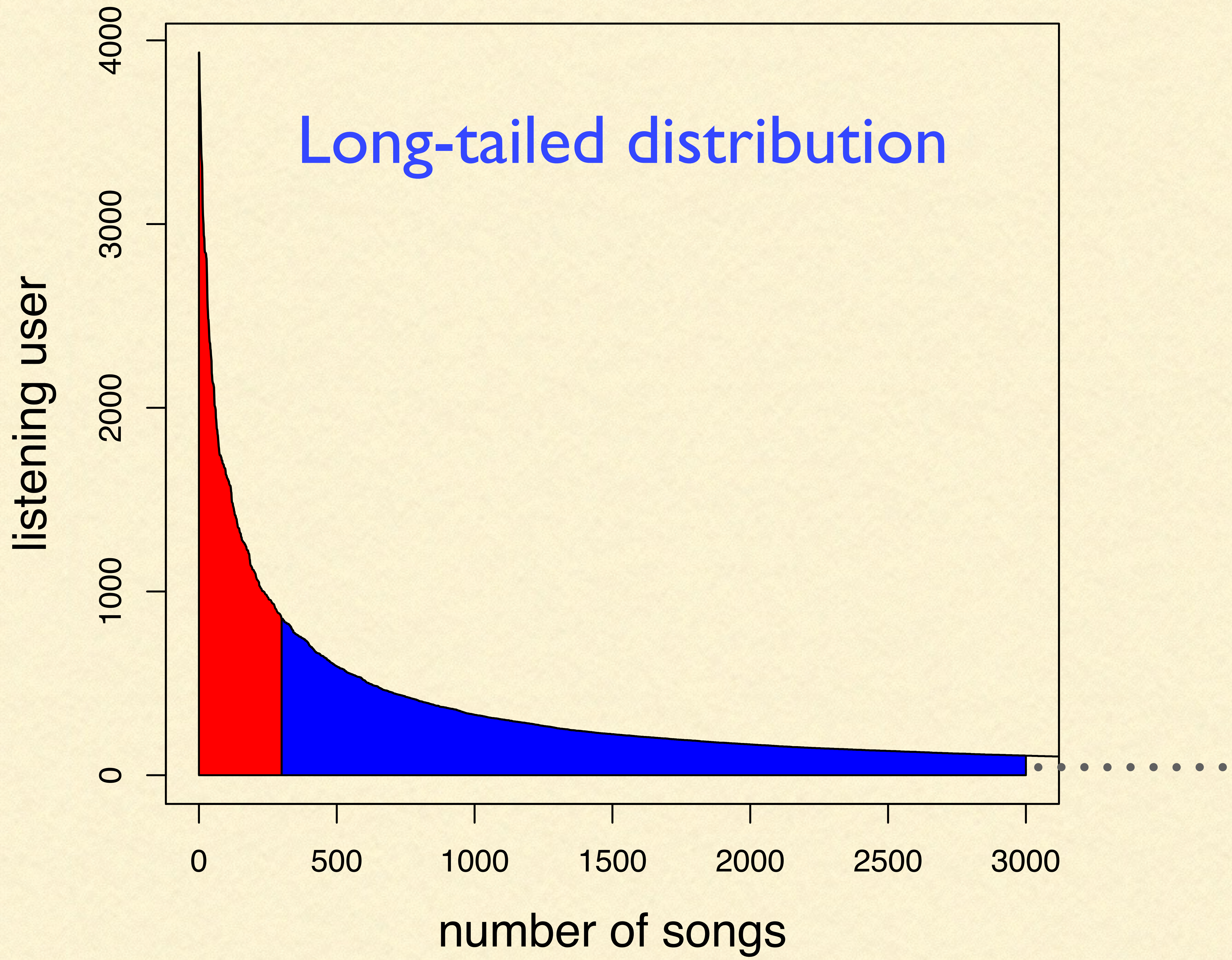
Goal Generate a playlist (top-N recommendations)

Classical Problem in Recommendation

- To get a sensible performance,
  a traditional recommendation
  algorithm tends to return
  popular songs

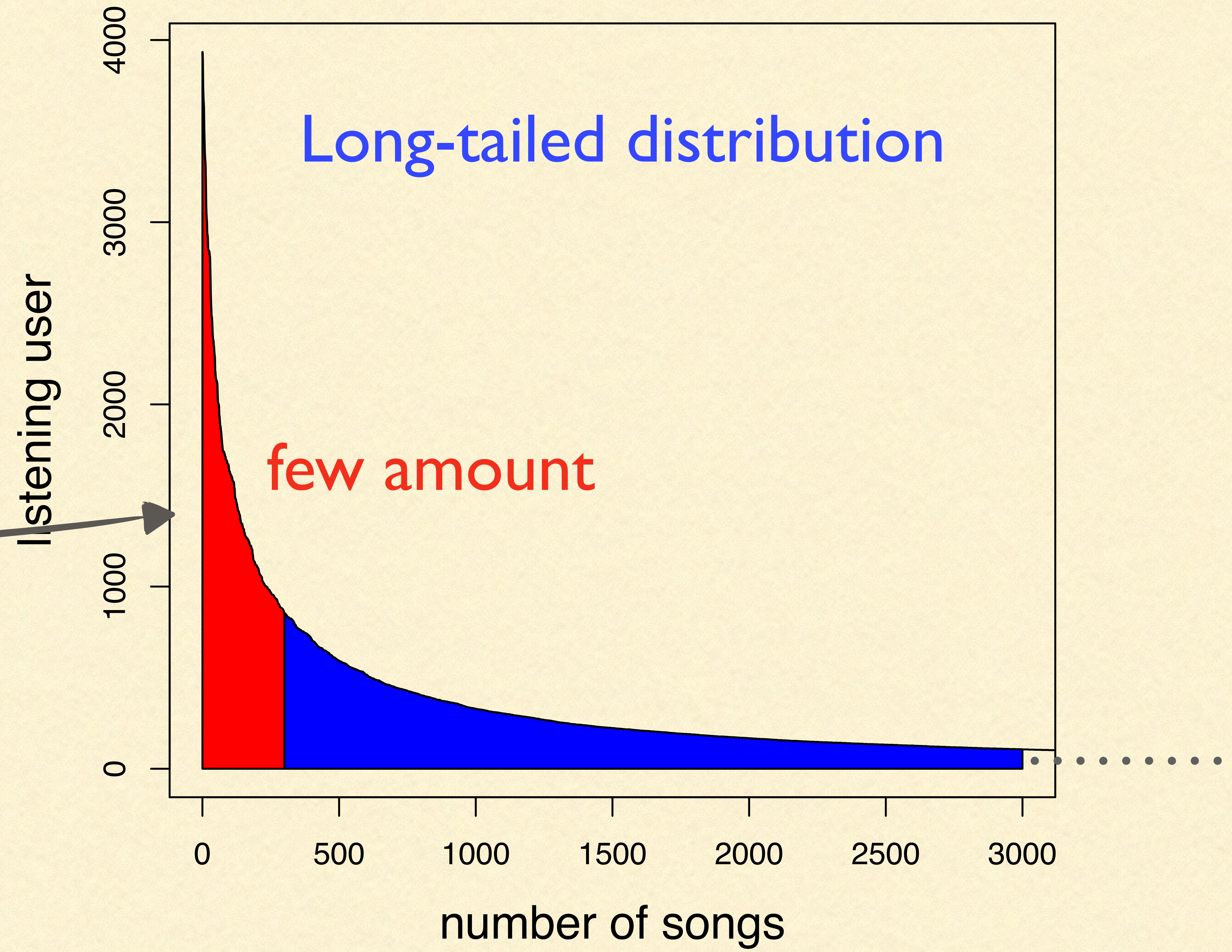
Classical Problem in Recommendation

- To get a sensible performance, a traditional recommendation algorithm tends to return **popular** songs



Classical Problem in Recommendation

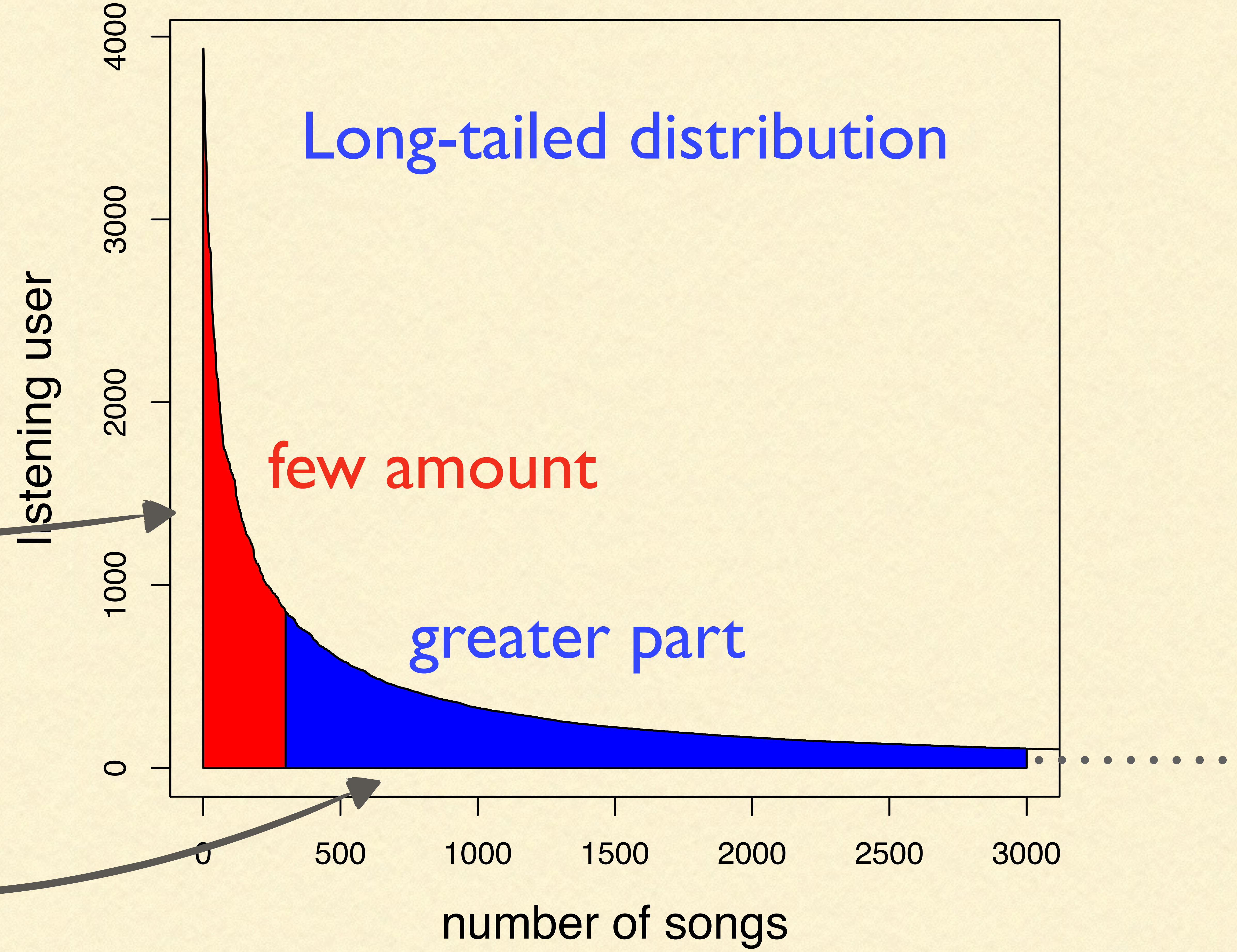
- To get a sensible performance, a traditional recommendation algorithm tends to return **popular** songs



Classical Problem in Recommendation

- To get a sensible performance, a traditional recommendation algorithm tends to return **popular** songs

unpopular music



The Goal of Recommendation

- To help users
  - discover novel music
  - recall the music users have listened to
  - …

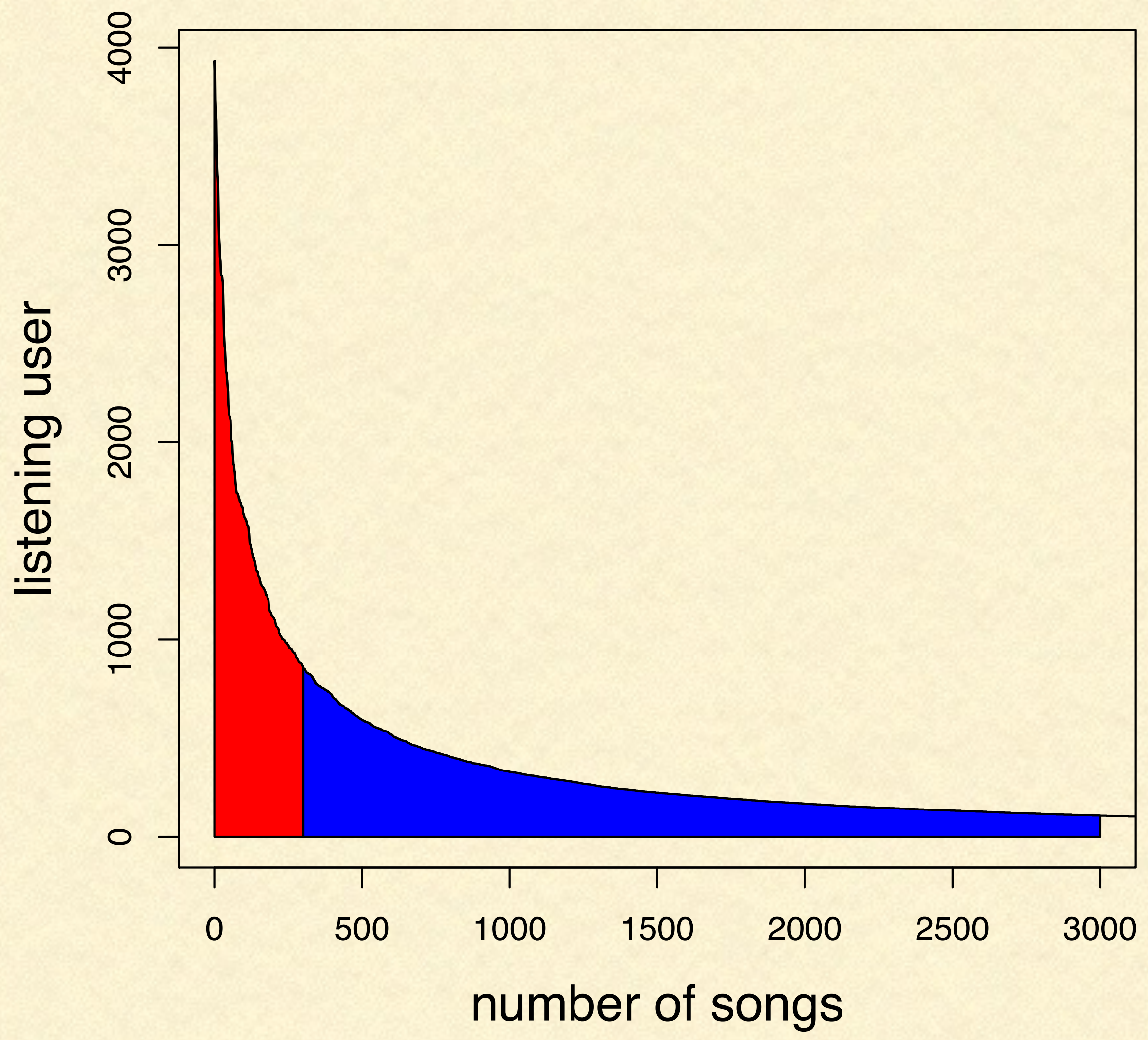
The Goal of Recommendation

- To help users
  - discover novel music
  - recall the music users have listened to

- Our goal
  - explore more novel music as much as possible

The Goal of Recommendation

- To help users
  - discover novel music
  - recall the music users have listened to



- Our goal
  - explore more novel music as much as possible

— which usually leans to **unpopular** music

The Goal of Recommendation

- To help users
  - discover novel music
  - recall the music users have listened to

It's a trade-off between
Item Popularity and Recommendation Quality.

- Our goal
  - explore more novel music as much as possible

— which usually leans to **unpopular** music

The Goal of Recommendation

- To help users
  - discover novel music
  - recall the music users have listened to

It's a trade-off between
Item Popularity and Recommendation Quality.

- Our goal
  - explore more novel music as much as possible
  - **keep a reasonable performance at the same time**

Problem — Music Recommendation

Music Dataset Given a user's listening history

Goal Generate a playlist

Satisfy

1. Explore novel music as much as possible
2. Keep a Reasonable Performance

Highlights

- A Cost-sensitive Factorization Machine (FM)
- Two cost functions
- Experiments are conducted on a real-world dataset.

Highlights

- A Cost-sensitive Factorization Machine (FM)

Cost-sensitive Learning + FM

- Two cost functions
- Experiments are conducted on a real-world dataset.

Recommendation Algorithm

- **Factorization Machines** [Rendle 2012]

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{j=1}^{p} w_j x_j + \sum_{j=1}^{p} \sum_{j'=j+1}^{p} x_j x_{j'} \sum_{f=1}^{k} v_{j,f}\, v_{j',f}$$

Recommendation Algorithm

- **Factorization Machines** [Rendle 2012]

$$\hat{y}(\mathbf{x}) := \underbrace{\boxed{w_0 + \sum_{j=1}^{p} w_j x_j}}_{\text{Linear Regression}} + \sum_{j=1}^{p} \sum_{j'=j+1}^{p} x_j x_{j'} \sum_{f=1}^{k} v_{j,f}\, v_{j',f}$$

Recommendation Algorithm

- **Factorization Machines** [Rendle 2012]

$$\hat{y}(\mathbf{x}) := \underbrace{\boxed{w_0 + \sum_{j=1}^{p} w_j x_j}}_{\text{Linear Regression}} + \underbrace{\boxed{\sum_{j=1}^{p} \sum_{j'=j+1}^{p} x_j x_{j'} \sum_{f=1}^{k} v_{j,f}\, v_{j',f}}}_{\text{Matrix Factorization / Latent Topic}}$$

Objective Function - Coordinate Descent

$$OPT(S) = \underset{\Theta}{\operatorname{argmin}} \sum_{(\mathbf{x},y)\in S} l(\hat{y}(x|\Theta), y) + \sum_{\theta\in\Theta} \lambda_\theta \theta^2$$

model parameter

minimize the entire loss

regularization term

Objective Function - Coordinate Descent

$$OPT(S) = \underset{\Theta}{\operatorname{argmin}} \sum_{(\mathbf{x},y)\in S} l(\hat{y}(x|\Theta), y) + \sum_{\theta\in\Theta} \lambda_\theta \theta^2$$

model parameter

minimize the entire loss

regularization term

+ Cost-Sensitive Learning

Traditional Cost

	prediction +1	prediction -1
actual +1	no error	false reject
actual -1	false accept	no error

	+1	-1
+1	0	1
-1	1	0

Cost-Sensitive Learning

prediction

actual	+1	-1
+1	no error	false reject
-1	false accept	no error

	+1	-1
+1	0	1
-1	1	0

	+1	-1
+1	0	100
-1	1	0

Coordinate Descent with Cost-Sensitive Learning

$$OPT(S) = \underset{\Theta}{\operatorname{argmin}} \sum_{(\mathbf{x},y)\in S} l(\hat{y}(x|\Theta), y) + \sum_{\theta\in\Theta} \lambda_\theta \theta^2$$

$$l_c(\hat{y}, y) = c_{\mathbf{x}}(\hat{y} - y)^2 \rightarrow \text{RMSE}$$

each record has its own penalty value

Instance-level Cost-Sensitive Learning

$$OPT(S) = \operatorname*{argmin}_{\Theta} \sum_{(\mathbf{x},y)\in S} l(\hat{y}(x|\Theta), y) + \sum_{\theta\in\Theta} \lambda_\theta \theta^2$$

$$\downarrow$$

$$l_c(\hat{y}, y) = c_{\mathbf{x}}(\hat{y} - y)^2$$

$$\downarrow$$

$$\frac{\partial}{\partial\theta} OPT(S) = \sum_{(\mathbf{x},y)\in S} 2(\hat{y}(x) - y) h_\theta(x) c_{\mathbf{x}} + 2\lambda_\theta \theta$$

$$\theta^* = -\frac{\sum_{(\mathbf{x},y)\in S} (g_{(\theta)}(\mathbf{x}) - y) h_\theta(x) c_{\mathbf{x}}}{\sum_{(\mathbf{x},y)\in S} h_\theta^2(x) c_{\mathbf{x}} + \lambda_\theta}$$

Highlights

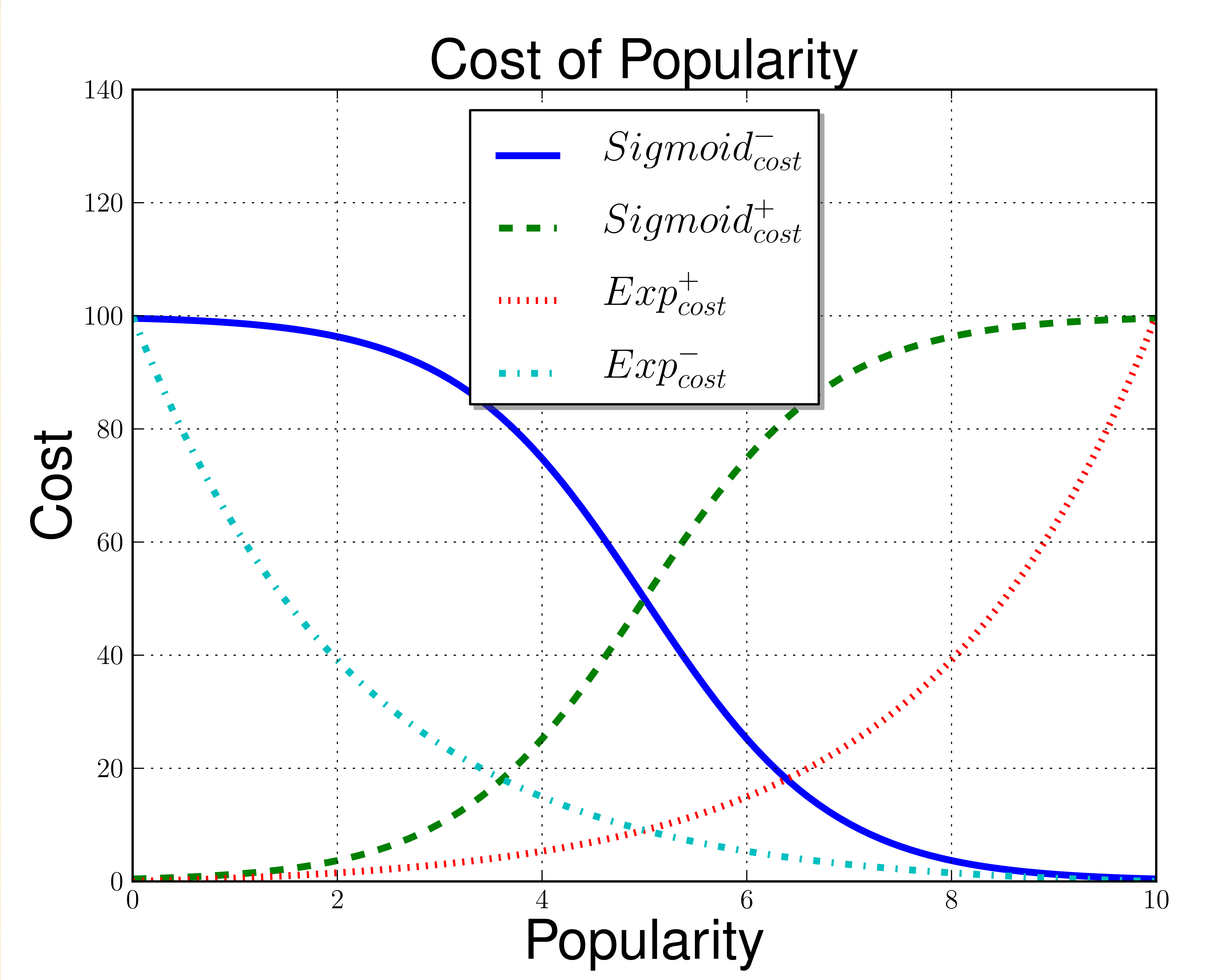
- A Cost-sensitive Factorization Machine (FM)
- Two cost functions
- Experiments are conducted on a real-world dataset.

Cost Function

$$\frac{\partial}{\partial \theta} OPT(S) = \sum_{(\mathbf{x},y)\in S} 2(\hat{y}(x) - y) h_\theta(x) c_{\mathbf{x}} + 2\lambda_\theta \theta$$

$$Sigmoid_{cost}(p_{\mathbf{x}}) = \frac{c_{\max}}{1 + e^{\pm(4e\frac{p_{\mathbf{x}}}{p_{\max}} - 2e)}}$$

$$Exp_{cost}(p_{\mathbf{x}}) = e^{\pm(\frac{p_{\mathbf{x}}}{p_{\max}}) \log w_{\max}}$$

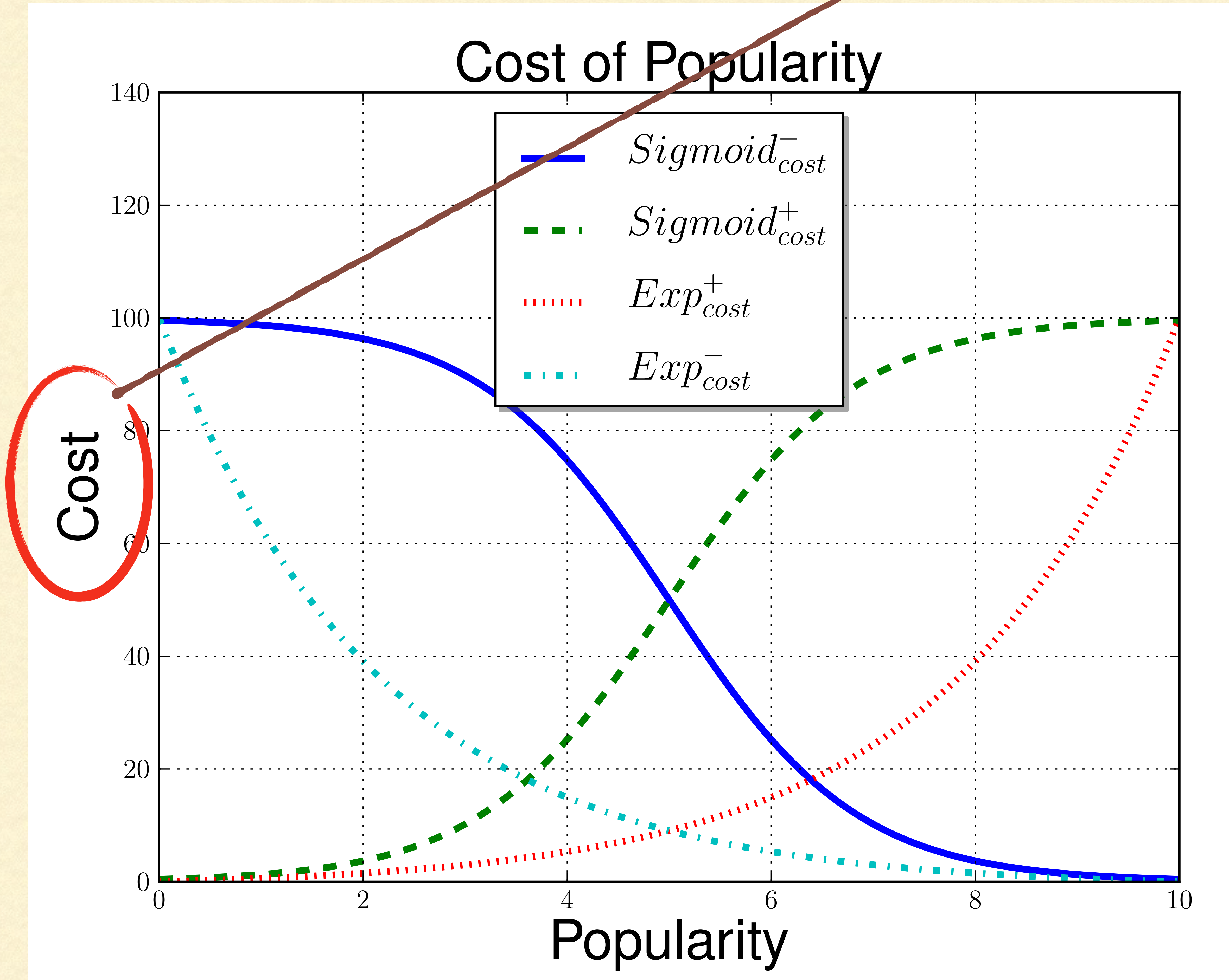


Cost Function

$$\frac{\partial}{\partial\theta} OPT(S) = \sum_{(\mathbf{x},y)\in S} 2(\hat{y}(x) - y)h_\theta(x)c_{\mathbf{x}} + 2\lambda_\theta\theta$$

$$Sigmoid_{cost}(p_{\mathbf{x}}) = \frac{c_{\max}}{1 + e^{\pm(4e\frac{p_{\mathbf{x}}}{p_{\max}} - 2e)}}$$

$$Exp_{cost}(p_{\mathbf{x}}) = e^{\pm(\frac{p_{\mathbf{x}}}{p_{\max}})\log w_{\max}}$$

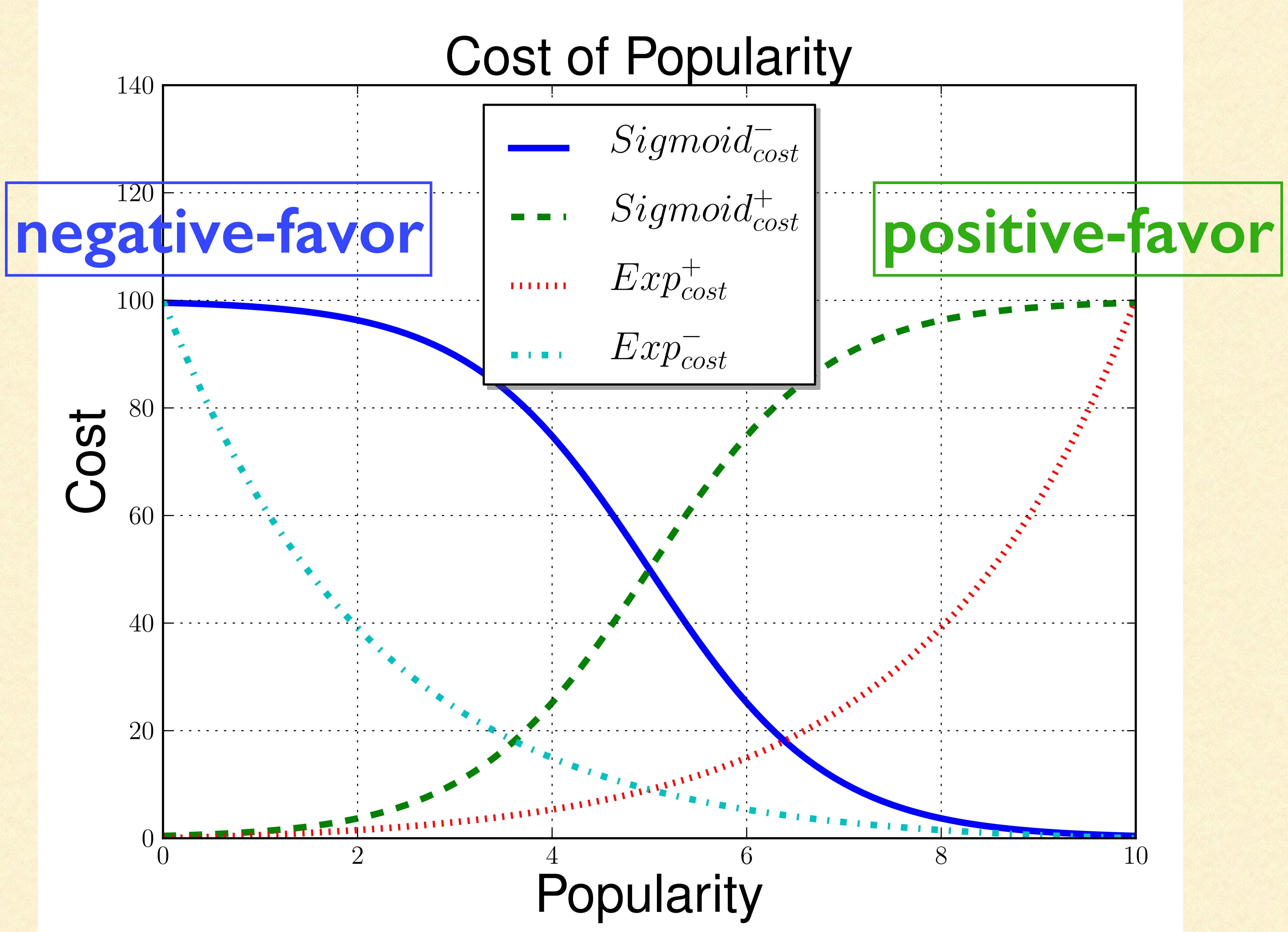


Cost Function

$$\frac{\partial}{\partial\theta} OPT(S) = \sum_{(\mathbf{x},y)\in S} 2(\hat{y}(x) - y)h_\theta(x)c_{\mathbf{x}} + 2\lambda_\theta\theta$$

$$Sigmoid_{cost}(p_{\mathbf{x}}) = \frac{c_{\max}}{1 + e^{\pm(4e\frac{p_{\mathbf{x}}}{p_{\max}} - 2e)}}$$

$$Exp_{cost}(p_{\mathbf{x}}) = e^{\pm(\frac{p_{\mathbf{x}}}{p_{\max}})\log w_{\max}}$$



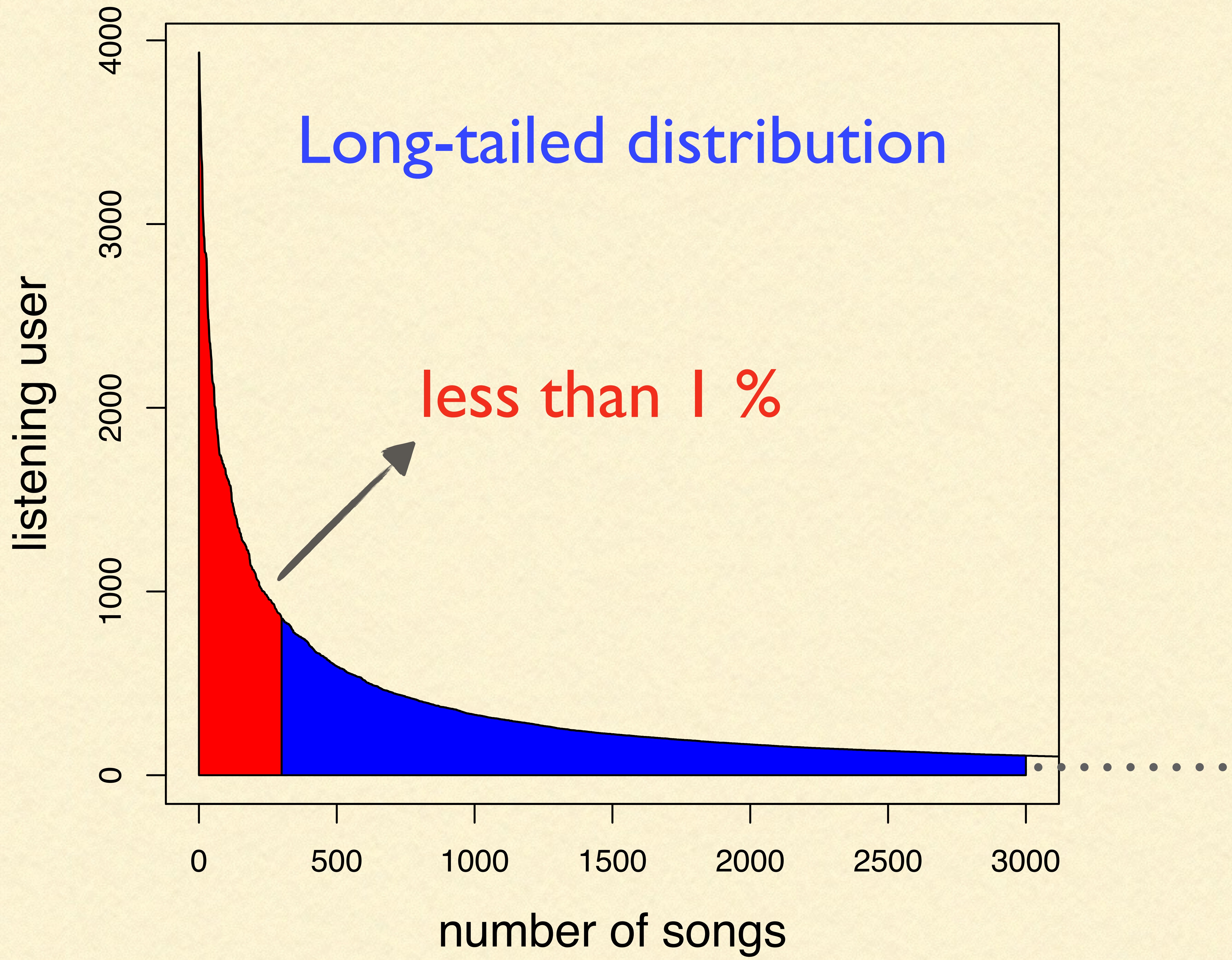
Highlights

- A Cost-sensitive Factorization Machine (FM)
- Two cost functions
- Experiments are conducted on a real-world dataset.

Dataset

KKBOX-5K

- 5,000 users
- 30,000 songs
- 1,800,000 listening records



Performance Measurement

Satisfy

1. Explore novel music as much as possible
2. Keep a Reasonable Performance

Performance Measurement

Satisfy

1. Explore novel music as much as possible

↓

Average Popularity

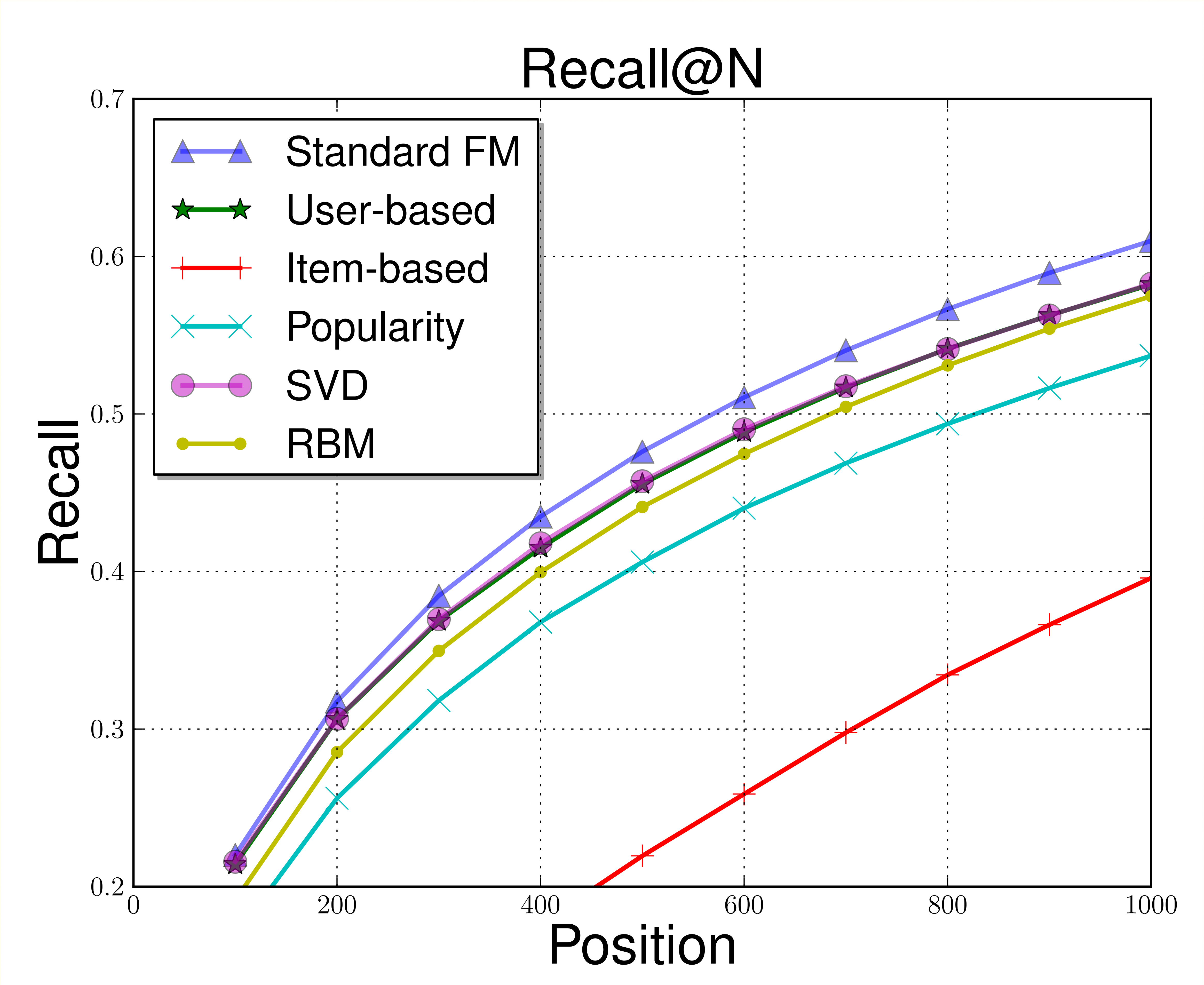
(number of listening users)

2. Keep a Reasonable Performance

↓

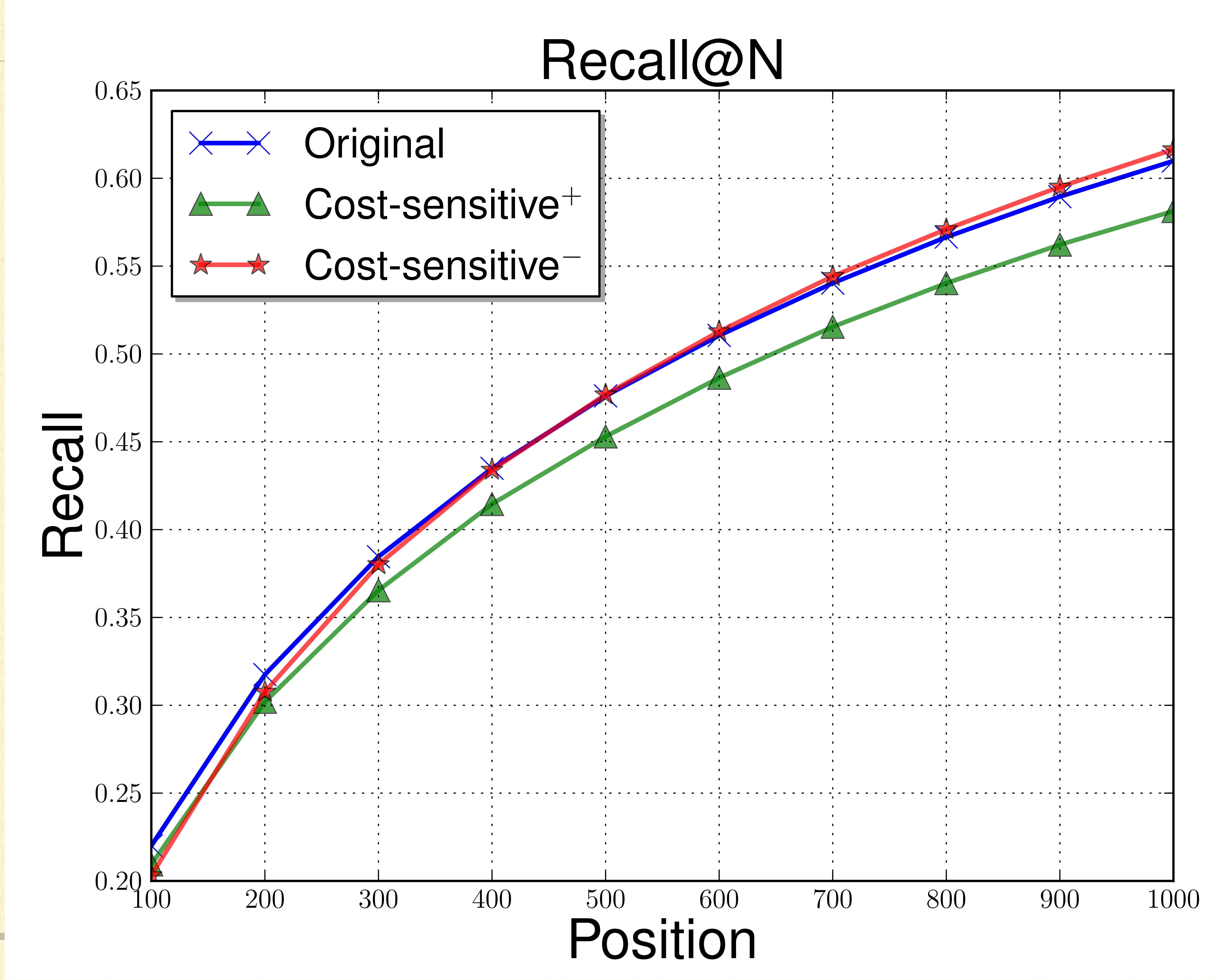
Recall

Performance Comparison **w/o** Cost-Sensitive

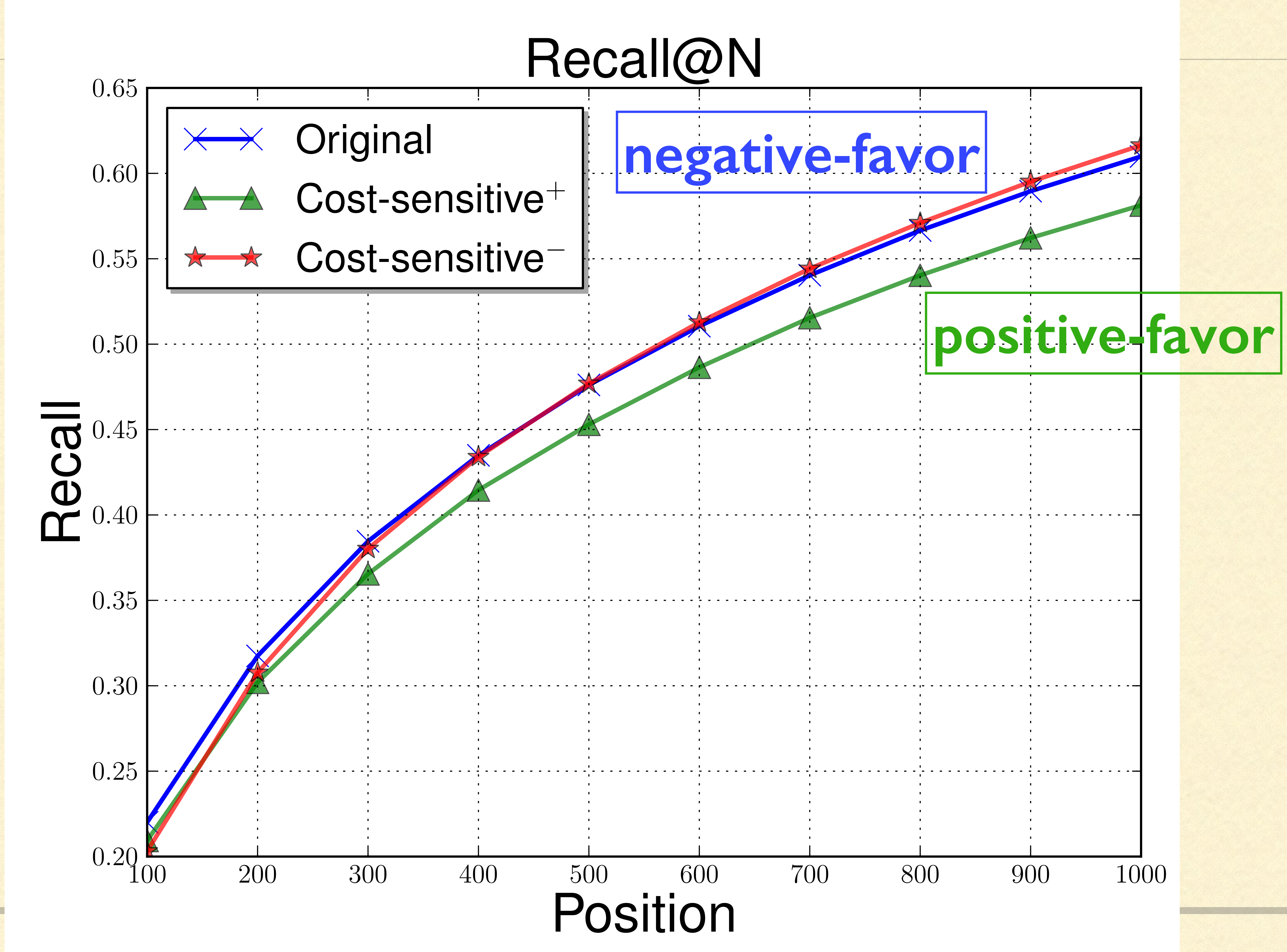


- Recommending only popular songs can obtain a feasible performance, which means users usually tend to listen to popular music

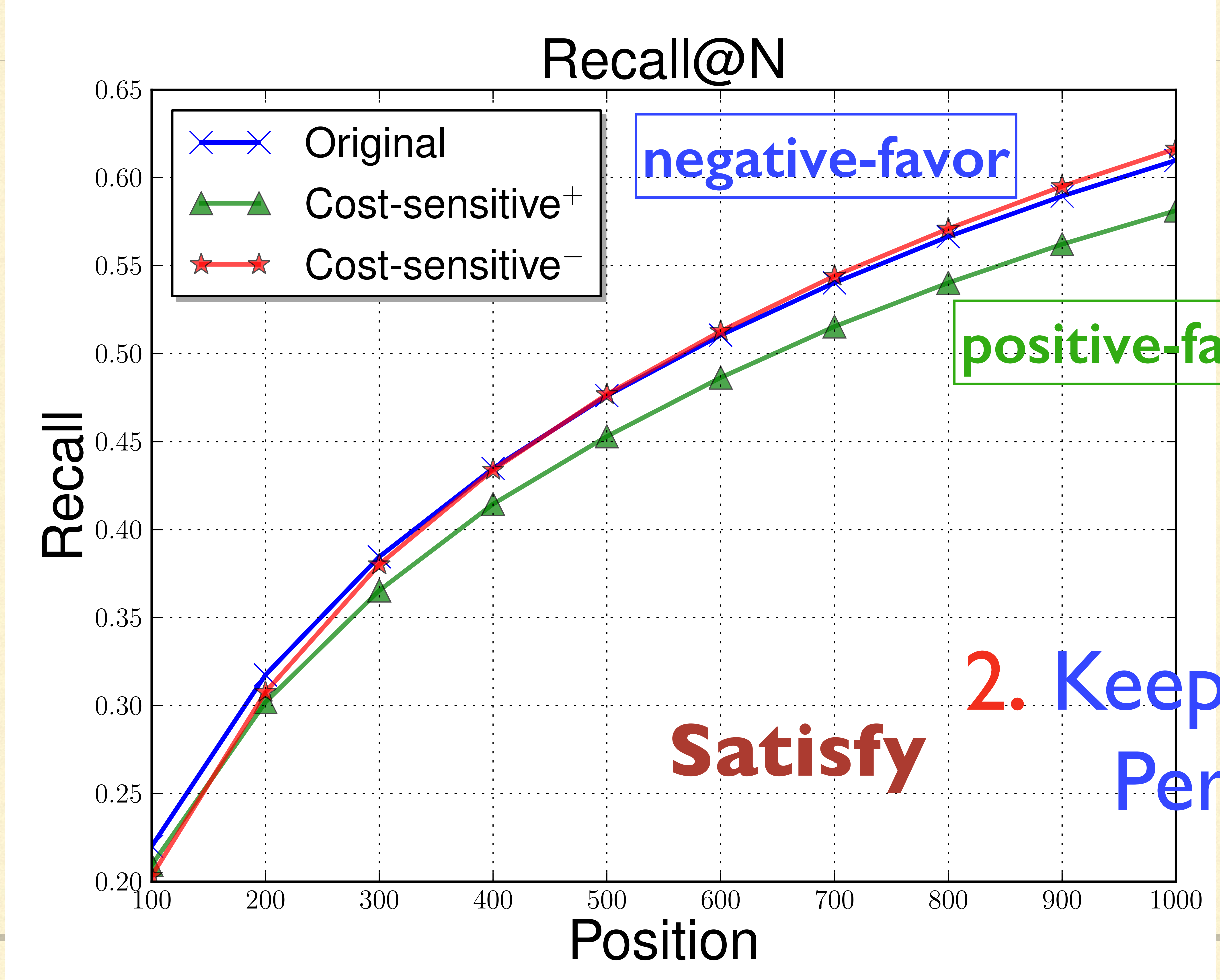
Original FM v.s. Cost-Sensitive FM



Original FM v.s. Cost-Sensitive FM

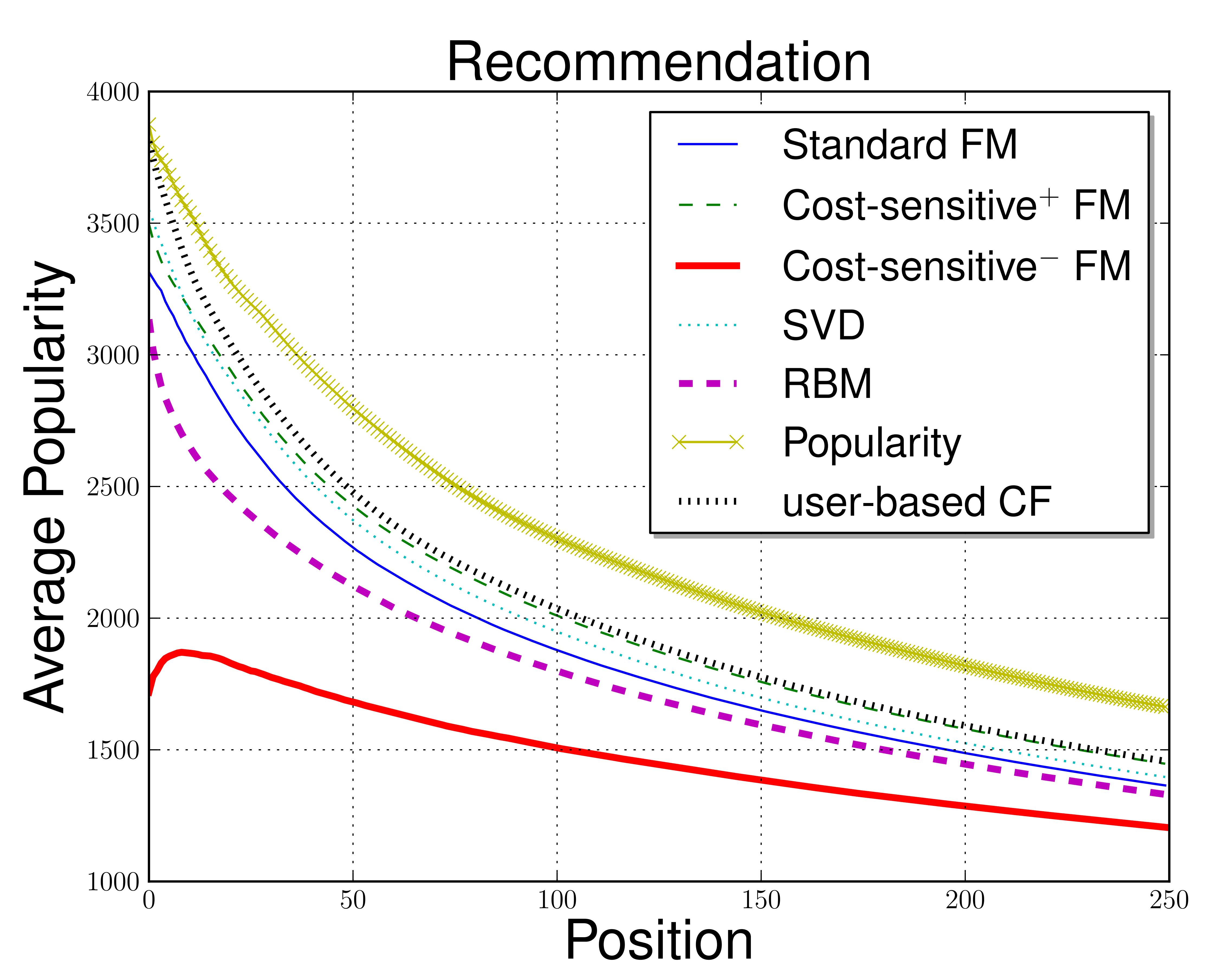


Original FM v.s. Cost-Sensitive FM



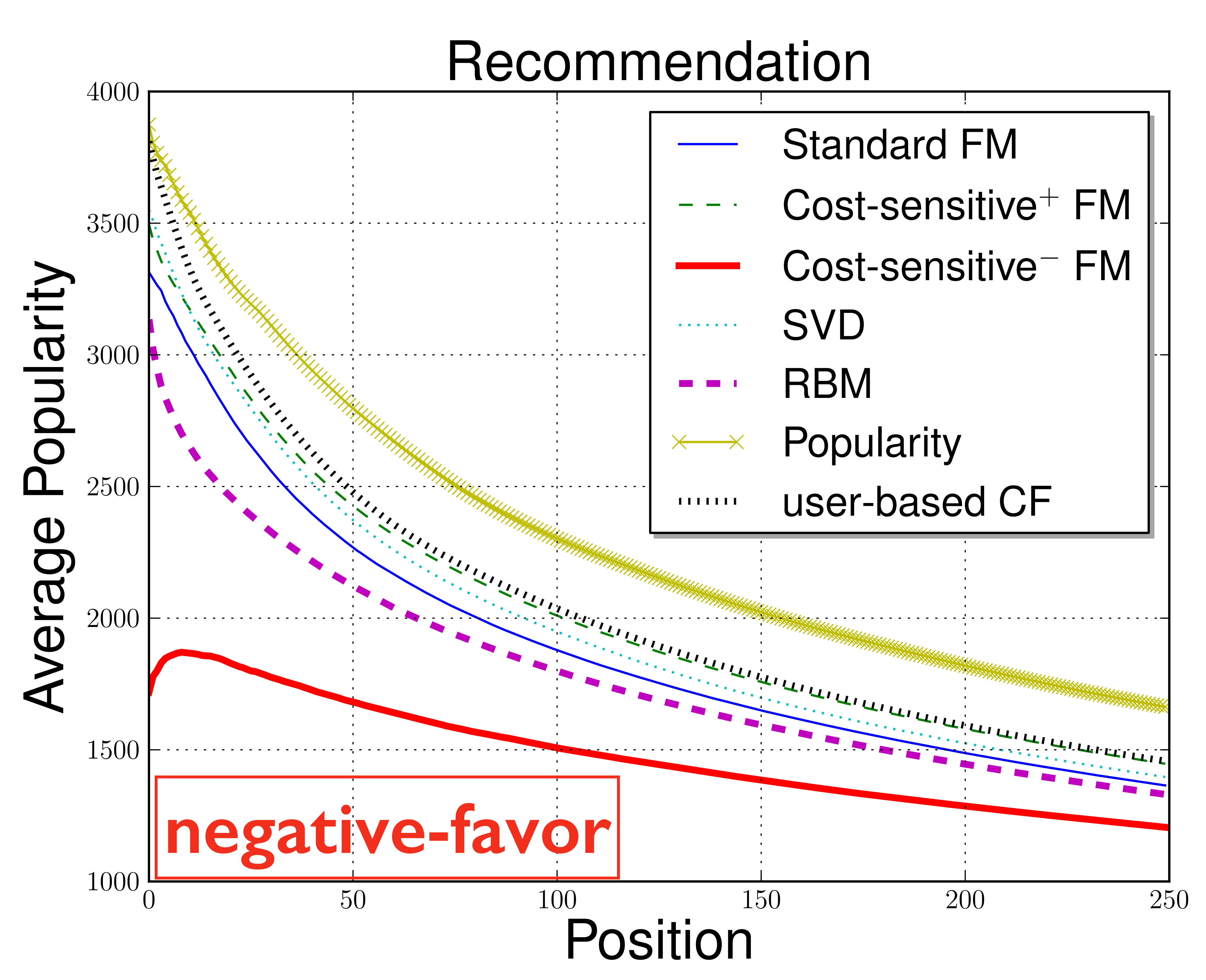
2. Keep a Reasonable Performance

Average Popularity of Top-N Recommendations



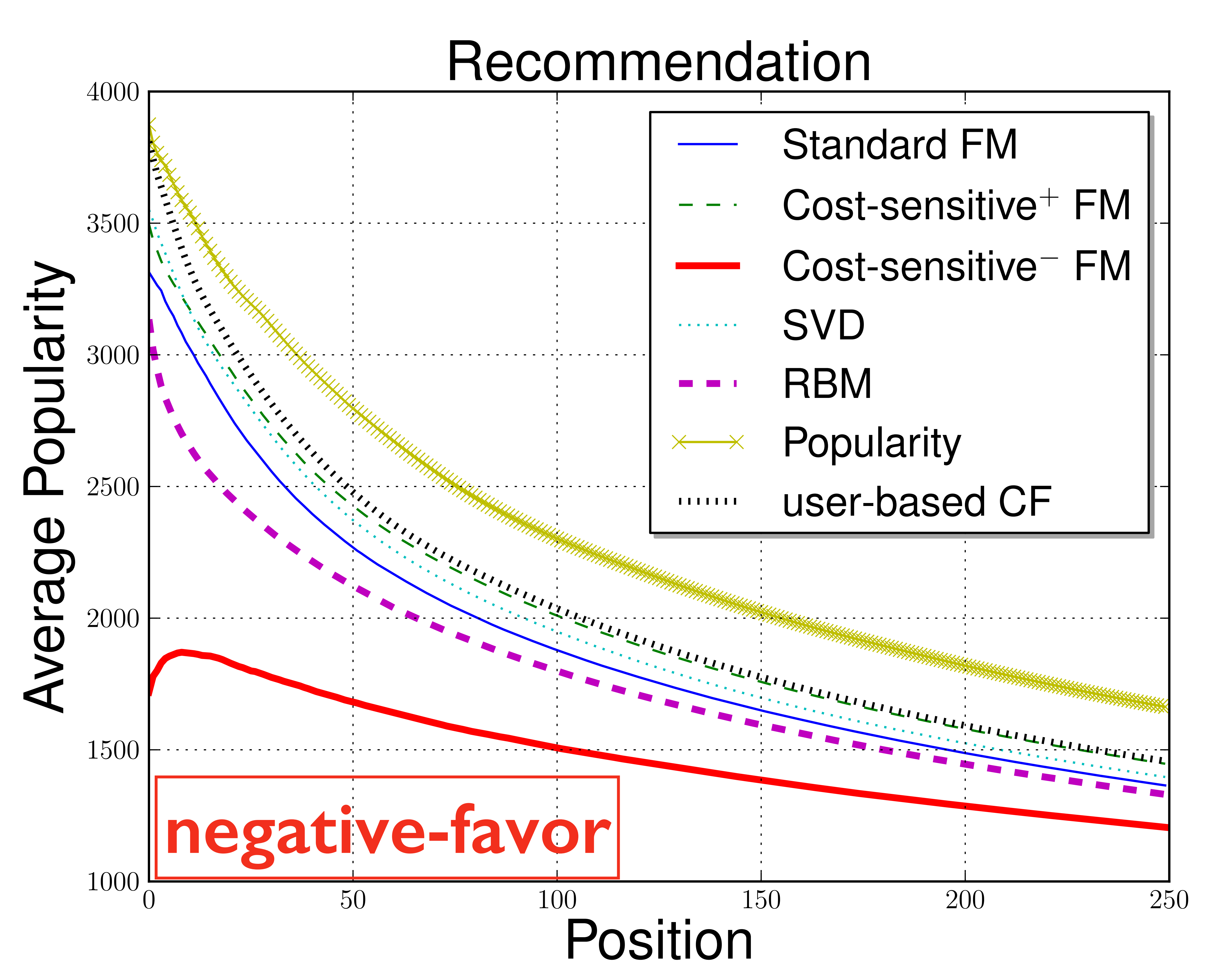
- Negative-Favour cost-sensitive FM could receive lower average popularity among top-N recommendations

Average Popularity of Top-N Recommendations



- Negative-Favour cost-sensitive FM could receive lower average popularity among top-N recommendations

Average Popularity of Top-N Recommendations



- Negative-Favour cost-sensitive FM could receive lower average popularity among top-N recommendations

Satisfy

1. Explore novel music as much as possible

Conclusion

- This is a preliminary research about cost-sensitive learning for recommendation problem and FM model.
- The experimental result shows that the proposed instance-level cost-sensitive FM could achieve the different goals with different weighting functions.

Any Question?

– *ChihMing*

chagnecandy at gmail.com