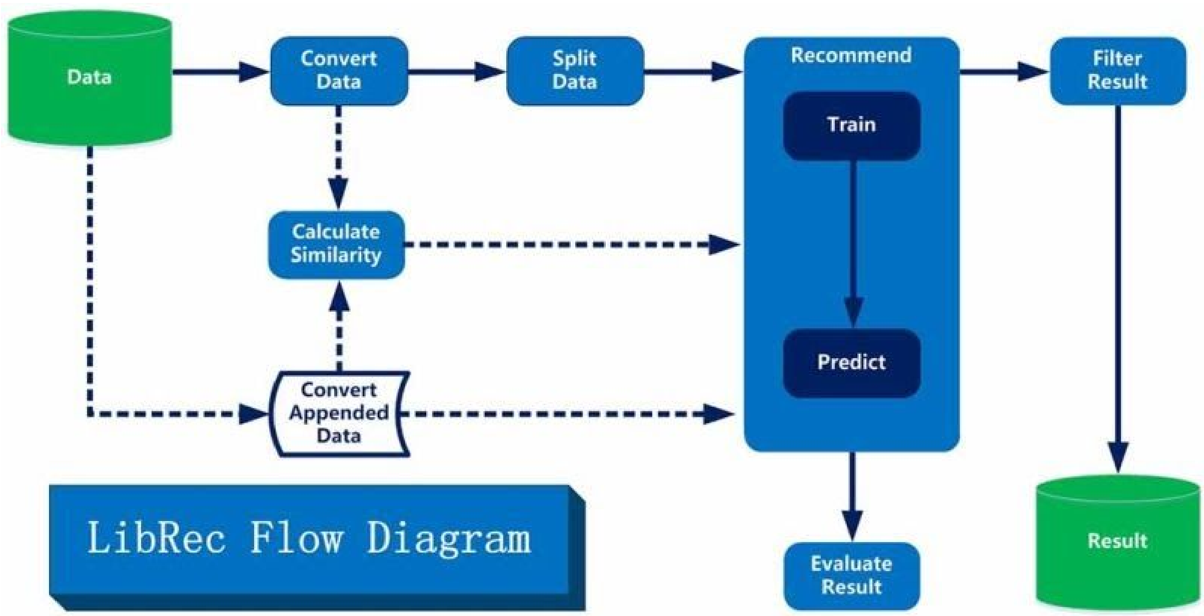
Librec
(https://www.librec.net/)
È una libreria java per recommender system, implementa una serie di algoritmi di raccomandazione “state-of-art”, che puntano a svolgere due compiti fondamentali per i RS:
-predizione dei rating
-rankare gli item

È presente anche una versione in python (https://pypi.org/project/librec/#description) (https://github.com/massquantity/LibRecommender), però è documentata molto peggio.

JAVA
La versione in java ha diverse features:
-Oltre 70 algoritmi (sotto c’è una tabella con tutti i nomi)
-Una grande modularità per i componenti
-Facile per iniziare, ci sono un po’ di demo e video che insegnano come iniziare ad utilizzare questa libreria

Librec FLOW



1) Data:
(https://www.librec.net/doc/librec-v2.0/index.html?overview-summary.html)

Librec

Data Set	Basic Meta					User Context			Other Contexts	
	Users	Items	Ratings (Scale)		Density	Users	Links (Type)		Items	Labels
Ciao	7,375	99,746	278,483	[1, 5]	0.0379%	7,375	111,781	Trust	General	
Douban	129,490	58,541	16,830,839	[1, 5]	0.222%	129,490	1,692,952	Friendship	Movie	
Epinions (665K)	40,163	139,738	664,824	[1, 5]	0.0118%	49,289	487,183	Trust	General	
Epinions (510K)	71,002	104,356	508,960	[1, 5]	0.00687%			Trust	General	
Epinions (Extended)	120,492	755,760	13,668,320	[1, 5]	0.015%			Trust Distrust	General	
Flixster	147,612	48,794	8,196,077	[0.5, 5.0]	0.1138%	787,213	11,794,648	Friendship	Movie	
FilmTrust	1,508	2,071	35,497	[0.5, 4.0]	1.14%	1,642	1,853	Trust	Movie	
Jester	59,132	140	1,761,439	Explicit	21.28%				Joke	
MovieLens 100K	943	1,682	100,000	[1, 5]	6.30%				Movie	Tag
MovieLens 1M	6,040	3,706	1,000,209	[1, 5]	4.47%				Movie	Tag
MovieLens 10M	71,567	10,681	10,000,054	[1, 5]	1.308%				Movie	Tag

2) Convert Data:
Un data convertor è un'interfaccia che converte un file di dati da un formato iniziale a un formato obiettivo.
Classi:
AbstractDataConvertor:La classe che implementa l'interfaccia
TextDataConvertor: da CSV a un formato X
ArffDataConvertor: da ARFF a un formato X
JDBCDataConvertor

3) Split Data:
Un'interfaccia che divide i dati in input.
Classi:
AbstractDataSplitter:La classe che implementa l'interfaccia
GivenNDataSplitter:divide i dati in test set e train set dato un determinato numero
GivenTestSetDataSplitter:divide il test set dato il suo percorso in input
KCVDataSplitter:K-fold cross validation
LOOCVDataSplitter:Leave one out
RatioDataSplitter:divide in train set, test set e valid set in base al rateo

4) Convert Appended Data:
Un'interfaccia che processa e salva dati "appender".
Classi:
DocumentDataAppender
SocialDataAppender

5) Similarità:
(https://github.com/guoguibing/librec/tree/3.0.0/core/src/main/java/net/librec/similarity)

Similarità calcolabili:
-CPC (Constrained Pearsons Correlation)

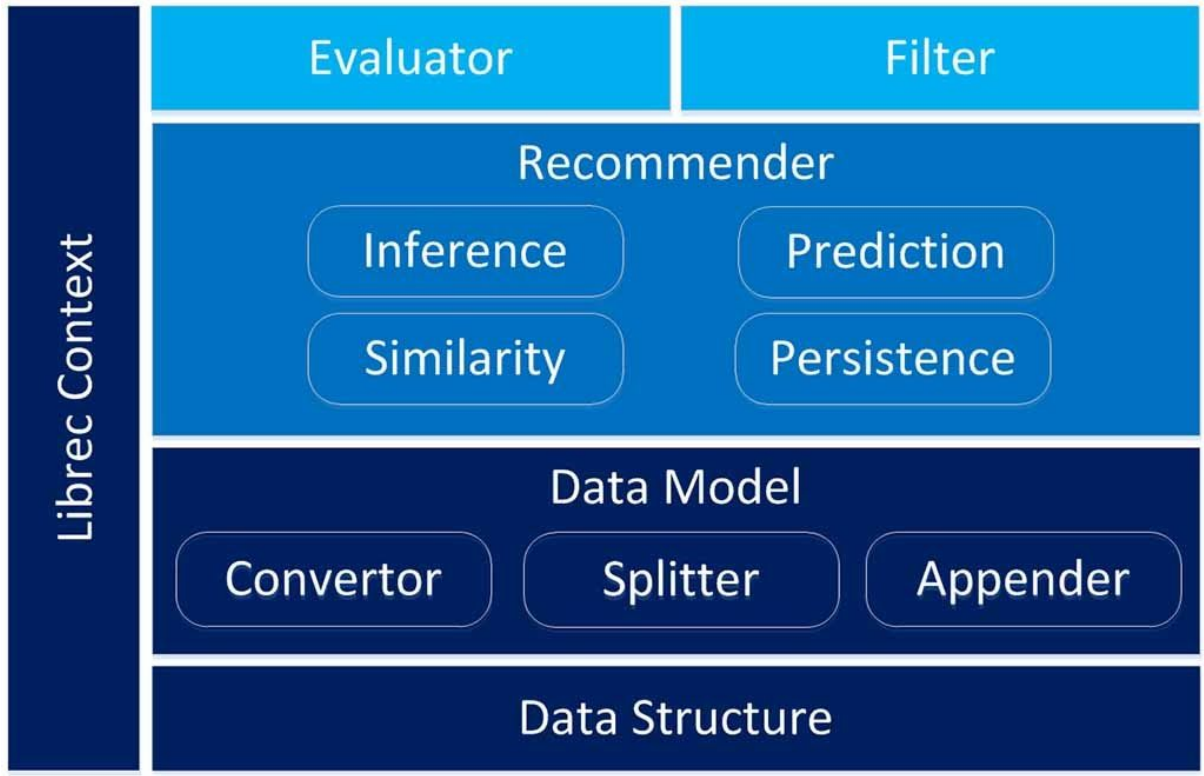
-Coefficiente di similarità del coseno
-Similarità di Jaccard
-Similarità di Jaccard Estesa
-Similarità del coseno binaria
-Coefficiente di Correlazione di Pearson
-MSE
-MSD (Mean Squared Difference)
-KRCC (Rank di correlazione di Kendall)
-Coefficiente di similarità del “dado” (Dice Coefficient Similarity)

6)Recommender:
Contenente interfacce e classi per implementare meccanismi basilari per il RS.

7)Evaluate Result:
Interfaccia che valuta la qualità delle raccomandazioni.
Classi:
AbstractRecommenderEvaluator:La classe che implementa l’interfaccia
AUCEvaluator
AveragePrecisionEvaluator
AverageReciprocalHitRankEvaluator
DiversityEvaluator
HitRateEvaluator
IdealDCGEvaluator
MAEEvaluator
MPEEvaluator
MSEEvaluator
NormalizedDCGEvaluator
PrecisionEvaluato:
RecallEvaluator
ReciprocalRankEvaluator
RMSEEvaluator
Ognuna di queste classi valuta in base alla metrica presente nel proprio nome.
MAEE, MPEE, MSEE e RMSEE sono per valutare il ranking e le altre per valutare il rating.

8) Filter Result:
Interfaccia per filtrare i raccomandati.
Classe:
GenericRecommendedFilter: La classe che la implementa

Librec



ALGORITMI

(tabella presa da:
https://github.com/guoguibing/librec/blob/3.0.0/doc/E-wiki/AlgorithmList.md#recommender-algorithm-list)
(Documentazione: https://www.librec.net/doc/librec-v2.0/index.html?overview-summary.html)

Classi:
Gamma
SVD:Singular Value Decomposition
Stats
Shuffle:rimescola matrici sparse
Randoms
Maths
KernelSmoothing
Gaussian

superClass	directory path	short name	algorithm
MatrixRecommender	baseline	constantguess	ConstantGuessRecommender
MatrixRecommender	baseline	globalaverage	GlobalAverageRecommender
MatrixRecommender	baseline	itemaverage	ItemAverageRecommender
MatrixProbabilisticGraphicalRecommender	baseline	itemcluster	ItemClusterRecommender
MatrixRecommender	baseline	mostpopular	MostPopularRecommender
MatrixRecommender	baseline	randomguess	RandomGuessRecommender
MatrixRecommender	baseline	useraverage	UserAverageRecommender
MatrixProbabilisticGraphicalRecommender	baseline	usercluster	UserClusterRecommender
MatrixProbabilisticGraphicalRecommender	cf	bhfree	BHFreeRecommender
MatrixProbabilisticGraphicalRecommender	cf	bucm	BUCMRecommender
MatrixRecommender	cf	itemknn	ItemKNNRecommender
MatrixRecommender	cf	itemknn	ItemKNNRecommender
MatrixRecommender	cf	userknn	UserKNNRecommender
MatrixRecommender	cf	userknn	UserKNNRecommender
MatrixFactorizationRecommender	cf.ranking	aobpr	AoBPRRecommender
MatrixProbabilisticGraphicalRecommender	cf.ranking	aspectmodelranking	AspectModelRecommender
MatrixFactorizationRecommender	cf.ranking	bnppf	BNPPFRecommeder
MatrixFactorizationRecommender	cf.ranking	bpoissmf	BPoissMFRecommender
MatrixFactorizationRecommender	cf.ranking	bpr	BPRRecommender
MatrixFactorizationRecommender	cf.ranking	climf	CLIMFRecommender
MatrixFactorizationRecommender	cf.ranking	cofiset	CoFiSetRecommender
MatrixFactorizationRecommender	cf.ranking	eals	EALSRecommender
MatrixFactorizationRecommender	cf.ranking	fismauc	FISMaucRecommender
MatrixFactorizationRecommender	cf.ranking	fismrmse	FISMrmseRecommender
MatrixFactorizationRecommender	cf.ranking	gbpr	GBPRRecommender
MatrixProbabilisticGraphicalRecommender	cf.ranking	itembigram	ItemBigramRecommender

MatrixProbabilisticGraphicalRecommender	cf.ranking	lda	LDARecommender
MatrixFactorizationRecommender	cf.ranking	listrankmf	ListRankMFRecommender
MatrixFactorizationRecommender	cf.ranking	nmfitemitem	NMFItemItemRecommender
MatrixProbabilisticGraphicalRecommender	cf.ranking	plsa	PLSARecommender
MatrixFactorizationRecommender	cf.ranking	pnmf	PNMFRecommender
MatrixFactorizationRecommender	cf.ranking	rankals	RankALSRecommender
MatrixFactorizationRecommender	cf.ranking	rankpmf	RankPMFRecommender
MatrixFactorizationRecommender	cf.ranking	ranksgd	RankSGDRecommender
MatrixFactorizationRecommender	cf.ranking	slim	SLIMRecommender
MatrixFactorizationRecommender	cf.ranking	wbpr	WBPRRecommender
MatrixFactorizationRecommender	cf.ranking	wrmf	WRMFRecommender
MatrixProbabilisticGraphicalRecommender	cf.rating	aspectmodelrating	AspectModelRecommender
cf.rating.BiasedMFRecommender	cf.rating	asvdpp	ASVDPlusPlusRecommender
MatrixFactorizationRecommender	cf.rating	biasedmf	BiasedMFRecommender
MatrixFactorizationRecommender	cf.rating	bpmf	BPMFRecommender
FactorizationMachineRecommender	cf.rating	ffm	FFMRecommender
FactorizationMachineRecommender	cf.rating	fmals	FMALSRecommender
FactorizationMachineRecommender	cf.rating	fmftrl	FMFTRLRecommender
FactorizationMachineRecommender	cf.rating	fmsgd	FMSGDRecommender
MatrixProbabilisticGraphicalRecommender	cf.rating	gplsa	GPLSARecommender
MatrixFactorizationRecommender	cf.rating	irrg	IRRGRecommender
MatrixProbabilisticGraphicalRecommender	cf.rating	ldcc	LDCCRecommender
MatrixFactorizationRecommender	cf.rating	llorma	LLORMARecommender
MatrixFactorizationRecommender	cf.rating	mfals	MFALSRecommender
MatrixFactorizationRecommender	cf.rating	nmf	NMFRecommender
MatrixFactorizationRecommender	cf.rating	pmf	PMFRecommender

MatrixRecommender	cf.rating	rbm	RBMRecommender
MatrixFactorizationRecommender	cf.rating	remf	ReMFRecommender
MatrixFactorizationRecommender	cf.rating	rfrec	RFRecRecommender
cf.rating.BiasedMFRecommender	cf.rating	svdpp	SVDPlusPlusRecommender
MatrixProbabilisticGraphicalRecommender	cf.rating	urp	URPRecommender
TensorRecommender	content	convmf	ConvMFRecommender
TensorRecommender	content	efm	EFMRecommender
TensorRecommender	content	hft	HFTRecommender
TensorRecommender	content	tfidf	TFIDFRecommender
TensorRecommender	content	topicmfat	TopicMFATRecommender
TensorRecommender	content	topicmfmt	TopicMFMTRecommender
FactorizationMachineRecommender	context.ranking	dlambdafm	DLambdaFMRecommender
SocialRecommender	context.ranking	sbpr	SBPRRecommender
TensorRecommender	context.rating	cptf	CPTFRecommender
SocialRecommender	context.rating	rste	RSTERecommender
SocialRecommender	context.rating	socialmf	SocialMFRecommender
SocialRecommender	context.rating	sorec	SoRecRecommender
SocialRecommender	context.rating	soreg	SoRegRecommender
cf.rating.BiasedMFRecommender	context.rating	timesvd	TimeSVDRecommender
SocialRecommender	context.rating	trustmf	TrustMFRecommender
SocialRecommender	context.rating	trustsvd	TrustSVDRecommender
MatrixFactorizationRecommender	cf.ranking	fismauc	FISMaucRecommender
MatrixFactorizationRecommender	cf.ranking	fismrmse	FISMrmseRecommender
MatrixFactorizationRecommender	cf.rating	biasedmf	BiasedMFRecommender
MatrixFactorizationRecommender	cf.rating	biasedmf	BiasedMFRecommender
MatrixFactorizationRecommender	cf.rating	nmf	NMFRecommender

MatrixFactorizationRecommender	cf.rating	pmf	PMFRecommender
MatrixRecommender	ext	associationrule	AssociationRuleRecommender
MatrixRecommender	ext	bipolarslopeone	BipolarSlopeOneRecommender
MatrixRecommender	ext	external	ExternalRecommender
MatrixRecommender	ext	personalitydiagnosis	PersonalityDiagnosisRecommender
cf.ranking.RankSGDRecommender	ext	prankd	PRankDRecommender
MatrixRecommender	ext	slopeone	SlopeOneRecommender
MatrixRecommender	hybrid	hybrid	HybridRecommender
MatrixRecommender	cf	itemknn	ItemKNNRecommender
MatrixFactorizationRecommender	cf.rating	pmf	PMFRecommender
MatrixRecommender	cf	userknn	UserKNNRecommender
MatrixRecommender	nn.ranking	cdae	CDAERecommender
MatrixRecommender	nn.rating	autorec	AutoRecRecommender
MatrixFactorizationRecommender	poi	rankgeofm	RankGeoFMRecommender
AbstractRecommender	poi	usg	USGRecommender

STRUTTURE DATI

(https://github.com/guoguibing/librec/tree/3.0.0/core/src/main/java/net/librec/math/structure)
le strutture dati elementari utilizzate sono:
Map
BiMap (Map bidirezionale)
Lista
Matrice Sparsa
Matrice Densa
Matrice Sparsa di stringhe
“Tensor” Sparso (https://epubs.siam.org/doi/abs/10.1137/07070111x)
Vettore Sparso
Vettore Denso
Matrice Simmetrica
Matrice Diagonale

RAPPRESENTAZIONE DATI

(https://github.com/guoguibing/librec/blob/454b040cfbb0f64bf3f5f379c9e1f02d99b0d8df/librec/src/main/java/lib/rec/DataDAO.java) Classe DAO (Data Access Object):
User e Item sono entrambi rappresentati da una coppia {Raw ID, Inner ID} utilizzando la struttura Map e BiMap, il raw id è una Stringa, l’inner id è un int.
Sono messi in realzione nella struttura BiMap.

```
// user/item {raw id, inner id} map
private BiMap<String, Integer> userIds, itemIds;
```

Python

La documentazione è molto povera se non inesistente, c'è una spiegazione molto esaustiva però sul come vengano rappresentati i dati

Data Format

JUST normal data format, each line represents a sample. By default, model assumes that `user`, `item`, and `label` column index are 0, 1, and 2, respectively. But you need to specify `user`, `item`, and `label` column index if that's not the case. For Example, the `movielens-1m` dataset:

> 1::1193::5::978300760
> 1::661::3::978302109
> 1::914::3::978301968
> 1::3408::4::978300275

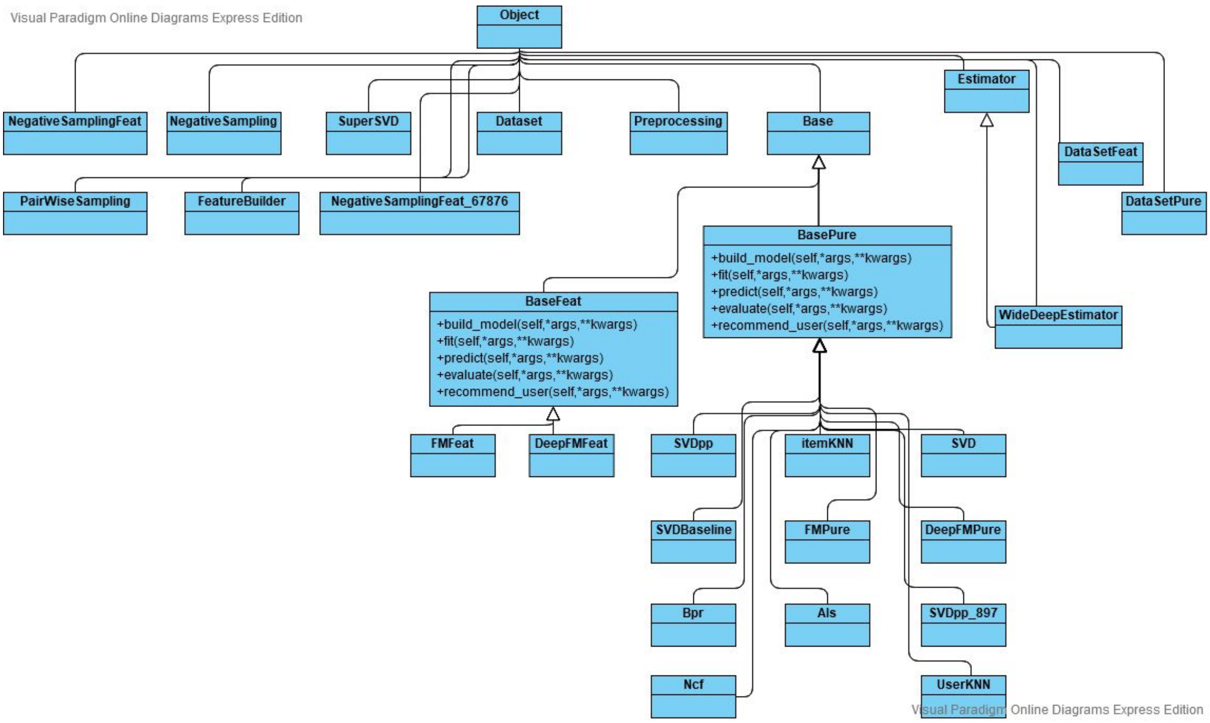
leads to the following settings in `conf` dict : `"user_col": 0, "item_col": 1, "label_col": 2, "sep": "::"` .

Besides, if you want to use some other meta features (e.g., age, sex, category etc.), `numerical` and `categorical` column index must be assigned. For example, `"numerical_col": [4], "categorical_col": [3, 5, 6, 7, 8]` , which means all features must be in a same table.

Il Librec flow rimane pressochè lo stesso, lo strutture dati anche, una cosa che varia molto sono gli algoritmi utilizzati, ci sono algoritmi puramente collaborative e algoritmi ibridi, è presente anche un file .py chiamato preprocessing, per processare il testo.
Gli algoritmi collaborative filtering sono:
userKNN/itemKNN (https://gyazo.com/452d2f15612d84bd44317fec59a61b7b)
SVD (https://datajobs.com/data-science-repo/Recommender-Systems-%5BNetflix%5D.pdf)
SVD ++ (https://dl.acm.org/doi/10.1145/1401890.1401944)
ALS (https://stanford.edu/~rezab/classes/cme323/S15/notes/lec14.pdf),
(https://stanford.edu/~rezab/classes/cme323/S15/notes/lec14.pdf),
(http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.379.6473&rep=rep1&type=pdf)
NCF(https://arxiv.org/pdf/1708.05031.pdf)
BPR (https://arxiv.org/ftp/arxiv/papers/1205/1205.2618.pdf)

Gli algoritmi ibridi sono:
Wide & Deep (https://arxiv.org/pdf/1606.07792.pdf)
FM (https://www.csie.ntu.edu.tw/~b97053/paper/Rendle2010FM.pdf)
DeepFM (https://arxiv.org/pdf/1703.04247.pdf)
YoutubeRec(https://static.googleusercontent.com/media/research.google.com/zh-CN//pubs/archive/45530.pdf)
DIN (https://arxiv.org/pdf/1706.06978.pdf)

Architettura



Data la presenza di algoritmi ibridi abbiamo l'interfaccia Base da cui derivano BasePure e BaseFeat da cui si possono implementare i metodi astratti per creare il modello, fare raccomandazioni, fare la valutazione e predire.
Da BasePure e BaseFeat derivano quasi tutti gli algoritmi utilizzabili in questo framework, da BasePure quelli puramente collaborativi, da BaseFeat quelli ibridi.
Abbiamo questa differenza anche nei dataset, con Dataset, DatasetFeat e DatasetPure.
La classe preprocessing serve per processare il Dataset in previsione degli algoritmi ibridi che prevedono l'utilizzo di euristiche content based.
SuperSVD e WideDeepEstimator sono i due algoritmi che non derivano da Base e che non implementano i suoi metodi astratti.