Data Mining 2013
Mining (Social) Network Data

Ad Feelders

Universiteit Utrecht

November 1, 2013

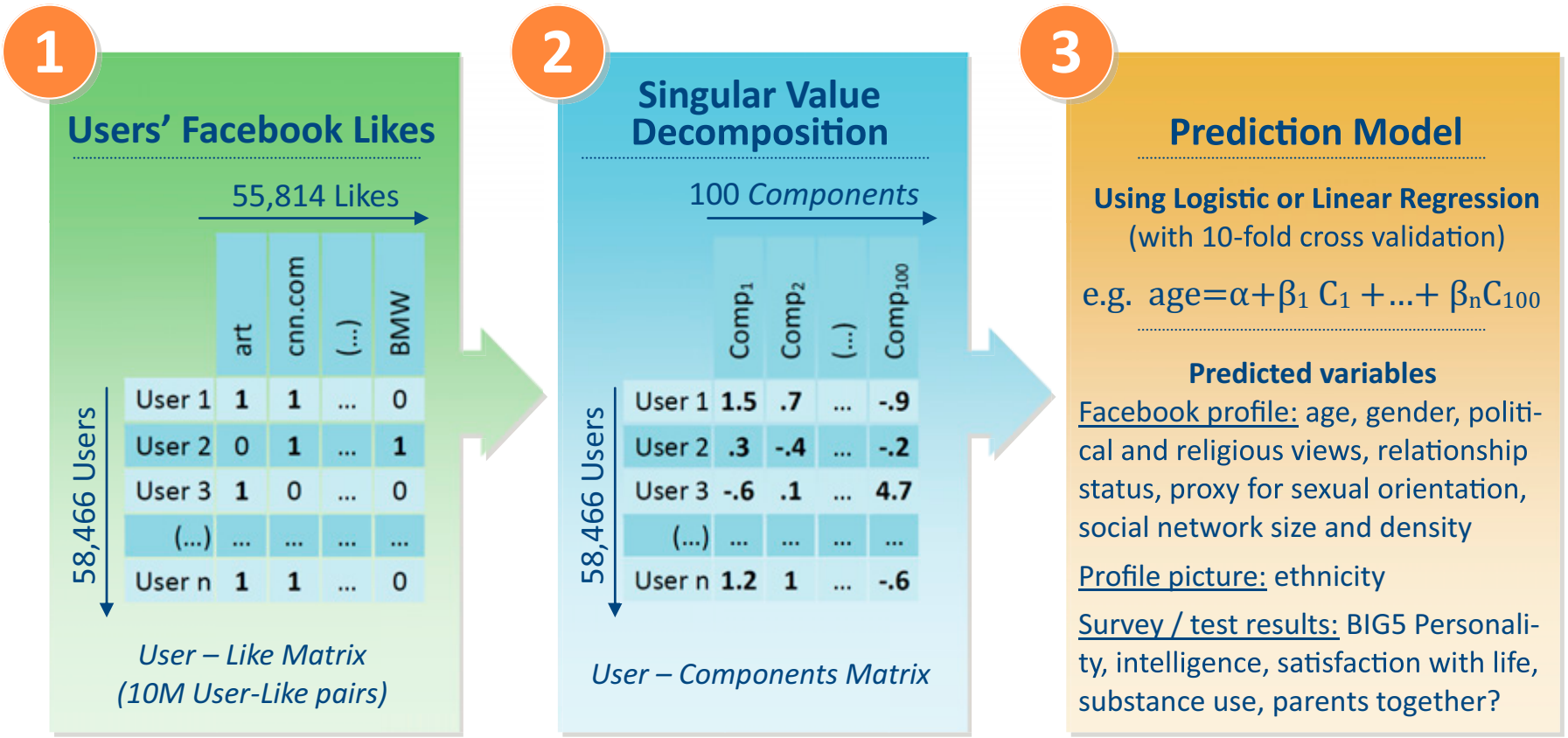
Example: Predicting Romantic Relationships

The latest offering from Facebooks data-science team teases out who is romantically involved with whom by examining link structures. It turns out that if one of your Facebook friends - lets call him Joe - has mutual friends that touch disparate areas of your life, and those mutual friends are themselves not extensively connected, its a strong clue that Joe is either your romantic partner or one of your closest personal friends.

`http://www.technologyreview.com/view/520771/now-facebook-can-see-inside-your-heart-too/`

Lars Backstrom and Jon Kleinberg: *Romantic Partnerships and the Dispersion of Social Ties: A Network Analysis of Relationship Status on Facebook*, Proc. 17th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW), 2014

Example: Mining facebook likes



M. Kosinski, D. Stillwell, T. Graepel: *Private traits and attributes are predictable from digital records of human behavior*, PNAS, March 11, 2013.

Example: Mining facebook likes

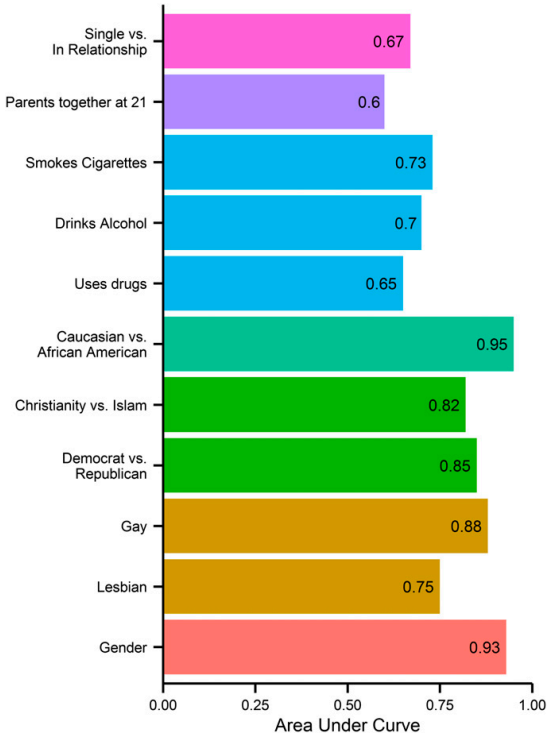


Fig. 2. Prediction accuracy of classification for dichotomous/dichotomized attributes expressed by the AUC.

AUC: probability of correctly classifying two random selected users, one from each class (e.g. male and female). Random guessing: AUC=0.5.

Example: Mining facebook likes

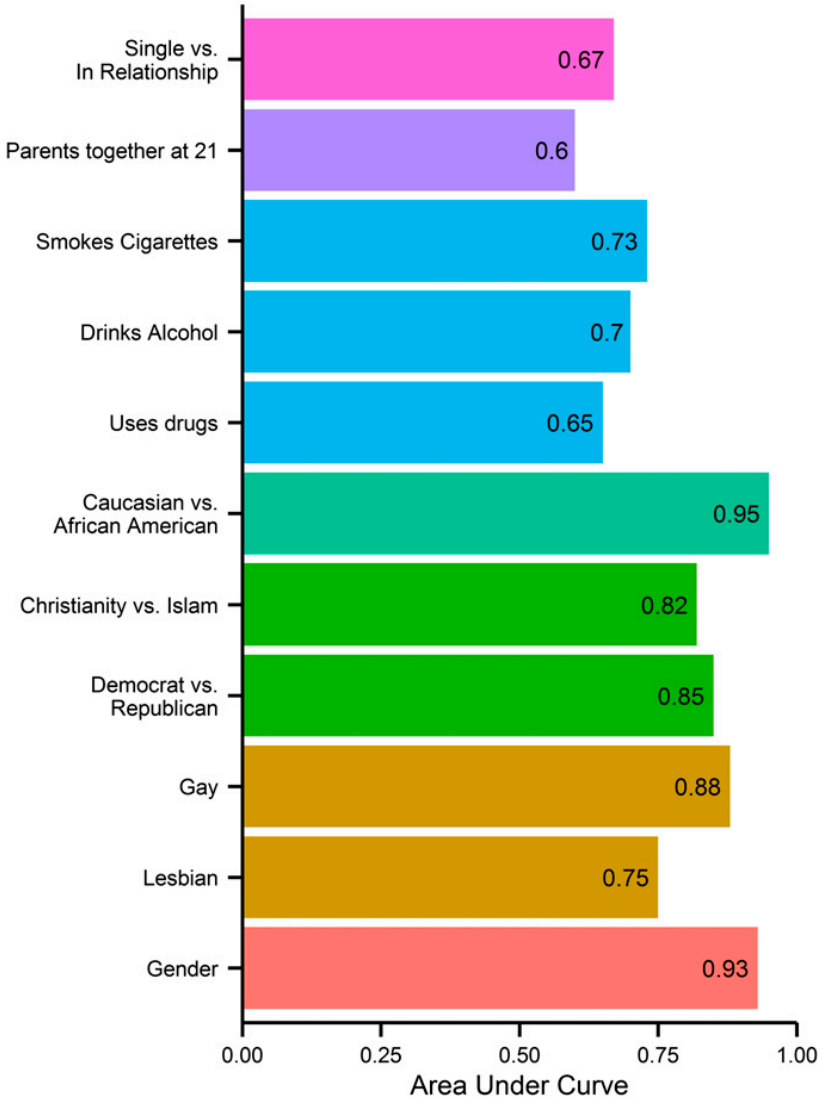


Fig. 2. Prediction accuracy of classification for dichotomous/dichotomized attributes expressed by the AUC.

Example: Mining facebook likes

Best predictors of high intelligence include:

- "Thunderstorms"
- "Science"
- "Curly Fries"

Best predictors of low intelligence include:

- "I love being a mom"
- "Harley Davidson"
- "Lady Antebellum"

The Node Classification Problem

Given a (social) network with linked nodes and labels for some nodes, how can we provide a high quality labeling for every node?

The existence of an explicit link structure makes the node classification problem different from traditional data mining classification tasks, where objects being classified are typically considered to be independent.

The Node Classification Problem

Two important phenomena:

- Homophily ("Birds of a feather"): a link between individuals (such as friendship) is correlated with those individuals being similar in nature. For example, friends often tend to be similar in characteristics like age, social background and education level.
- Co-citation regularity: similar individuals tend to refer or connect to the same things. For example, when two individuals have the same tastes in music, literature or fashion, co-citation regularity suggests that they may be similar in other ways or have other common interests.

Example: Facebook

$G = (V, E, W)$

- The set of nodes V represents users of Facebook.
- An edge $(i, j) \in E$ could represent:
  - A relationship (friendship, sibling, partner)
  - An interaction (wall post, private message, group message)
  - An activity (tagging a photo, playing games)
- Node attributes: demographics (age, location, gender, occupation), interests (hobbies, movies, books, music), etc.
- Edge weights W: strength of connection, e.g. number of messages exchanged.

Example: YouTube

$G = (V, E, W)$

- The set of nodes V represents users of YouTube.
- An edge $(i, j) \in E$ could represent:
  - subscription or friend relation
  - derived link: v_i and v_j are connected if the corresponding users have co-viewed more than a certain number of videos.
- Node attributes: demographics (age, location, gender, occupation), interests (hobbies, movies, books, music), etc.
- Edge weights W: strength of connection, e.g. the number of co-viewed video's.

Example: Papers and Citations

$G = (V, E, W)$

- The set of nodes V represents papers.
- An edge $(i, j) \in E$ could represent that paper v_i cites paper v_j.
- Node attributes: authors, title, word frequencies, topic of the paper.
- Edge weights W: Number of times v_i cites v_j.

Literature (not required)

The remainder of the slides is primarily based on:

Qing Lu and Lise Getoor, *Link-based Classification*, Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

Link attributes

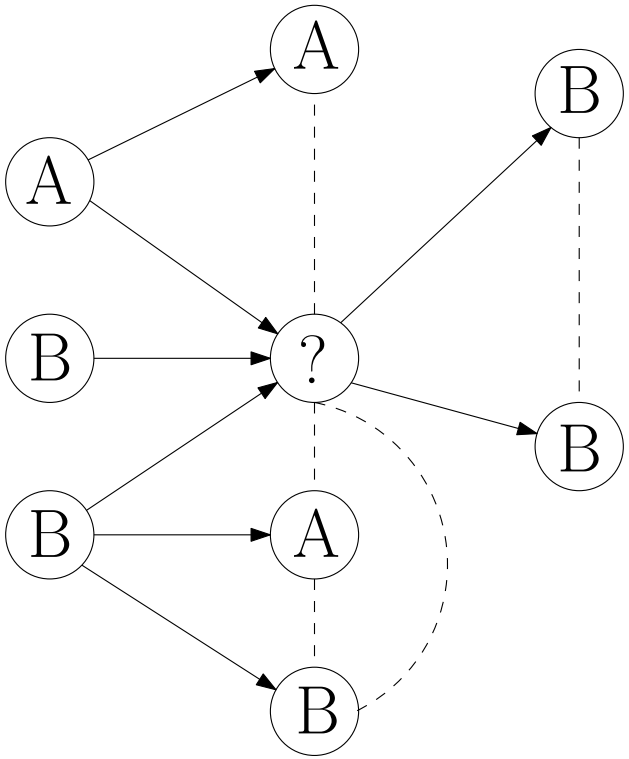
Link attributes are based on the class labels or categories of the linked objects.

Different statistics:

1. Mode-link: compute a single feature, the mode (majority class), from each set of linked objects from the in-links, out-links, and co-citation links.
2. Count-link: use the frequencies of the categories of the linked objects.
3. Binary-link: 1 if category occurs at least once, 0 otherwise.

Link attributes: example

Suppose there are two class labels, A and B:



Co-citation links are indicated by dashed lines.

Link attributes: example

The node labeled "?" on the previous slide would get the following values for the link attributes:

	in-A	in-B	out-A	out-B	co-A	co-B
Count-link	1	2	0	2	2	1
Mode-link	0	1	0	1	1	0
Binary-link	1	1	0	1	1	1

Logistic regression

Let C be a binary class label with values coded as 0 and 1.
$x = (x_1, \ldots, x_p)$ are attributes or features.

Logistic regression model:

$$P(C = 1|x) = \frac{e^{w_0 + \sum w_i x_i}}{1 + e^{w_0 + \sum w_i x_i}}$$

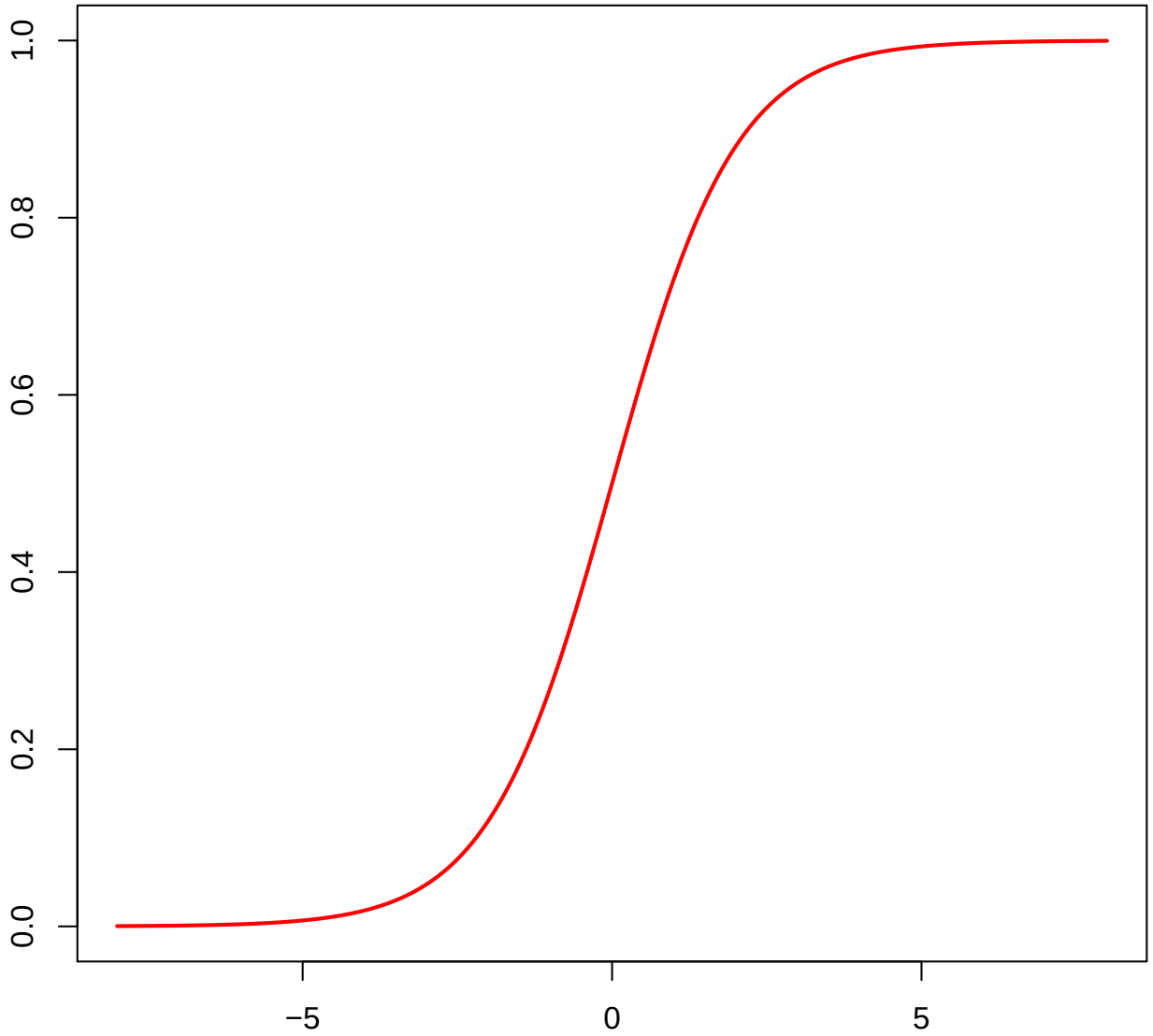
Coefficients $w_0, w_1, \ldots, w_p$ can be estimated from data with maximum likelihood estimation.

Logit transformation:

$$\ln \left\{ \frac{P(C = 1|x)}{P(C = 0|x)} \right\} = w_0 + \sum_{i=1}^{p} w_i x_i$$

Hence, logistic regression produces a *linear* decision boundary.

Logistic response function: $\frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}}$



Logistic regression

Let x denote the object attributes and z the link attributes.

$w^{(o)}$ are the weights for the object attributes, and $w^{(\ell)}$ are the weights for the link attributes.

Estimate 2 logistic regression models:

$$P(C = 1|x) = \frac{e^{w_0^{(o)} + \sum w_i^{(o)} x_i}}{1 + e^{w_0^{(o)} + \sum w_i^{(o)} x_i}} \qquad \text{(object attributes)}$$

$$P(C = 1|z) = \frac{e^{w_0^{(\ell)} + \sum w_i^{(\ell)} z_i}}{1 + e^{w_0^{(\ell)} + \sum w_i^{(\ell)} z_i}} \qquad \text{(link attributes)}$$

Logistic regression

To estimate the weights w, regularized maximum likelihood is applied, that is, we maximize the function

$$\mathcal{L}(w) - \lambda \left(w_0^2 + \sum_{i=1}^{p} w_i^2 \right)$$

with respect to w, where $\mathcal{L}$ is the log-likelihood function and $\lambda \geq 0$ is a regularization parameter that punishes large weights in order to prevent overfitting. The best value for λ is usually selected using cross-validation (cf. selection of nmin and minleaf in trees).

Prediction

Logistic regression is a model for binary classification. For classification problems with k possible class labels, one often fits k one-against-all binary models. To make predictions one then selects the class label with highest posterior probability.

The overall prediction rule is:

$$\hat{C}(x, z) = \arg \max_{i \in 1,\ldots,k} P(C = i|x)P(C = i|z)$$

Link-based Classification

In classifying new cases we run into the problem that the link attributes are not observed: to predict the class label of an object, we need the class labels of its neighbors!

Use an Iterative Classification Algorithm:

1. Using only the object attributes, assign an initial class label to each object in the test set.
2. Iteratively apply the full model to classify each object until the stopping criterion has been satisfied:
   - Compute the link statistics, based on the current assignments to linked objects.
   - Compute the posterior probability for the class variable for this object.
   - The class label with the largest posterior probability is chosen as the new label for the current object.

Experiments: Data

The algorithm was evaluated on 3 data sets: Cora, WebKB and CiteSeer.

The CiteSeer data set contains about 3,600 papers from six categories:

1. Agents
2. Artificial Intelligence
3. Database
4. Human Computer Interaction
5. Machine Learning
6. Information Retrieval.

There are 7,522 citations in the data set.

Experiments: Data

After stemming and removal of stop words and rare words, the dictionary contains 3,000 words. Hence, there are 3,000 attributes in the "content-only" model!

The data set is split into 3 separate equally sized parts. Set 1 to fit the logistic regression models with different values for the regularization parameter λ, set 2 to select the best value for λ, and set 3 to estimate the error of the selected model.

Experiments: Data

The WebKB data.

Classes are topics of Web Pages from 4 CS departments:

1. student
2. faculty
3. staff
4. department
5. course
6. project
7. other

Links are hyperlinks between pages.
Attributes are word frequencies.

Experiments: Modeling

In the one-against-all all approach we learn a binary classification model for each class, for example, "Machine Learning" ($\mathrm{ML}{=}1$) against "not Machine Learning" ($\mathrm{ML}{=}0$).

$$\ln\left\{\frac{P(\mathrm{ML}=1|x)}{P(\mathrm{ML}=0|x)}\right\} = w_0 + \sum_{i=1}^{3{,}000} w_i x_i,$$

where x_i is for example the number of times the word "data" appears in the article.

With 3,000 attributes, regularization to avoid overfitting is indeed a good idea!

Experiments

Table 1. Summary of average accuracy, precision, recall and F1 measure using different link-based models on Cora, CiteSeer and WebKB. The random iteration ordering strategy is used.

Cora							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.674	0.649	0.74	0.728	0.717	0.754	**0.758**
Avg. Precision	0.662	0.704	0.755	0.73	0.717	0.747	**0.759**
Avg. Recall	0.626	0.59	0.689	0.672	0.679	0.716	**0.725**
Avg. F1 Measure	0.643	0.641	0.72	0.7	0.697	0.731	**0.741**
CiteSeer							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.607	0.618	0.634	0.644	0.658	0.664	**0.679**
Avg. Precision	0.551	0.55	0.58	0.579	**0.606**	0.597	0.604
Avg. Recall	0.552	0.547	0.572	0.573	0.601	0.597	**0.608**
Avg. F1 Measure	0.551	0.552	0.575	0.575	0.594	0.597	**0.606**
WebKB							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.862	0.848	0.832	0.863	0.851	0.871	**0.877**
Avg. Precision	0.876	0.86	0.864	0.876	0.878	**0.879**	0.878
Avg. Recall	0.795	0.79	0.882	0.81	0.772	0.811	**0.83**
Avg. F1 Measure	0.832	0.821	0.836	0.84	0.82	0.847	**0.858**

Experiments

Table 2. Average accuracy using in-links, out-links, co-links separately, and all (in+out+co) links with **mode-link**, **binary-link** and **count-link** models on Cora, CiteSeer and WebKB

	Mode-Link				Binary-Link				Count-Link			
data set	in	out	co	all	in	out	co	all	in	out	co	all
Cora	0.687	**0.717**	0.668	**0.717**	0.695	0.732	0.686	**0.754**	0.694	0.729	0.688	**0.758**
CiteSeer	0.632	0.651	0.628	**0.658**	0.629	0.659	0.624	**0.664**	0.631	0.644	0.636,,	**0.679**
WebKB	0.853	**0.857**	0.843	0.851	0.857	0.847	0.857	**0.871**	0.866	0.863	0.868	**0.877**

Order of Processing

In the iterative step there are many possible orderings of the objects.

You can process the objects:

1. In random order.
2. Order on number of links.
3. Order on class posterior probability.
4. Order on number of different categories to which an object is linked (link diversity).

Convergence

Influence of order on convergence.

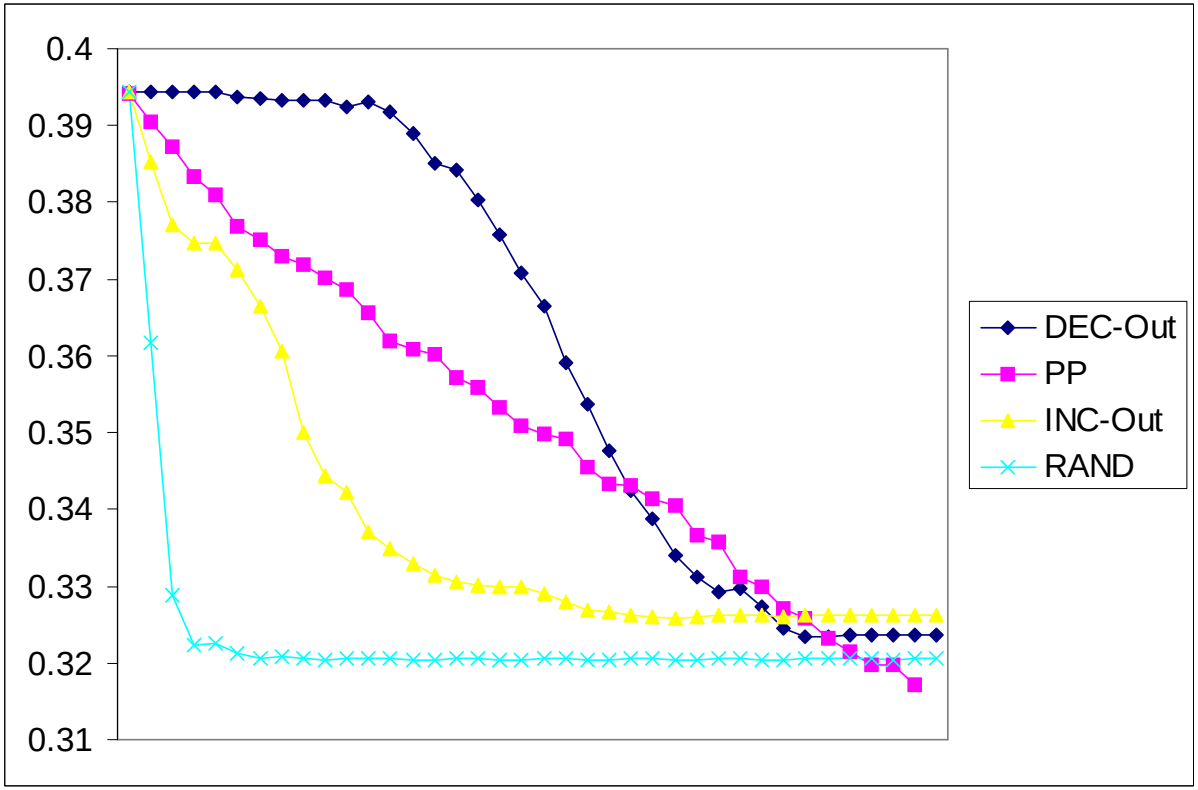


Figure 1. The convergence rates of different iteration methods on the CiteSeer data set.