Learning to Rank Social Update Streams

Liangjie Hong*, Lehigh University
Ron Bekkerman, LinkedIn Corp.
Joseph Adler, LinkedIn Corp.
Brian D. Davison, Lehigh University

1

* Part of this work was done when the author was on an internship at LinkedIn Corp.

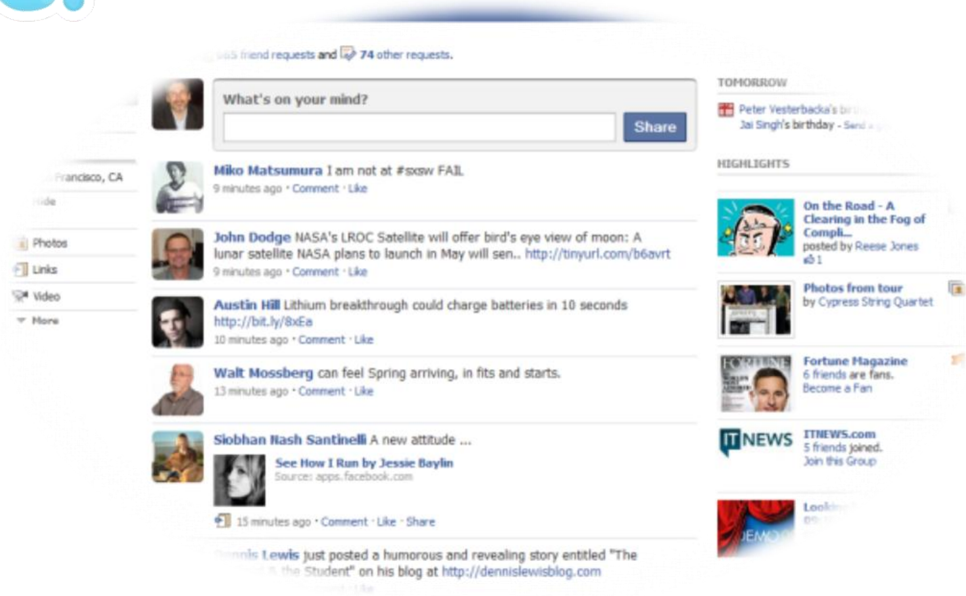
Overview

- Social Update Streams
- Overview of LinkedIn
- Social Stream Ranking & Dataset
- Methods
- Experiments
- Conclusion

ONE SLIDE TAKEAWAY

- Task
  - Improve user engagement by re-ranking social updates
- Main results
  - We demonstrate that recommender systems + preference-based learning can be used to re-rank social updates.
  - A linear model can achieve 60% of the performance of latent factor models, on average.
  - A tensor factorization model with regression on explicit features works the best.
  - The cold-start problem makes it impossible to model some kinds of interactions.

Social Update Streams



Social Update Streams

Problems?

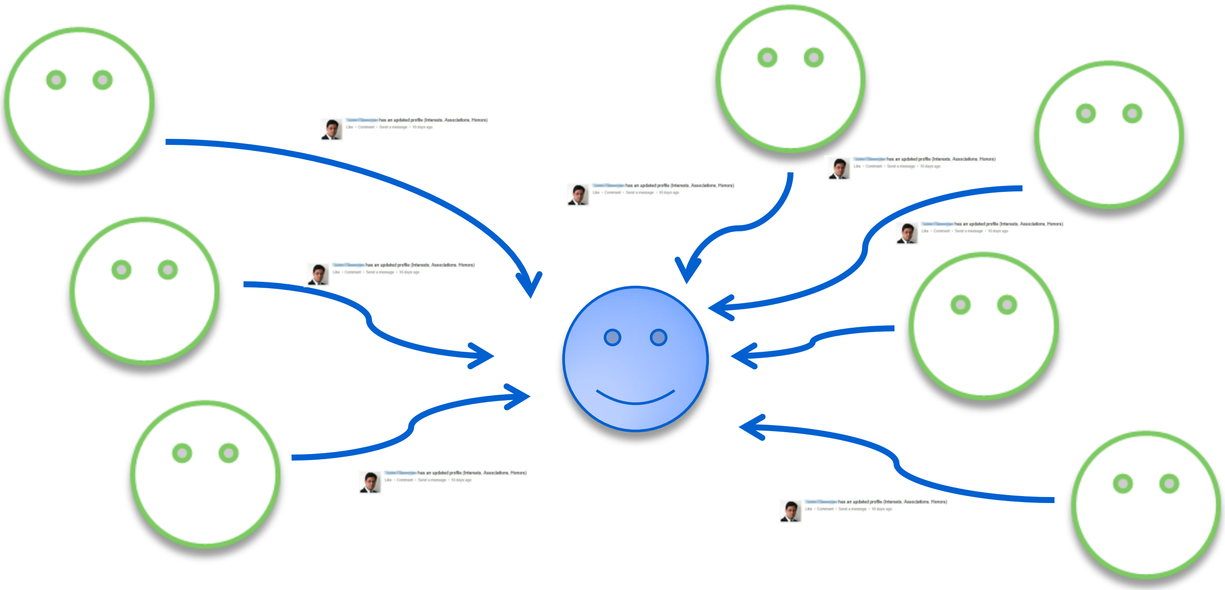
Social Update Streams

Information overload

Social Update Streams

Information shortage

Social Update Streams



Overview of LinkedIn

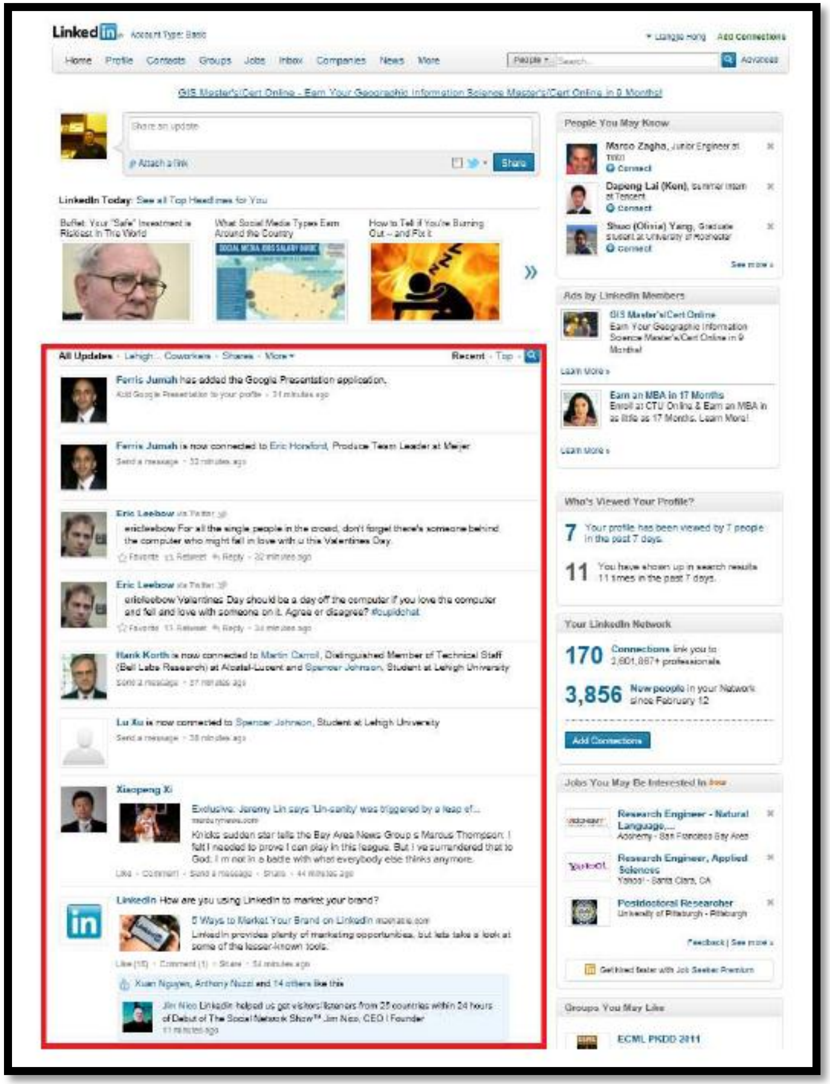
Overview of LinkedIn

- Founded in Dec. 2002, launched in May 2003
- 160M[1] users in 200 countries and territories
- Biggest social network for professionals

[1] As of March 2012

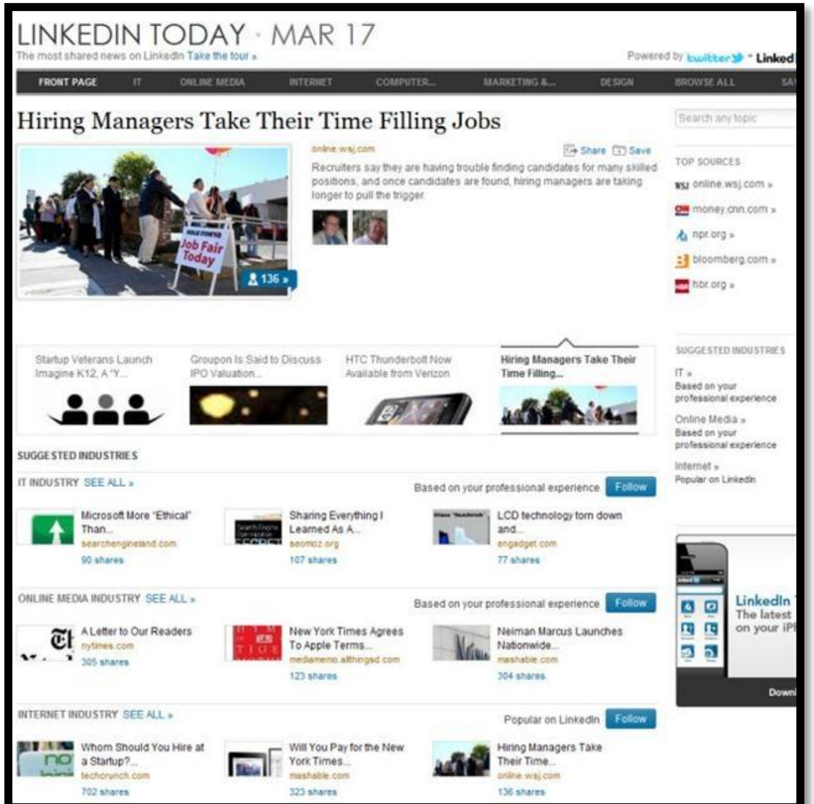
Overview of LinkedIn

LinkedIn Homepage

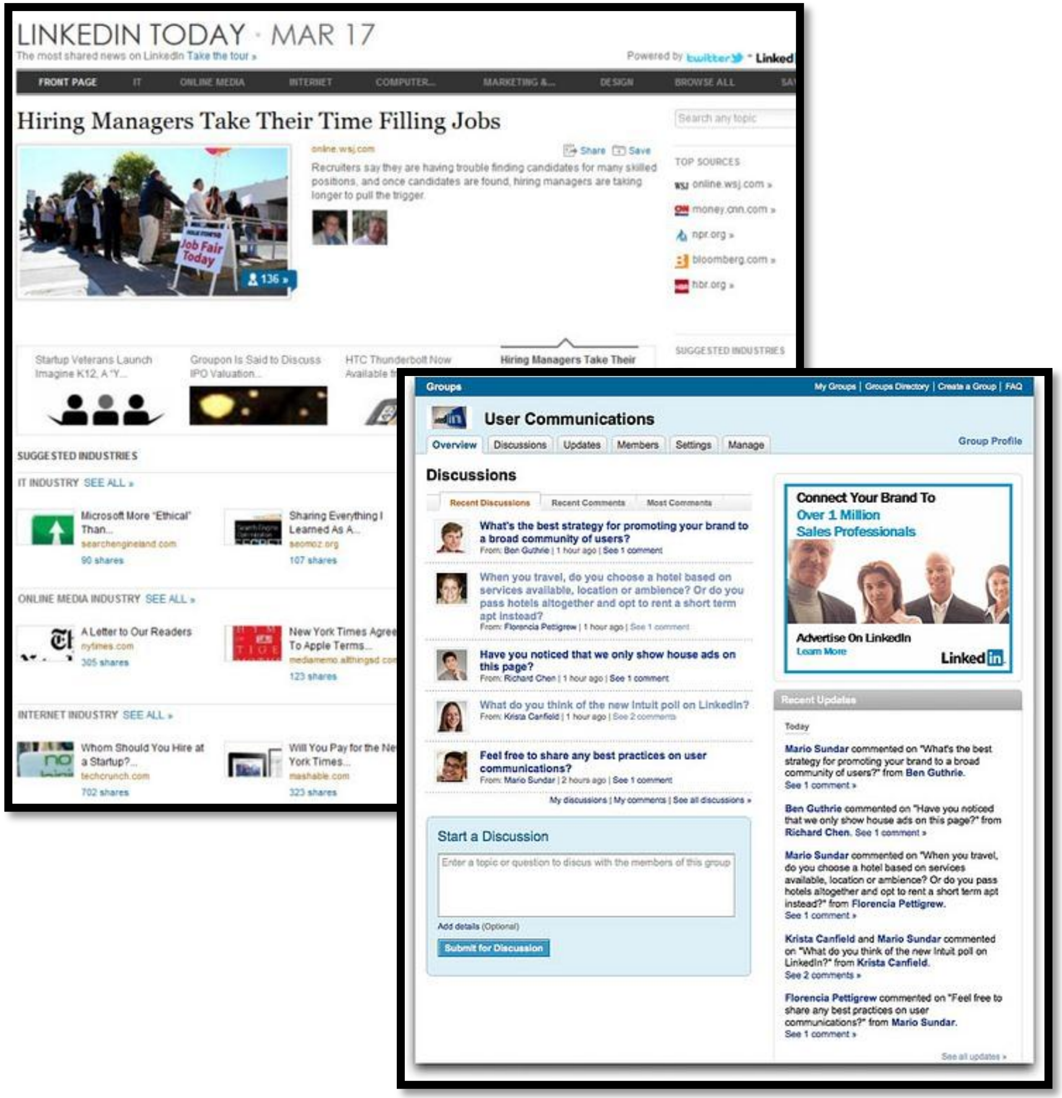


LinkedIn Homepage

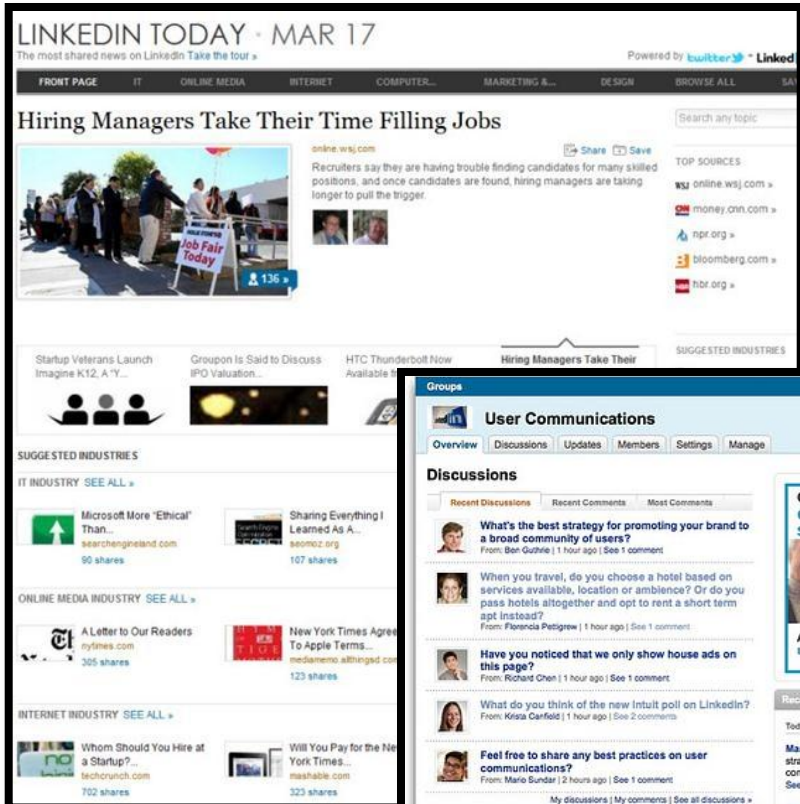
OVERVIEW OF LINKEDIN

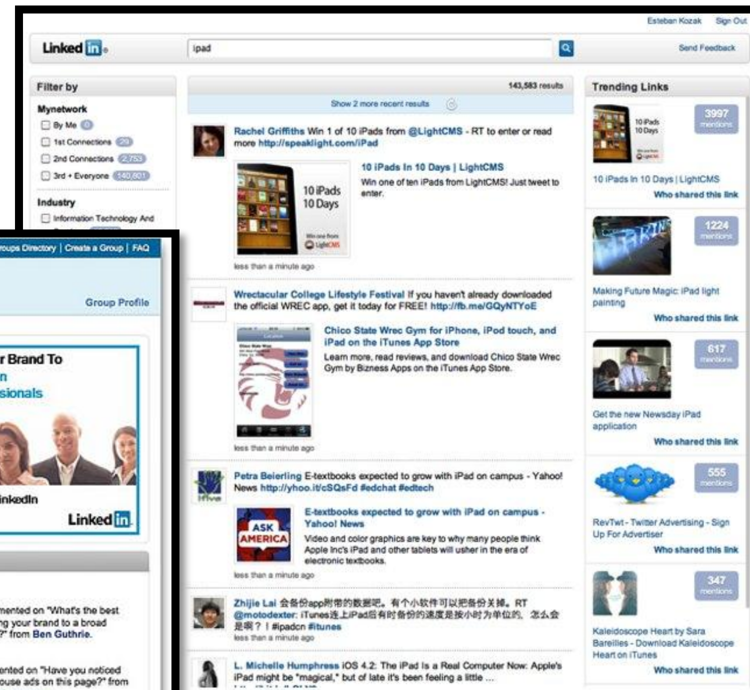


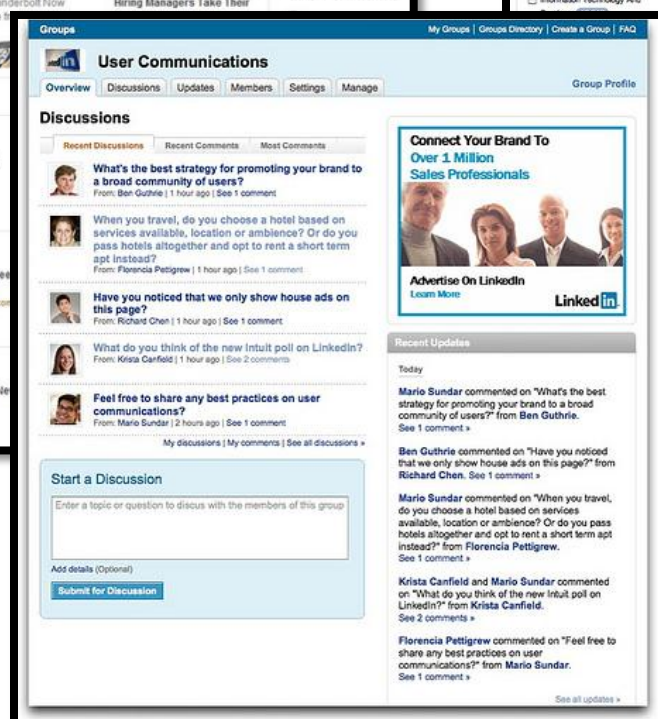
Overview of LinkedIn



Overview of LinkedIn







Problem Definition

For a given recipient and updates from his/her social connections (senders), we want to re-rank these updates to optimize user engagement.

DATASET

Data Summary	April, 2011	September, 2011
Impressions	3M-4M	10M-20M
Updates	30M-40M	100M-200M
Clicked Updates	3M-4M	10M-20M
Non-clicked Updates	27M-36M	90-180M
Distinct Updates	10M-20M	20M-30M
Recipients	1M-2M	4M-5M
Producers	4M-5M	6M-7M

The numbers are obfuscated for commercial reason.

Evaluation Metric

- Precision@*k*
  - $\frac{\text{\# of clicks in top } k \text{ positions}}{k}$
- Average Precision (AP) for ranked list *i*
  - $\frac{\sum_{k=1}^{m} \text{Precision@ } k \times l_k}{\text{\# of clicks for ranked list of ranked list } i}$
    - l_k: position *k* is clicked.
    - m: total number of positions evaluated.
- Mean Average Precision (MAP)
  - average AP across all ranked lists

Methods

- Linear Models
  - Feature Model
  - Bias Model
  - Hybrid Model
- Latent Factor Models
  - Matrix Factorization
  - Tensor Factorization
  - Regression-based Tensor Factorization

METHODS

- Linear Models
  - Feature Model
  - Bias Model
  - Hybrid Model
- Latent Factor Models
  - Matrix Factorization
  - Tensor Factorization
  - Regression-based Tensor Factorization

From the simplest to the most complex

Methods

Linear Models: Feature Model

$$f_i^{(1)} = \boldsymbol{\beta}_{r(i)}^T \boldsymbol{\phi}_{r(i)} + \boldsymbol{\alpha}_{r(i)}^T \boldsymbol{\phi}_i$$

- utilize explicit features.
- f_i represents the estimation of user's click on update *i.*
- $r(i)$ is the recipient of update *i.*
- ϕ is a feature vector.
- β and α are coefficients.

METHODS

Linear Models: Latent Bias Model

$$f_i^{(2)} = \mu + b_i + b_{t(i)} + b_{r(i)} + b_{c(i)} + b_{s(i)}$$

- utilize categorical features.
- $t(i)$ is the type of update *i.*
- $c(i)$ is the type of sender of update *i.*
- $s(i)$ is the sender of update *i.*

METHODS

Linear Models: Latent Bias Model

$$f_i^{(2)} = \boxed{\mu} + \boxed{b_i + b_{t(i)} + b_{r(i)} + b_{c(i)} + b_{s(i)}}$$

- utilize categorical features.
- $t(i)$ is the type of update *i.*
- $c(i)$ is the type of sender of update *i.*
- $s(i)$ is the sender of update *i.*

METHODS

Combining Feature and Bias

$$f_i^{(3)} = f_i^{(1)} + f_i^{(2)}$$

Incorporating Temporal Effects

$$f_i^4 = f_i^{(*)} + \zeta \times t_{\mathrm{recency}}$$

METHODS

Learning through L_2-regularized logistic regression

$$l_1(y_i, f_i^{(*)}) = \log\left[1 + \exp(-y_i f_i^*)\right]$$

METHODS

Linear Model Summary

- Simple
- Fast
- Intuitive

METHODS

Linear Model Summary

- Simple
- Fast
- Intuitive

Does not exploit user-user, user-item interactions at all

Methods

Latent Factor Model: Matrix Factorization

How to utilize pair-wise interactions?

Methods

Latent Factor Model: Matrix Factorization

- user-item interaction?

	?	?
	?	?
	?	?
	?	?

Methods

Latent Factor Model: Matrix Factorization

- user-user interaction?

		
	?	?
	?	2
	1	?
	?	4

METHODS

Latent Factor Model: Matrix Factorization

- user-user interaction?
- user-item interaction?

Methods

Latent Factor Model: Matrix Factorization

$$f_i = \mu + b_i + b_{t(i)} + b_{r(i)} + b_{c(i)} + b_{s(i)} + \boldsymbol{\eta}_{r(i)}^T \boldsymbol{\eta}_{s(i)}$$

Methods

Latent Factor Model: Matrix Factorization

$$f_i = \boxed{\mu + b_i + b_{t(i)} + b_{r(i)} + b_{c(i)} + b_{s(i)}} + \boxed{\boldsymbol{\eta}_{r(i)}^T \boldsymbol{\eta}_{s(i)}}$$

Latent Bias Model

Matrix Factorization

- very similar to basic MF model used in SVD++

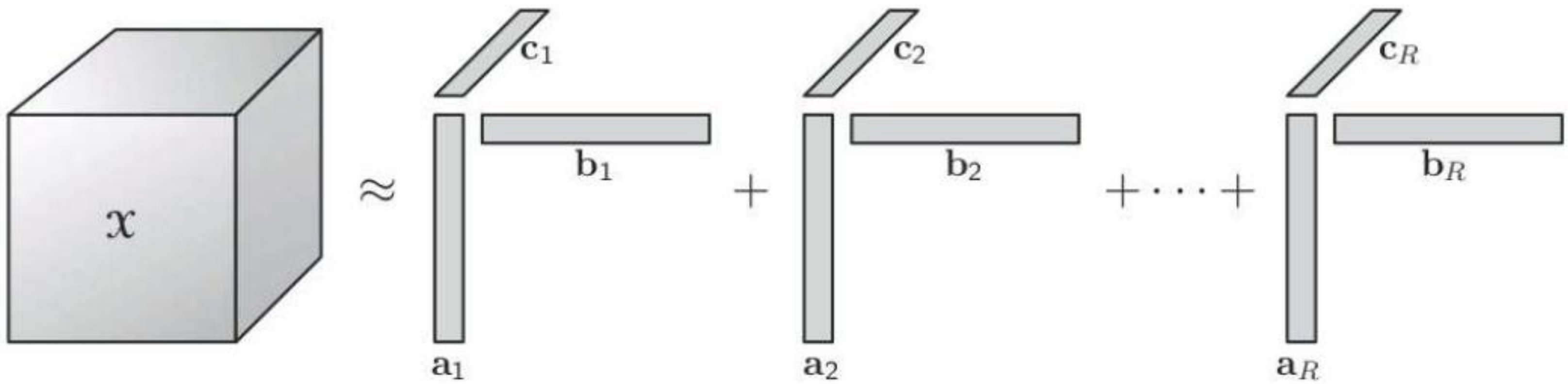
[Koren 2010]

Methods

Higher-order interactions?

METHODS

Latent Factor Model: Tensor Factorization



- Recipient-Type-Sender relationships
- CP decomposition

METHODS

Latent Factor Model: Tensor Factorization

$$f_i = \boxed{\mu + b_i + b_{t(i)} + b_{r(i)} + b_{c(i)} + b_{s(i)}} + \boxed{\sum_k \boldsymbol{\eta}_{r(i),k} \boldsymbol{\eta}_{s(i),k} \boldsymbol{\eta}_{t(i),k}}$$

Latent Bias Model

Tensor Factorization

METHODS

How about other explicit features?

METHODS

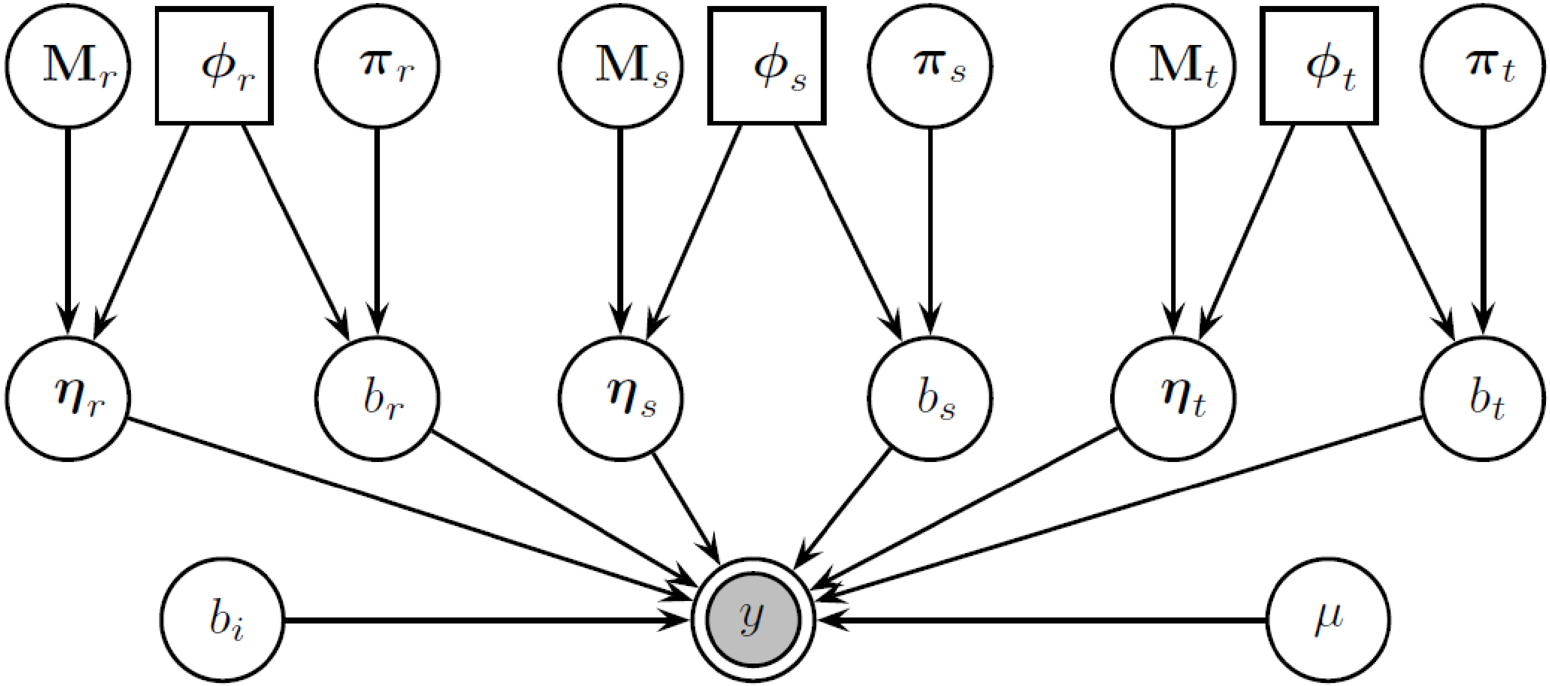
How about other explicit features?

- Regression-based latent factor models
  - another layer of regression
  - replacing zero-mean with regression-based mean

$$\boldsymbol{\eta}_{x(*)} = \mathbf{M}_x \boldsymbol{\phi}_{x(*)} + \epsilon_x \qquad x \in \{\mathcal{R}, \mathcal{S}, \mathcal{T}\}$$

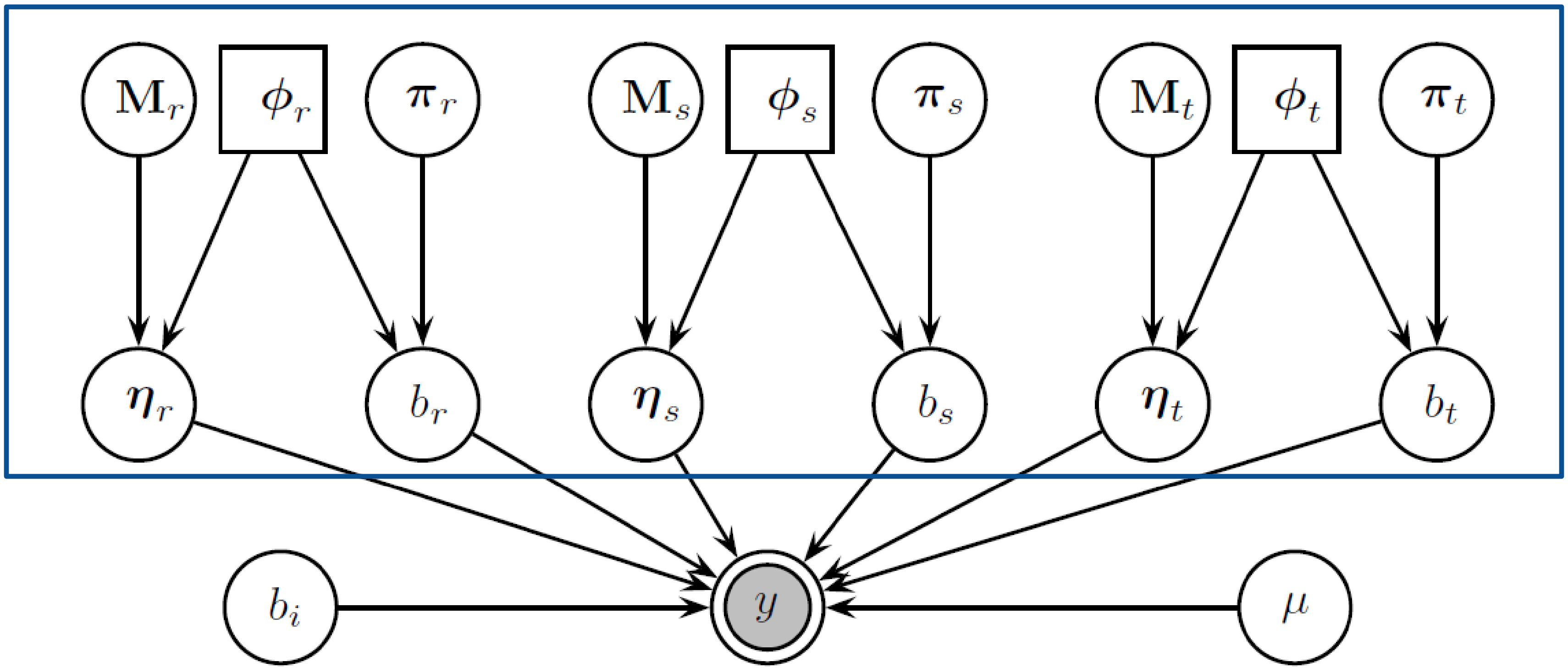
$$b_{x(*)} = \boldsymbol{\pi}_x^T \boldsymbol{\phi}_{x(*)} + \epsilon_{b_x}$$

METHODS

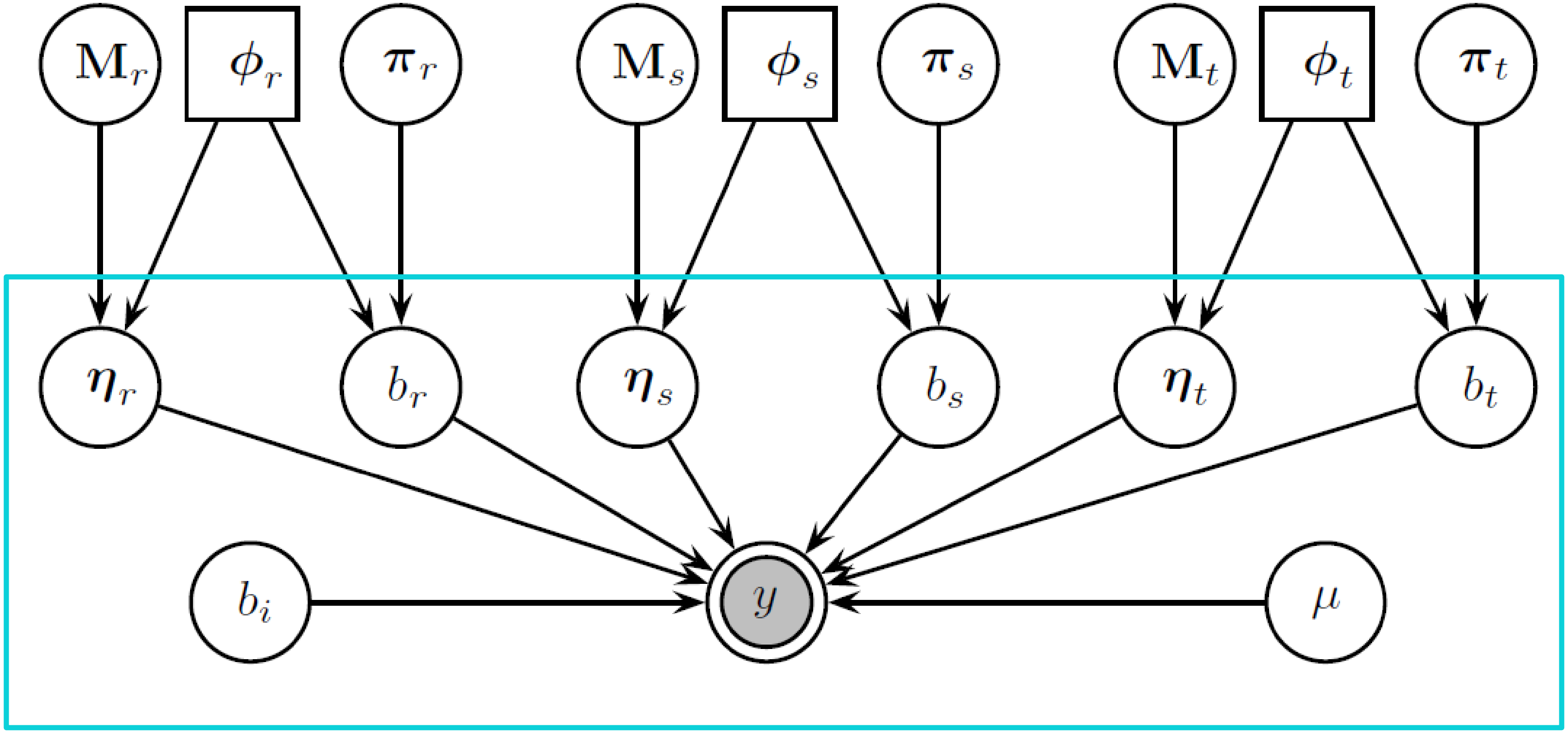


$\mathbf{M}_r$
ϕ_r
$\boldsymbol{\pi}_r$
$\mathbf{M}_s$
ϕ_s
$\boldsymbol{\pi}_s$
$\mathbf{M}_t$
ϕ_t
$\boldsymbol{\pi}_t$
$\boldsymbol{\eta}_r$
b_r
$\boldsymbol{\eta}_s$
b_s
$\boldsymbol{\eta}_t$
b_t
b_i
y
μ

METHODS


$\mathbf{M}_r$
ϕ_r
$\boldsymbol{\pi}_r$
$\mathbf{M}_s$
ϕ_s
$\boldsymbol{\pi}_s$
$\mathbf{M}_t$
ϕ_t
$\boldsymbol{\pi}_t$
$\boldsymbol{\eta}_r$
b_r
$\boldsymbol{\eta}_s$
b_s
$\boldsymbol{\eta}_t$
b_t
b_i
y
μ

METHODS



$\mathbf{M}_r$
ϕ_r
$\boldsymbol{\pi}_r$
$\mathbf{M}_s$
ϕ_s
$\boldsymbol{\pi}_s$
$\mathbf{M}_t$
ϕ_t
$\boldsymbol{\pi}_t$
$\boldsymbol{\eta}_r$
b_r
$\boldsymbol{\eta}_s$
b_s
$\boldsymbol{\eta}_t$
b_t
b_i
y
μ

METHODS

Learning procedure

- Maximum A Posterior (MAP)
- Stochastic Gradient Descent

Methods

Going beyond pointwise learning

- Optimizing Bayesian Personalized Ranking (BPR)

$$\sum_{m \in \mathcal{O}_{i,+}} \sum_{n \in \mathcal{O}_{i,-}} \sigma\Big(f_m - f_n\Big)$$

[Rendle et al. 2009]

EXPERIMENTS

Models

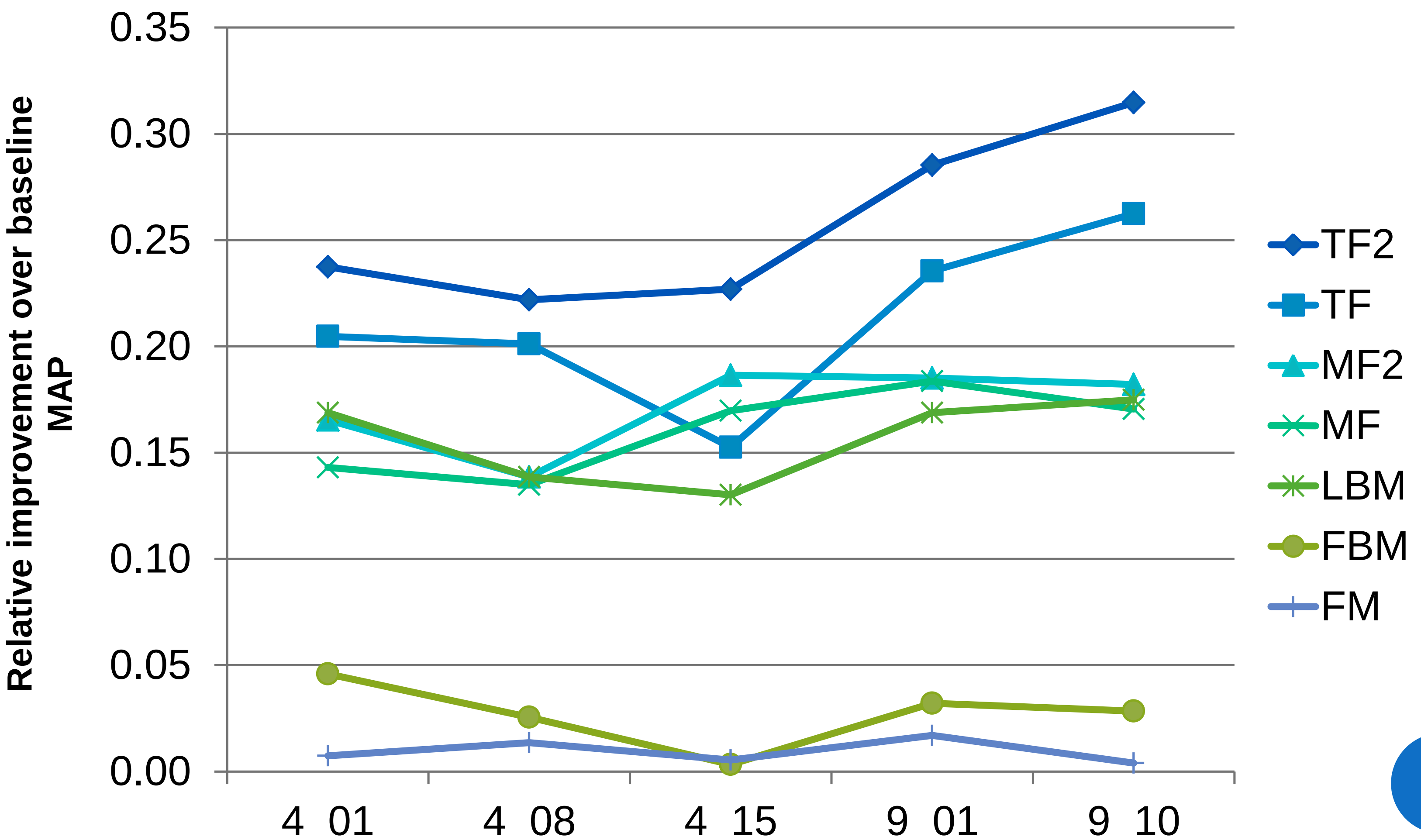
Methods
Baseline (BL)
Feature Model (FM)
Latent Bias Model (LFM)
Feature Bias Model (FBM)
Matrix Factorization (MF)
Tensor Factorization (TF)
Matrix Factorization with Features (MF2)
Tensor Factorization with Features (TF2)

Experiments

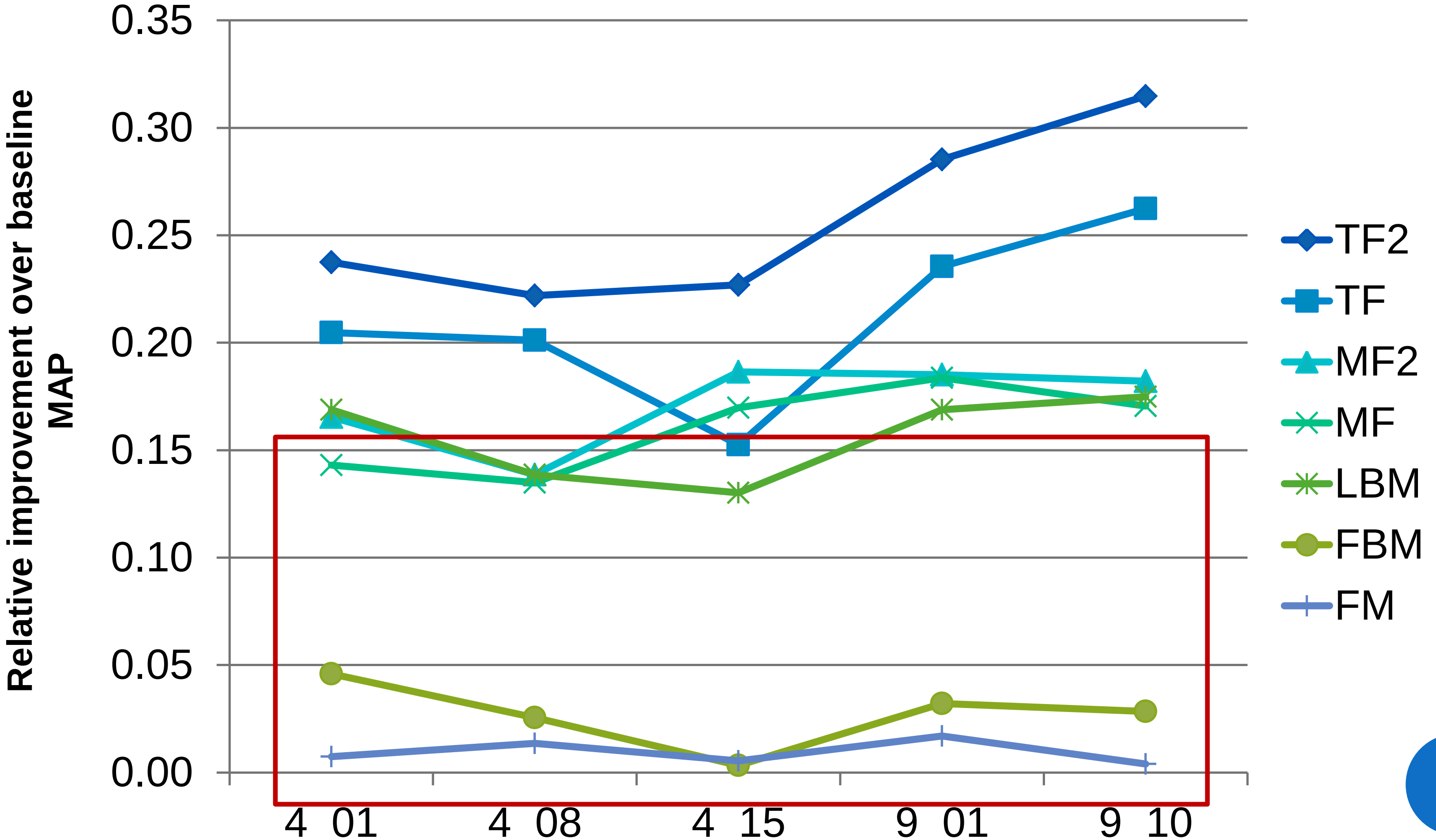
Models

Features	Comments
Seniority	the seniority level of a user
Visiting	how frequently a user visits LinkedIn
PageRank	discretized PageRank scores
Connectedness	how well a user is connected to others
Social strength	social strength between recipient and sender
Professionalism	how professional an update's language is
Recency	the freshness of an update

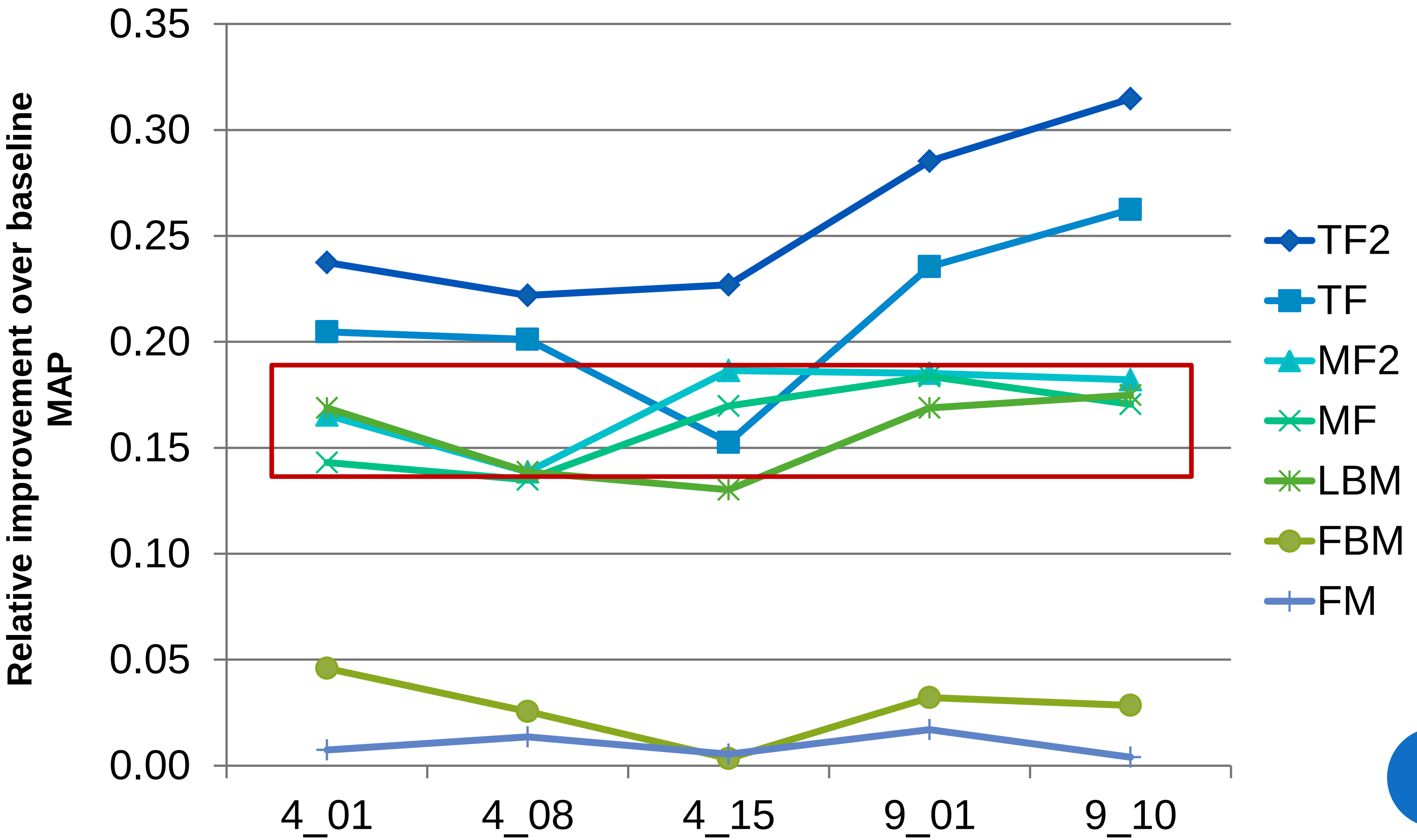
Experiments: Pairwise Learning



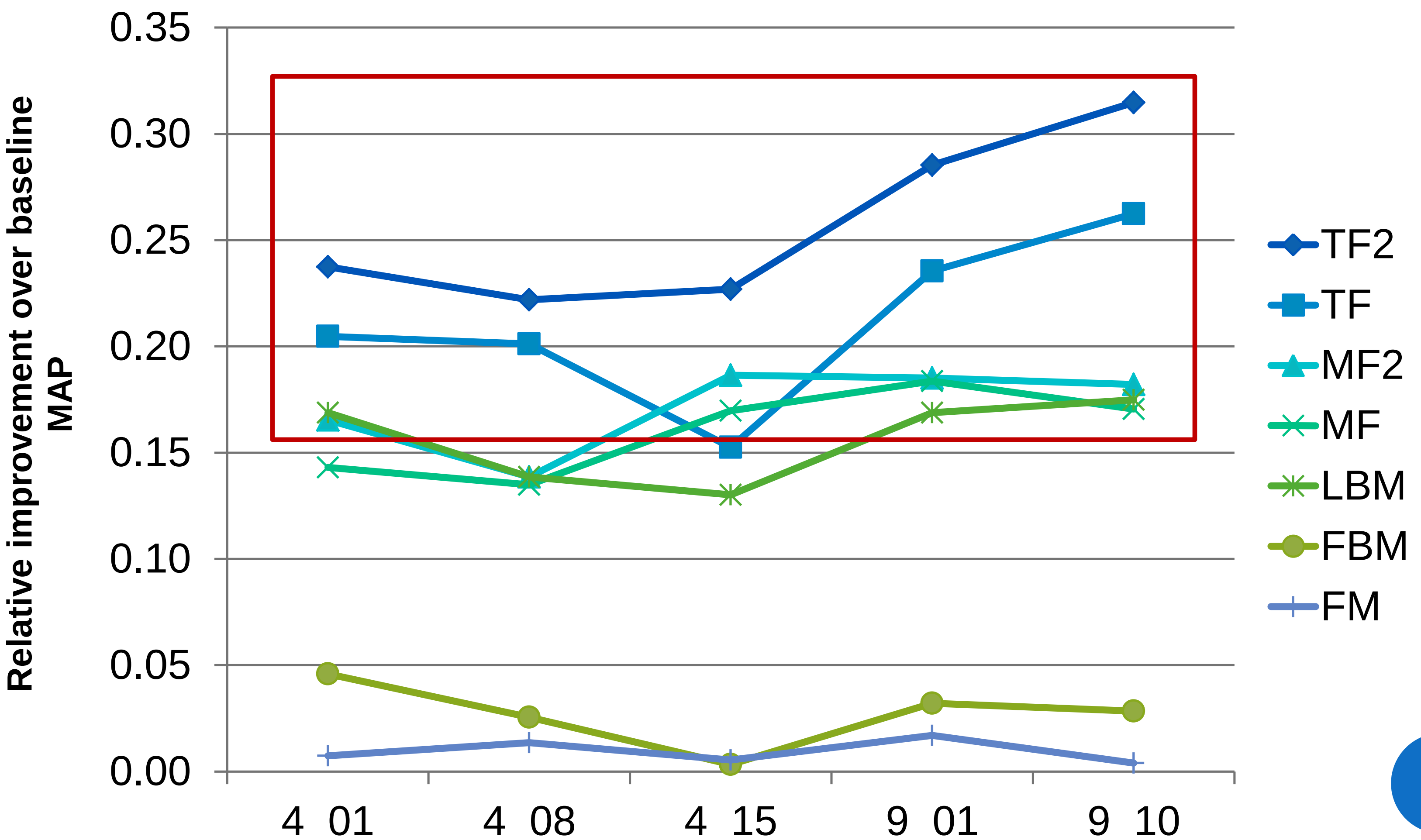
Experiments: Pairwise Learning



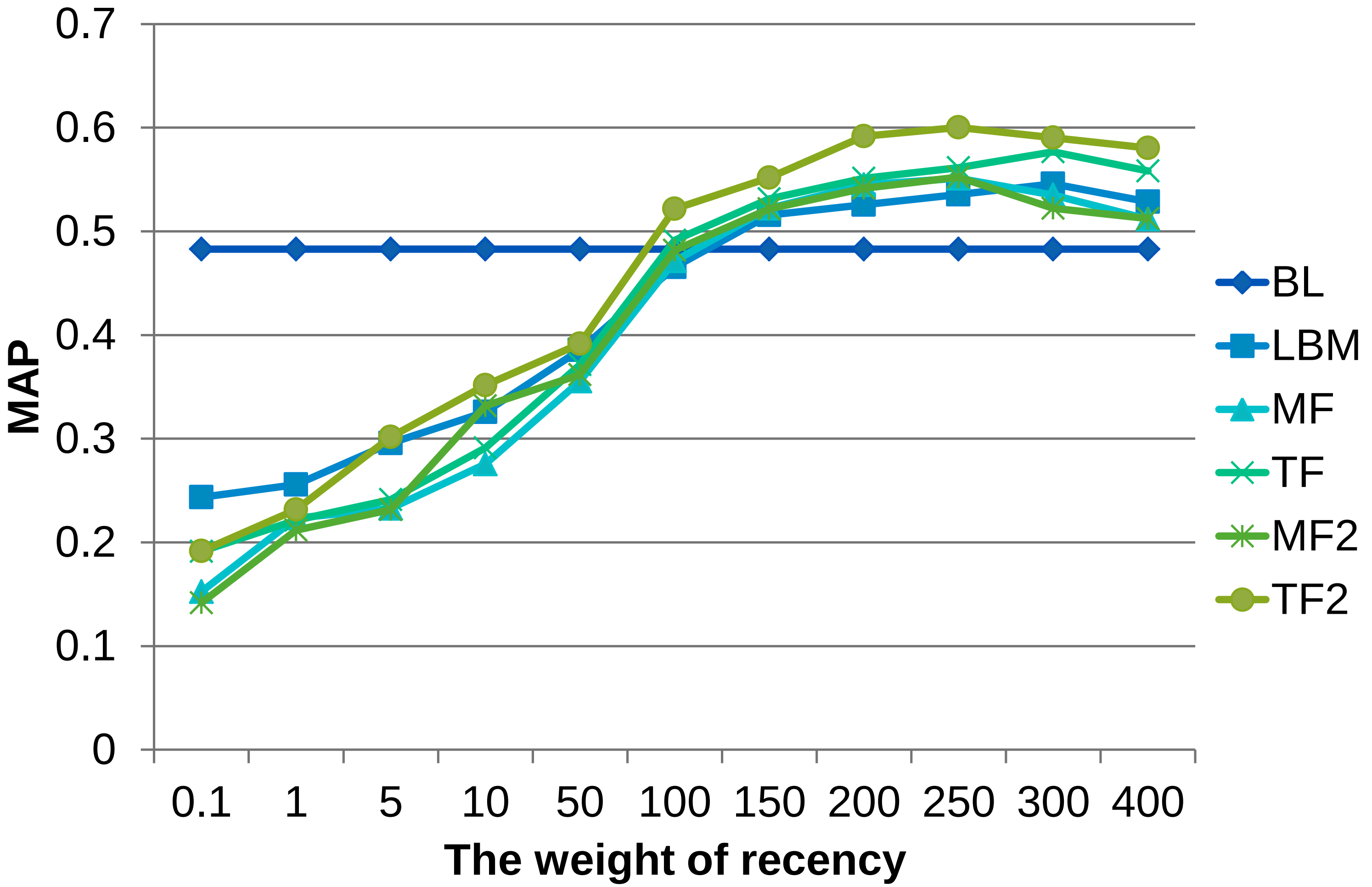
Experiments: Pairwise Learning



Experiments: Pairwise Learning



Experiments: Parameter Sensitivity



EXPERIMENTS

Example of highly ranked types of updates

Type Description	**Bias b_t**
Job Seeker Product Update	0.5765
Joining Sub-Group	0.5407
Company News	0.4592
Joining Group	0.2625
Profile Picture Update	0.2516
Initiating Direct Ads Campaign	0.2253
Profile Update	0.1394

Conclusions

- We demonstrate that recommender systems + preference-based learning can be used to re-rank social updates.
- A linear model can achieve 60% of the performance of latent factor models, on average.
- A tensor factorization model with regression on explicit features works the best.
- The cold-start problem makes it impossible to model some kinds of interactions.

Thank you.

Liangjie Hong

PhD candidate

WUME Lab

Lehigh University

lih307@cse.lehigh.edu

EXPERIMENTS: COMPARISON

Training/Testing	BL	FM	LBM	FBM
4_01(Tr.)/4_08(Te.)	0.5278	0.5317	**0.5943**	0.5520
4_08(Tr.)/4_15(Te.)	0.5435	0.5509	**0.6040**	0.5574
4_15(Tr.)/4_22(Te.)	0.5218	0.5246	**0.5823**	0.5235
9_01(Tr.)/9_10(Te.)	0.4829	0.4911	**0.5457**	0.4984
9_10(Tr.)/9_18(Te.)	0.4779	0.4798	**0.5432**	0.4915
9_18(Tr.)/9_25(Te.)	0.4768	0.4803	**0.5329**	0.4886

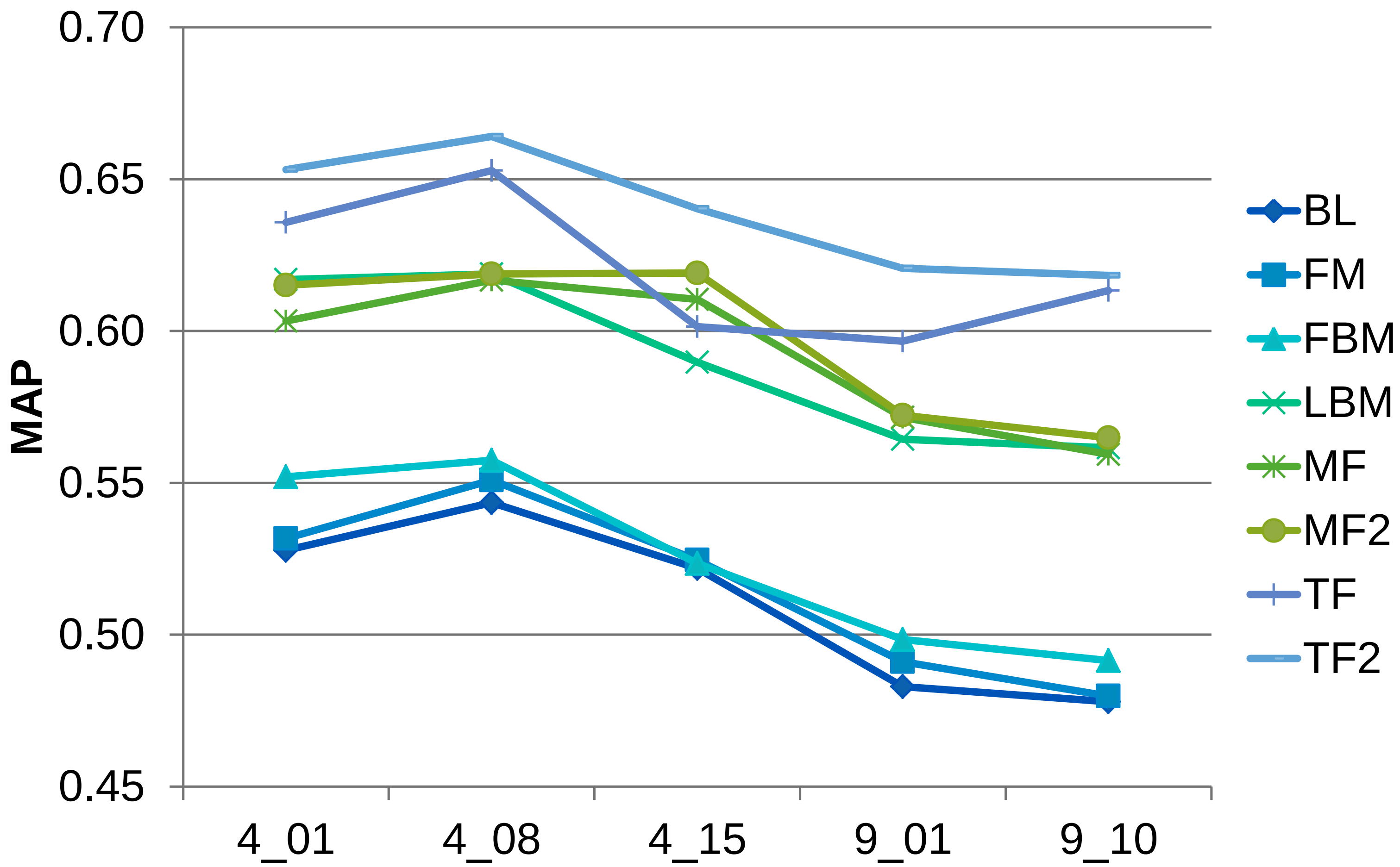
Training/Testing	MF	TF	MF2	TF2
4_01(Tr.)/4_08(Te.)	0.5955	0.6258	0.5951	**0.6336**
4_08(Tr.)/4_15(Te.)	0.6079	0.6228	0.6088	**0.6535**
4_15(Tr.)/4_22(Te.)	0.5962	0.6014	0.5991	**0.6312**
9_01(Tr.)/9_10(Te.)	0.5511	0.5766	0.5523	**0.6003**
9_10(Tr.)/9_18(Te.)	0.5412	0.5833	0.5449	**0.6109**
9_18(Tr.)/9_25(Te.)	0.5359	0.5799	0.5362	**0.5992**

EXPERIMENTS

The effects of pairwise learning

Training/Testing	LBM	MF	MF2	TF	TF2
4_01(Tr.)/4_08(Te.)	0.6169	0.6033	0.6151	0.6358	**0.6532**
4_08(Tr.)/4_15(Te.)	0.6188	0.6168	0.6188	0.6528	**0.6641**
4_15(Tr.)/4_22(Te.)	0.5897	0.6104	0.6191	0.6014	**0.6402**
9_01(Tr.)/9_10(Te.)	0.5644	0.5716	0.5723	0.5966	**0.6207**
9_10(Tr.)/9_18(Te.)	0.5593	0.5621	0.5607	0.5999	**0.6183**

Experiments



EXPERIMENTS

Parameter Sensitivity