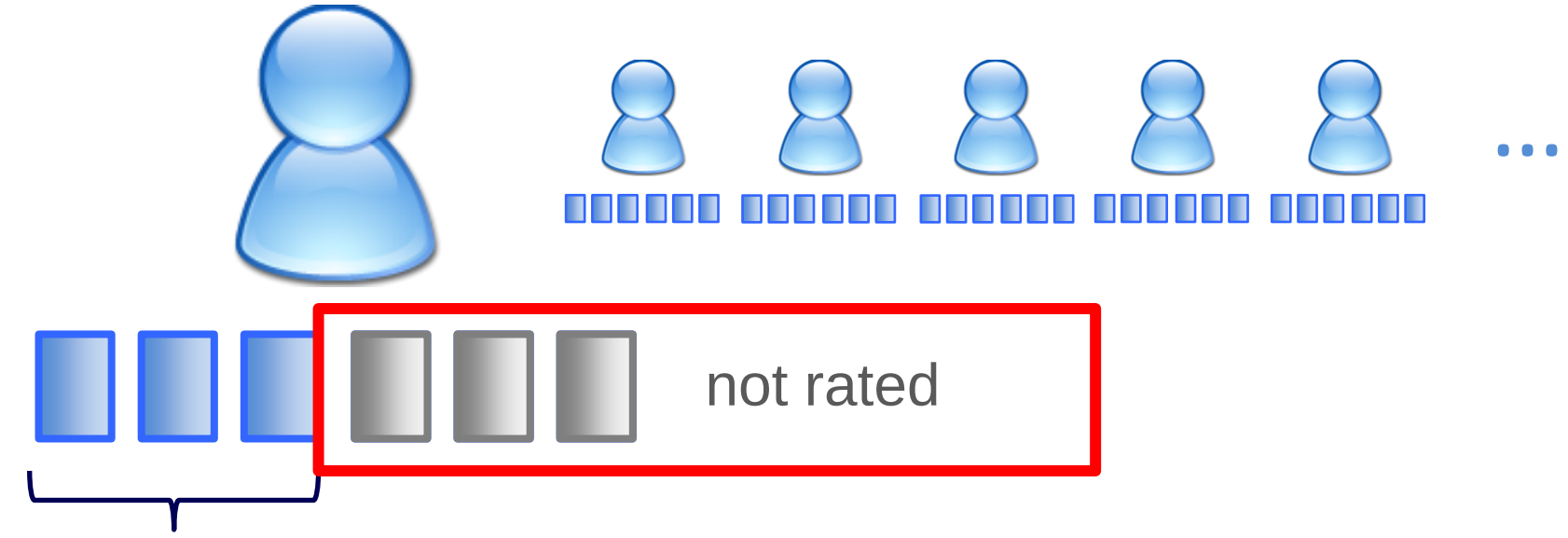
Track 2

Sebastian Stange,
Martin Köppelmann,
Caroline Fetzer

Track 2 – Assignment 2011

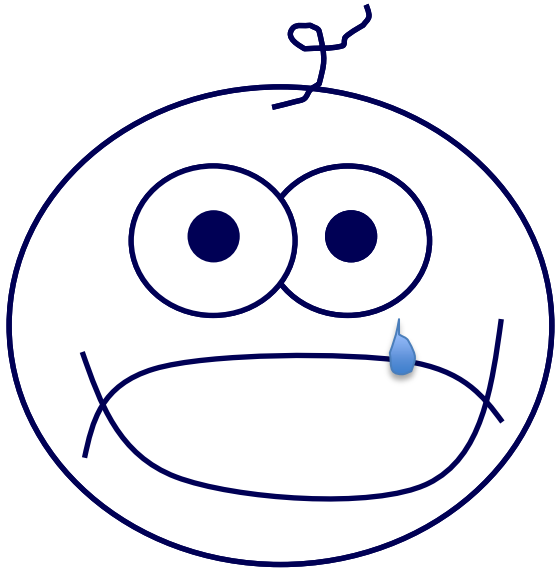
#User	#Items	#Ratings	#Ratings to predict
249.012	296.111	61.944.406	607.032



Results till now

10.127% error rate

(2.475% No. 1)



strategy

use Machine learning with:

- many different attributes
- own trainingssample
- own validationsample

Attributes

user attributes
number of ratings
number of rated genres out of the 50 most rated genres / 50
number of high rated genres out of the 50 most rated genres / 50
average rating of the user
RMSE of users ratings

Attributes

track attributes
number of ratings
number of ratings >= 80
number of genres that are within the 50 most rated genres / number of genres
number of ratings missing to 20
6 more....

Attributes

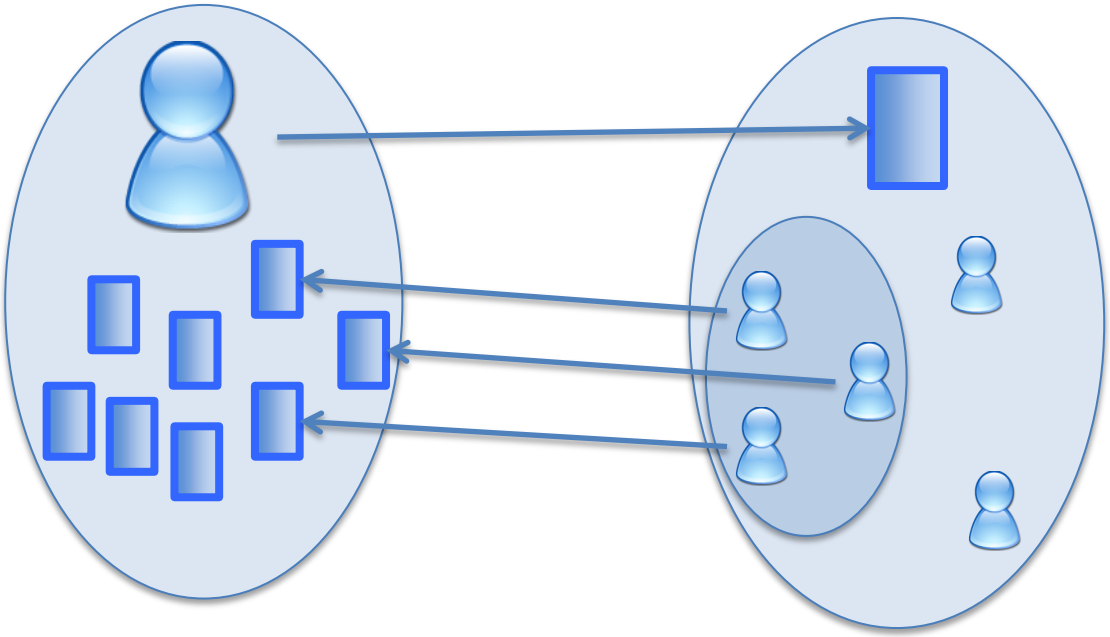
user/item attributes:
number of tracks rated of the user from the same album
number of tracks rated of the user from the same artist / #tracks of this artist
rating for genre of track
number of users rated this song and another song rated from the given user (CF)
6 more….

Attributes

Attribute	Cluster 0 (0.07)	1 (0.04)	2 (0.11)	3 (0.03)	4 (0.12)	5 (0.09)	((
getRatingCountForUser							
mean	544.0508	4735.5801	1445.9896	786.8194	1261.5473	852.8721	73
std. dev.	270.7776	5638.1886	1067.5144	557.913	904.6745	511.6272	39
getRatingRatioForUserInMostRatedGenres							
mean	0.1023	0.9384	1	0.2059	0.6587	0.0965	
std. dev.	0.0395	0.071	0.3232	0.1289	0.1941	0.0266	
getHighRatingRatioForUserInMostRatedGenres							
mean	0.0771	0.1358	0.1615	0.1384	0.3502	0.0777	
std. dev.	0.0382	0.1437	0.145	0.0798	0.207	0.033	
getAverageRatingForUser							
mean	68.4676	33.8681	41.5902	87.6189	56.4741	44.2436	4
std. dev.	10.8146	20.5535	19.4215	3.6347	17.7279	15.8869	
getStdvRatingForUser							
mean	24.0267	30.1343	32.7018	11.0899	29.9102	33.6255	3
std. dev.	5.5945	8.0249	7.0178	5.3799	8.498	6.0862	

2 Attributes

cf attributes:
Average rating of track by all users in same user cluster
Rating probability of track in the same user cluster
number of users rated this song and another song rated from the given user



3 Results for different Classifiers

Classifer	Own Validation	Yahoo
Naive Bayes	17,68%	17,56%
Logistic	15,92%	15,26%
REPTree	14,82%	14,52%
J48	14,37%	14,61%
SMO	15,86%	15,21%
PART	10,47%	10,13%

3 Differences between classifiers

	J48	REPTree	NaiveBayes	SMO	PART	Logistic
J48	0,0%	11,6%	17,5%	13,2%	10,5%	13,2%
REPTree	11,6%	0,0%	17,8%	13,7%	11,5%	13,6%
NaiveBayes	17,5%	17,8%	0,0%	12,2%	15,9%	11,8%
SMO	13,2%	13,7%	12,2%	0,0%	11,9%	1,6%
PART	10,5%	11,5%	15,9%	11,9%	0,0%	12,0%
Logistic	13,2%	13,6%	11,8%	1,6%	12,0%	0,0%

4 Merging – confidence based

1.)

$$norm(x_i) = \frac{x_i * 3}{\sum_{i=1}^{6} x_i}$$

$$stdv(x) = \sqrt{[\sum_{i=1}^{6} (x_i - 0{,}5)^2] * 6}$$

2.)

$$norm(x_i) = \frac{x_i * 3}{\sum_{i=1}^{6} x_i}$$

$$(1\text{-}prob_{max1})^2 + (1\text{-}prob_{max2})^2 + (1\text{-}prob_{max3})^2 + prob_{max4}^2 + prob_{max5}^2 + prob_{max6}^2$$

Merging – confidence based

Excel Demo

Merging – weighted average

1.) sum up probabilities of classifiers

2.) add performance of classifier as weights

3.) add confidence as weights

→ merged results outperfomed best classifier

Contributions

1) analyze data
2) sampling own trainingset and testset
3) pick useful attributes (28)
   - create user clusters for cf
4) analyze performance of different classifiers
5) merge predictions of different classifiers

Lessons learned

1) Keep track of performance
   for immediate feedback
2) know your data
3) produce a representative sample
4) end with attributes earlier & tweak more
5) make use of parallelism,
   but keep it clean and simple