A Java Toolbox for Analysis of Massive Data Streams using Probabilistic Graphical Models

Masegosa, Andres; Martinez, Ana M.; Ramos-López, Darío; Langseth, Helge; Nielsen, Thomas Dyhre; Salmerón, Antonio; Cabanas de Paz, Rafael; Madsen, Anders Læsø

Publication date:
2016

Link to publication from Aalborg University

Citation for published version (APA):
A Java Toolbox for Analysis of Massive Data Streams using Probabilistic Graphical Models

Andrés R. Masegosa, Ana M. Martínez, Darío Ramos-Lopez, Helge Langseth, Thomas D. Nielsen, Antonio Salmerón, Rafael Cabañas & Anders L. Madsen

1 Department of Computer and Information Science, NTNU, Norway
2 Department of Computer Science, Aalborg University, Denmark
3 Department of Mathematics, University of Almería, Spain
4 Hugin Expert A/S, Aalborg, Denmark

Presentation

Data mining frameworks

- PGMs
- AMIDST
- Sta4onary data sets
- Data streams

- MLlib | Apache Spark/Flink
- MOA
- Elvira
- Infer.net
- Hugin
- Weka
- R
- R Libs
- Matlab
- Apache SAMOA
- Vowpal Wabbit

Academic and Industrial partners

Description

- Analysis of big data streams: A complete collection of algorithms for inference and learning of both static and dynamic Bayesian networks from streaming data. Existing software systems for PGMs only focus on stationary datasets.
- Distributed parallel algorithms: AMIDST provides parallel multi-core and distributed implementations of Bayesian parameter learning, using streaming variational Bayes and variational message passing.

Main Features

- Java 8 based
- Latent variable models
- Integration

Code example

Learn hidden naive Bayes model from data stream

```java
// We create a MAP algorithm
PMap<FactorGraph, FactorGraph> algorithm = new PMap();
// We fix the map structure
algorithm.setMapStructure(\text{Marginal\text{\_}Network\text{\_}Structure});
// We fix the size of the window
algorithm.setWindowSize(100);
// We activate the output
algorithm.setOutput(true);
// We set the data which is going to be used for leaning the parameters
algorithm.setDataStream(data);
// We perform the leaning
algorithm.runLearning();
// We can open the data stream using the static class DataStreamLoader
DataStreamLoader.openFromFile("datasets/simulated\/WasteIncineratorSample.arff");
```

Use-case: Risk prediction in credit operations

Concept drift Correlated with Unemployment Rate

And much more...

amidst.eu
amidst.github.io/toolbox/

AMIDST project has received funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no 619209.