

Data Mining 2018

Mining (Social) Network Data

Ad Feelders

Universiteit Utrecht

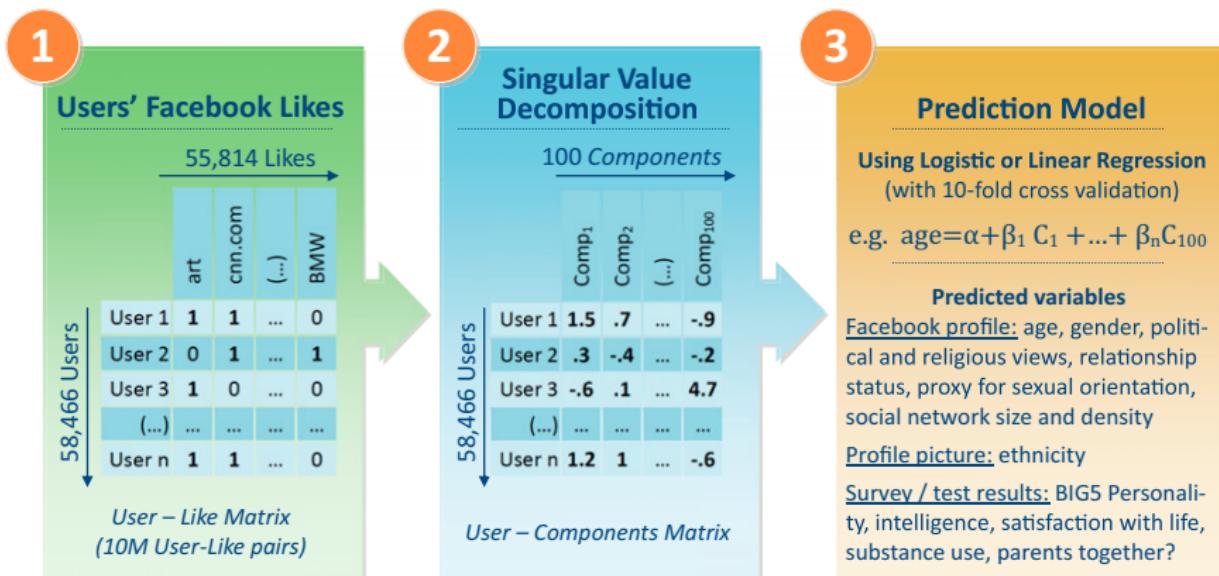
Example: Predicting Romantic Relationships

The latest offering from Facebook's 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

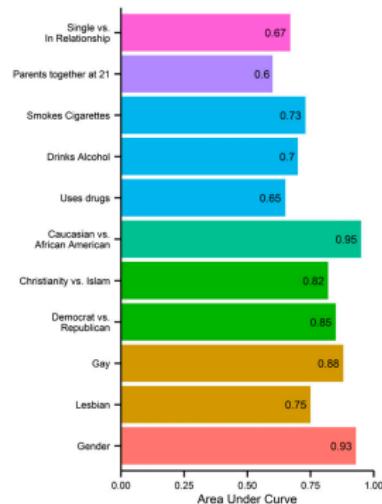


Fig. 2. Prediction accuracy of classification for dichotomous/dichotomized attributes expressed by the AUC.

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

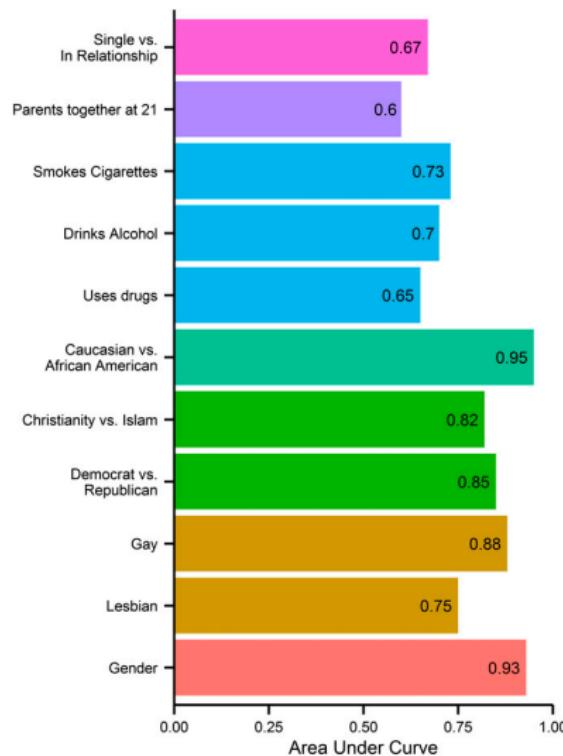


Fig. 2. Prediction accuracy of classification for dichotomous/dichotomized attributes expressed by the AUC

Example: Mining facebook likes

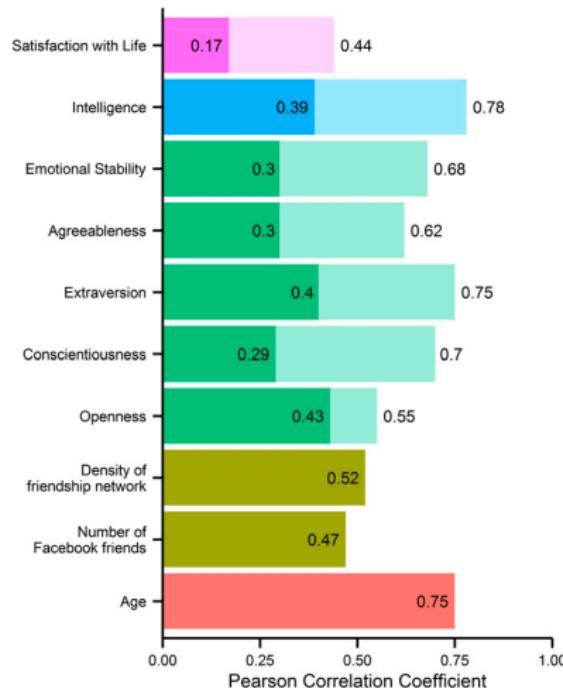


Fig. 3. Prediction accuracy of regression for numeric attributes and traits expressed by the Pearson correlation coefficient between predicted and actual attribute values; all correlations are significant at the $P < 0.001$ level. The transparent bars indicate the questionnaire's baseline accuracy, expressed in terms of test-retest reliability.

Example: Mining facebook likes

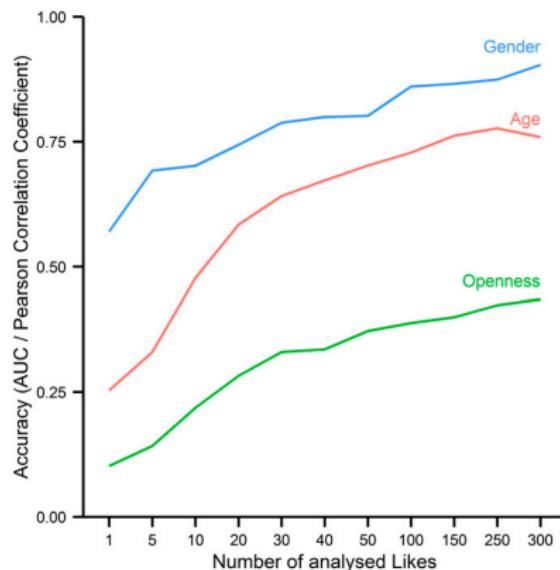


Fig. 4. Accuracy of selected predictions as a function of the number of available Likes. Accuracy is expressed as AUC (gender) and Pearson's correlation coefficient (age and Openness). About 50% of users in this sample had at least 100 Likes and about 20% had at least 250 Likes. Note, that for gender (dichotomous variable) the random guessing baseline corresponds to an AUC = 0.50.

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”

Example: predicting personality from Twitter

J. Golbeck, C. Robles, M. Edmondson, K. Turner: *Predicting Personality from Twitter*, IEEE International Conference on Social Computing, 2011.

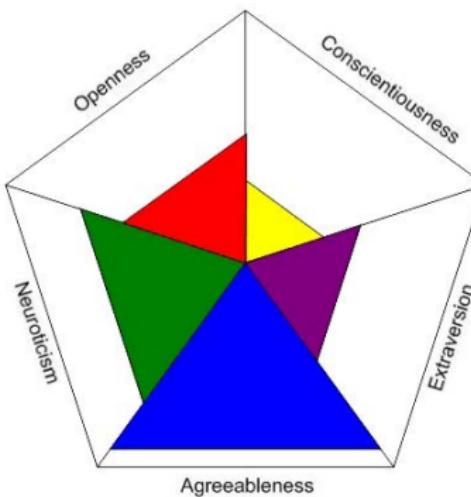


Fig. 1: A person has scores for each of the five personality factors. Together, the five factors represent an individual's personality.

Example: predicting personality from Twitter

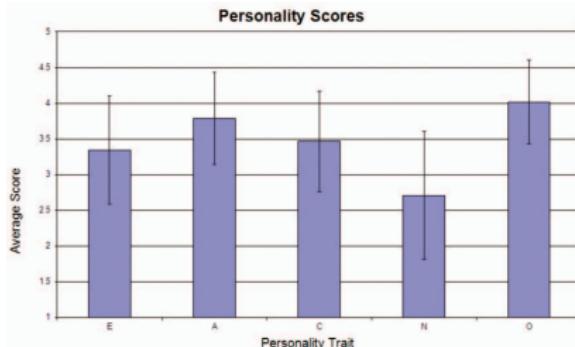


Fig. 2: Average scores on each personality trait shown with standard deviation bars.

TABLE I: Average scores on each personality factor on a normalized 0-1 scale

	Agree.	Consc.	Extra.	Neuro.	Open.
Average	0.697	0.617	0.586	0.428	0.755
Stdev	0.162	0.176	0.190	0.224	0.147

Example: predicting personality from Twitter

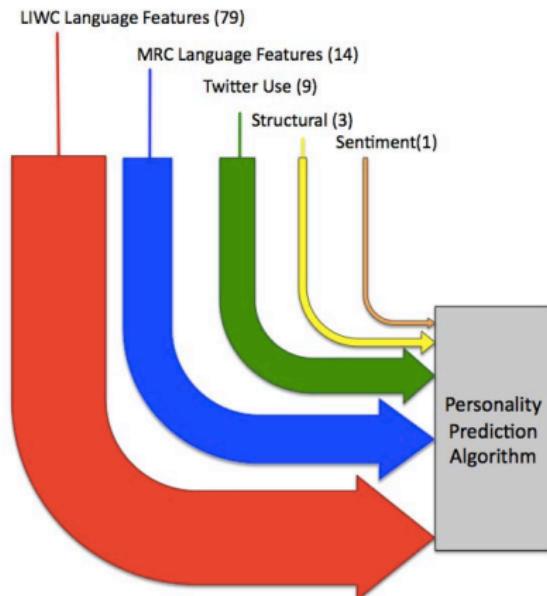


Fig. 4: Features used for predicting personality.

Example: predicting personality from Twitter

TABLE II: Pearson correlation values between feature scores and personality scores. Significant correlations are shown in bold for $p < 0.05$. Only features that correlate significantly with at least one personality trait are shown.

Language Feature	Examples	Extro.	Agree.	Consc.	Neuro.	Open.
“You”	(you, your, thou)	0.068	0.364	0.252	-0.212	-0.020
Articles	(a, an, the)	-0.039	-0.139	-0.071	-0.154	0.396
Auxiliary Verbs	(am, will, have)	0.033	0.042	-0.284	0.017	0.045
Future Tense	(will, gonna)	0.227	-0.100	-0.286	0.118	0.142
Negations	(no, not, never)	-0.020	0.048	-0.374	0.081	0.040
Quantifiers	(few, many, much)	-0.002	-0.057	-0.089	-0.051	0.238
Social Processes	(mate, talk, they, child)	0.262	0.156	0.168	-0.141	0.084
Family	(daughter, husband, aunt)	0.338	0.020	-0.126	0.096	0.215
Humans	(adult, baby, boy)	0.204	-0.011	0.055	-0.113	0.251
Negative Emotions	(hurt, ugly, nasty)	0.054	-0.111	-0.268	0.120	0.010
Sadness	(crying, grief, sad)	0.154	-0.203	-0.253	0.230	-0.111
Cognitive Mechanisms	(cause, know, ought)	-0.008	-0.089	-0.244	0.025	0.140
Causation	(because, effect, hence)	0.224	-0.258	-0.155	-0.004	0.264
Discrepancy	(should, would, could)	0.227	-0.055	-0.292	0.187	0.103
Certainty	(always, never)	0.112	-0.117	-0.069	-0.074	0.347

Example: predicting personality from Twitter

Perceptual Processes						
Hearing	(listen, hearing)	0.042	-0.041	0.014	0.335	-0.084
Feeling	(feels, touch)	0.097	-0.127	-0.236	0.244	0.005
Biological Processes	(eat, blood, pain)	-0.066	0.206	0.005	0.057	-0.239
Body	(cheek, hands, spit)	0.031	0.083	-0.079	0.122	-0.299
Health	(clinic, flu, pill)	-0.277	0.164	0.059	-0.012	-0.004
Ingestion	(dish, eat, pizza)	-0.105	0.247	0.013	-0.058	-0.202
Work	(job, majors, xerox)	0.231	-0.096	0.330	-0.125	0.426
Achievement	(earn, hero, win)	-0.005	-0.240	-0.198	-0.070	0.008
Money	(audit, cash, owe)	-0.063	-0.259	0.099	-0.074	0.222
Religion	(altar, church, mosque)	-0.152	-0.151	-0.025	0.383	-0.073
Death	(bury, coffin, kill)	-0.001	0.064	-0.332	-0.054	0.120
Fillers	(blah, imean, youknow)	0.099	-0.186	-0.272	0.080	0.120

Example: predicting personality from Twitter

Punctuation							
Commas		0.148	0.080	-0.24	0.155	0.170	
Colons		-0.216	-0.153	0.322	-0.015	-0.142	
Question Marks		0.263	-0.050	0.024	0.153	-0.114	
Exclamation Marks		-0.021	-0.025	0.260	0.317	-0.295	
Parentheses		-0.254	-0.048	-0.084	0.133	-0.302	
Non-LIWC Features							
GI Sentiment		0.177	-0.130	-0.084	-0.197	0.268	
Number of Hashtags		0.066	-0.044	-0.030	-0.217	-0.268	
Words per tweet		0.285	-0.065	-0.144	0.031	0.200	
Links per tweet		-0.061	-0.081	0.256	-0.054	0.064	

Example: predicting personality from Twitter

TABLE III: Mean Absolute Error on a normalized scale for each algorithm and personality trait.

	Agree.	Consc.	Extra.	Neuro.	Open.
ZeroR	0.129980265	0.146204953	0.160241663	0.182122225	0.11923333
GaussianProcess	0.130675423	0.14599073	0.160315335	0.18205923	0.11922558

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: 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

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

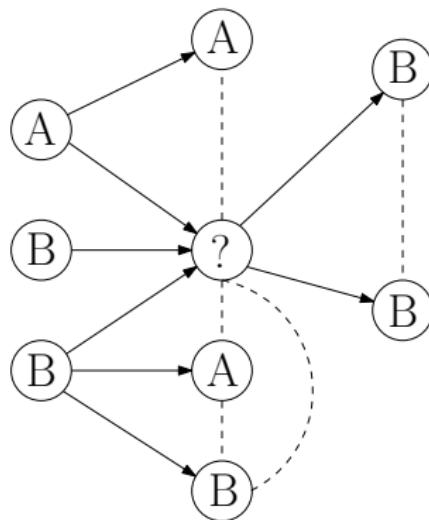
Link attributes are based on the class labels or categories of the linked objects.

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.

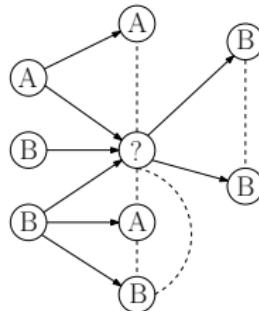
Link attributes: example

Suppose there are two class labels, A and B:



Co-citation links are indicated by dashed lines.

Link attributes: example



	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 Y be a binary class label with values coded as 0 and 1.
 $x = (x_1, \dots, x_m)$ are attributes or features.

Logistic regression model:

$$P(Y = 1|x) = \frac{e^{\beta_0 + \sum \beta_j x_j}}{1 + e^{\beta_0 + \sum \beta_j x_j}}$$

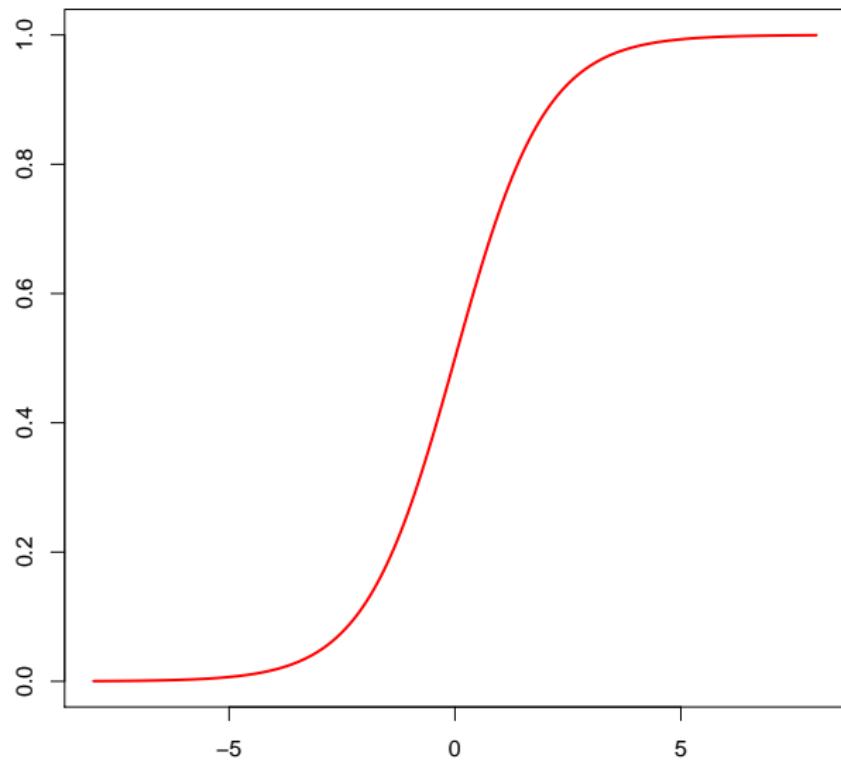
Coefficients $\beta_0, \beta_1, \dots, \beta_m$ can be estimated from data with maximum likelihood estimation.

Logit transformation:

$$\ln \left\{ \frac{P(Y = 1|x)}{P(Y = 0|x)} \right\} = \beta_0 + \sum_{j=1}^m \beta_j x_j$$

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.

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

Estimate 2 logistic regression models:

$$P(Y = 1|x) = \frac{e^{\beta_0^{(o)} + \sum \beta_j^{(o)} x_j}}{1 + e^{\beta_0^{(o)} + \sum \beta_j^{(o)} x_j}} \quad (\text{object attributes})$$

$$P(Y = 1|z) = \frac{e^{\beta_0^{(\ell)} + \sum \beta_j^{(\ell)} z_j}}{1 + e^{\beta_0^{(\ell)} + \sum \beta_j^{(\ell)} z_j}} \quad (\text{link attributes})$$

Logistic regression

To estimate the weights β , regularized maximum likelihood is applied, that is, we maximize the function (L_2 penalty)

$$\mathcal{L}(\beta) - \lambda \sum_{j=1}^m \beta_j^2$$

or (L_1 penalty; “LASSO”)

$$\mathcal{L}(\beta) - \lambda \sum_{j=1}^m |\beta_j|$$

with respect to β , 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.

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:

$$\hat{C}(x, z) = \arg \max_{k \in 1, \dots, K} \hat{P}(Y = k|x)\hat{P}(Y = k|z)$$

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” ($\text{ML}=1$) against “not Machine Learning” ($\text{ML}=0$).

$$\ln \left\{ \frac{P(\text{ML} = 1|x)}{P(\text{ML} = 0|x)} \right\} = \beta_0 + \sum_{j=1}^{3,000} \beta_j x_j,$$

where x_j 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!

Accuracy, Precision and Recall

prediction/truth	in the class	not in the class
in the class	true positives (TP)	false positives (FP)
not in the class	false negatives (FN)	true negatives (TN)

TP, FP, FN, TN are counts of documents. The sum of these four counts is the total number of test documents N_{test} .

- Accuracy is the fraction of correct predictions:

$$\text{Accuracy} = \frac{TP + TN}{N_{\text{test}}}$$

- Precision:

$$P = TP / (TP + FP)$$

- Recall:

$$R = TP / (TP + FN)$$

A combined measure: F

- Precision and recall only measure a single aspect of performance.
We can easily get a recall of 1 simply by classifying all documents as *in the class*.
- F_1 allows us to trade off precision against recall.
- Definition:
$$F_1 = \frac{2P \times R}{P + R}$$
- This is the *harmonic mean* of P and R .

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

data set	Mode-Link				Binary-Link				Count-Link			
	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.

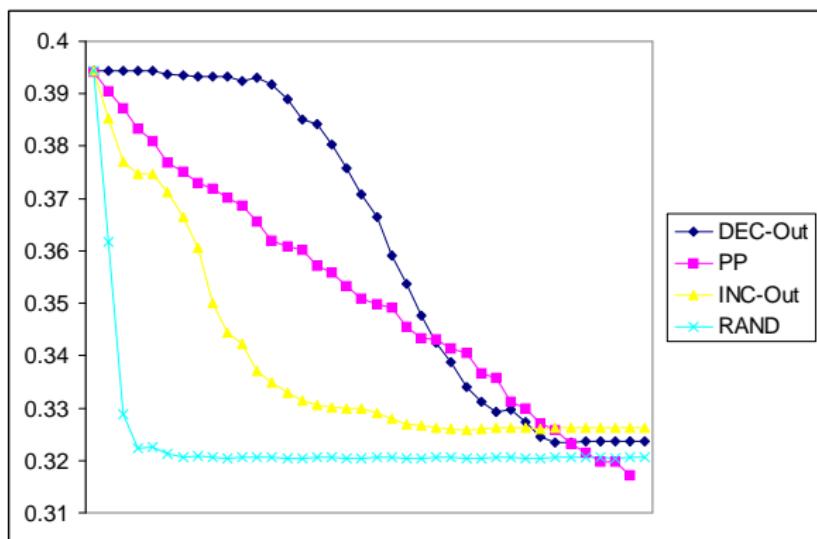


Figure 1. The convergence rates of different iteration methods on the CiteSeer data set.