< AIIDE 2011, Oct. 14th, 2011 >

Detecting Real Money Traders in MMORPG by Using Trading Network

Atsushi FUJITA   Hiroshi ITSUKI   Hitoshi MATSUBARA

Future University Hakodate, JAPAN

fujita@fun.ac.jp

Help the online game operators

- Focusing on Massively Multiplayer Online RPG
  - Thousands of players co-exist in one virtual “world”
    - cf. millions of registered players

Operators' issue: Grasp the virtual world

- To facilitate further growth
  - Effect of features
    - Extended game fields, one-shot events
  - Influential players
    - Mentoring, intermediation, trades

- To maintain the order of the virtual world
  - Harassments between players
    - Player killing, occupation of specific locations
  - Causes that lead unfairness and crisis of virtual economy
    - Real Money Trading, use of bots, cheat

RMT: Real Money Trading

- Real money ⇔ Virtual properties
  - Currency, items, status, functions, avatars, etc.
  - Observed in other online services, e.g., SNS, auction

- Two opposing attitudes (sometimes ambivalent)
  - **Positive**: Means of augmenting the real world
    - e.g., Second Life
  - **Negative**: Source of serious problems
    - e.g., Most MMORPGs in Japan

Task & given situation

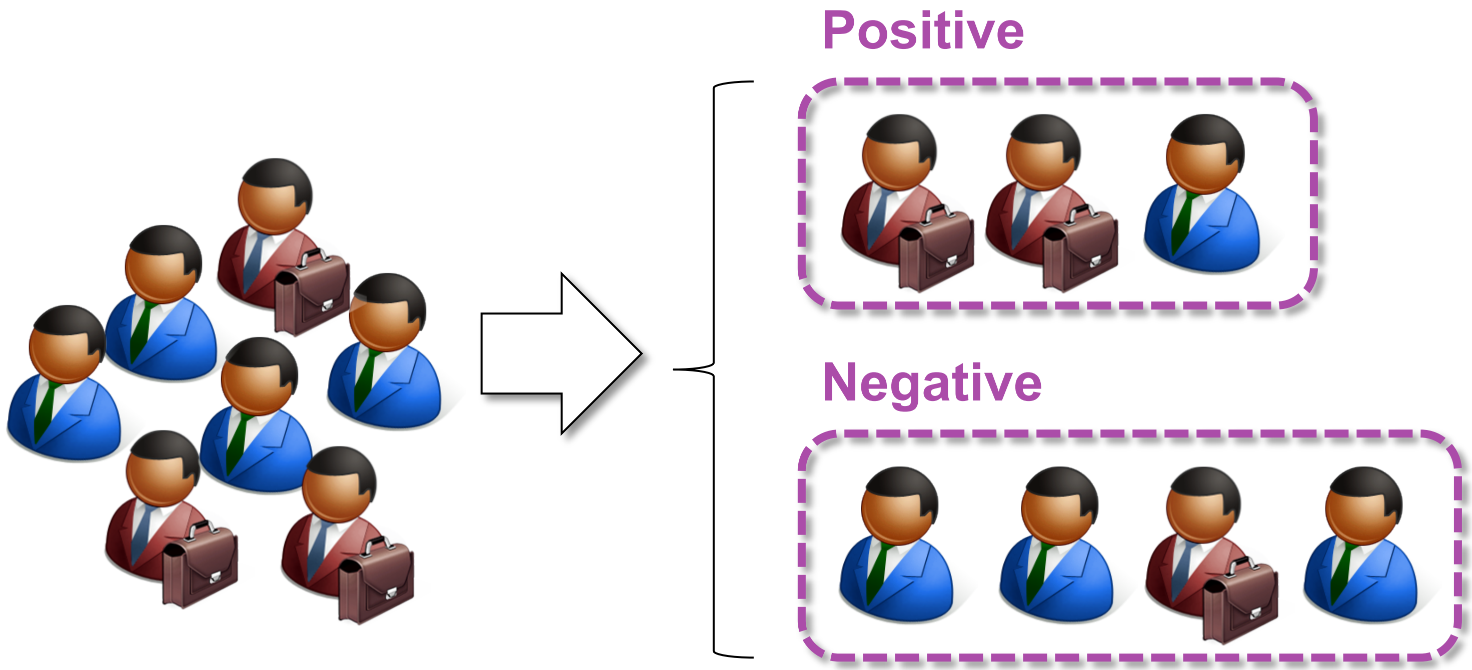
- Automatic detection of RMTers
  - Actual log data is available
    - Now with TECMO KOEI GAMES CO., LTD.
    - Prefer title independent features
  - Operators want no arms race
    - Desire un-cheatable features
  - Operators' verification is indispensable
    - To avoid ruling out honest players
    - The amount of human resource depends on situation
      - Title (scale, seriousness) and budget for operation
      - Prefer unsupervised or semi-supervised methods

Outline

1. Introduction
2. Approach
3. Procedure
4. Experiment
5. Conclusion

As a binary classification

- Classify each character into RMTer or non-RMTer
  - Supervised machine learning [Ahmad+, 09]
    - Naïve Bayes, k-NN, AdaBoost, etc.
    - Various features (incl. those specific to the title)
  - Not flexible: Too much/less positive class

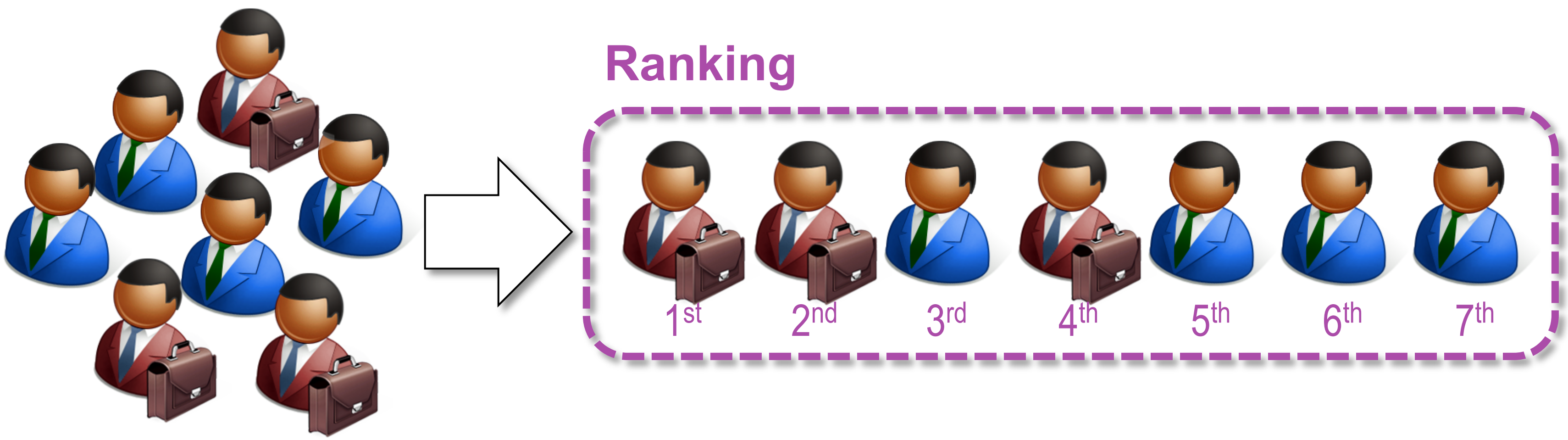


As a ranking problem

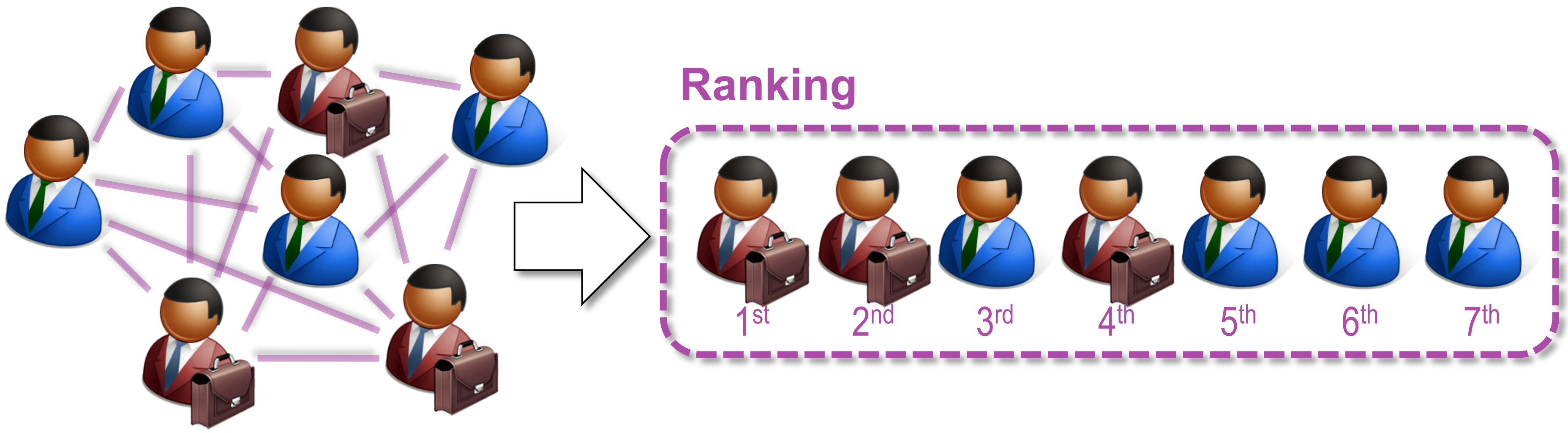
- Sort characters according to their suspiciousness
  - Using cumulative features [Itsuki+, 10]
    - Handled currency
    - Volume of actions
    - Activity hours

    dealing with enormous virtual currency

    working hard to earn virtual currency
  - Not thoroughly studied



Ranking characters by using trading network

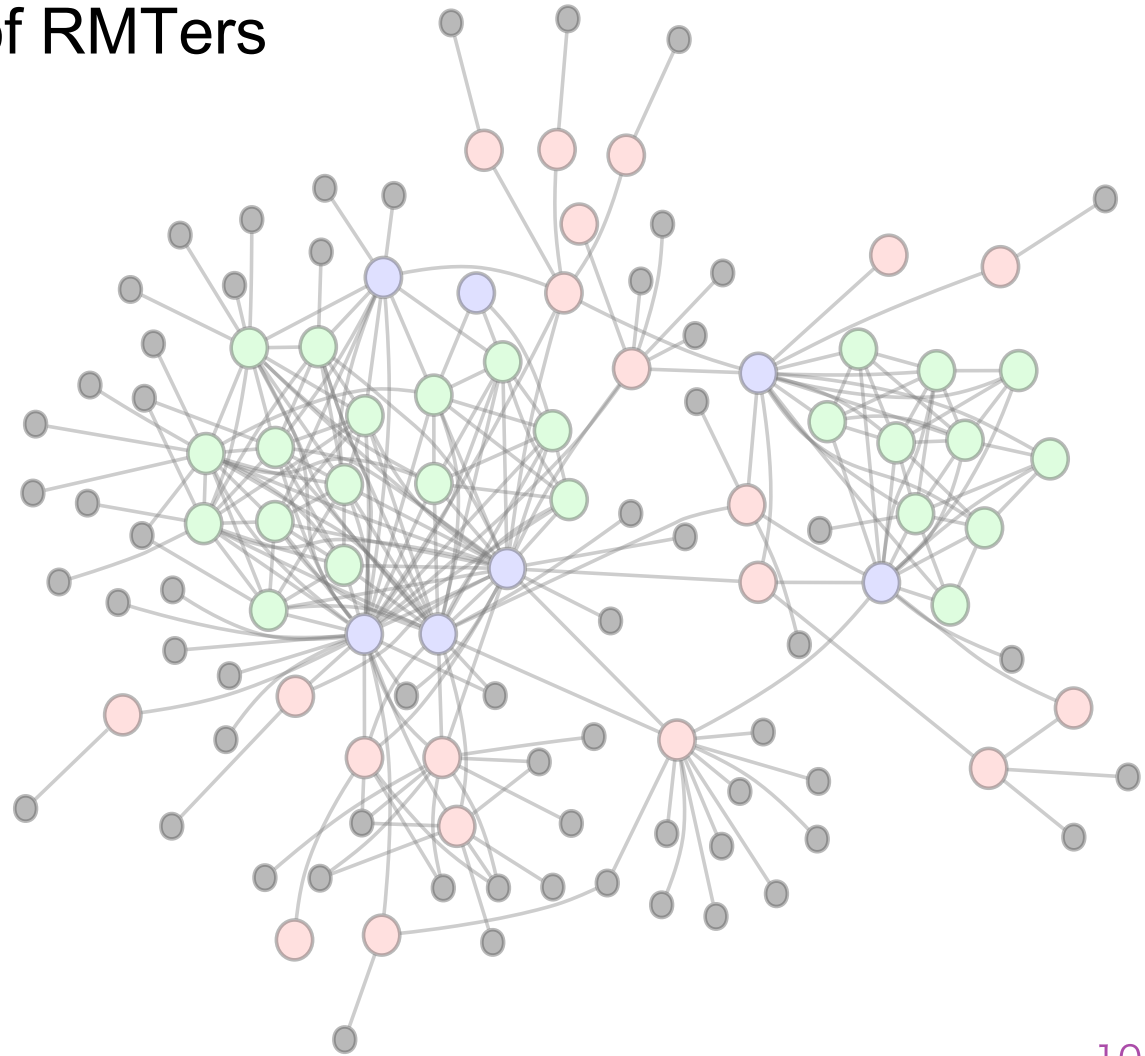


- Connection between pairs of characters
  - Extremely low exchange rate, e.g., full of wallet = 

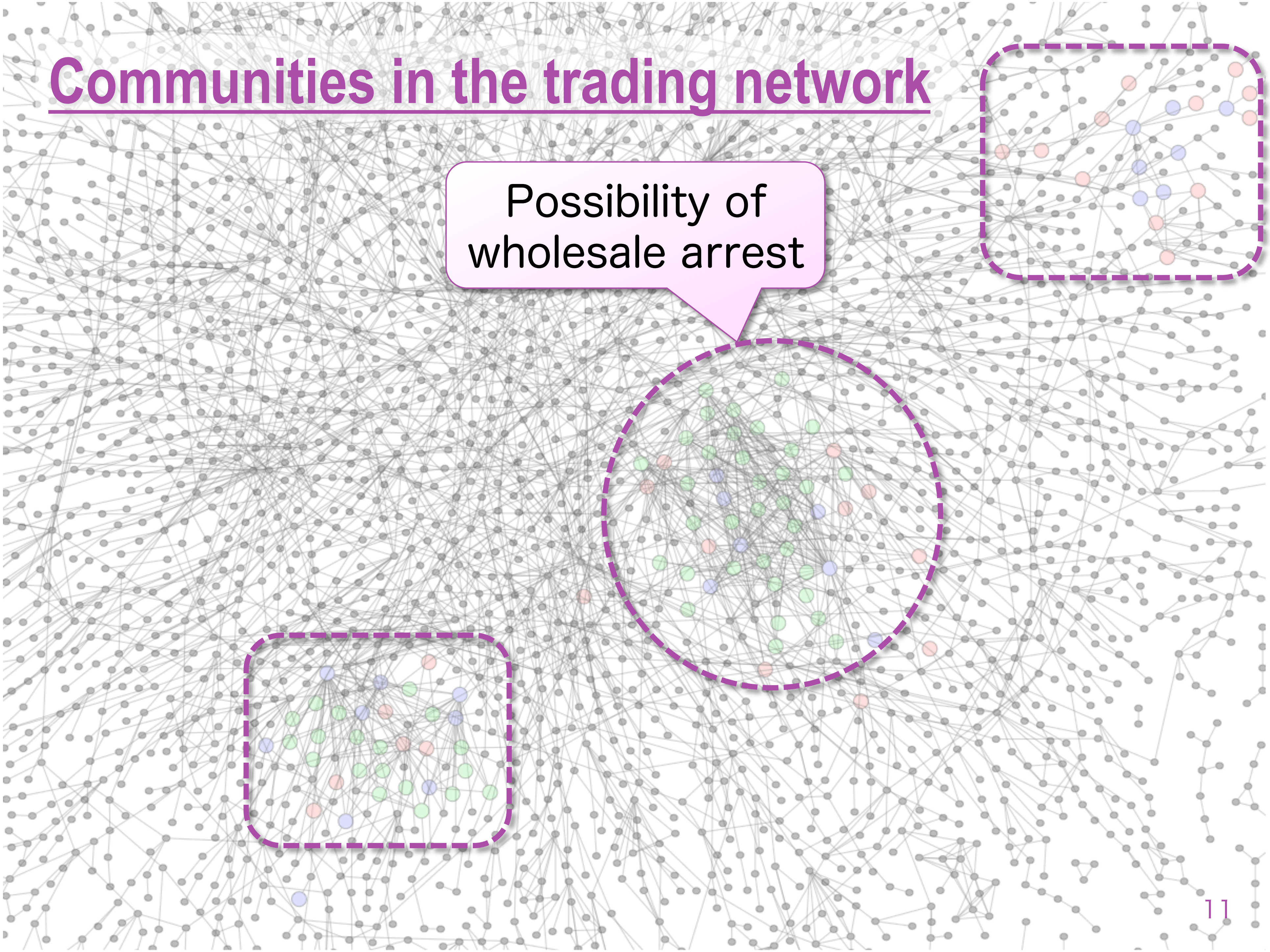
    - Division of RMT labor & frequent trade
  - Infrastructure for trading → log data are available
- Volume of individual trade

Trading sub-network (from our data)

- RMTers and their trading partners in one timeframe
  - Division of labor of RMTers
    - Typical roles
      - Seller
      - Earner
      - Collector
  - Tight connection

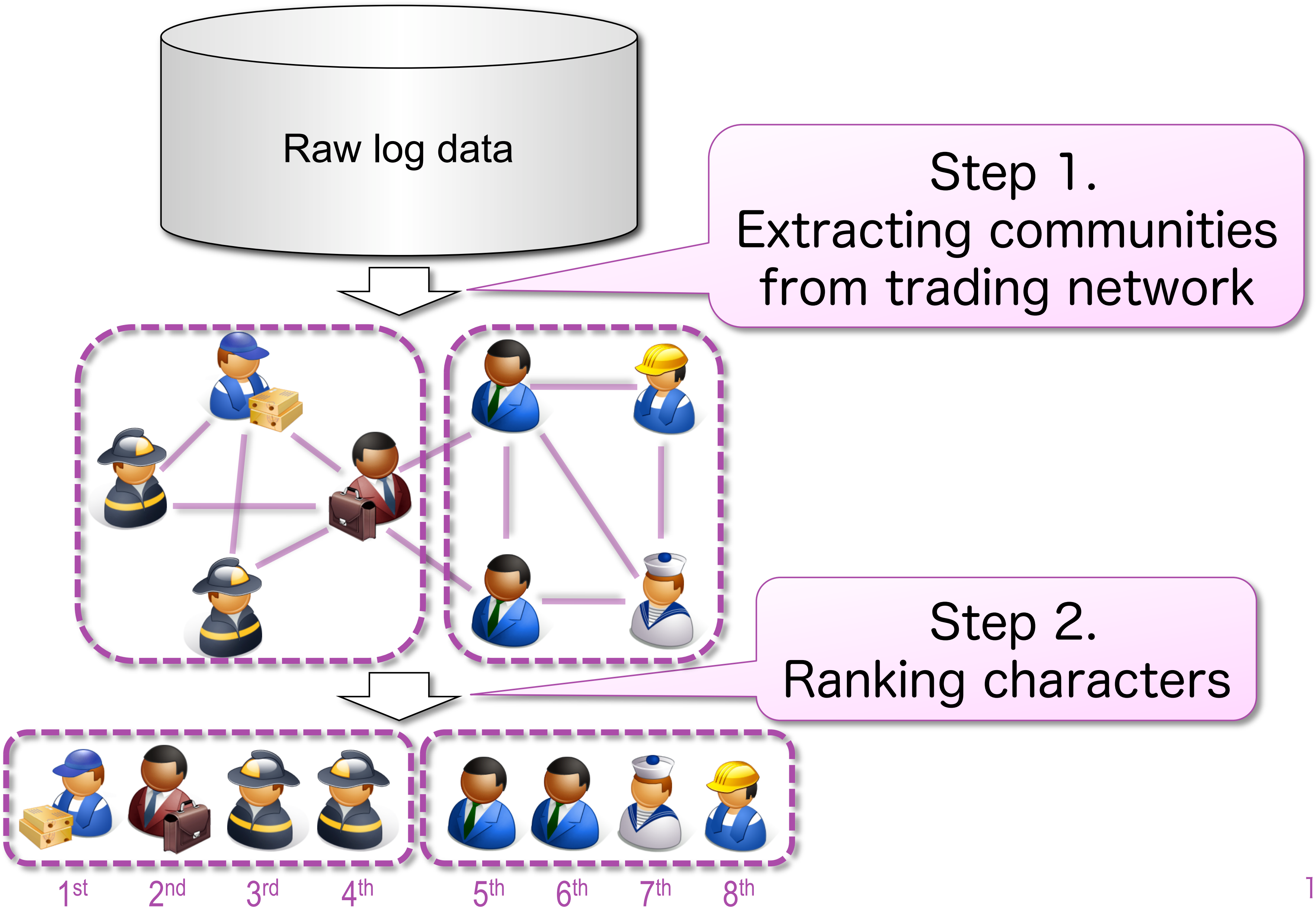


Communities in the trading network



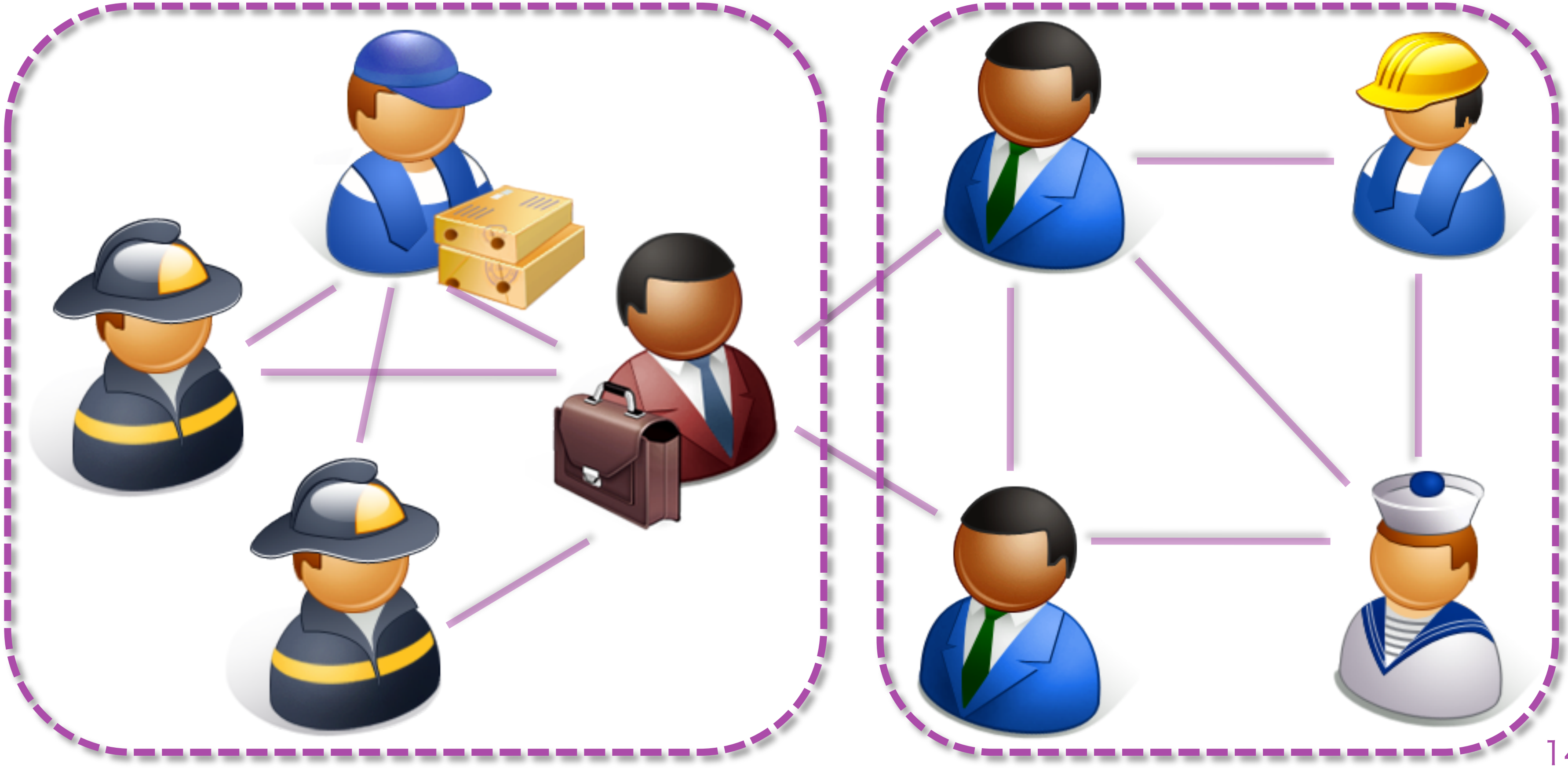
Outline

Overview

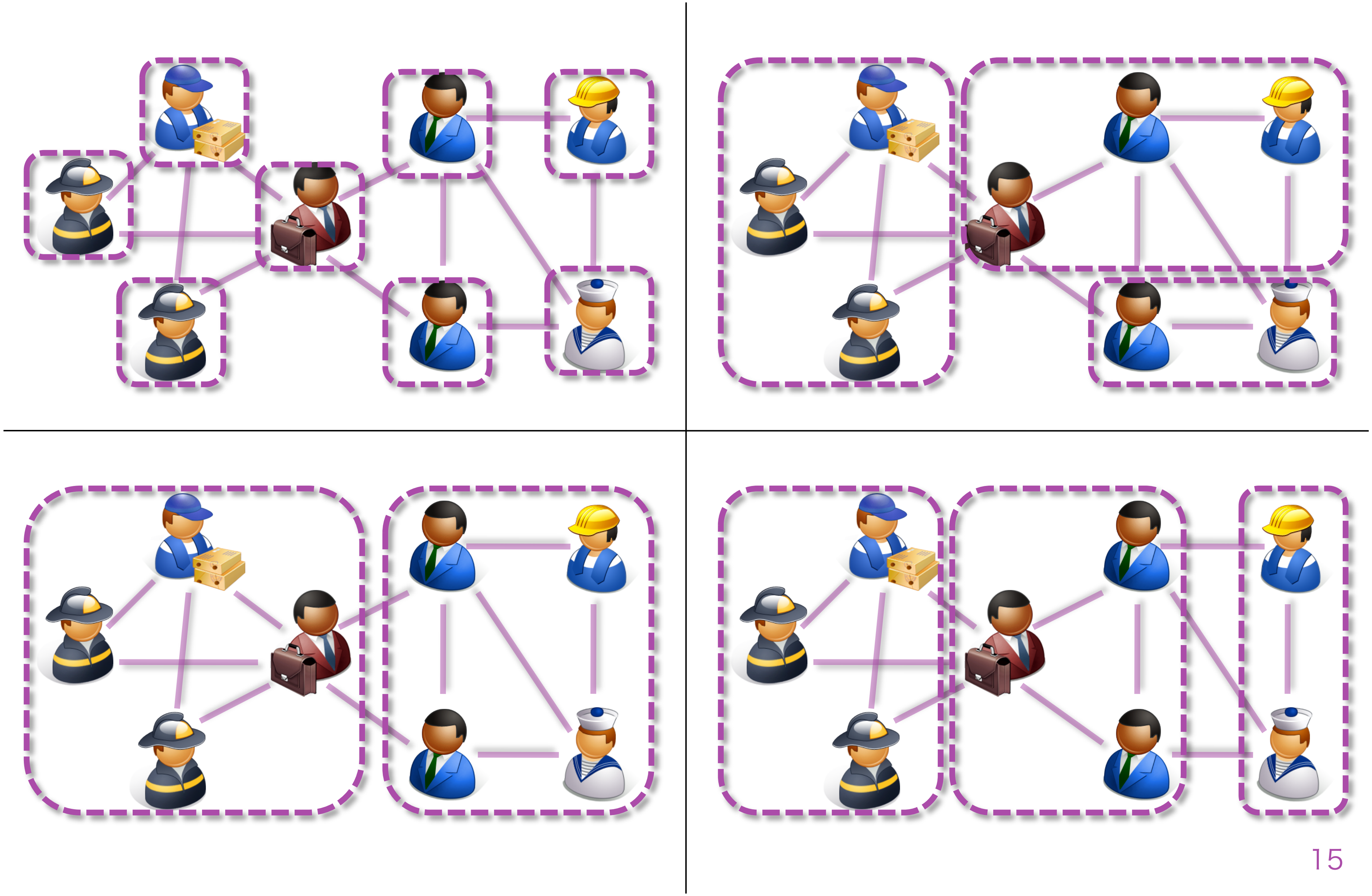

Raw log data
Step 1.
Extracting communities from trading network
Step 2.
Ranking characters
1st
2nd
3rd
4th
5th
6th
7th
8th

Step 1. Community extraction

- Graph partitioning / graph clustering
  - Node: Character
  - Edge: Trade between two characters



Which division is best?



Quality of a given division of network

- Modularity [Newman+, 04]

$$Q = \sum_i (e_{ii} - a_i^2)$$

Many edges in each community → Large Q

$$\frac{\sum_{e \in E_i} (\text{weight of } e)}{\sum_{e \in E} (\text{weight of } e)}$$

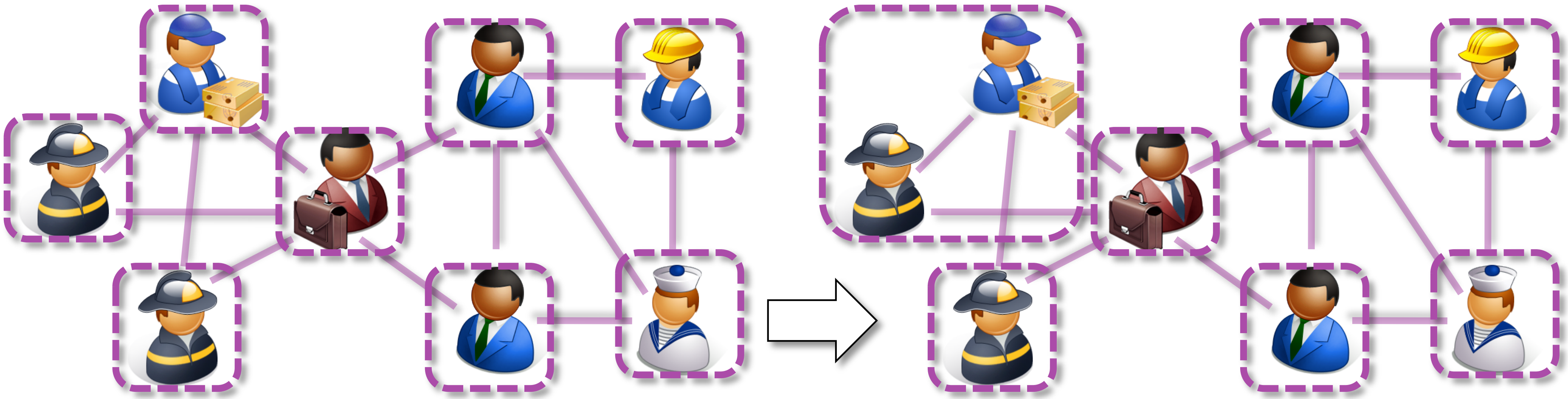
Expected value of link ratio: to avoid a trivial solution

$$\frac{\sum_{e \in A_i} (\text{weight of } e)}{\sum_{e \in E} (\text{weight of } e)}$$

E: Set of all edges in the network
E_i: Set of edges within i^{th} community
A_i: Set of edges connecting to a node in i^{th} community

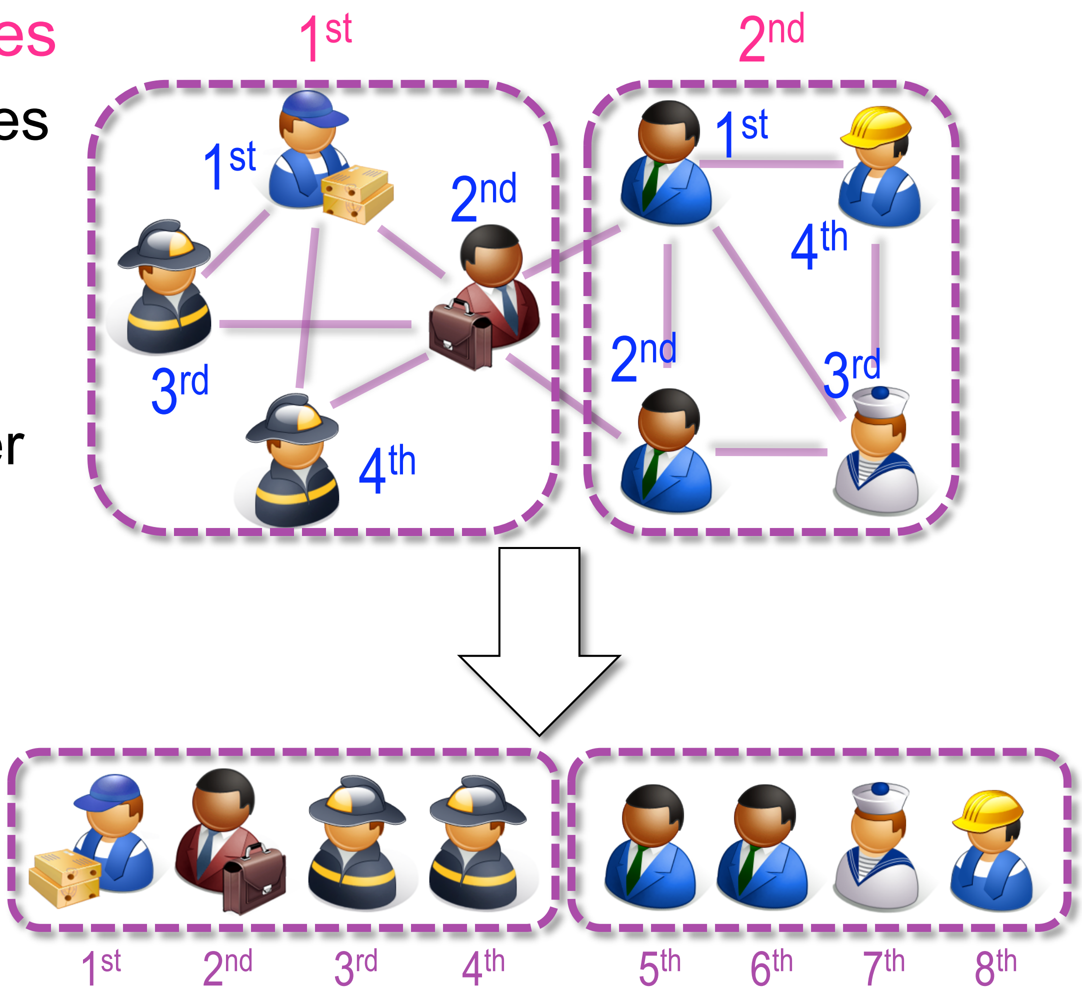
Community extraction algorithm

- Finding a partitioning that maximizes Q: NP-hard
- A bottom-up greedy algorithm [Clauset+, 04]
  - 1. Regard each node as a community and calculate ΔQ for each connected community pair
  - 2. Merge two communities whose ΔQ is largest (and >0)
  - 3. Update ΔQ for the merged communities
  - 4. Repeat steps 2 & 3 while Q gains



Step 2. Ranking characters

- Frequent and/or large-scale trades → RMT
  1. Ranking communities
     - In-community trades
  2. Ranking characters in each community
     - Trades of individual character



Outline

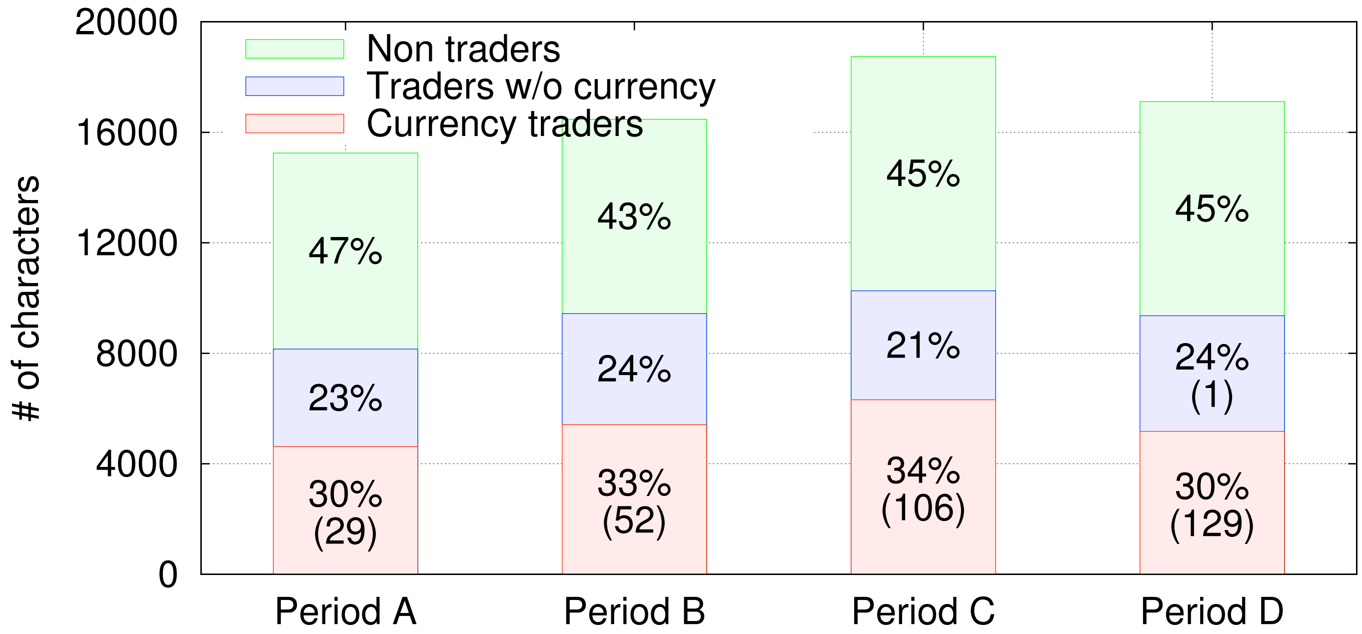
1. Introduction
2. Approach
3. Procedure
4. <u>Experiment</u>
5. Conclusion

Application to a real MMORPG

- "Uncharted Waters Online"
  - Exploration, naval battle and trading in mid-ages
  - RMT is prevalent

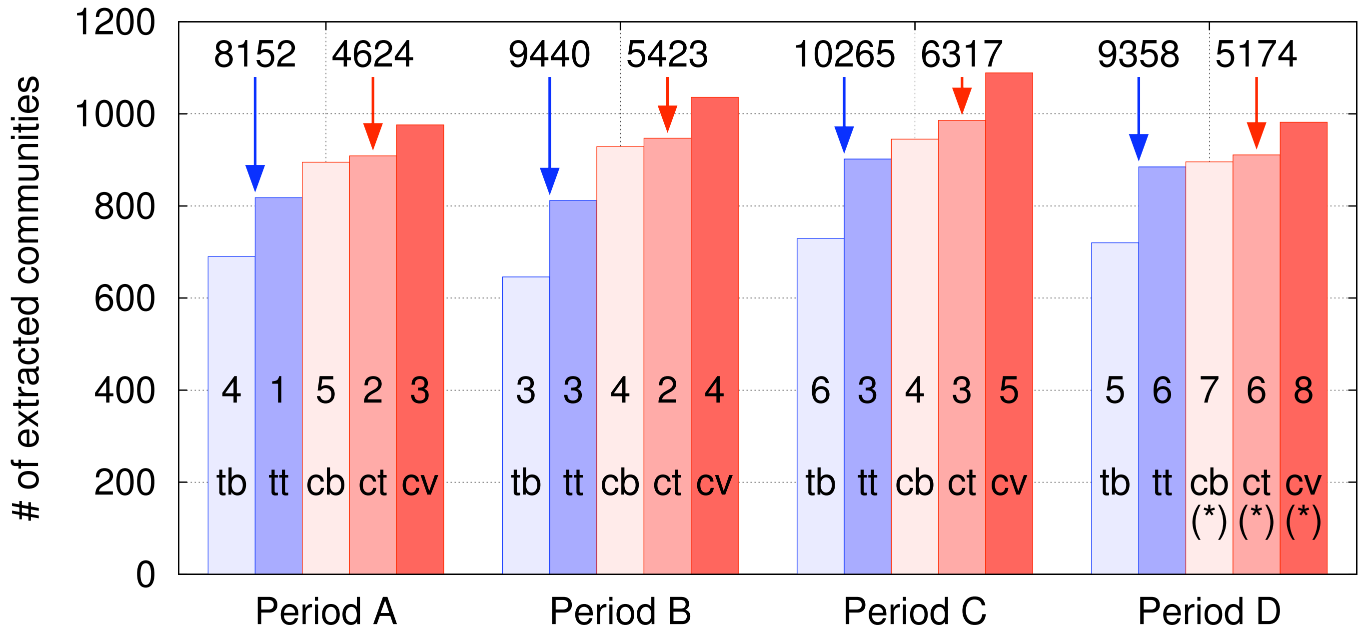
- 4 timeframes (15～23 days, no overlap)
  - RMTers are identified (& banned) manually
    - 29～130 (<1%) within 15,249～18,745 characters
  - Actual action log data in the same period
    - 300～480 million records

Traders and RMTers



- Obs.
  - Half of all characters traded something
  - 1/3 of all characters traded virtual currency
  - Most of RMTers traded virtual currency
    - Only 1 exception in period D

# of extracted communities



- Target characters and weight of trade
  - All traders — tb: binary | tt: # of times
  - Currency traders — cb: binary | ct: # of times | cv: volume
- Obs.
  - Weights of trades / focusing on currency → fine-grained
  - RMTers → only a few communities (1-8)

Evaluation metrics for RMTer detection

- Available human resource is unknown
  - It varies depending on the situation

- Two measures
  - Balance between Recall and Precision
    - Recall (R): how exhaustively RMTers are identified
    - Precision (P): how correctly system identifies RMTers
  - Avg. Precision at various recall

1 RMTer is found
2 RMTers are found
3 RMTers are found
…
All RMTers are found

Parameter selection of proposed method (1/2)

- Representation of trading network
  - All traders

All traders	tb: binary	tt: # of times	
Currency traders	cb: binary	ct: # of times	cv: volume

- Measure for in-community trades
  - **tt**: # of trade transactions
  - **ct**: # of currency transactions
  - **cv**: Total volume of traded currency
- Measure for trades of individual character
  - **tt**: # of trade transactions
  - **ct**: # of currency transactions
  - **cv**: Total volume of traded currency

Parameter selection of proposed method (2/2)

- 45 combinations → 10
  - Representation of trading network (5)
    - Different network achieved the best result in different period
  - Measure for in-community trades (3 → 1)
    - Volume of traded currency (**cv**) > # of transactions (**tt**, **ct**)
  - Measure for trades of individual character (3 → 2)
    - Traded currency (**ct**, **cv**) > All trade (**tt**)
- Implications
  - Large amount of currency is exchanged for RMT
    - RMTers dealt with more than 1/3 of total currency trades
  - Virtual currency is popular in RMT
    - Buyers want virtual currency

Baselines: direct assessment of each char.

- Sort characters based on handled currency (cv)
- Two supervised methods (w/o constants)
  - Naïve Bayes
    - with multinomial distribution [McCallum+, 98]

$$Score(c) = \sum_{a \in A(c)} freq(a, c) \log \frac{P(a|\mathrm{RMTer})}{P(a|\text{non-RMTer})}$$

  - Support Vector Machines [Vapnik, 99]
    - with linear kernel (SVM^{light} is used)

$$Score(c) = \sum_{x_i \in X} y_i \alpha_i K(x_i, c)$$

{+1: RMTer, -1: non-RMTer}

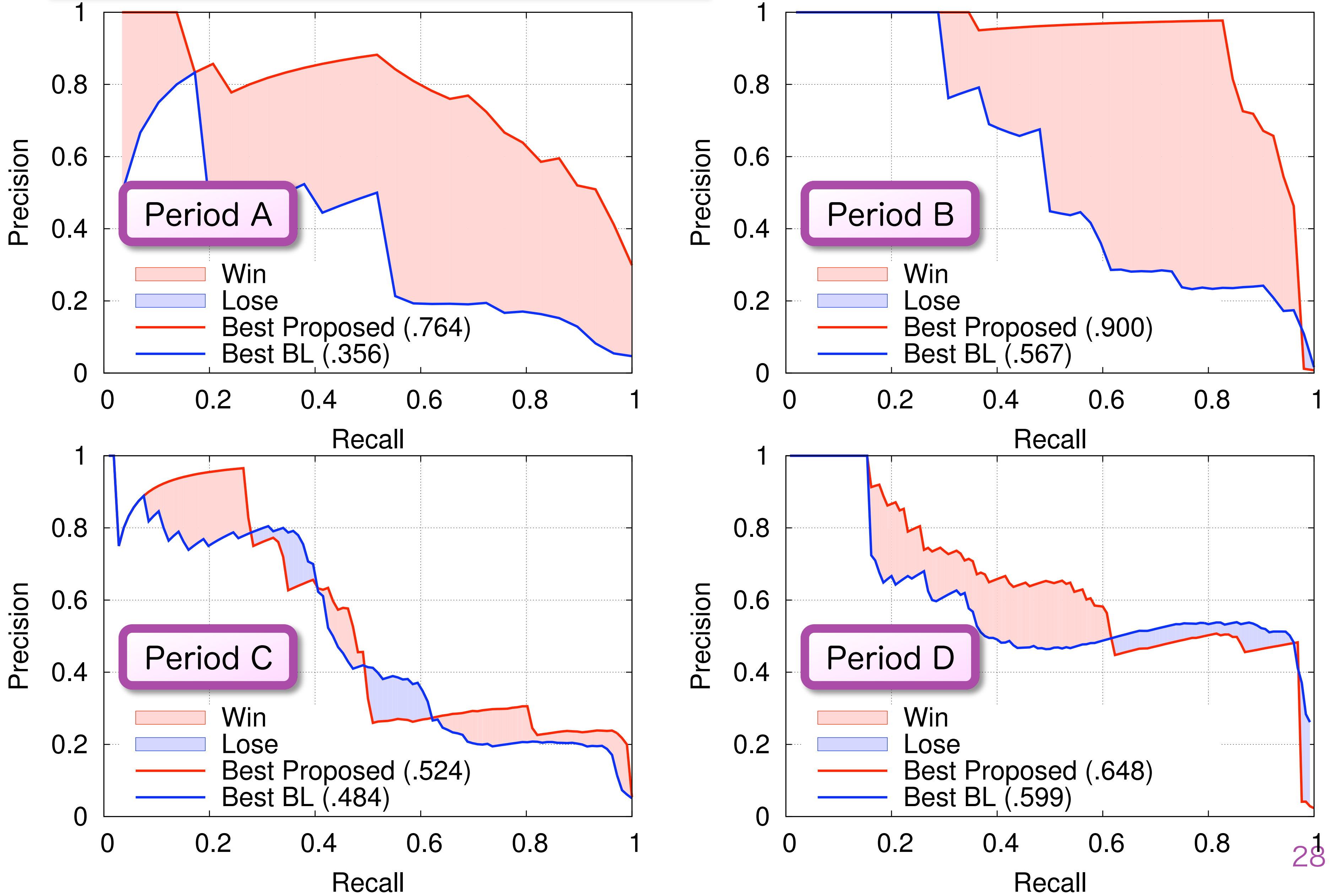
  - Feature: all of 338 types of actions
    - trade, attack to other player, find an item, invest for a ship

Average Precision

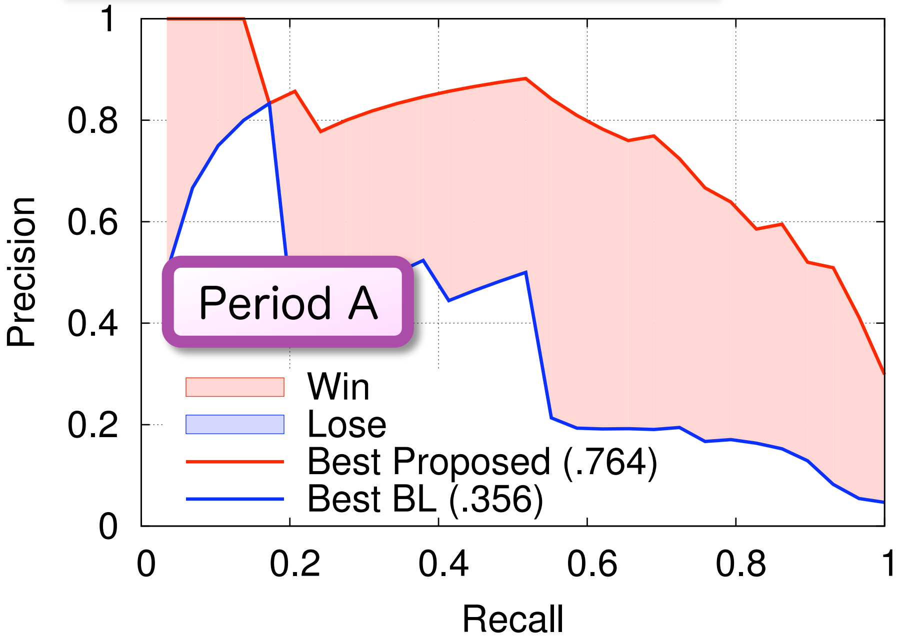
- Several versions beat all the baselines
  - But nothing significantly wins in all periods

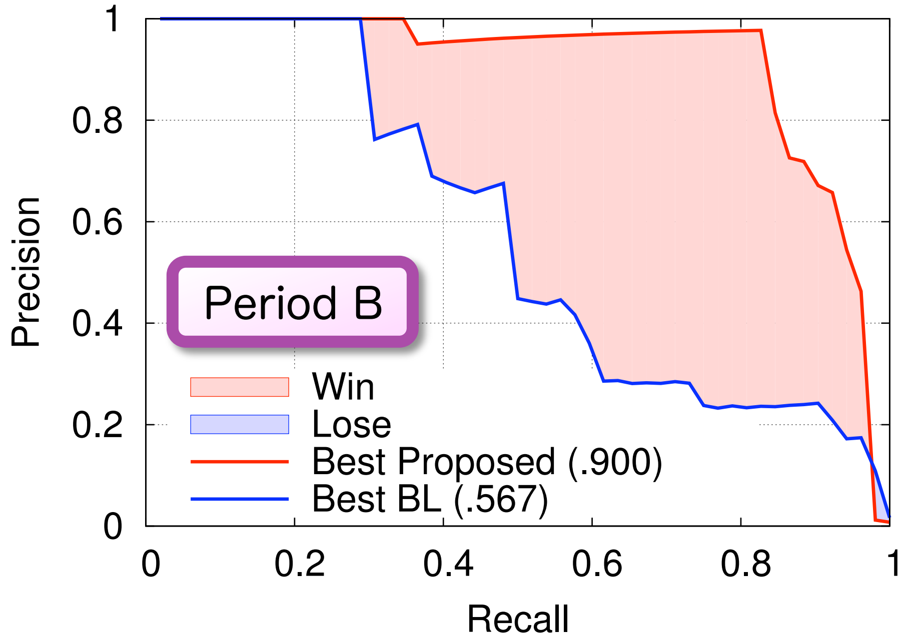
Model	Target char. set	Period A $N = 29$	Period B $N = 52$	Period C $N = 106$	Period D $N = 130$
cv	Currency traders	0.320	0.440	0.484	*0.466
MNB	All chars	0.239	0.305	0.342	0.357
	Traders	0.273	0.367	0.381	0.416
	Currency traders	0.336	0.391	0.420	*0.469
SVMs	All chars	0.340	0.198	0.438	0.517
	Traders	0.310	0.567	0.408	0.553
	Currency traders	0.356	0.554	0.421	*0.599
Proposed	tb.cv.ct	0.385	**0.900**	0.499	0.404
	tb.cv.cv	0.393	**0.860**	0.503	0.388
	tt.cv.ct	0.328	**0.882**	0.459	**0.648**
	tt.cv.cv	0.362	**0.837**	0.448	**0.624**
	cb.cv.ct	0.167	**0.883**	**0.524**	*0.570
	cb.cv.cv	0.179	**0.832**	**0.510**	*0.554
	ct.cv.ct	**0.764**	0.626	**0.515**	*0.557
	ct.cv.cv	**0.756**	0.606	0.498	*0.540
	cv.cv.ct	**0.522**	0.573	**0.513**	*0.547
	cv.cv.cv	**0.547**	0.564	0.498	*0.529

Recall - Precision curves



Successful cases

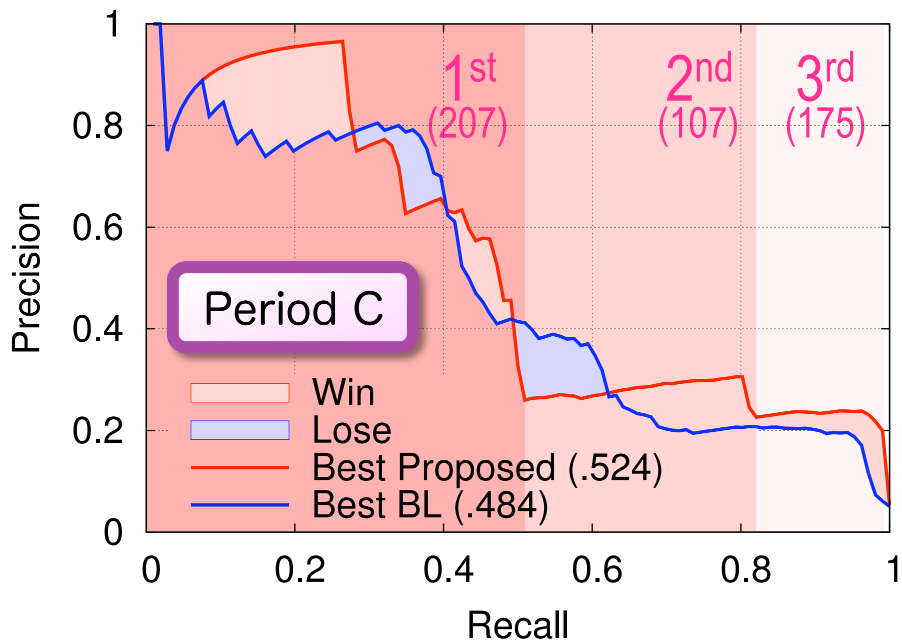


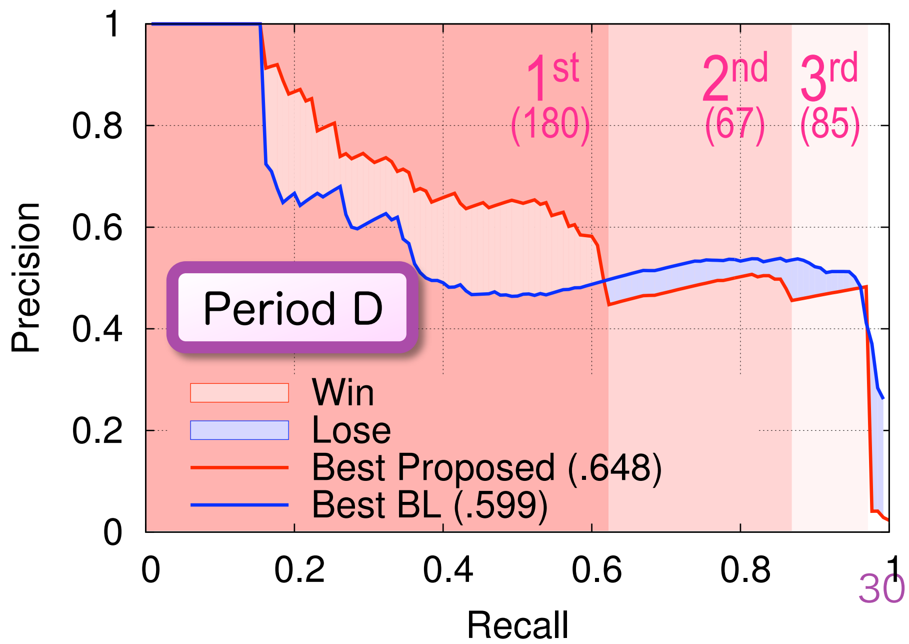


- Significant improvement
  - Both on R-P curves and Avg. Prec.
  - Most RMTers → a single, small, and top-rank community
    - Period A: 29 RMTers → **28** + 1
    - Period B: 52 RMTers → **50** + 1 + 1
  - Some are still difficult to detect

Relatively unsuccessful cases

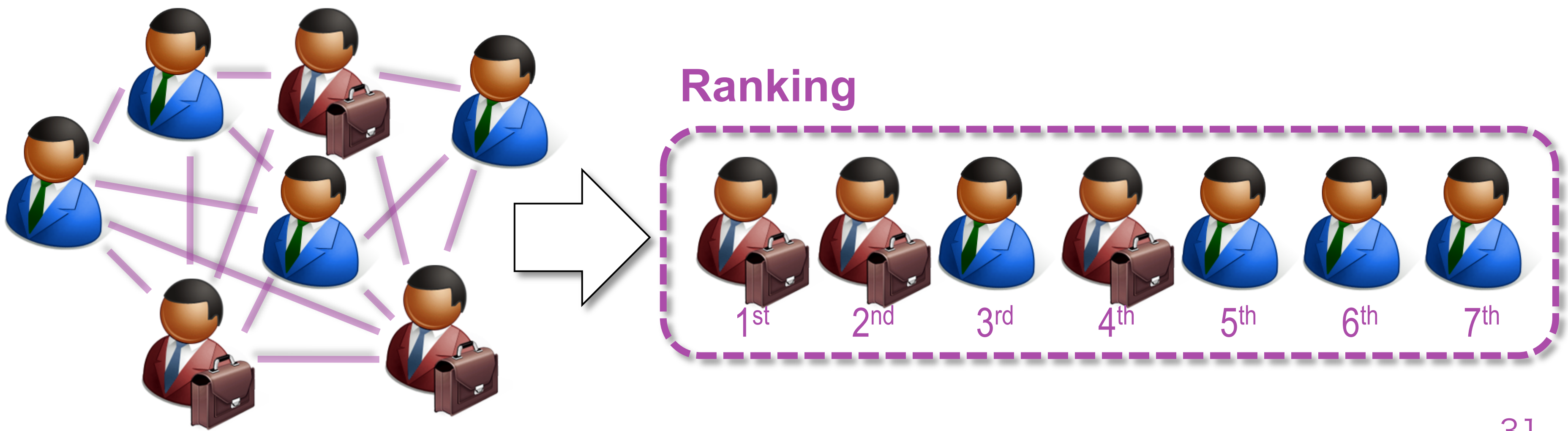
- Weak for plural RMTer communities
  - Period C: 106 RMTers → **53** + **33** + **19** + 1
  - Period D: 130 RMTers → **80** + **32** + **14** + 2 + 1 + 1
- Need a more intelligent ranking
  - e.g., Combination of ranks (community, character)
  - e.g., Re-ranking based on operators' judge





Conclusion

- Detection of RMTers in MMORPG
  - As a ranking problem
  - Wholesale arrest thru capturing communities
    - Low exchange rate → division of labor & frequent trade
  - Evaluation using actual log data
    - Better performance than separately assessing each char.
    - w/ a room of further improvement



Future work

- Technical aspect
  - Further investigation into trading network
    - Mixture models [Newman+, 07]
    - Augmentation with other components [Ahmad+, 11]
  - Apply state-of-the-art machine learning techniques

- Evaluation
  - Is arms race really overcome?
    - e.g., Robustness against disposal use of characters
  - Application to other MMORPGs