Computational Intelligent Methods for Trusting in Social Networks

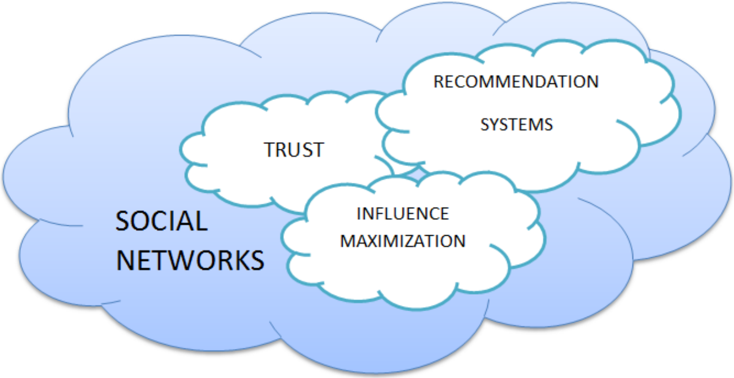
Thesis dissertation

José David Núñez González

Advisor: Prof. Manuel Graña
Computational Intelligence Group
University of the Basque Country (UPV/EHU)
Donostia

Feb 26, 2016

Abstract


RECOMMENDATION
SYSTEMS
TRUST
SOCIAL
NETWORKS
INFLUENCE
MAXIMIZATION

Abstract

- The **contents** of the thesis deals with:
  - proposal of **new feature extraction approaches**
  - **new influence maximization heuristic**
  - their **application** to social networks

Outline

Outline

1. Introduction
   - Introduction
2. Trust Prediction
   - State of the Art
   - Experimental work
3. Recommendation Systems
   - State of the Art
   - Recipe Generation experiments
   - Product Recommendation experiments
4. Influence Maximization
   - State of the Art
   - Experimental work
5. Conclusions
   - Conclusions

Definitions

Social Network

A social network is a social structure made up of a set of actors that are related according to some criterion.

$$G(V, E)$$

G: Social Graph
V: actors
E: relationships

Definitions

Social Network

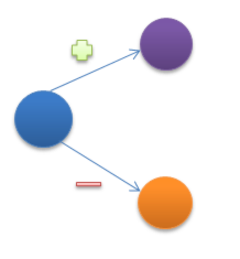
A social network is a social structure made up of a set of actors that are related according to some criterion.

$$G(V, E, W)$$

G: Social Graph
V: actors
E: relationships
W: labels

Definitions

Web of Trust



$$G(V, E, W)$$

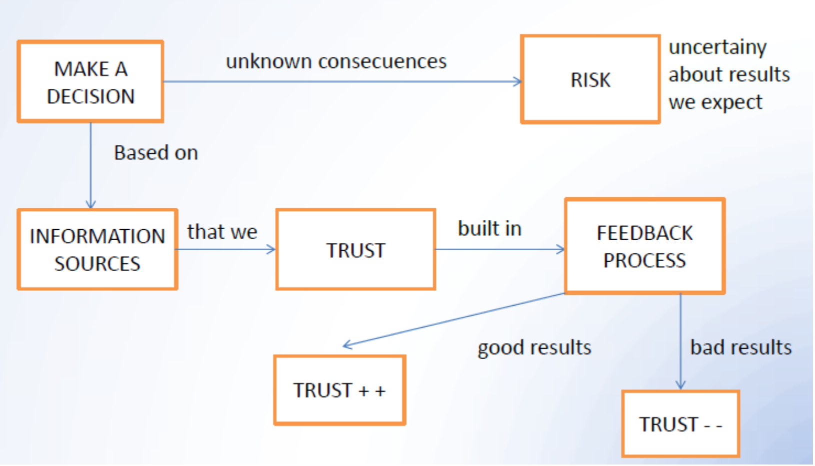
V: actors	**E**: relationships	**W**: trust values

Definitions

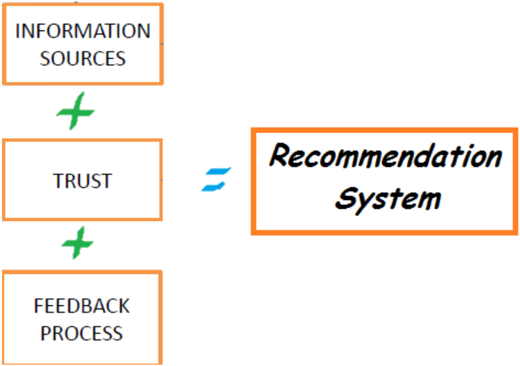
Trust

- the degree of subjective belief about the behaviors of a particular entity.
- the expectation that a service will be provided or a commitment will be fulfilled.
- confidence that one will find what is desired from another, rather than what is feared.

Trust


MAKE A DECISION
unknown consecuences
RISK
uncertainy about results we expect
Based on
INFORMATION SOURCES
that we
TRUST
built in
FEEDBACK PROCESS
good results
bad results
TRUST + +
TRUST - -

Recommendation Systems

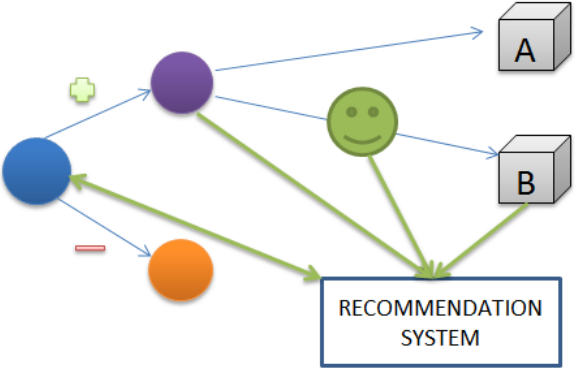


Definitions

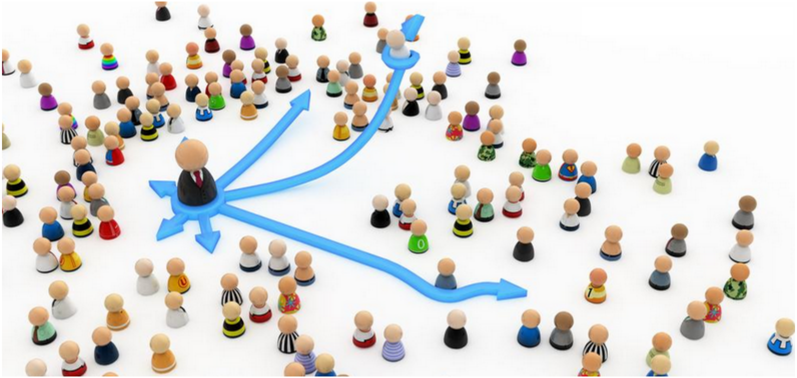
Recommendation Systems

- Information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item.

Recommendation Systems



Influence Maximization

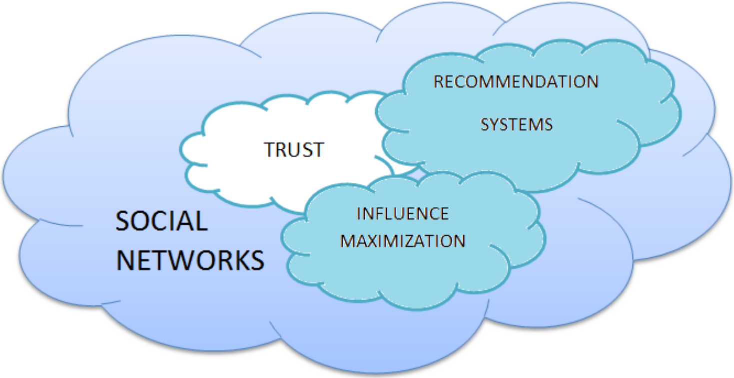


Definitions

Influence Maximization

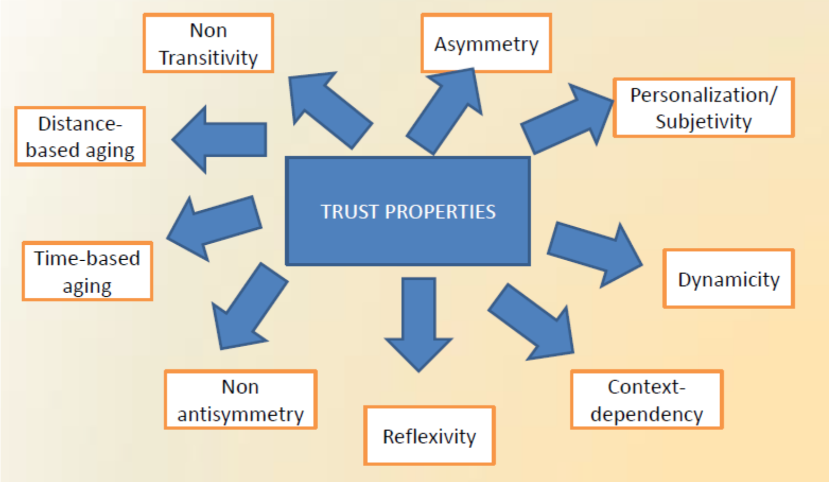
- Find the minimum K seed nodes (users) in a social network that could maximize the spread of influence.

Abstract


RECOMMENDATION
SYSTEMS
TRUST
SOCIAL
NETWORKS
INFLUENCE
MAXIMIZATION

Outline

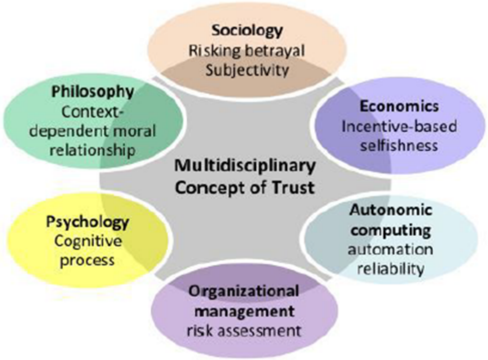
Trust - Properties



Trust - Metrics

- Binary state metrics
- Discrete scale metric
- Probabilistic metric
- Hybrid metric
- Negative values

Trust - Applications



Jin-Hee Cho, *Ananthram Swami, and Ing-Ray Chen*
A Survey on Trust Management for Mobile Ad Hoc Networks
IEEE COMMUNICATIONS SURVEYS & TUTORIALS,
VOL. 13, NO. 4, 2011 (PAGE 563)

Outline

Trust Prediction - Problem definition

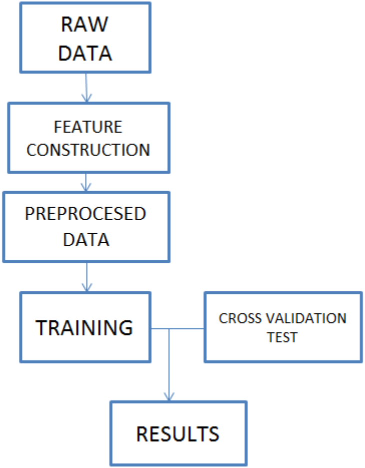
- Given a Web of Trust (WoT) associated to some social system, specificed by weighted graph G = (U;E;T), where there is a trust value associated with each edge, we want to predict how a User A would trusts another User B, positive or negatively.

Trust Prediction - Databases

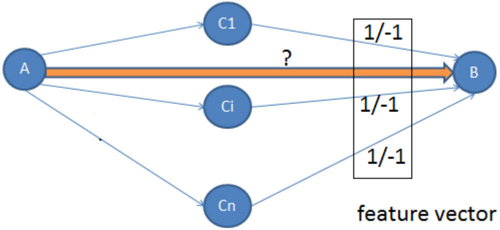
- Epinions and Wikipedia Voted Network databases
- Imbalanced databases

User Id	User Id	Trust Value
245	246	1
245	247	-1

Trust Prediction - Pipeline



Trust Prediction - Feature extraction I



$$L_{AB} = \{C \mid (C, A, t_{AC} \in D) \wedge (C, B, t_{CB} \in D)\}$$

Trust Prediction - Results (Epinions)

Table 3.2: Results of cross-validation experiments on the raw reputation features of the Epinions Database after one SMOTE iteration of database balancing - 10 and 3 features.

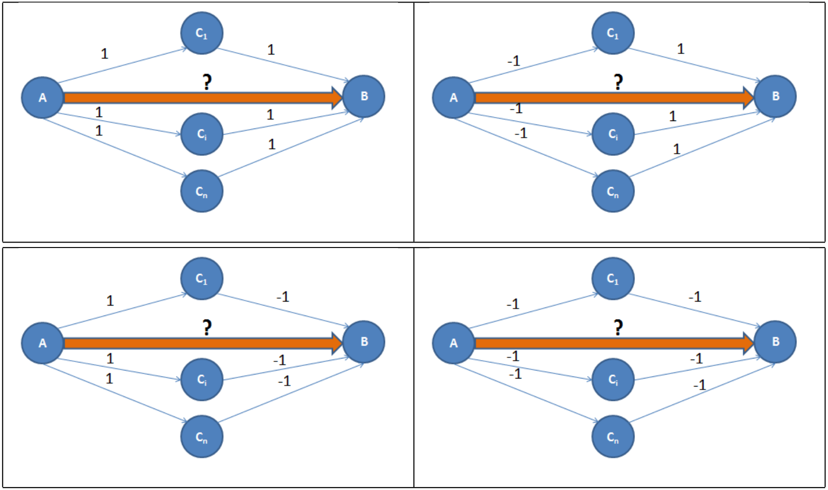
	10 features					3 features				
		Trust		Notrust			Trust		Notrust	
Classif.	OA	R	P	R	P	OA	R	P	R	P
NB	85.77	88.2	94.0	74.7	58.5	90.02	100.0	90.0	0.0	0.0
MLP	90.03	97.8	90.7	54.9	84.9	89.45	97.5	90.4	53.1	82.6
RBFC	90.08	97.6	90.9	56.1	84.0	89.49	97.4	90.4	53.6	82.3
RBFN	89.84	97.3	90.9	56.2	82.1	89.49	97.4	90.4	53.6	82.3
SVM	90.15	98.1	90.6	54.1	86.5	89.49	97.4	90.4	53.6	82.3
AdaBoost	89.74	97.5	90.7	54.8	82.8	89.49	97.4	90.4	53.6	82.3
JRip	90.09	98.0	90.7	54.5	85.7	89.49	97.4	90.4	53.6	82.3
J48	90.12	98.0	90.7	54.5	85.7	89.49	97.4	90.4	53.6	82.3

Trust Prediction - Results (Wikipedia)

Table 3.5: Results of cross-validation experiments on the raw reputation features of the Wikipedia Database after two SMOTE iteration - 10 and 3 features.

	10 features					3 features				
		Trust		Notrust			Trust		Notrust	
Classif.	OA	R	P	R	P	OA	R	P	R	P
NB	71.31	81.2	73.6	56.5	56.5	90.02	100.0	60.0	0.0	0.0
MLP	71.03	75.8	75.8	63.9	63.9	69.84	79.1	72.9	55.9	64.2
RBFC	71.27	76.1	76.0	64.1	64.2	69.84	79.1	72.9	55.9	64.2
RBFN	70.97	69.1	79.1	73.7	61.5	69.84	79.1	72.9	55.9	64.2
SVM	70.97	70.1	79.0	73.4	61.8	69.84	79.1	72.9	55.9	64.2
AdaBoost	67.46	88.9	67.3	35.4	68.1	67.42	87.1	67.7	38.1	66.3
JRip	71.27	73.2	77.6	68.4	63.1	69.84	79.1	72.9	55.9	64.2
J48	71.25	72.3	78.1	69.6	62.7	69.84	79.1	72.9	55.9	64.2

Trust Prediction - Feature extraction II



Trust Prediction - Feature extraction II

$$L_{CB}^{++} = \{C \in L_{AB} \,|\, t_{AC} = +1 \wedge t_{CB} = +1\},$$

$$L_{CB}^{+-} = \{C \in L_{AB} \,|\, t_{AC} = +1 \wedge t_{CB} = -1\},$$

$$L_{CB}^{-+} = \{C \in L_{AB} \,|\, t_{AC} = -1 \wedge t_{CB} = +1\},$$

$$L_{CB}^{--} = \{C \in L_{AB} \,|\, t_{AC} = -1 \wedge t_{CB} = -1\}.$$

Trust Prediction - Feature extraction II

$$P\left(t_{CB} = +1 \left| t_{AC} = +1\right.\right) = \frac{\left|L_{CB}^{++}\right|}{\left|L_{AB}\right|},$$

$$P\left(t_{CB} = -1 \left| t_{AC} = +1\right.\right) = \frac{\left|L_{CB}^{+-}\right|}{\left|L_{AB}\right|},$$

$$P\left(t_{CB} = +1 \left| t_{AC} = -1\right.\right) = \frac{\left|L_{CB}^{-+}\right|}{\left|L_{AB}\right|},$$

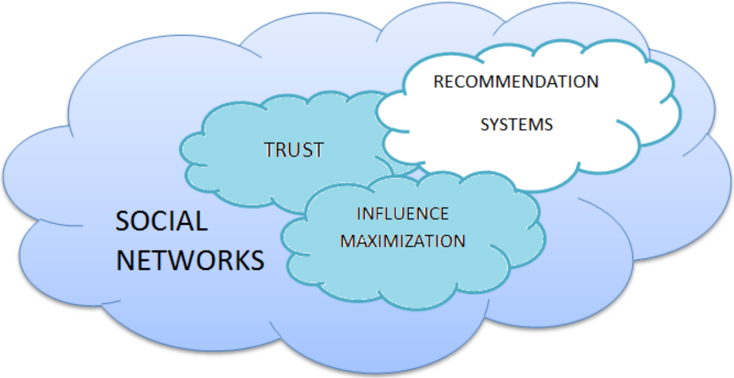
$$P\left(t_{CB} = -1 \left| t_{AC} = -1\right.\right) = \frac{\left|L_{CB}^{--}\right|}{\left|L_{AB}\right|}.$$

Trust Prediction - Results

Table 3.6: Average performance results of cross-validation experiments with different classifiers over the probabilistic reputation features. (OA) Overall Accuracy, (F1) F1 score, (AUC) area under the ROC.

	Wikipedia			Epinions		
	OA	F1	AUC	OA	F1	AUC
NB	100	98.3	0.973	100	98.7	0.983
MLP	99.99	99.1	0.981	100	99.2	0.991
RBFC	100	98.7	0.965	100	99.3	0.971
RBFN	99.99	98.6	0.966	100	99.4	0.976
AdaBoost	100	99.4	0.986	100	99.7	0.989
JRip	99.99	98.4	0.977	100	98.8	0.975
J48	99.99	98.1	0.962	100	98.2	0.972

Abstract


RECOMMENDATION
SYSTEMS
TRUST
SOCIAL
NETWORKS
INFLUENCE
MAXIMIZATION

Outline

Recommendation Systems - Techniques

Colaborative Filtering: new recommendations from similarities among users and past ratings.
Content based Filtering: based on user profile and his/her own ratings.
Demographic Filtering
Utility Filtering
Knowledge based Filtering

Recommendation Systems - Colaborative Filtering

- **Memory-based:** K-NN among similar users
- **Model-based:** K-NN among similar item rated previously by users

Recommendation Systems - Content-based Filtering

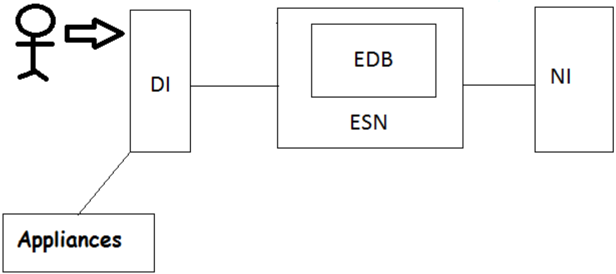
similar items that users rated positively previously

Recommendation Systems - Identified problems

Cold start
Sparsity
Subjectivity
Scalability

Outline

Recipe Generation - Context



Recipe Generation - Context

- Recipe
  - $5x\,[Time, Temperature]$
  - 10 variables $[r_1, ..., r_{10}] \in \mathbb{R}^{10}$
- Satisfaction
  - $1x\,[fragance, softness, baking, crust]$
  - 4 variables $[s_1, ..., s_4] \in \mathbb{R}^4$

Recipe Generation - Problem definition

- Direct prediction: What will be the satisfaction feedback obtained from the user for a given recipe?
    - Direct mapping: $\phi(R) = S$ to predict the satisfaction of the user with the quality of the bread resulting from a proposed recipe (first question): $\phi : \mathbb{R}^{10} \longrightarrow \mathbb{R}^{4}$
- Inverse recommendation: Which is the recipe that I need to get a specific satisfaction?
    - Inverse mapping: $\phi^{-1}(S) = R$ that looks for the recipe that would provide the desired satisfaction parameter values (second question):$\phi^{-1} : \mathbb{R}^{4} \longrightarrow \mathbb{R}^{10}$

Recipe Generation - Generated database

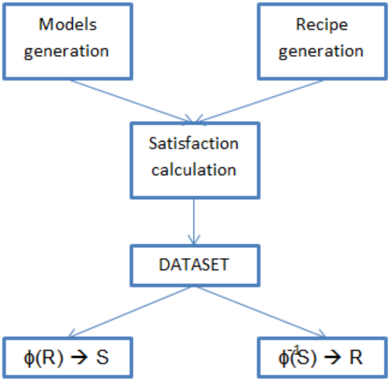
r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	s_1	s_2	s_3	s_4
0.2	0.4	1	0.5	0.9	0.7	1	0.2	0.7	1	2	3	4	3

synthetic database

100,000 instances

40 non-linear models

Recipe Generation - Pipeline



Recipe Generation - Results

	1 hidden unit	525 hidden units
s1	1.4490	0.4972
s2	1.7756	0.4790
s3	1.6259	0.5639
s4	1.1084	0.4832

Table 4.1: Average cross-validation error results of satifaction prediction for given recipes: $\phi(R) \rightarrow S_j$

Recipe Generation - Results

	1 hidden unit	525 hidden units
r1	0.4382	0.2816
r2	0.3910	0.2887
r3	0.4298	0.2919
r4	0.4080	0.2659
r5	0.4433	0.2923
r6	0.4063	0.2837
r7	0.3936	0.2903
r8	0.4743	0.2885
r9	0.4308	0.2911
r10	0.4456	0.2688

Table 4.2: Average cross-validation error results of recipe recommendation for desired satisfactions: $\phi^{-1}(S) \rightarrow R_i$

Outline

1. Introduction
   - Introduction
2. Trust Prediction
   - State of the Art
   - Experimental work
3. Recommendation Systems
   - State of the Art
   - Recipe Generation experiments
   - Product Recommendation experiments
4. Influence Maximization
   - State of the Art
   - Experimental work
5. Conclusions
   - Conclusions

Product Recommendation - Problem definition

Let have a social system with $u \in U$ users, and a catalog of items (products) $i \in I$ belonging to several categories C, the ratings of the products by (some of) the users are stored in a matrix R of size $\mid U \mid x \mid I \mid$. The recommendation problem consists in the prediction of the rating $R_{u,i}$ that a user u would give to a product i using the information provided by the social system and/or the ratings given by other users.

Product Recommendation - Database

Epinions Database is a bipartite graph based on Web of Trust and Items' Ratings.
24 Domains of products.
Selection of 5,000 users and items.
Removal of sparse values.

Product Recommendation - Feature construction I

Algorithm 5.1 Algorithm extracting the Web of Trust for each target user.

Given $G_u(U,T)$, ratings R
For each target user u_i in $G_u(U,T)$
  if $(\exists t_{ij} \in T)$ then
    $[trustedUsers]_{u_i} \leftarrow u_j$
    $R_i \leftarrow R(u_j)$
  endif

Product Recommendation - Feature construction II

Algorithm 5.2 Algorithm for extraction of target user similar users based on ratings.

For each target user u_i in $G_r(\{\{U \cup I\}, R\})$
  For each R_c matrix rating for category c
    $\psi \Lambda \phi^T = SVD(R_c)$
    $\Phi = \phi * \Lambda$ /*Each row of Φ are eigenvectors from a user i
    for each user /*Get distances
      $d(u_i, u_j) = \left\|\Phi_i - \Phi_j\right\|$
    end
  end
end
Select α most similiar users in D_u
$[SimilarUsers]_{u_i} \leftarrow \alpha$ similar users
$R_i \leftarrow R(u_j)$ for u_j in $[SimilarUsers]_{u_i}$

Product Recommendation - Results

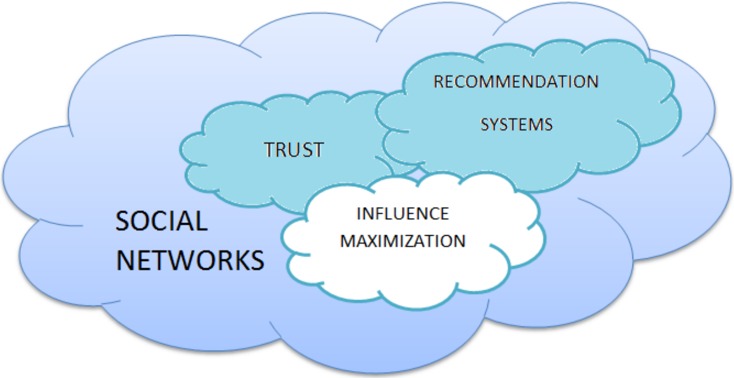
	MAE	RMSE	RAE	RRSE
Linear Regression	0.37	0.79	32.13%	56.87%
Multilayer Perceptron	0.59	0.94	51.67%	67.42%
Support Vector Regression	0.81	0.34	30.09%	57.96%
KNN	0.36	0.79	31.60%	56.84%
Additive Regression	*0.30*	*0.96*	*25.91%*	*68.69%*
Random Tree	0.36	0.79	31.60%	56.84%

Table 5.1: Results of features extracted from Web of Trust

	MAE	RMSE	RAE	RRSE
Linear Regression	0.74	1.49	44.47%	79.11%
Multilayer Perceptron	0.97	1.64	58.15%	87.23%
Support Vector Regression	*0.56*	*1.14*	*33.43%*	*60.41%*
KNN	1.25	1.39	85.10%	84.32%
Additive Regression	*0.56*	*1.14*	*33.43%*	*60.41%*
Random Tree	0.74	1.49	44.47%	79.11%

Table 5.2: Results of features extracted from user distances

Abstract

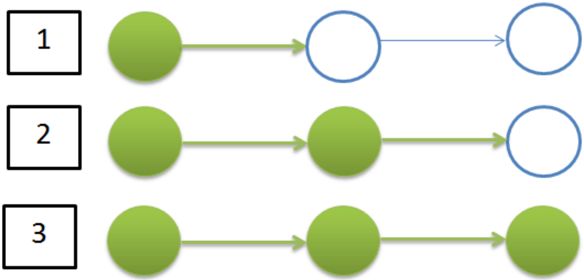

RECOMMENDATION
SYSTEMS
TRUST
SOCIAL
NETWORKS
INFLUENCE
MAXIMIZATION

Outline

Influence Maximization - Solving the problem

- Exhaustive Search
  - #NP hard
- Heuristic Search
  - $1 - \frac{1}{e}$ aproximation

Influence Maximization - ICM Model



Influence Maximization - LTM Model

Outline

Influence Maximization - Problem definition

- Given a Web of Trust (WoT) associated to some social system, specificed by weighted graph G = (U;E;T), where there is a trust value associated with each edge, find the minimal subset that maximizes the spread of influence. (IM-seed nodes).

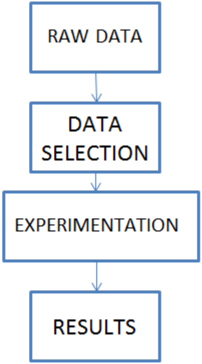
$$S* = minmax\sigma(S)$$
$$\mid S \mid S \subset V$$

Influence Maximization - Databases

- Epinions and synthetic databases
- Subgraphs of different sizes

User Id	User Id	Trust Value
245	246	1
...	...	...

Influence Maximization - Pipeline



Influence Maximization - Harmony Search

Algorithm 6.1 Harmony Search algorithm adapted to Influence Maximization

1. Given probabilistic social graph $G = (V, E, W)$
2. Initialize HS parameters and HM_0
3. while $t \leq NI$
   (a) for $i = 1 \dots N$ //Improvise new harmony $\mathbf{x}'$,
      i. if $r \leq HMCR$
         A. $x'_i \overset{\in}{\leftarrow} \{x_i^1, \dots, x_i^{HMS}\}$,
         B. if $r < PAR$ then $x'_i \leftarrow x'_i + \alpha_i$; $\alpha_i \sim U[-BW, BW]$;
      ii. otherwise $x'_i \overset{\in}{\leftarrow} X_i$
   (b) Evaluate harmony $f(\mathbf{x}')$
   (c) If $f(\mathbf{x}') > f(\mathbf{x}^{HMS})$
      i. replace $\mathbf{x}^{HMS}$ in HM, and sort HM.
4. Return best Harmony

Influence Maximization - New Method

Algorithm 6.2 Proposed IM heuristic (IMH) solution algorithm

1. Given social graph $G = (V, E, W)$, with adjacency matrix $A_b = \left[a^b_{v,v'}\right]$
2. $S_0 = \left\{ v \middle| \sum_{v' \neq v} a^b_{v',v} = 0 \right\}$
3. $R_0 = V - \{S_0 \cup \sigma(S_0, A_b)\}$
4. $A_0 = \left[a^0_{v,v'}\right]$ s.t. $a^0_{v,v'} = 0$ if $v \notin R_0 \vee v' \notin R_0$; otherwise $a^0_{v,v'} = a^b_{v,v'}$
5. $t = 0$
6. iterate until $R_t = \emptyset$
   (a) $v^* = \arg\max_{v \in R_t} \{\sigma_1(\{v\}, A_t)\}$
   (b) $S_{t+1} = S_t \cup \{v^*\}$
   (c) $R_{t+1} = R_t - \{\{v\} \cup \sigma_1(\{v\}, A_t)\}$
   (d) $A_{t+1} = \left[a^{t+1}_{v,v'}\right]$ s.t. $a^{t+1}_{v,v'} = 0$ if $v \notin R_t \vee v' \notin R_t$; otherwise $a^{t+1}_{v,v'} = a^t_{v,v'}$
   (e) $t \leftarrow t + 1$
7. Return S_t

Influence Maximization - Results

density

Density	New method	Greedy
0.000011	0.018 sec.	5.587 sec.
0.00011	0.144 sec.	5.817 sec.
0.0011	0.031 sec.	$>$ 5 min.
0.011	0.050 sec.	$>$ 5 min.

Table 6.1: Comparison of speed using matrix of different density

Influence Maximization - Results

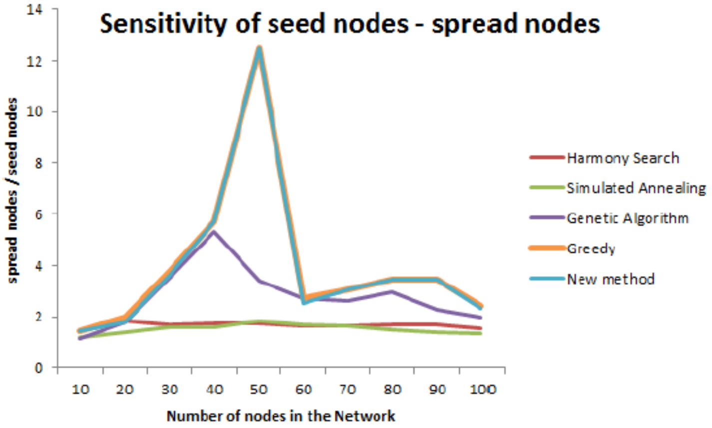
size

Sizes	New method	Greedy
1000	0.065 sec	3.586 sec
2000	0.119 sec	35.938 sec
3000	0.182 sec	> 5 min
4000	0.363 sec	> 5 min
5000	0.446 sec	> 5 min
6000	0.706 sec	> 5 min
7000	1.702 sec	> 5 min
8000	1.914 sec	> 5 min
9000	2.012 sec	> 5 min
10000	2.435 sec	> 5 min

Table 6.2: Comparison of speed using matrix of different sizes

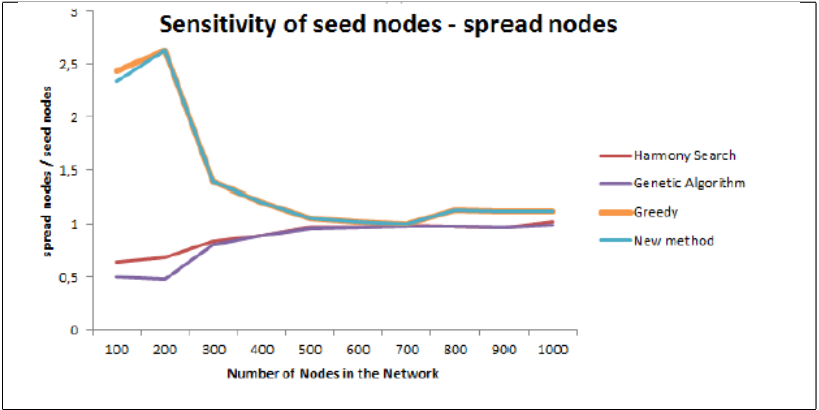
Influence Maximization - Results

Synthetic database



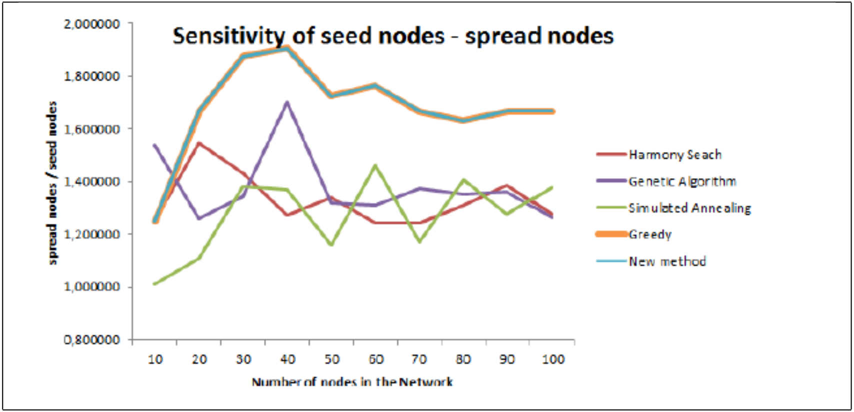
Influence Maximization - Results

Synthetic database



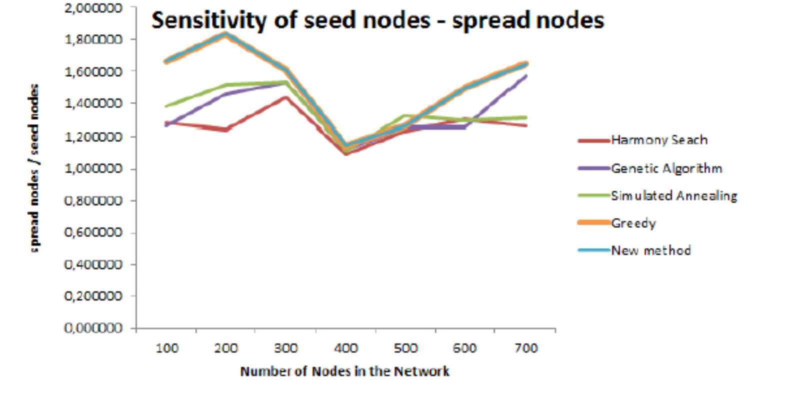
Influence Maximization - Results

Epinions database



Influence Maximization - Results

Epinions database



Outline

Thesis Goals

1. Study of basic social mechanisms such as the propagation of Trust and influence, and the generation of innovations from social interactions.
2. Study of the applications of Machine Learning algorithms in social networks, for Trust prediction, recommendation systems, and social computing.
3. Study of Computational Intelligence algorithms for social networks.
4. Creation of a computational substrate for experimentation, and validation given by real and synthetic databases.

Conclusions - Trust

1. Trust Prediction problem as Machine Learning problem is rather straightforward compared with other works in the literature (Graph Algebra).
2. Attempts to improve results applying a SMOTE approach do not improve the overall accuracy, but provide improvements on the minority class.

Conclusions - Recommendation Systems

1. Application of regression ELM to build a breadmaker recommender system which is an instance of the social intelligence in the Internet of Things framework of the SandS European Project.
2. Methods for Recommendation Systems in Social Networks based on Colaborative Filtering.
3. Better results are obtained with the Web of Trust provided by user explicit statements.

Conclusions - Influence Maximization

1. A new heuristic search method for Influence Maximization (IMH).
2. It is guaranteed to terminate covering the entire graph.
3. Proposed IMH method is always faster than Greedy algorithm.

Future Work

1. **Trust**: imbalanced databases and feature definition.
2. **Recommendation Systems**: Real datasets. Sparse computational methods.
3. **Influence Maximization**: Scalability.

Research activities

THANK YOU

QUESTIONS ?