Linking Named Entities in Tweets with Knowledge Base via User Interest Modeling

Wei Shen[†], Jianyong Wang[†], Ping Luo[‡], Min Wang[♯]
[†]Tsinghua University, Beijing, China
[‡]HP Labs China, Beijing, China
[♯]Google Research, Mountain View, USA

KDD 2013
Chicago, Illinois USA

Outline

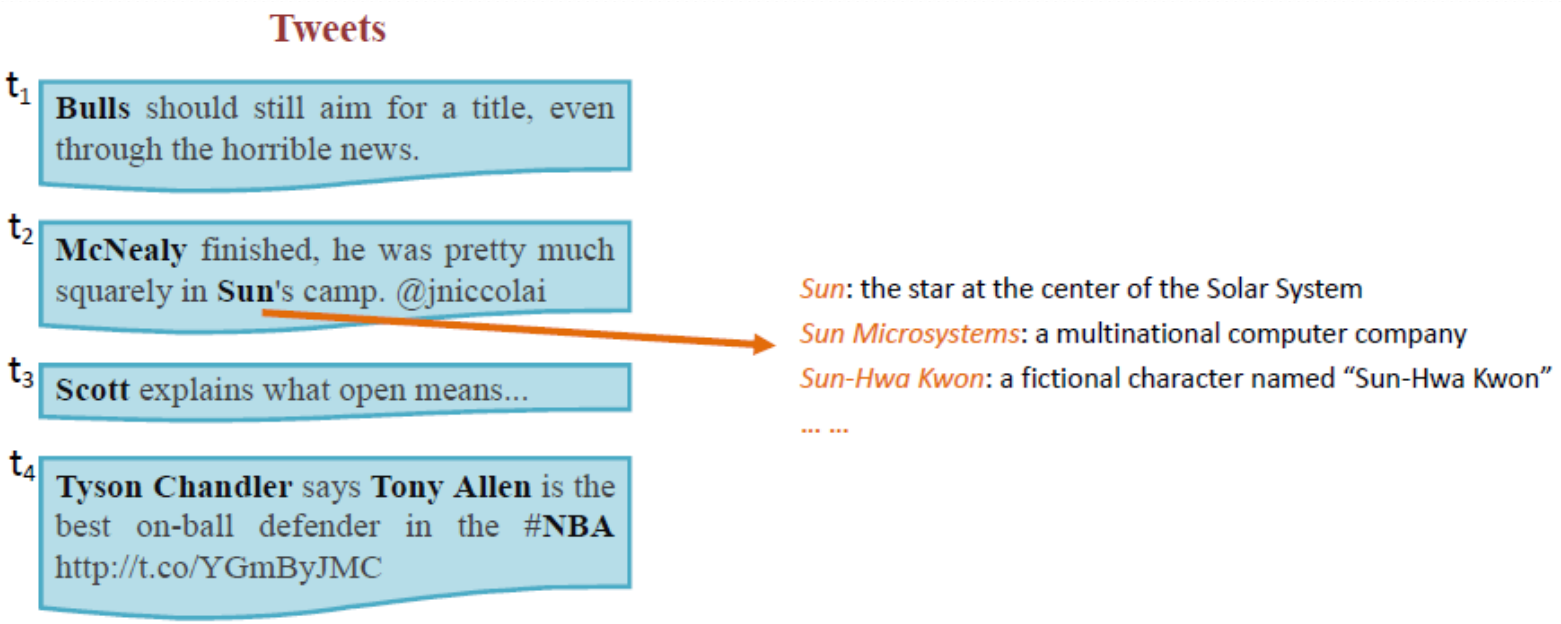
- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

Outline

- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

Motivation

- Twitter: important information source
- Tweets: social status updates about topics ranging from daily life to news events



Motivation

- Many large scale knowledge bases have emerged
  - Dbpedia, YAGO, Freebase, Probase, and etc.

- Bridging these knowledge bases with the collection of tweets

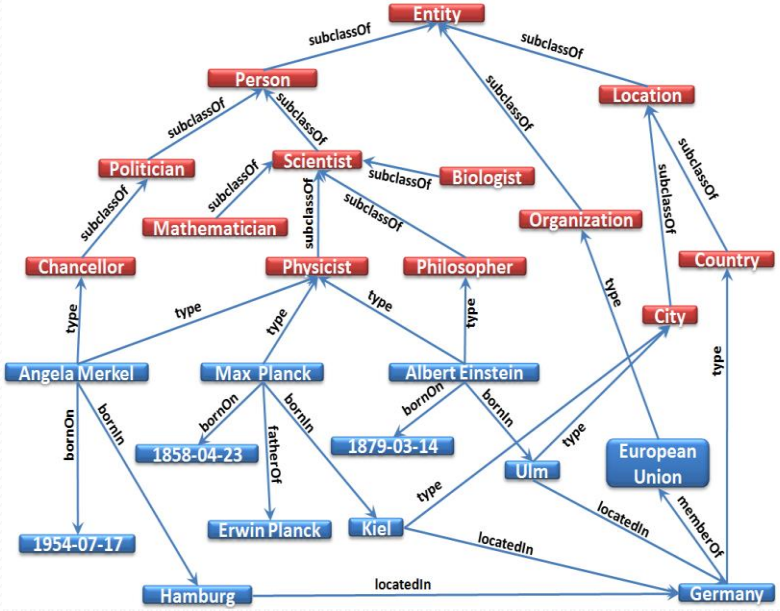


Figure 1: An example of YAGO knowledge base

Source: From Information to Knowledge:Harvesting Entities and Relationships from Web Sources. PODS'10.

Outline

- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

Problem definition

- Tweet entity linking
  - link the textual named entity mentions detected from tweets with their mapping entities existing in a knowledge base

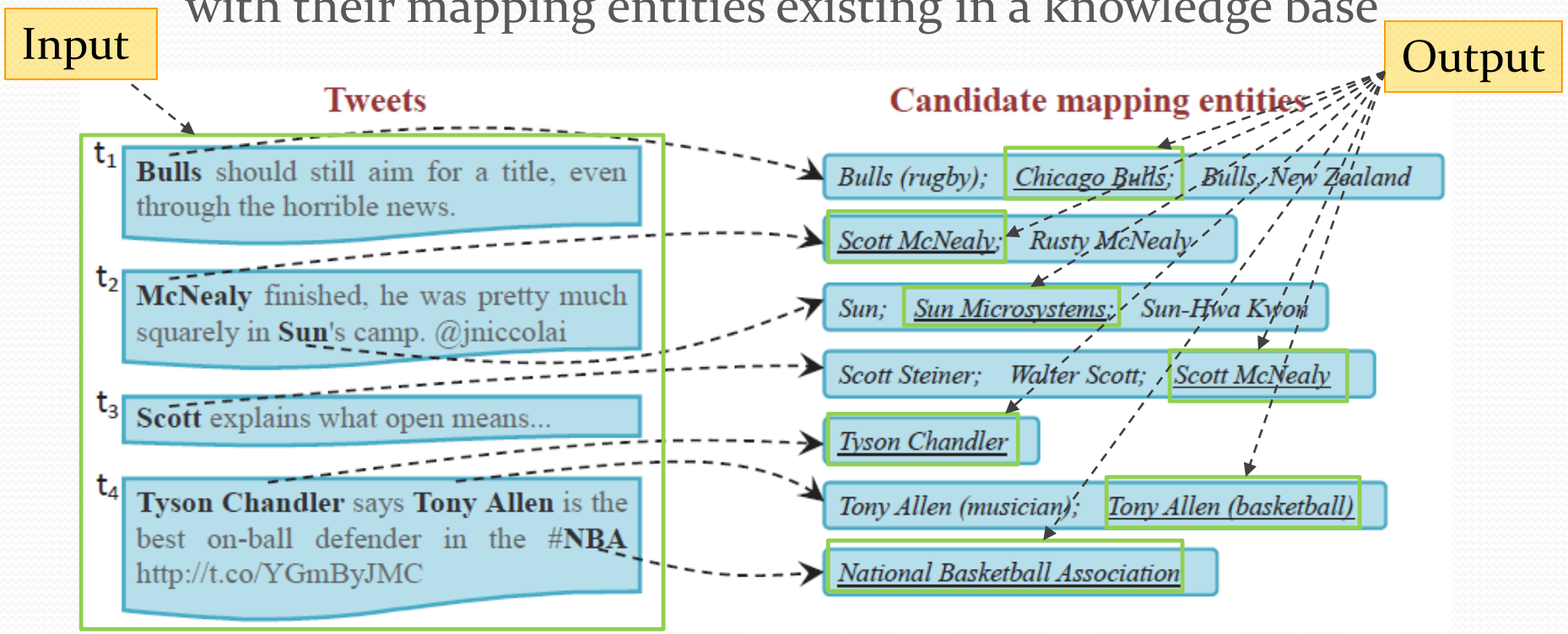


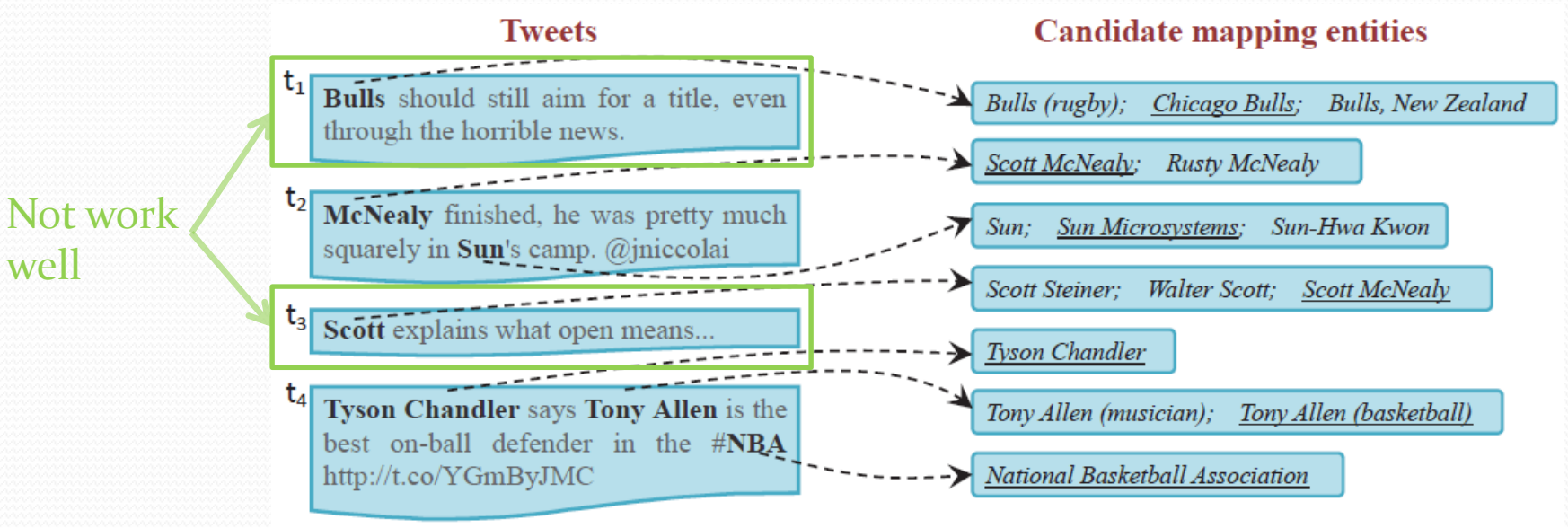
Figure 2: An illustration of the tweet entity linking task. Named entity mentions detected in tweets are in bold; candidate mapping entities for each entity mention are generated by a dictionary-based method and ranked by their prior probabilities in decreasing order; true mapping entities are underlined.

Applications

- Twitter user interest discovery
- Twitter users Recommendation
- Tweets recommendation and re-ranking
- Entity information collection from Twitter
  - e.g., products and celebrities

Tweet entity linking

- Challenge
  - noisy, short, and informal nature of tweets
- Previous entity linking methods (EACL'06, EMNLP'07, KDD'09, SIGIR'11, EMNLP'11, and WWW'12)
  - focus on linking entities in Web documents
  - Context Similarity
  - Topical Coherence

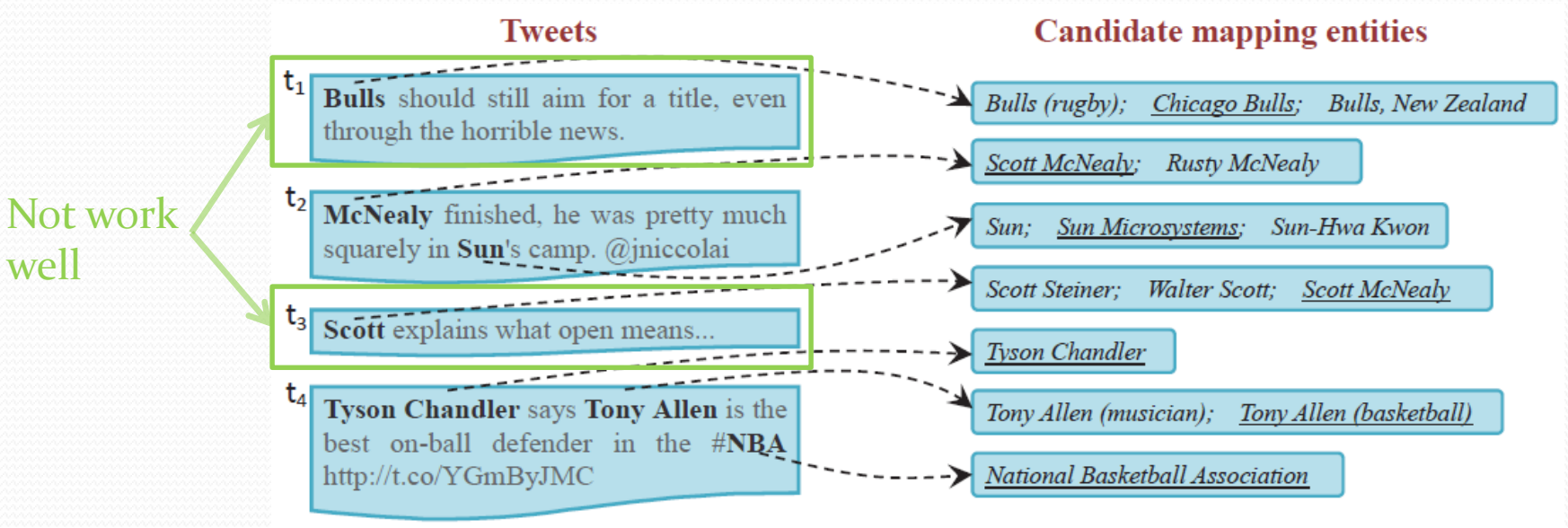


Tweet entity linking

- Challenge
  - noisy, short, and informal nature of tweets
- Previous entity linking methods (EACL'06, EMNLP'07, KDD'09, SIGIR'11, EMNLP'11, and WWW'12)
  - focus on linking entities in Web documents
  - Context Similarity
  - Topical Coherence
- We can increase the linking accuracy, if we
  - combine intra-tweet local information
  - with inter-tweet user interest information

Tweet entity linking

- Challenge
  - noisy, short, and informal nature of tweets
- Previous entity linking methods (EACL'06, EMNLP'07, KDD'09, SIGIR'11, EMNLP'11, and WWW'12)
  - focus on linking entities in Web documents
  - Context Similarity
  - Topical Coherence



Outline

- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

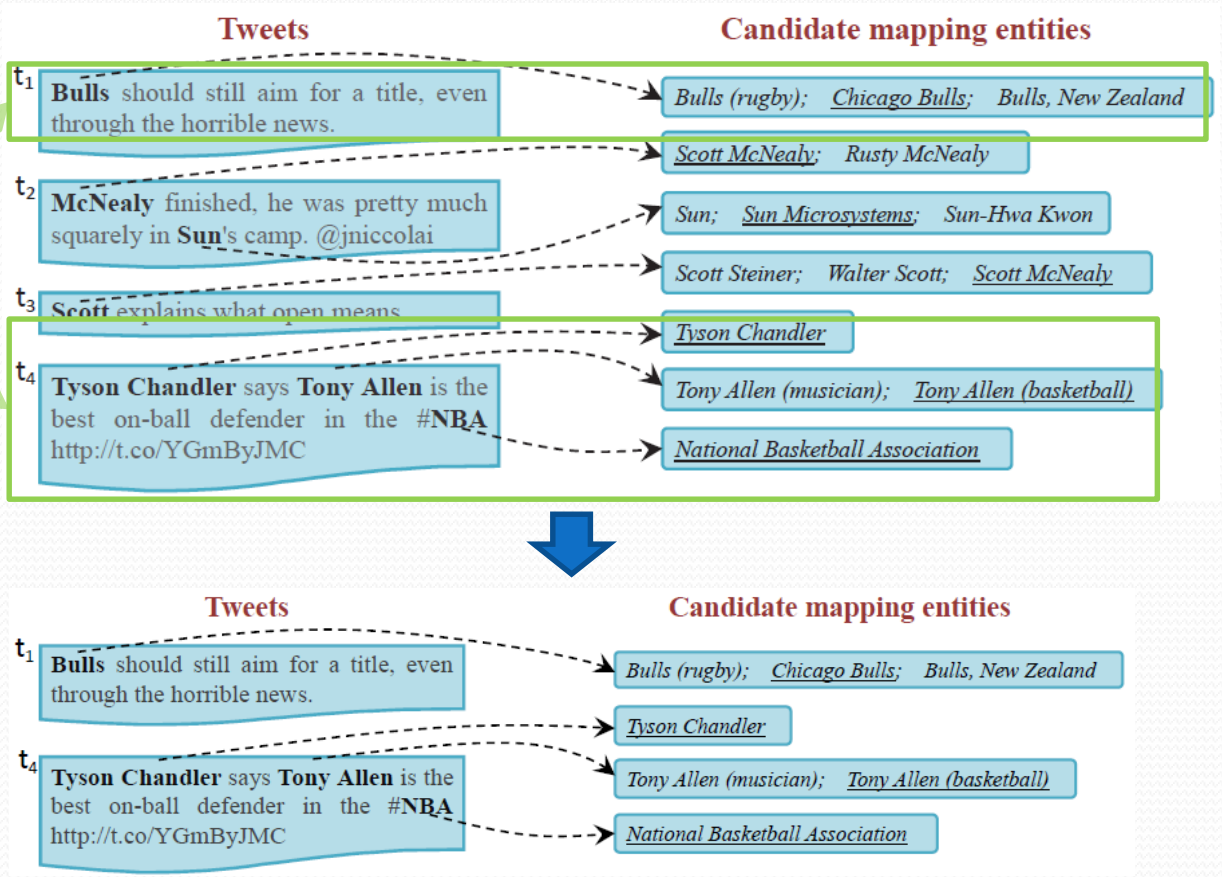
KAURI Framework

- Assumption 1.
  - Each Twitter user has an underlying topic interest distribution over various topics of named entities.
- Assumption 2.
  - If some named entity is mentioned by a user in his tweet, that user is likely to be interested in this named entity.
- Assumption 3.
  - If one named entity is highly topically related to the entities a user is interested in, that user is likely to be interested in this named entity as well.

Graph construction

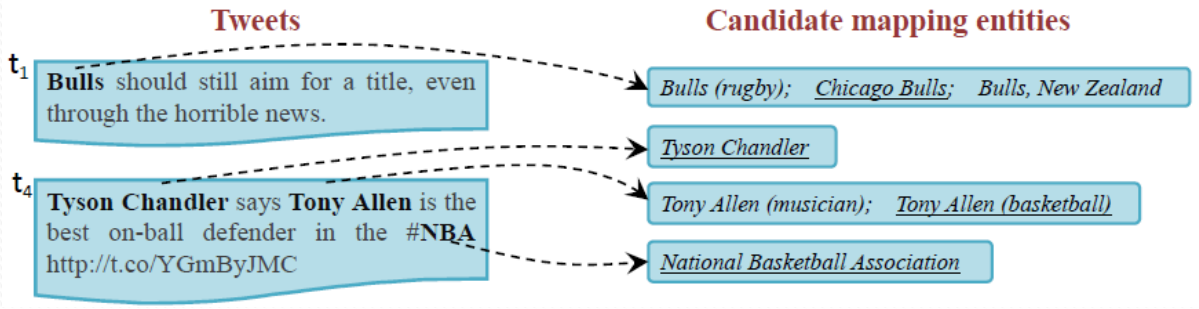
- For each Twitter user,
  - we construct a graph

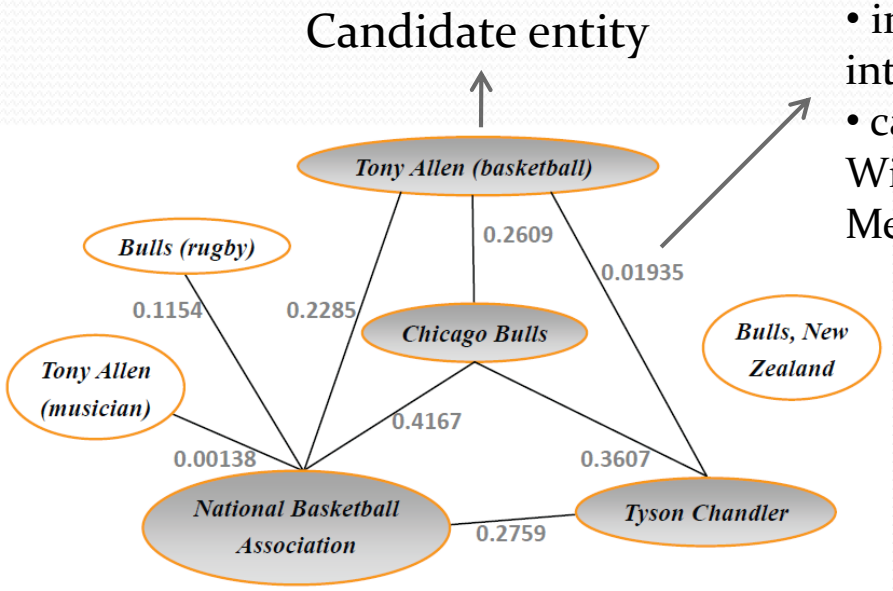
Running example: Example 1



Example 1

Graph construction





Weight:
- indicating the strength of interdependence
- calculated using the Wikipedia Link-based Measure [1].

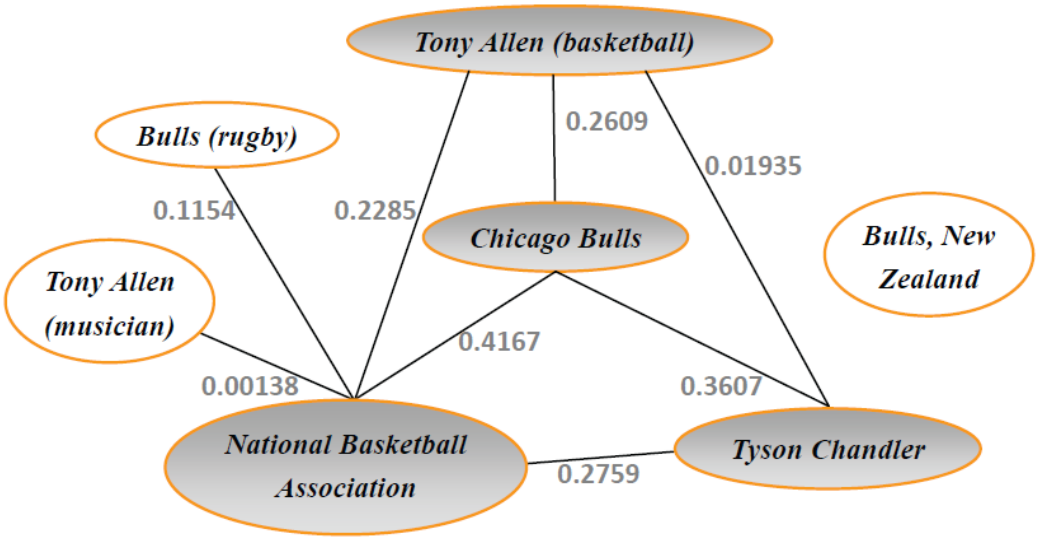
[1] D. Milne and I. H. Witten. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *WIKIAI'08*.

Graph construction

- Assumption 1.
  - Each Twitter user has an underlying topic interest distribution over various topics of named entities.

$r^i_{j,q}$: candidate mapping entity

$s^i_{j,q}$: interest score indicating the strength of the user's interest in it



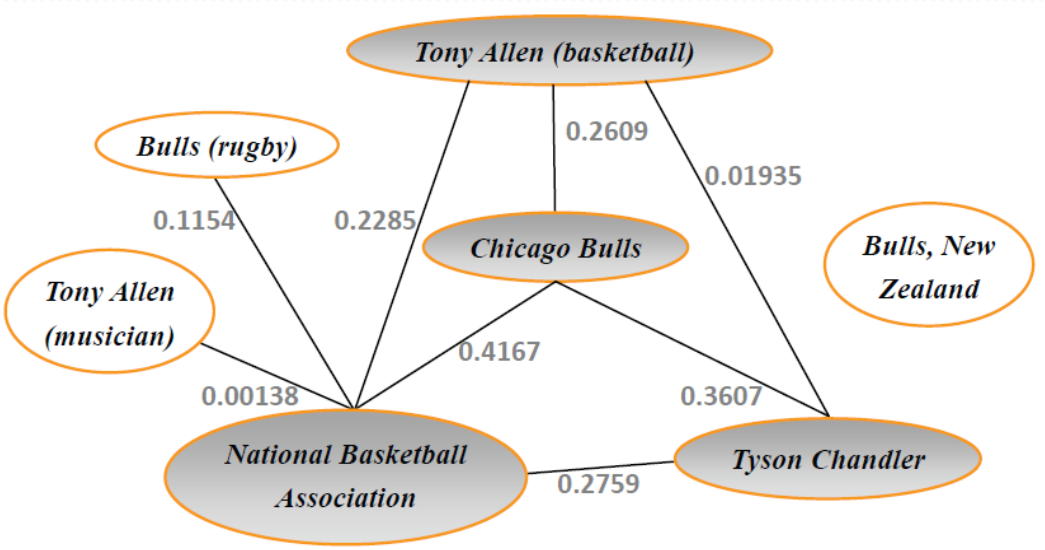
Graph construction

- Assumption 2.
  - If some named entity is mentioned by a user in his tweet, that user is likely to be interested in this named entity.

$r_{j,q}^{i}$: candidate mapping entity

$s_{j,q}^{i}$: interest score indicating the strength of the user's interest in it

$p_{j,q}^{i}$: initial interest score estimated from intra-tweet local information



Initial interest score estimation

The initial interest score

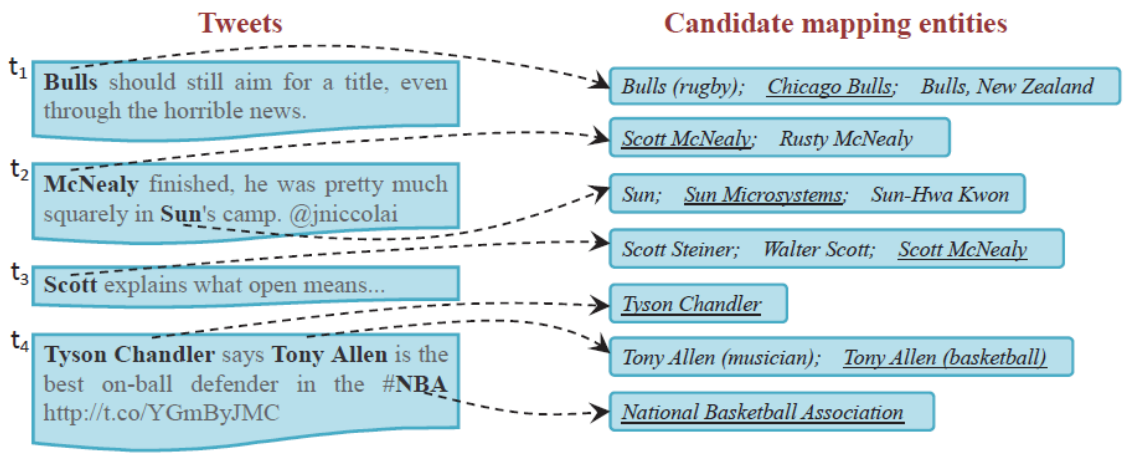
Context similarity

$$p^i_{j,q} = \alpha * Pp(r^i_{j,q}) + \beta * Sim(r^i_{j,q}) + \gamma * Coh(r^i_{j,q})$$

Prior probability

Topical coherence in tweet

entity frequency in the Wikipedia article corpus



Initial interest score estimation

The initial interest score

Context similarity

bag of words cosine similarity

$$p^i_{j,q} = \alpha * Pp(r^i_{j,q}) + \beta * Sim(r^i_{j,q}) + \gamma * Coh(r^i_{j,q})$$

Prior probability

entity frequency in the Wikipedia article corpus

Topical coherence in tweet

estimated using the *iterative substitution algorithm* proposed in [2]

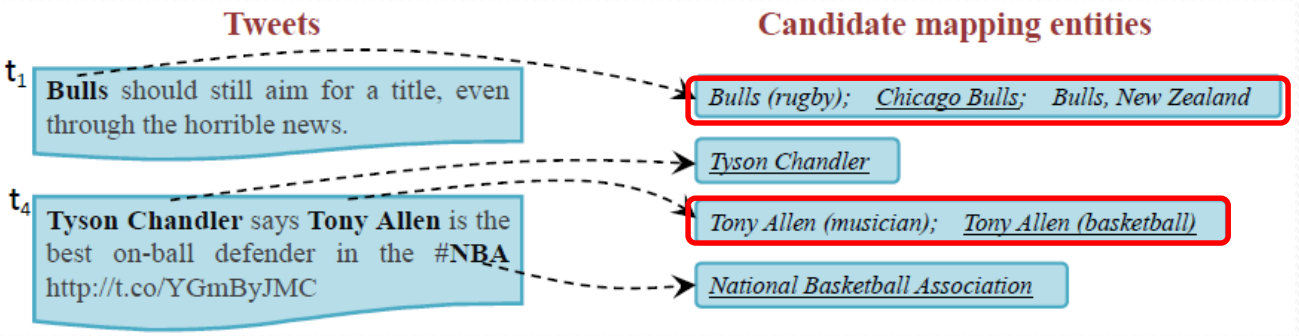
$\overrightarrow{w} = \langle \alpha, \beta, \gamma \rangle$ ← Weight vector

$\alpha + \beta + \gamma = 1$

- We utilize the **max-margin** technique to automatically learn the weight vector which gives proper weights for those three intra-tweet local features.

[2] W. Shen, J. Wang, P. Luo, and M. Wang. Liege: Link entities in web lists with knowledge base. In *SIGKDD'12*.

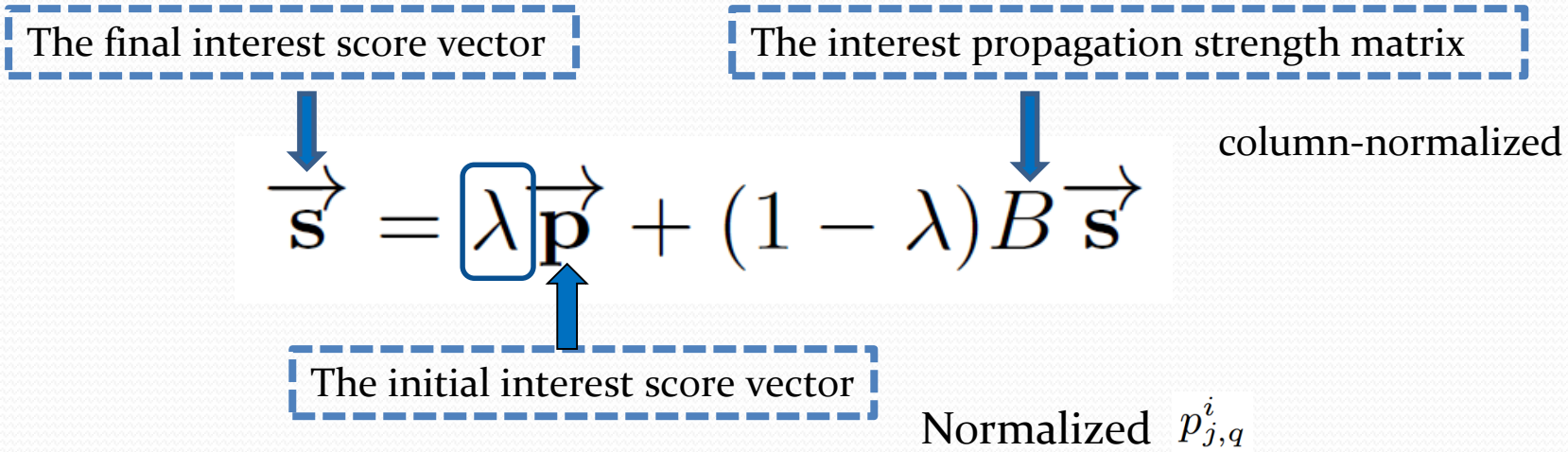
Initial interest score estimation



$r^i_{j,q}$	*Bulls (rugby)*	*Chicago Bulls*
$p^i_{j,q}$	0.13	0.0492
$r^i_{j,q}$	*Bulls, New Zealand*	*Tyson Chandler*
$p^i_{j,q}$	0.0208	0.318
$r^i_{j,q}$	*Tony Allen (musician)*	*Tony Allen (basketball)*
$p^i_{j,q}$	0.145	0.155
$r^i_{j,q}$	*National Basketball Association*	
$p^i_{j,q}$	0.402	

Table 2: The initial interest scores for candidate mapping entities in Example 1

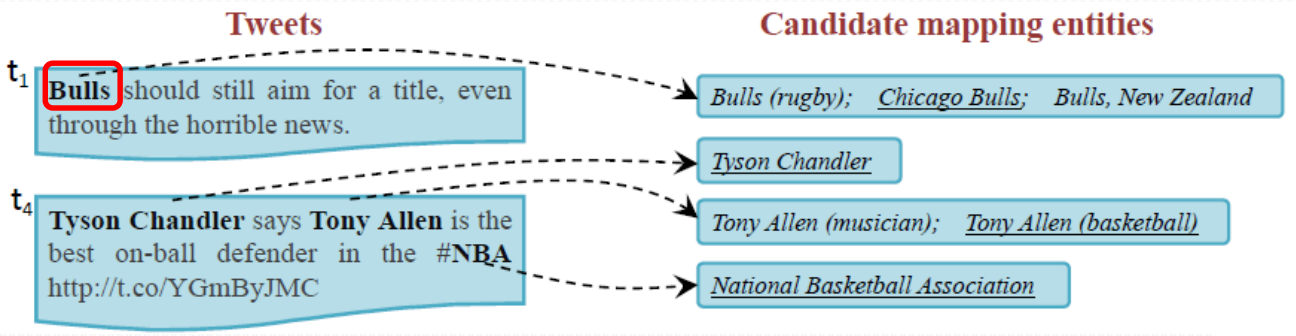
User interest propagation algorithm

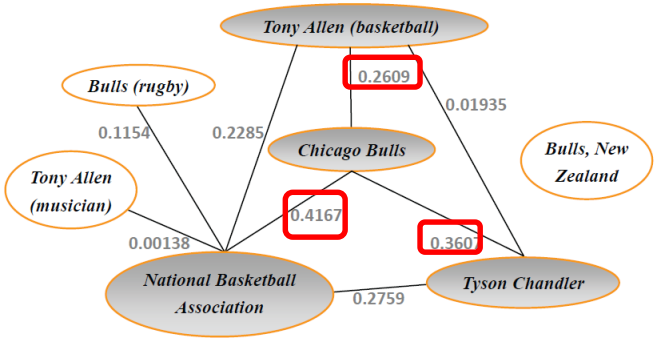


- Initialization: $\vec{\mathbf{s}} = \vec{\mathbf{p}}$

- Then apply this formula iteratively until $\vec{\mathbf{s}}$ stabilizes within some threshold

User interest propagation algorithm





$r^i_{j,q}$	Bulls (rugby)	Chicago Bulls
$p^i_{j,q}$	0.13	0.0492
$r^i_{j,q}$	Bulls, New Zealand	Tyson Chandler
$p^i_{j,q}$	0.0208	0.318
$r^i_{j,q}$	Tony Allen (musician)	Tony Allen (basketball)
$p^i_{j,q}$	0.145	0.155
$r^i_{j,q}$	National Basketball Association	
$p^i_{j,q}$	0.402	

Table 2: The initial interest scores for candidate mapping entities in Example 1

$r^i_{j,q}$	Bulls (rugby)	Chicago Bulls
$s^i_{j,q}$	0.0624	0.189
$r^i_{j,q}$	Bulls, New Zealand	Tyson Chandler
$s^i_{j,q}$	0.00682	0.194
$r^i_{j,q}$	Tony Allen (musician)	Tony Allen (basketball)
$s^i_{j,q}$	0.0476	0.122
$r^i_{j,q}$	National Basketball Association	
$s^i_{j,q}$	0.297	

Table 3: The final interest scores for candidate mapping entities in Example 1

Outline

- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

Experimental setting

- Data sets

# Twitter user	20
# tweets	3818
# tweets having at least one named entity mention	1721
# named entity mentions	2918
# uncertain named entity mentions	241
# test named entity mentions	2677
# linkable named entity mentions	2240
# unlinkable named entity mentions	437

Table 4: A summary of the gold standard data set

- Weight learning:
  - Two-fold cross validation

Experimental results

Method	Linkable		Unlinkable		All	
	#	Accu.	#	Accu.	#	Accu.
LINDEN	1852	0.827	353	0.808	2205	0.824
$\text{LOCAL}_{\beta=0,\gamma=0}$	1784	0.796	355	0.812	2139	0.799
$\text{LOCAL}_{\gamma=0}$	1795	0.801	355	0.812	2150	0.803
$\text{LOCAL}_{\beta=0}$	1862	0.831	355	0.812	2217	0.828
LOCAL_{full}	1863	0.832	355	0.812	2218	0.829
$\text{KAURI}_{\beta=0,\gamma=0}$	1882	0.840	356	0.815	2238	0.836
$\text{KAURI}_{\gamma=0}$	1894	0.846	357	0.817	2251	0.841
$\text{KAURI}_{\beta=0}$	1913	0.854	371	0.849	2284	0.853
KAURI_{full}	**1923**	**0.858**	**373**	**0.854**	**2296**	**0.858**

Table 5: Experimental results over the data set

$$p^i_{j,q} = \alpha * Pp(r^i_{j,q}) + \beta * Sim(r^i_{j,q}) + \gamma * Coh(r^i_{j,q})$$

LINDEN is the model proposed in [3] to address the task of linking entities in Web documents.

[3] W. Shen, J. Wang, P. Luo, and M. Wang. Linden: linking named entities with knowledge base via semantic knowledge. In *WWW'12*.

Incremental update

Time	1	2	3	4	5
Tweet index	11-160	21-170	31-180	41-190	51-200
# mentions	1979	1988	2005	2013	2016
# added nodes	827	598	726	775	780
# added edges	172296	139398	122824	217440	201368
Accuracy	0.853	0.859	0.859	0.855	0.858
Incremental annot. time (s)	35.58	30.22	28.13	45.36	42.50
Incremental annot. time per mention (ms)	17.98	15.20	14.03	22.53	21.08

Table 6: Incremental update performance of KAURI

Outline

- Motivation
- Problem Definition
- KAURI Framework
- Experiments
- Conclusion

Conclusion

- **A novel problem**
  - tweet entity linking
- **KAURI**
  - a graph-based framework that unifies
    - intra-tweet local information
    - with inter-tweet user interest information
- **Good performance**
  - significantly outperforms the baseline methods in terms of accuracy
  - efficient and scales well to tweet stream

Thanks!

Question?