Pattern Based Recommendation

Work in Progress

Koen Verstrepen, prof. dr. Bart Goethals

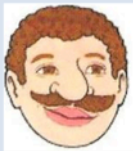
DBDBD
29/11/2013
Rotterdam

Universiteit Antwerpen

Binary, Positive-only data

						
	1	1		1		
	1		1			1
		1		1		
		1	1		1	1

What else does Anna like?

						
	1	1		1		
Anna 	1	?	1	?	?	1
		1		1		
		1	1		1	1

Notation

e_k

$\mathcal{U}$ \ $\mathcal{E}$						
	1	1		1		
u_i 	1	$\hat{p}(e_k\|E_{u_i})$	1			1
		1		1		
		1	1		1	1

E_{u_i} ={  ,  ,  }

U_{e_k} ={  ,  ,  }

User-based Nearest Neighbors

- Oldest
- Most intuitive

							cos(., )
Anna	1		1			1	1
Wendy		1	1		1	1	0.6
Carlos	1	1		1			0.3
Joe		1		1			0

$\mathcal{N}(u_i)$

User-based Nearest Neighbors

							cos(., )
	1		1			1	1
		1	1		1	1	0.6
	1	1		1			0.3
		1		1			0
$\hat{p}(e_k \mid E_{u_i})$		0.9		0.3	0.6		

Item-based Nearest Neighbors

	1		1		?	1
		1	1		1	1
	1	1		1		
		1		1		
cos(. ,)		0.4		0.5	0	
cos(. ,)		0.4		0	0.7	
cos(. ,)		0.4		0	0.7	
$\hat{p}(e_k \mid E_{u_i})$		1.2		0.5	1.4	

- Different result
- can be better explained(?!)

Explainability : state-of-the-art

p( | )		
user-based	$\hat{p}(e_k \mid E_{u_i}) = \sum_{u_n \in \mathcal{N}(u_i)} sim(u_i, u_n) \cdot \lvert E_{u_n} \cap \{e_k\} \rvert$	
item-based	$\hat{p}(e_k \mid E_{u_i}) = \sum_{e_i \in E_{u_i}} sim(e_i, e_k) \cdot \lvert \mathcal{N}(e_i) \cap \{e_k\}$	 

We propose: Unifying framework

Algorithm 1: Unifying, pattern based framework

input : $\mathcal{D}$

output: $\mathcal{Q} = \{\hat{p}\left(e_k|E_{u_i}\right) | u_i \in \mathcal{U} \wedge e_k \in (\mathcal{E} \setminus E_{u_i})\}$

1 compute $\mathcal{P}$ ◁ `s.1`

2 **for** $u_i \in \mathcal{U}$ **do**

3 Select $\mathcal{P}(u_i) \subseteq \mathcal{P}$ ◁ `s.2`

4 **for** $e_k \in (\mathcal{E} \setminus E_{u_i})$ **do**

5 Select $\mathcal{P}(u_i, e_k) \subseteq \mathcal{P}(u_i)$ ◁ `s.3`

6 $\hat{p}\left(e_k|E_{u_i}\right) \leftarrow 0$

7 **for** $P \in \mathcal{P}(u_i, e_k)$ **do**

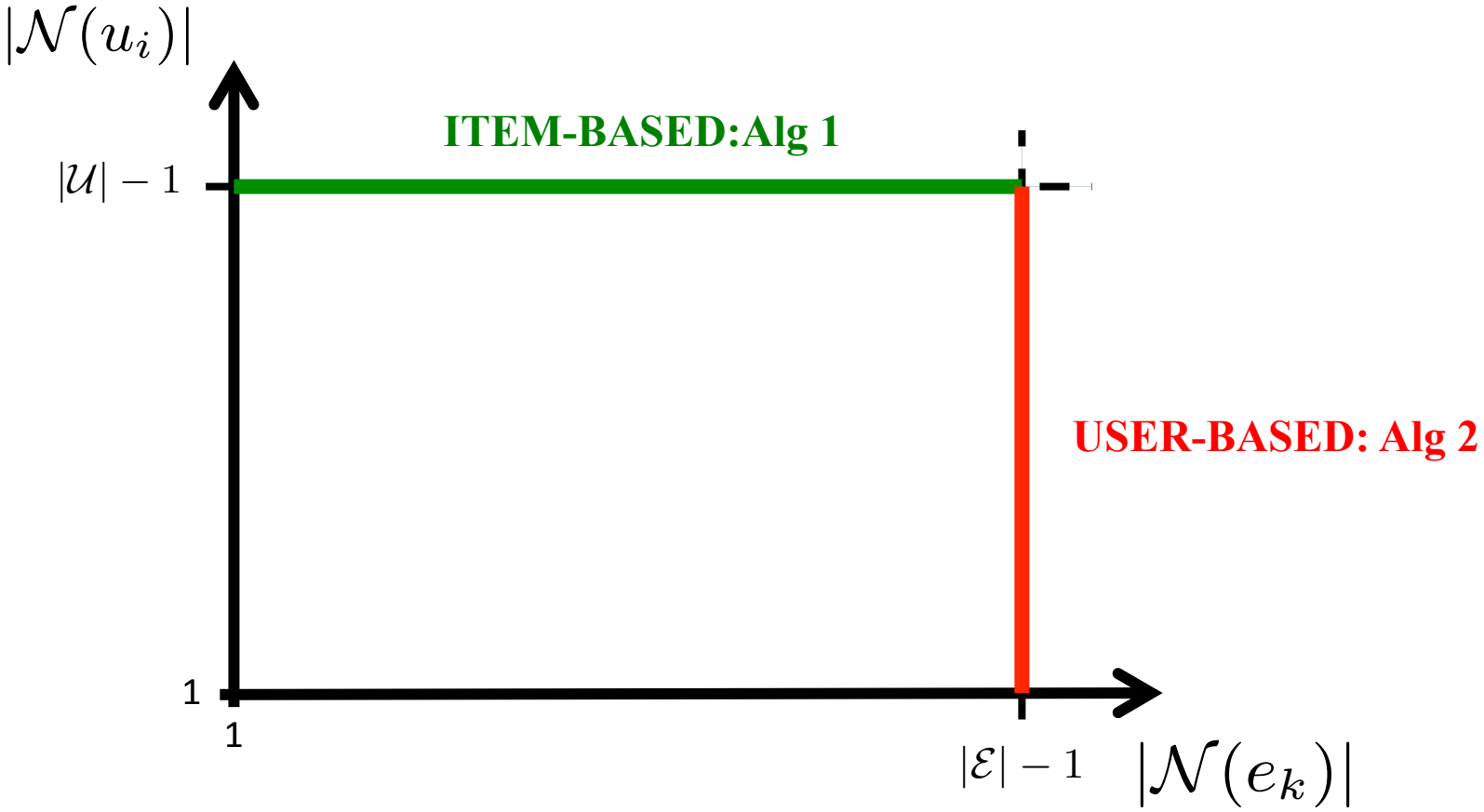
8 $\hat{p}\left(e_k|E_{u_i}\right) \leftarrow \hat{p}\left(e_k|E_{u_i}\right) + w(P, u_i, e_k)$ ◁ `s.4`

9 $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{\hat{p}\left(e_k|E_{u_i}\right)\}$

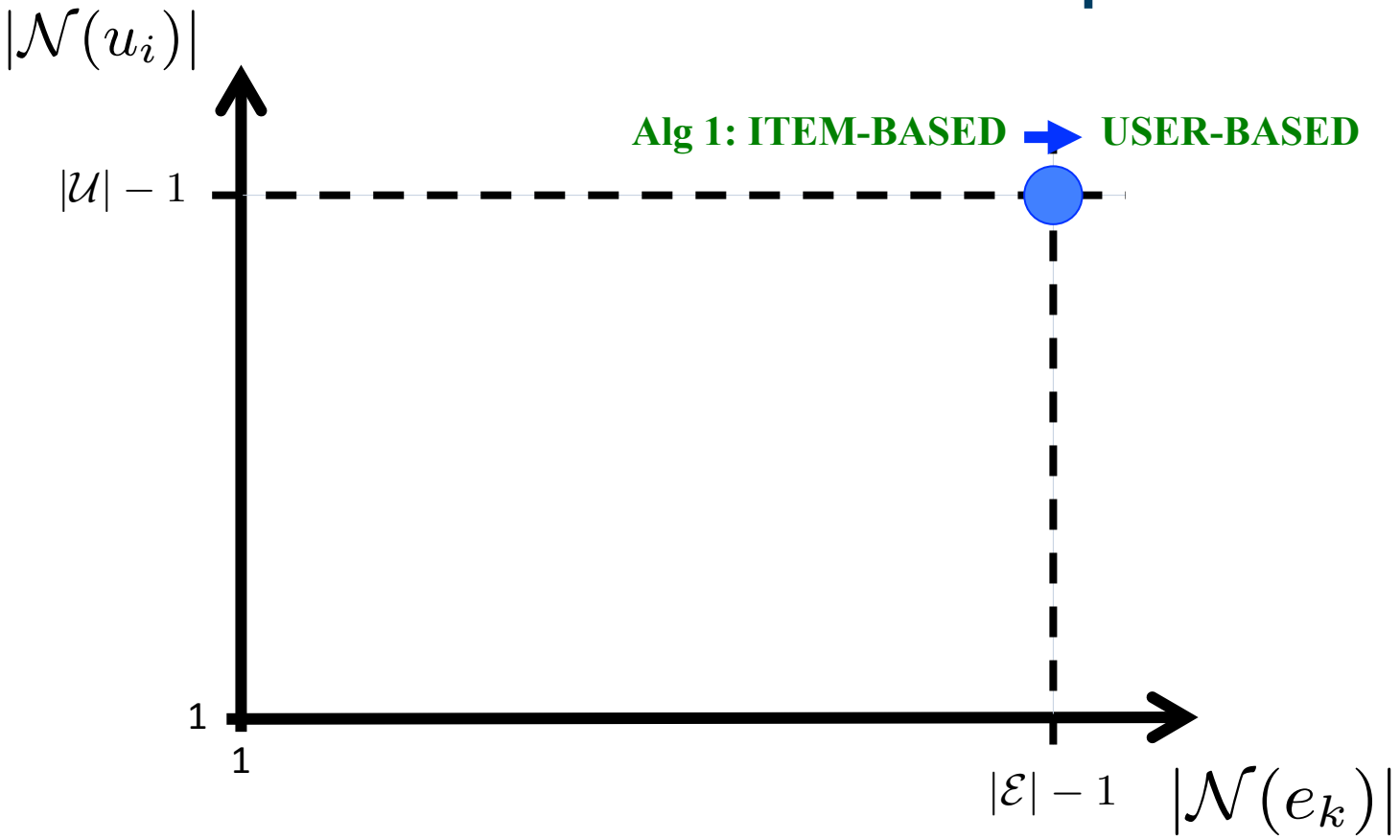
determines the type of algorithm

different flavors of the same algorithm

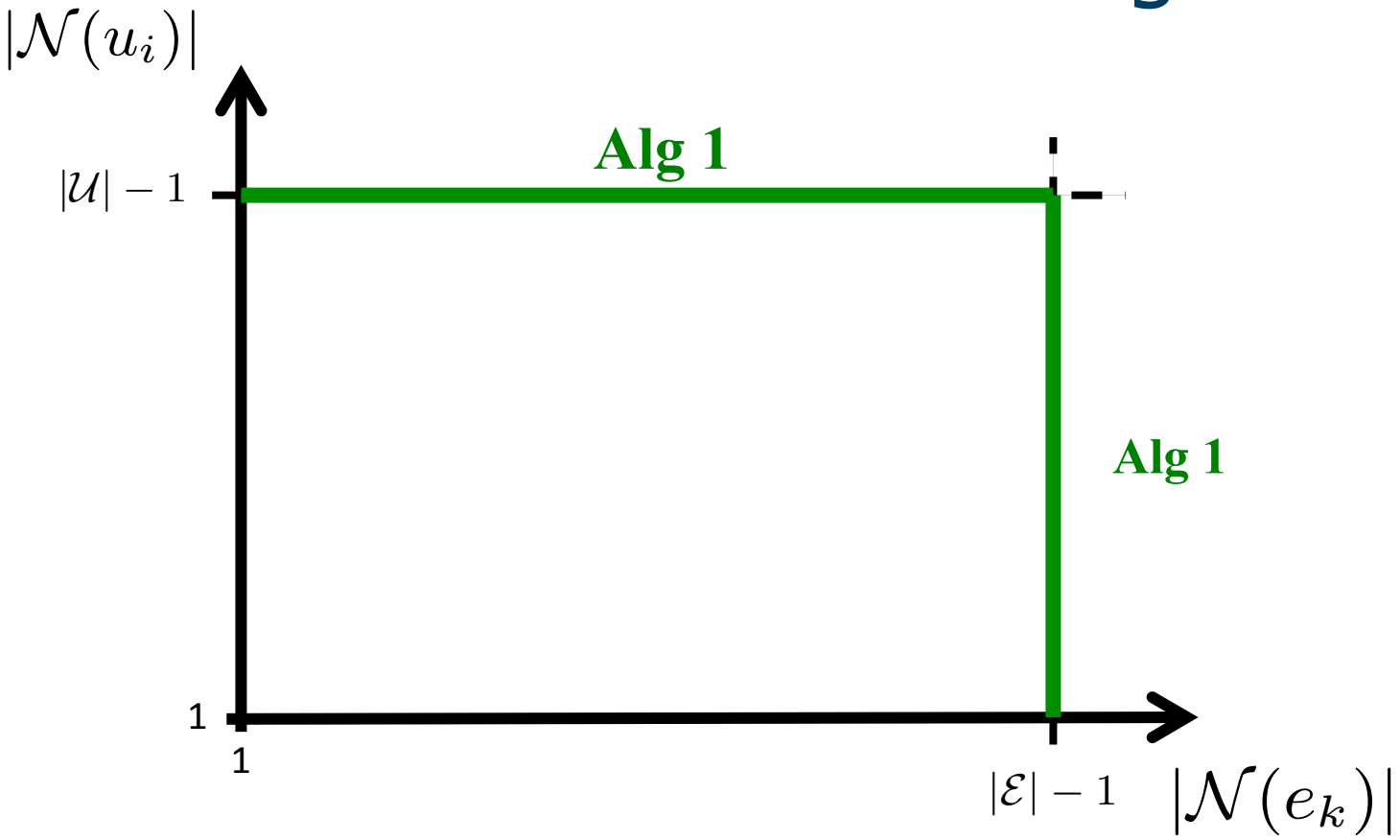
And there is more



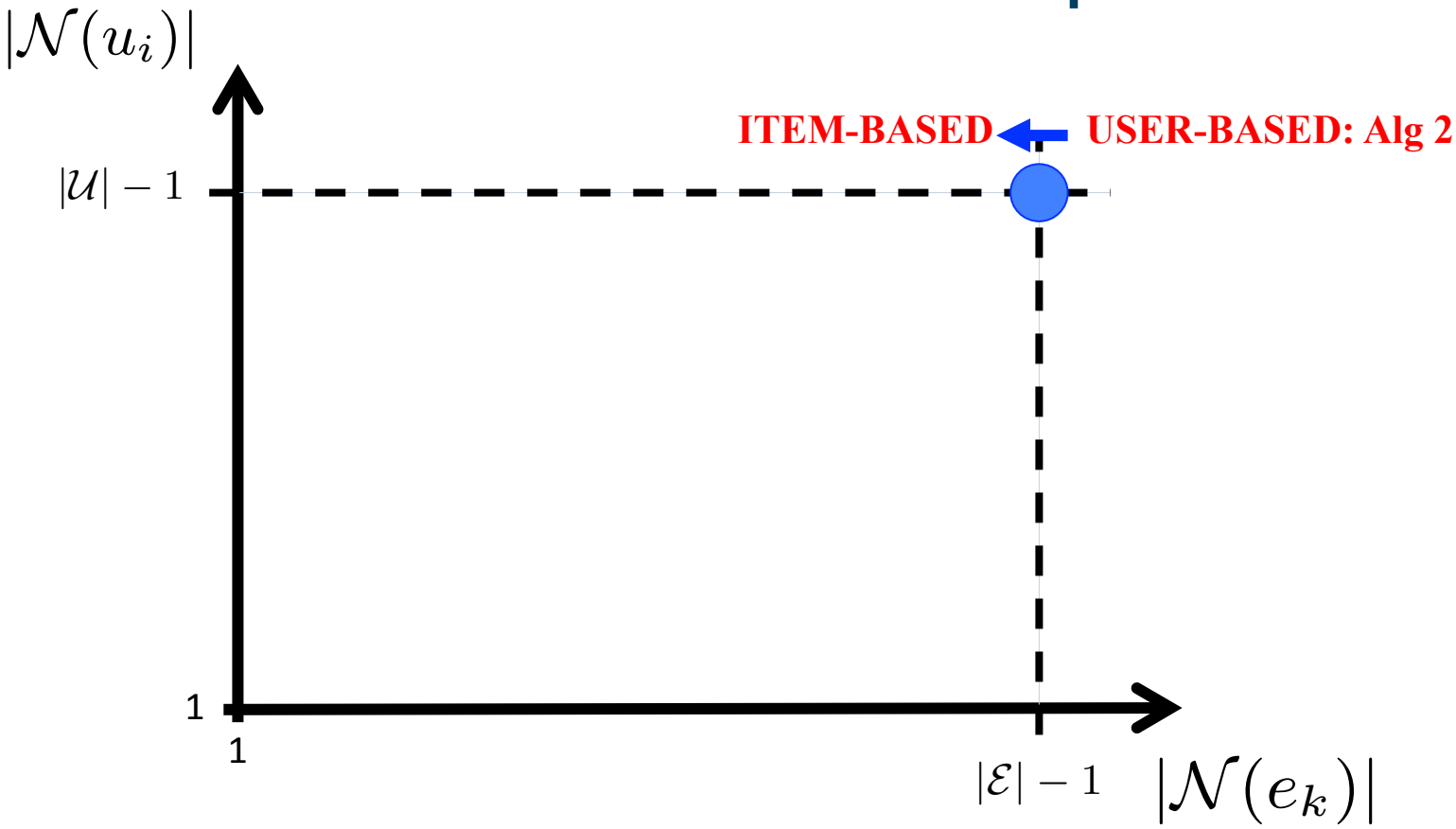
Proven Equivalence



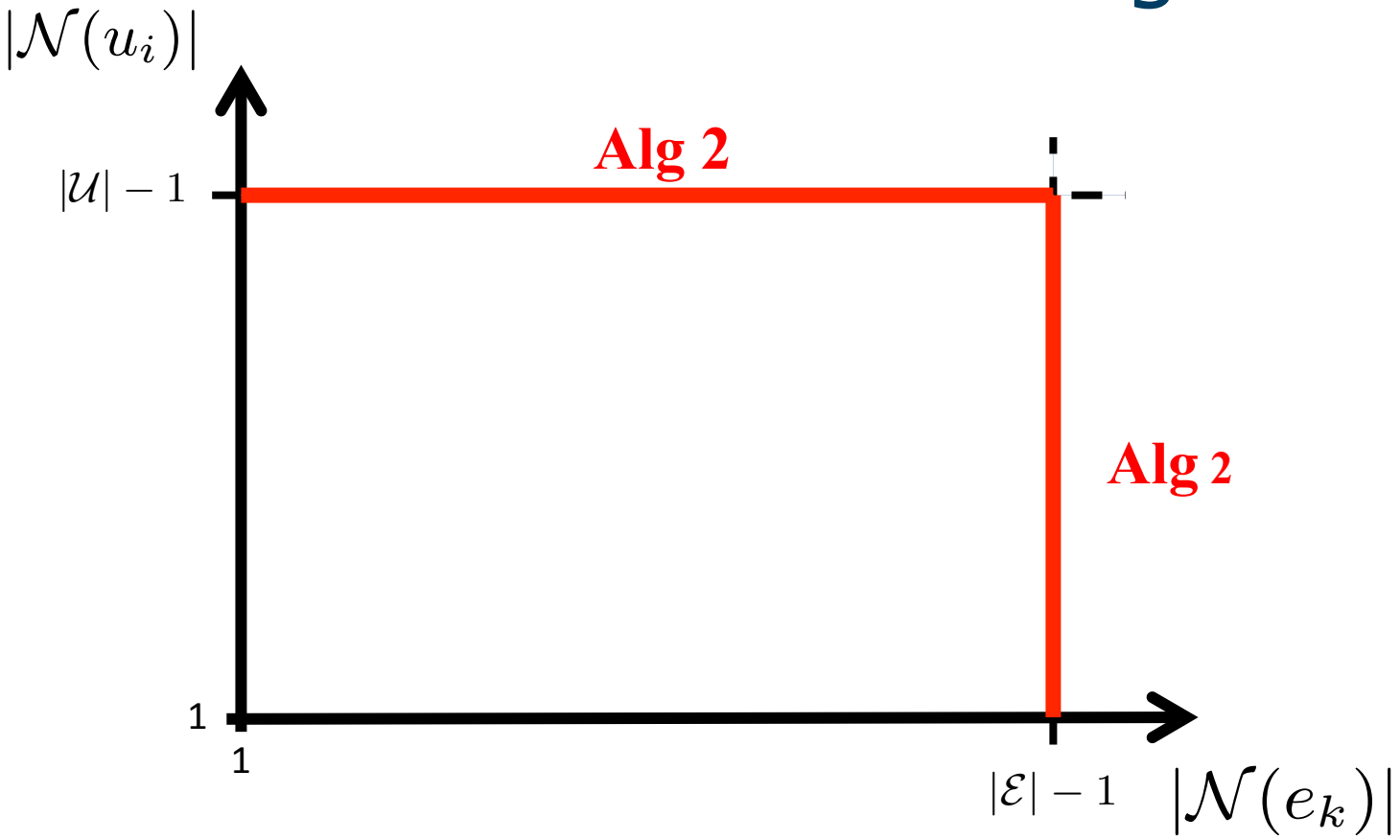
Novel algorithms



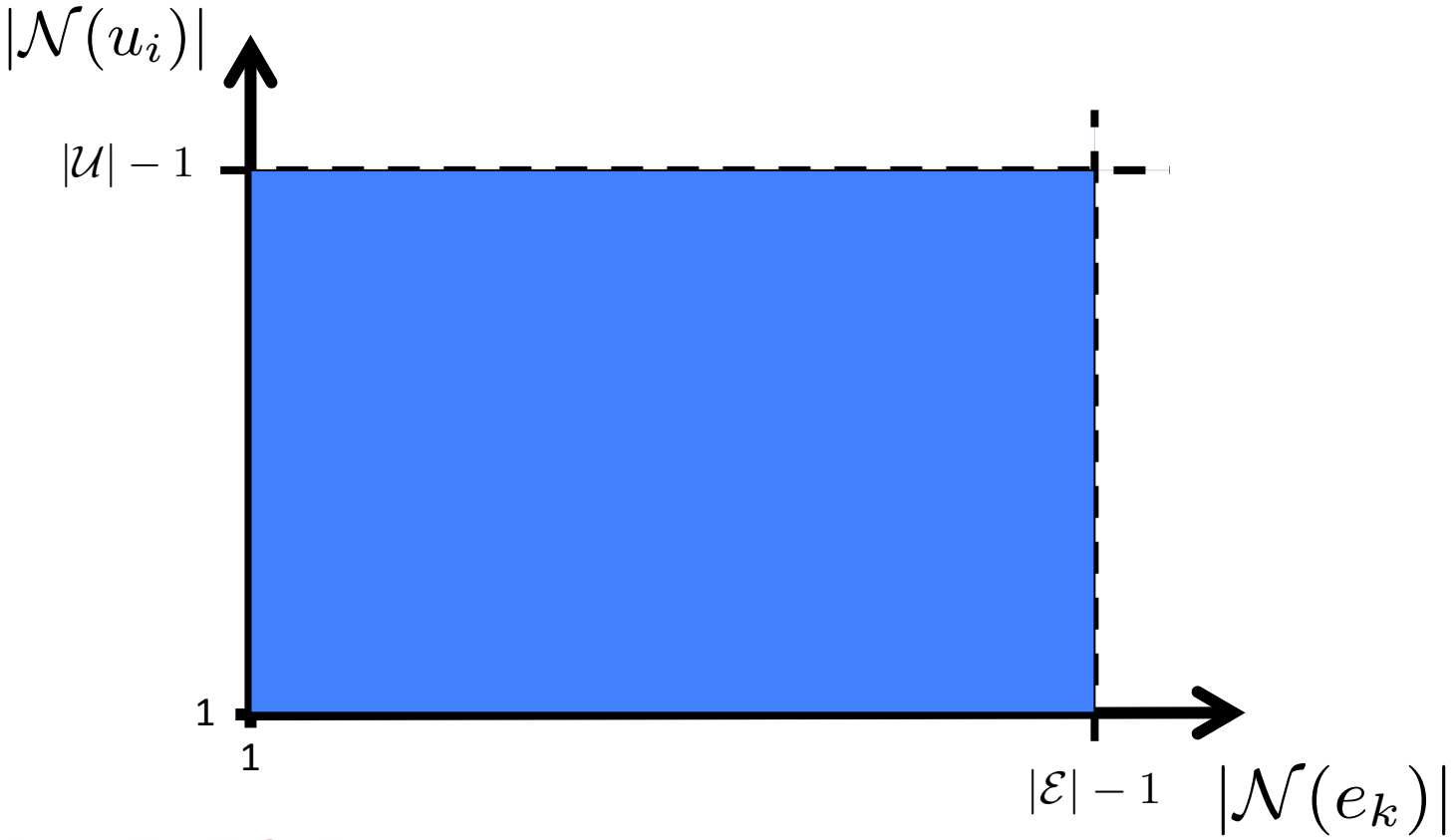
Equivalence



Novel algorithms



Further expansion of parameter space



Remember: Explainability

p( | )		
user-based	$\hat{p}(e_k \mid E_{u_i}) = \sum_{u_n \in \mathcal{N}(u_i)} sim(u_i, u_n) \cdot \lvert E_{u_n} \cap \{e_k\} \rvert$	
item-based	$\hat{p}(e_k \mid E_{u_i}) = \sum_{e_i \in E_{u_i}} sim(e_i, e_k) \cdot \lvert \mathcal{N}(e_i) \cap \{e_k\}$	 

Also user based algorithms are explainable

$$\mathcal{P} = \{E_u | u \in \mathcal{U}\}$$

$$\hat{p}\left(e_k | E_{u_i}\right) = \sum_{P \in \mathcal{P}(u_i)} w(P, u_i, e_k).$$

$$\vdots$$

$$\hat{p}\left(e_k | E_{u_i}\right) = \sum_{e_i \in E_{u_i}} \sum_{P \in \mathcal{P}(u_i)} F\left(P, u_i, e_i\right) w(P, u_i, e_k)$$

Summary

- Unifying framework
- Item-based ←→ User-based
  1. Novel algorithms
  2. Also user-based naturally explainable