AMIDST: Analysis of MassIve Data STreams

Masegosa, Andres; Martinez, Ana M.; Borchani, Hanen; Ramos-López, Darío ; Nielsen, Thomas Dyhre; Langseth, Helge; Salmerón, Antonio; Madsen, Anders Læsø

Published in:
The 27th Benelux Conference on Artificial Intelligence (BNAIC 2015)

Publication date:
2015

Document Version
Early version, also known as pre-print

AMIDST: Analysis of MassIve Data STreams

Andrés R. Masegosa [a] Ana M. Martínez [b] Hanen Borchani [b]

Darío Ramos-López [c] Thomas D. Nielsen [b] Helge Langseth [a]

Antonio Salmerón [c] Anders L. Madsen [b,d]

[a] *Department of Computer and Information Science, The Norwegian University of Science and Technology, Norway*
[b] *Department of Computer Science, Aalborg University, Denmark*
[c] *Department of Mathematics, University of Almería, Spain*
[d] *HUGIN EXPERT A/S, Aalborg, Denmark*

Abstract

The Analysis of MassIve Data STreams (AMIDST) Java toolbox provides a collection of scalable and parallel algorithms for inference and learning of hybrid Bayesian networks from data streams. The toolbox, available at `http://amidst.github.io/toolbox/` under the Apache Software License version 2.0, also efficiently leverages existing functionalities and algorithms by interfacing to software tools such as HUGIN and MOA.

1 Purpose

The Analysis of MassIve Data STreams (AMIDST) toolbox offers a scalable framework for data stream analysis based on probabilistic graphical models (PGMs). Where other software systems developed for PGMs only focus on mining *stationary* data sets [2], AMIDST provides contributions to efficient data analysis using PGMs for mining both stationary and streaming data that may be subject to concept drift (see Figure 1). AMIDST relies on the Bayesian network (BN) [3] framework, and its immediate extensions, as particular types of PGMs.

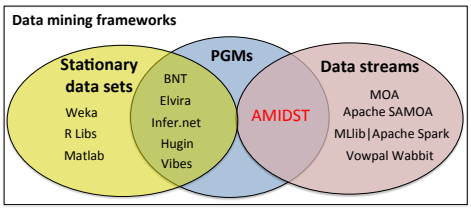

Figure 1: A non-exhaustive taxonomy of data mining software frameworks.

2 AMIDST

AMIDST is an open source toolbox available at `http://amidst.github.io/toolbox/` under the Apache Software License version 2.0. It was developed within the context of the AMIDST research project (`http://amidst.eu/`) by Andrés R. Masegosa, Ana M. Martínez, Hanen Borchani, Darío Ramos-López, Thomas D. Nielsen, Helge Langseth, Antonio Salmerón, and Anders L. Madsen.

The intended user groups consist, on the shorter time horizon, of the industrial AMIDST project partners, namely, the German multinational automotive corporation *Daimler AG* and the Spanish bank

Banco de Crédito Social Cooperativo S.A. These companies will use AMIDST for real time identification and interpretation of manoeuvres in traffic and risk prediction in credit operations, respectively, and the end-users will test the developed algorithms on real-world extremely large data streams. Moreover, since AMIDST is generic, i.e., can be employed in a vast range of industrial contexts with varying data characteristics, additional intended user groups may subsequently include other industrial companies, institutions, and/or individuals that are interested in performing efficient analysis and prediction based on information captured in streaming data. Besides using AMIDST, user groups are encouraged to incorporate their potential developments and collaborations via AMIDST GitHub Fork and Pull requests.

AMIDST makes use of Java 8's functional programming style to support parallel processing on mutli-core CPUs. It supplies several functionalities for inference and learning hybrid BNs from streaming data, including:

- Parallel processing of data streams using Java 8's functionalities.
- Implementations for BN representations both in standard format and as conjugate exponential family models [6]. Discrete and continuous variables, having multionomial, Gaussian, and conditional linear Gaussian distributions, are also supported.
- An inference engine including implementations of the *variational message passing* [6] and parallel *importance sampling* [4] algorithms.
- A learning engine consisting of multi-core parallel implementations of the *streaming variational Bayes* [1] and *maximum likelihood estimation* [5] algorithms.
- Links to existing software tools such as HUGIN[1] and MOA (Massive Online Analysis)[2]. This allows the toolbox to efficiently exploit well-established systems and also broaden the AMIDST user groups.

3 Demonstration

During the proposed demonstration, we plan to first give a general introduction to the issues related to data stream analysis and discuss why scalability is important within this context. Next, we will describe the AMIDST toolbox design and architecture, show how to install it, and run several examples to illustrate its main functionalities. A special emphasis will be given to the scalability features of the toolbox and how they are supported using the functional programming style of the Java 8 API. A regular desktop or laptop computer may be used for the demo. There are no special system requirements since AMIDST is based on Java technology and is platform independent. The demo is expected to take approximately 20 minutes and extra time may be devoted to particular audience queries.

Acknowledgments

This work was performed as part of the AMIDST project funded by the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 619209.

References

[1] Tamara Broderick, Nicholas Boyd, Andre Wibisono, Ashia C. Wilson, and Michael I. Jordan. Streaming variational Bayes. In *Advances in Neural Information Processing Systems*, pages 1727–1735, 2013.

[2] Kevin Murphy. Software for graphical models: A review. *International Society for Bayesian Analysis Bulletin*, 14(4):13–15, 2007.

[3] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, CA., 1988.

[4] Antonio Salmerón, Darío Ramos-López, et al. Parallel importance sampling in conditional linear Gaussian networks. In *Conferencia de la Asociación Española para la Inteligencia Artificial*, volume in press, 2015.

[5] F. W. Scholz. *Maximum Likelihood Estimation*. John Wiley & Sons, Inc., 2004.

[6] John M. Winn and Christopher M. Bishop. Variational message passing. *Journal of Machine Learning Research*, 6:661–694, 2005.

[1] http://www.hugin.com

[2] http://moa.cms.waikato.ac.nz