Referências Bibliográficas


Christophe Prieur Nicolas Pissard. Thematic vs. social networks in web 2.0 communities: A case study on flickr groups. 2007.


Figura A.1: Classes e propriedades da ontologia SIOC-Core
Figura A.2: Classes e propriedades da ontologia SIOC-Types
Figura A.3: Contexto Flickr completo. Crédito: http://soldierant.net/
B
Módulos de software da framework

O software integrante da framework proposta na seção 3.2 encontra-se disponível como um projeto open-source em http://code.google.com/p/recfwk/. Neste endereço são disponibilizados – além de downloads e exemplos de uso – mais detalhes sobre os pacotes, classes e interfaces integrantes dessa framework, que descrevemos rapidamente a seguir.

B.1 Packages

recfwk.engine Provides implementations for the main framework interfaces.
recfwk.filters Provides interfaces and implementations for data filters.
recfwk.io Provides implementations for data input and output.
recfwk.model Provides basic entities for implementing and conducting experiments with recommenders: Items, Recommendations, Users etc.
recfwk.util Provides misc utils: I/O, statistics and probability, performance measuring, string manipulation etc.
recfwk.vis Provides interfaces and implementations for representing performance results and experiment parameters graphically.

B.2 Package recfwk.engine

Provides implementations for the main framework interfaces.
Contains all implementations not falling on the other categories such as data filters, basic entity models, data loaders etc.

B.2.1 Class Summary

BaseConfig Holds the most important config parameters.
ExperimentRecorder Records experiment results to disk
**SetRetrievalEvaluator** Evaluates how well a recommender suggests items that should belong to a given set, by verifying whether a recommended item is indeed on the training set (repeated hold-out technique)

**B.3**
**Package recfwk.filters**
Provides interfaces and implementations for data filters.

**B.3.1**
**Interface Summary**
- **Filter**: filters a stream of data tuples

**B.3.2**
**Class Summary**
- **RandomSampleFilter**: random filter; randomly selects a given percentage of all filtered tuples

**B.4**
**Package recfwk.io**
Provides implementations for data input and output.

**B.4.1**
**Class Summary**
- **CSVItemTupleReader**: Reads data from text comma-separated files.
- **CSVItemTupleWriter**: writes a list of tuples to disk as comma-separated text files

**B.5**
**Package recfwk.model**
Provides basic entities for implementing and conducting experiments with recommenders: Items, Recommendations, Users etc.

**B.5.1**
**Interface Summary**
- **ItemSimilarity**: Stores the similarity between two content items
- **ItemTupleReader**: Reads data tuples
- **ItemTupleWriter**: Persists data tuples
- **Recommender**: Provides recommendations to target items.
B.5.2
Class Summary

Item  Holds basic info about a content item
Recommendation  Represents a recommendation made.
RecommendedItem  Represents a recommended item.

B.6
Package recfwk.util

Provides misc utils: I/O, statistics and probability, performance measuring, string manipulation etc.

B.6.1
Class Summary

IOUtils  I/O-related utility methods that don’t have a better home.
RandomUtil  Provides helpers for common random/statistics functions.
StopWatch  Performance helper for measuring time lapses.
StringUtils  Misc string utils.

B.7
Package recfwk.vis

Provides interfaces and implementations for representing performance results and experiment parameters graphically.

B.7.1
Interface Summary

BasicPlot  Provides basic plot functions.
PlotBivariatePerformance  Plots experimental data where you have a series two variables and an associated performance rate.
PlotHistogram  Plots frequency histograms of given a series of entities or events and their associated count of occurrence.
PlotUnivariatePerformance  Plots (line charts) experimental data where you have a series of performance rates and the associated value of an experiment variable.