Weighting Schemes in Collaborative Filtering

June 2007

Authors Paper Title and Bibtex Cite	Weighting	Applied to	Formula	Results	What is this like?
Breese et al Empirical Analysis of Predictive …breese98	inverse user frequency : items rated highly by many users in the data set are given low weights	all item ratings	raw rating * log(n/n_j). where n_j is num of users who rated item j; n is num users	improvements shown	inverse document frequency:
breese98	high correlates (nearer 1) are more valuable than low correlates	PC correlation weight	w=w^p if w >=0; w=-(-w^p) if q <0 with typical p value of 2.5	some improvement on ranking	
herlocker et al An Algorithmic framework…. herl99	significance weighting	PC correlation weight	weight * num co-rated items/50 if num co-rated less than threshold	improvement under some conditions	like a document length normalisation?
herl99	variance weighting	all items	for n ratings of item i $v_i = \frac{\mathrm{var}_i - \mathrm{var}_{\min}}{\mathrm{var}_{\max}}$ where var_i is $\sum_{i=1}^{n} \frac{r_{u,i} - \bar{r}}{n-1}$	no effect	"increase the influence of items with high variance in ratings and decrease the influence of items with low variance"-thereby giving more distinguished/influential items more influence I think
Yu et al Feature weighting and instance …yu03	inverse user frequencey (same as Breese)	items	as before: raw rating * log(n/nj). Where nj is num of users who rated item j; n num of users	reduced accuracy of prediction	contradicts Breese results?

Authors Paper Title and Bibtex Cite	Weighting	Applied to	Formula	Results	What is this like?
yu03	entropy-based where the more diverse the range of ratings an item has the more information we have	Items..weighting is applied to items when calculating predictions	weight=$H_j/H_{j,max}$ where H_j the entropy of item j is $-\sum_i p_{i,j}.\log_2 p_{i,j}$ $H_{j,max}$ is max. entropy	only slight improvement- perhaps because variance of entropy across movies is not large	similar to Herlocker's variance weighting but calculated differently?
yu03	mutual info. between item & test (target) item: item which is more similar to test item is given more credence/weight	Items- weighting is applied to items when calculating predictions	$I(X,Y)=\sum_x \sum_y p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)$	improvement	is it like pre-selecting neighbourhood of items that are similar to target item? And giving the ratings for these more credence? Almost like some 'relevance feedback' idea I think
Jin et al An automatic weighting scheme… Jin04	learns weights for items using a "leave one out" approach	items	Probabilistic optimisation problem where appropriate weights for items are found by maximizing the likelihood for each user to be similar to at least one of the other users		
Karypis Evaluation of Item-based Top-N … karypis01	column normalisation: divide by total num of users who have rated item	items(weighting is part of similarity fun)	sim(u, v)=freq(uv)/freq(v)*freq(u)^p, where p takes a value between 0 and 1	improvement	"to suppress the influence of items that are being rated frequently", like an IDF weighting … less credence to items that are rated by most people?

Authors Paper Title and Bibtex Cite	Weighting	Applied to	Formula	Results	What is this like?
karypis01	Some type of row normalisation after items have been chosen - customers that rated many items will "weigh less" during the item similarity calculations	items(weighting is part of similarity fun)	Not sure about this – unclear exactly what was done here – poorly explained	mixed	Is original intuition/hypothesis valid?-customers who rate more, have less influence
Symeonidis Collaborative Filtering: Fallacies and Insights … symeon06b	ratio which takes into account non co-rated items	PC correlations I think		improvement but used some data filling also	like normalising by document length as considering what other items exist in the *user1* by *user2* space but are not co-rated by these two users
Wang A user-item relevance model for log-based wang06	same as karypis TF/IDF I think but using a probabilistic framework and log ratings (implicit) rather than explicit ratings	**items** in relation to target user's items (weighting is part of similarity fn)	Probabilistic – uses smoothing (read again- hard to understand approach)	Improvement	“provides a probabilistic justification of using TFxIDF like item weighting in Collaborative Filtering” claim it is like TFxIDF from IR and smoothing from IR

Authors Paper Title and Bibtex Cite	Weighting	Applied to	Formula	Results	What is this like?
deBruyn et al Offering collaborative-like recommendations	not quite "pure" CF..defines an attraction weighting based on number of times an item has been recommended	items	$a_i = 1 + \ln\left(\frac{1 + c_i/c_{avg}}{1 + s_i/s_{avg}}\right)$ where c_i is num times item *i* was recommended by system and s_i is num items that are v. similar to item *i* (> certain threshold). c_{avg} and s_{avg} are average values of c_i and s_i in the database		"the more an item is recommended by CF the higher its attraction index. The denominator is a correction for uniqueness"
O'Donovan and Smyth odonovan05	Some neighbour users are given more weight/influence (correlation * trust measure) based on how many correct recommendations they have given user in past.	in the calc of predictions:	coarse grained-looking at the percentage of correct predictions a user has made in general (profile level trust) or fine grained: looking at % predictions for item that were correct	improvements	
Rashid et al Influence in Ratings-Based … Rashid05	User influence	users	Regression analysis using 7 influence factors/features of dataset, i.e. num ratings, degree of agreement, rarity of rated items, std. dev., degree of similarity with top neighs; aggregated popularity of rated items; aggregated movie popularity*entropy	Not compared to another standard CF technique	

Authors Paper Title and Bibtex Cite	**Weighting**	**Applied to**	**Formula**	**Results**	**What is this like?**
Ding et al. Time weight CF Ding05	Weight items by time factor so that older data gets less weight (at prediction stage)	To each item in neighbourhood at prediction stage	Weight assigned to each item is: $e^{-\lambda t}$ Where $\lambda = 1/T_0$ and $T_0 = 1/2f(0)$ and have to learn T_0		*Learning*
Cheung and Tian Learning User Similarity .. Cheung04	Learn optimal values of weights representing similarity between users				To deal with rating subjectivity *Learning*
Lemire Scale and Translation invariant … Lemire05a	Scale and translation invariance: normalize users w.r.t. mean, amplitude and possibly number of their ratings	users	? notation is so complex it's hard to say what is going on		