Number 109 April 2017

ERCIM NEWS

www.ercim.eu

Special theme:

Autonomous Vehicles

Also in this issue:

Research and Society:

Virtual Research Environments

ERCIM News is the magazine of ERCIM. Published quarterly, it reports on joint actions of the ERCIM partners, and aims to reflect the contribution made by ERCIM to the European Community in Information Technology and Applied Mathematics. Through short articles and news items, it provides a forum for the exchange of information between the institutes and also with the wider scientific community. This issue has a circulation of about 6,000 printed copies and is also available online.

ERCIM News is published by ERCIM EEIG
BP 93, F-06902 Sophia Antipolis Cedex, France
Tel: +33 4 9238 5010, E-mail: contact@ercim.eu
Director: Philipp Hoschka, ISSN 0926-4981

Contributions
Contributions should be submitted to the local editor of your country

ERCIM News online edition
http://ercim-news.ercim.eu/

Next issue
July 2017, Special theme: Blockchain Engineering

Subscription
Subscribe to ERCIM News by sending an email to en-subscriptions@ercim.eu or by filling out the form at the ERCIM News website: http://ercim-news.ercim.eu/

Contents

JOINT ERCIM ACTIONS

RESEARCH AND SOCIETY

This section on Virtual Research Environments has been coordintated by Keith G. Jeffery and Pierre Guisset (ERCIM)

SPECIAL THEME

The special theme section “Autonomous Vehicles” has been coordinated Erwin Schoitsch (AIT) and Fawzi Nashashibi (Inria)

RESEARCH AND INNOVATION

This section features news about research activities and innovative developments from European research institutes

CALLS, IN BRIEF

Event

Calls

In Brief

An Integrated Software Development Lifecycle for Intelligent Automotive Software: The W Model

by Fabio Falcini and Giuseppe Lami (ISTI-CNR)

Deep learning is becoming crucial to the development of automotive software for applications such as autonomous driving. The authors have devised a framework that supports a robust, disciplined development lifecycle for such software, and a comprehensive integration with traditional automotive software engineering.

Deep learning is a branch of machine learning based on a set of artificial neural network (ANN) algorithms that model high-level abstractions in input data by using a deep graph representation made of multiple processing layers [1].

The introduction of deep-learning technology on board cars is starting to have a significant effect on the automotive software engineering that needs to incorporate and harmonise the development of these technologies [2].

The software development process for on board automotive electronic control units (ECUs) is subject to proprietary OEM norms as well as several international standards. Among them, the most relevant and influential standards for deep learning are Automotive SPICE and ISO 26262. Needless to say, these standards are still far from addressing it with dedicated statements.

The Automotive SPICE standard – SPICE stands for “Software Process Improvement and Capability dEtermination” – provides a process framework that disciplines the software development activities. ISO 26262, titled ‘Road Vehicles – Functional Safety’, released in late 2011, targets safety-related development and its scope expectedly includes system, hardware and software engineering.

Both standards, as far as the software is concerned, rely conceptually on the traditional development lifecycle: the V-model. Also very relevant for deep learning is the ISO PAS ‘Safety of the Intended Functionality’ (ISO/TC 22 N356).

Because of its pervasive adoption and its holistic coverage of the automotive software development processes, Automotive SPICE standard can be recognised as the appropriate reference.

The software side of deep neural network (DNN) development is a highly iterative activity composed by a stream of steps in an end-to-end fashion, as shown in Figure 1.

Compared with traditional approaches, the deep learning development process needs the support of empirical design choices driven by heuristics. Development often starts from well-known learning algorithms, which have been proven effective in comparable problems or domains [3].

Automotive software engineering, while welcoming innovation and outstanding functional performances, remains strict in its request for a robust and predictable development cycle. For that reason, it is important to place deep learning in more controlled V-model perspective to address a lengthy list of challenges, such as requirements criteria for training, validation and test data sets, and much more.

The introduction of a more structured conception of the deep learning lifecycle is instrumental to reach a controlled development approach that cannot be addressed by the mere functional benchmarking obtained with validation activities. It is essential to pursue both fundamental directions: high performance at the functional level and a high-quality development process.

However, the central role that is played by data in this context has stimulated the authors to introduce of a new development lifecycle model: the W model. To support it, we employ the term ‘programming by example’ to highlight the

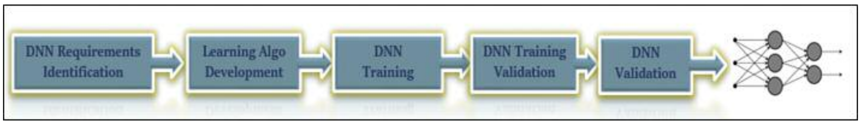

Figure 1: DNN development workflow.

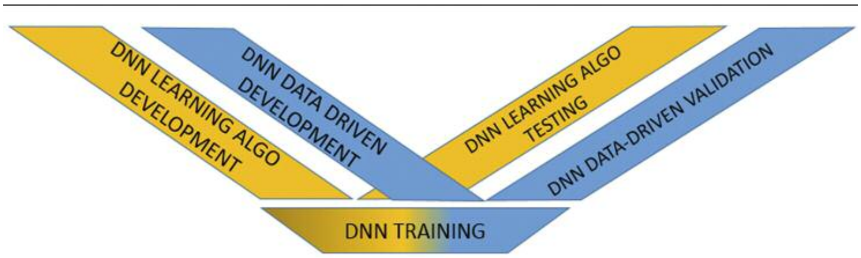

Figure 2: The W-model for deep learning.

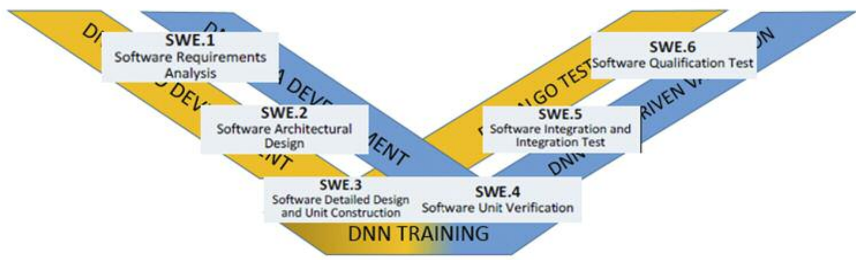

Figure 3: The W-model in Automotive SPICE 3.0 perspective.

importance of data in developing systems based on deep-learning technology.

The deep learning W-model is a framework lifecycle that conceptually integrates a V model for data development in the standard V perspective (Figure 2).

This lifecycle model acknowledges that both software development and data development drive deep learning. The design and creation of training, validation and test datasets, together with their exploitation, are crucial development phases because the DNN's functional behaviour is the combined result of its architectural structure and its automatic adaptation through training. By definition, deep learning moves away from feature engineering. This aspect makes the W model an appropriate, useful representation of this sophisticated paradigm.

By placing the software part of the V-model of ASPICE 3.0 on top of the coined W-model, the following diagram hints at the integration of deep learning along the process requirements of the Automotive SPICE model (Figure 3). The same approach may apply for ISO 26262.

As deep learning ushers in radical changes to automotive software development (characterised by stringent requirements in terms of rigor, control and compliance with standards), the W model is a promising basis for the comprehensive integration of deep learning with traditional automotive software engineering.

References:
[1] S. Haykin: "Neural Networks and Learning Machines", Prentice-Hall, 2009.
[2] F. Falcini, G. Lami: "Deep Learning in Automotive Software", IEEE Software Vol. 34 No. 3, May-June 2017.
[3] J. Schmidhuber: "Deep learning in neural networks: An overview", Neural Networks Vol 61, 85-117, Jan 2015.

Please contact:
Giuseppe Lami, Fabio Falcini
ISTI-CNR, Italy
giuseppe.lami@isti.cnr.it
fabio.falcini@isti.cnr.it

Reliable Vehicular-to-Vehicular Connectivity

by Lisa Kristiana, Corinna Schmitt, and Burkhard Stiller (University of Zurich)

Vehicular-to-vehicular (V2V) communications requires a reliable information interchange system. However, in a three-dimensional environment, inevitable obstructions (e.g., road topology and buildings) need to be considered in order to design data forwarding schemes. As one approach, Vehicle-to-Vehicle Urban Network (V2VUNet) introduces Vertical Relative Angle (VRA) as one of the significant factors in the case of a three-dimensional environment. VRA locates a participating vehicle's coordinates more precisely, when its position cannot be calculated based only on distance. Therefore, the increased location precision leads to several forwarding algorithms (e.g., prediction mobility and transmission area efficiency), which are expected to overcome frequent topology changes and re-routing.

The demand for reliable Vehicular-to-vehicular (V2V) communication increases on a daily basis [1]. V2V communication, as part of a Vehicular Ad hoc Network (VANET), offers advantages for safety (e.g., automatic braking system and traffic accident information) and non-safety applications (e.g., business and entertainment). V2V communication becomes interesting due to its flexibility to connect to other participating vehicles without any Road Side Unit (RSU) infrastructure. Additionally, V2V is a promising approach to extend the scalability issue due to high mobility behaviour and complex city environments.

Because of the high mobility behaviour of V2V, frequent disconnection of communication occurs due to overpasses, tunnels, and other obstructions leading to unstable routing issues for transmitted packets. Evaluating this unstable routing can be done by investigating short lifetime connections, low packet delivery ratio (PDR), and end-to-end (e2e) delays that are all highly influenced by the road topology. Therefore, the road topology impacts connection and reconnection establishment between vehicles. The road topology in a two-dimensional scenario, i.e., a road with intersections, and in a three-dimensional scenario, i.e., a road with overpasses, requires special calculations.

Vehicular-to-Vehicular Urban Network

Existing urban environments challenge the connectivity for V2V. Overcoming this challenge, the developed Vehicle-to-Vehicle Urban Network (V2VUNet) concept offers an optimisation for transmission between vehicles by selecting (1) the proper message route and (2) the best relay candidate in the network. Instead of using a flooding mechanism in route path establishment, the area of route request transmission is restricted by measuring the relative angles [2]. The Horizontal Relative Angle (HRA) is applied when the forwarding mechanism occurs in a two-dimensional environment and the Vertical Relative Angle (VRA) is applied in a three-dimensional environment to restrict the area for possible next relay candidates. This area restriction algorithm minimises the required time to build a complete route, which frequently changes, by broadcasting the route request to the closest vehicle (the relay candidate). Figure 1a shows the area restriction algorithms. The resulting number of relay candidates is now limited but still the best one has not been selected. Thus, the predictive forwarding al gorithm shown in Figure 1b is performed. It performs a more precise calculation with additional weight values θ_x and θ_z to select the best relay candidate. As an overall result the V2VUNet concept allows the lowest value HRA and VRA to be determined to find the best relay candidate by restricting the area of candidate