

1.1

AI for Inbound Logistics Optimisation in Automotive Industry

Nikolaos Evangeliou¹, George Stamatis¹, George Bravos¹,
Daniel Plorin² and Dominik Stark²

¹Information Technology for Market Leadership, Greece

²Audi AG, Germany

Abstract

Artificial intelligence (AI) is playing an increasing role in the logistical aspects of a production site in an automotive industry. The pre-calculation of critical situations in the delivery of parts to the supplier network faces increasing disruptions which have an impact on delivery reliability. The planning and control processes are currently implemented by employees and consequently causes a lot of effort and sometimes incorrect decisions which are mostly based on the experiences of employees. The processing and learning AI component will assess the disruption risk caused by natural disasters such as earthquakes, hurricanes or through manmade political or social actions such as strikes and propose countermeasures and assure material availability. Automatic and permanent screening of external sources (newsfeed, weather forecast, traffic situation) determine potential influence of road conditions, natural disasters, strikes etc. on the expected reliability of material replenishment. Finally, the processing and learning component will assess different countermeasures based on a machine learning algorithm, which will be feed with data collected from the sensing component.

Keywords: artificial intelligence (AI), inbound logistics, optimisation, machine learning, real time analytics, data fusion bus, decision support system, scikit-learn.

1.1.1 Introduction and Background

The focus of demonstrator use case in AI4DI is the design and implementation of a Material Planning Decision Support System (MPDSS) that operates in an automotive production site and aims to optimize the complete inbound logistics process. Towards this direction, the work centres around the employment of advanced data-driven methods to collect and consolidate all relevant information and to use it for the identification of critical parts in the supply of AUDI's production lines.

This information refers to AUDI's internal data and information, AUDI's partner data and information (e.g., OEM's supplier's stock levels), public data information (e.g. weather conditions/forecast, road condition), as well as historical decisions and recommendations in similar situations. Finally, the MPDSS evaluates all possible measures for securing part supply via assessing all available data and collecting decisions and recommendations, and autonomously prioritises the applicable measures. Part autonomy is only delivered during decision on any critical part, as the user can always take the final decision of which countermeasure to apply based on given assessment parameters (e.g., cost, efficiency, CO₂ footprint, etc.). While the data collection (from local and publicly available sites) occurs at the edge, decision support offered by the MPDSS occurs at the cloud side. Training and inference of the ML algorithms happens centrally in the cloud. The AI methodology to follow is supervised training, with the main challenges being the learning prediction.

1.1.2 Requirements – User Journeys

The user requirements of the MPDSS will be presented below as short user journeys.

Data Collection and Consolidation: The MPDSS should make use of all available information to identify critical parts, while minimising the necessary actions for the manual collection and consolidation of data.

This is achieved by collecting (i) AUDI's internal data and information; (ii) AUDI's partner data and information; (iii) Public data and information (e.g. weather information, political situations affecting road conditions, etc); and (iv) decisions and recommendations.

Identification of critical parts: The MPDSS should show only those parts that are critical enough to cause a supply bottleneck in the production line.

To achieve this, the system should provide the best possible assessment of criticality by (i) categorising parts and determine critical ones; (ii) prioritising them according to supply capability (how probably it is to obtain this part on time); (iii) visualising critical parts and relevant background information (based on historical data).

Recommendation of measures: The MPDSS should leverage optimisation algorithms to prioritise the different applicable measures for securing part supply and recommend the best-suited measure, taking into consideration certain parameters (e.g., cost, effectiveness, CO₂ footprint).

Autonomous decision making: The MPDSS should autonomously decide which measures are applicable based on given conditions that can be defined by the user either in advance or after the user visualises the suggested countermeasures (partly autonomous decision making). This feature gives a flexible definition of conditions.

Continuous improvement: The user should be able to rate the recommendations given by the MPDSS and this rating should be used to improve the AI routines of the system in the future. This is achieved by comparing user’s decision with MPDSS best-fit recommendation (when part autonomous operation).

1.1.3 Data Flow Principles and Architecture of the MPDSS

In this envisioned MPDSS, the data flow is depicted [1] in the following data flow diagram. A streaming platform collects information from AUDI internal data sources (such as warehouse databases) and external data streams (such as weather APIs, traffic condition APIs etc), and all information is fused

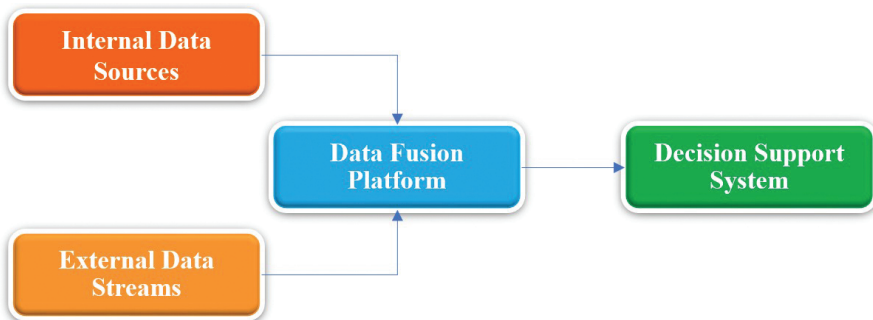


Figure 1.1.1 High level data flow diagram.

dynamically in a single dataset. To account for emergency situations such as traffic conditions or natural disasters, this dataset should be updated and queried continuously, so that decision support and alerting is provided in a real-time manner.

The Data Fusion Bus (DFB) is well-suited to account for the need of providing real time Machine Learning analytics. Brief reference to DFB and its rationale has been made below. DFB enables organizations in developing, deploying, operating, and managing a big data environment with emphasis on real-time applications. It combines the features and capabilities of several big data applications and utilities within a single platform.

The key capabilities of DFB [2] are:

- Real-time monitoring and event-processing, semantic fusion of events not coinciding in time.
- Data aggregation from heterogeneous data sources and data stores.
- Real time analytics, offering ready to use Machine Learning algorithms for classification, clustering, regression, anomaly detection.
- An extendable and highly customizable Interface REST API (and web app) for configuring analytics, manipulation, and filtering. It also includes functionality for managing the platform.

The technical architecture of MPDSS [3] will be a combination of well-known opensource tools and proprietary modules. ITML will leverage its in-house developed Data Fusion Bus, as depicted in Figure 1.1.2 below.

The main building blocks of the architecture are:

- **Support for multiple data streams and data stores:** Readily available interfaces are in place that allow for data acquisition for all well-established Relational Database Management System (RDBMSs), data streams (using MQTT), NoSQL databases, shared filesystems (HDFS Hadoop [4]). This functionality is supported by Kafka [5].
- **Data Fusion Bus**, comprised of the following sub-modules: (i) The Streaming Core of the platform is Apache Kafka. It relies on Kafka 's distributed messaging system to provide high fault-tolerance (Resiliency to node failures and support of automatic recovery) and elasticity - high scalability; (ii) Internal Store and Search Engine: When persistence of data within the platform is required, the Elastic stack (Elasticsearch and Logstash) is utilized. Data may flow either through Kafka connectors (usually in cases of stream data) or may be directly imported to Elasticsearch [6]. Elasticsearch also provides provide high fault-tolerance and scalability; and (iii) Identity management, authentication,

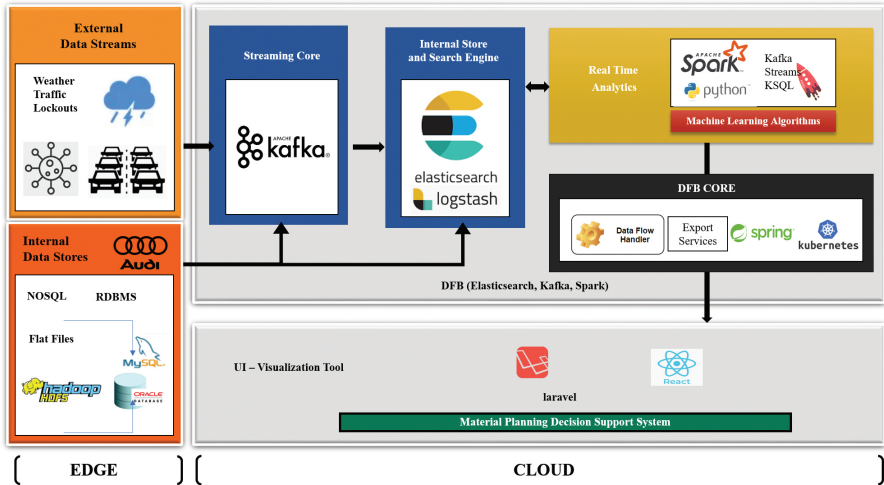


Figure 1.1.2 MPDSS architectural diagram.

authorization and accounting mechanisms that enhance the security of the platform. Moreover, the security mechanism includes dataset encryption and anonymization.

- **DFB Analytics Engine** supports batch processing and stream processing with Apache Spark [7], Kafka Streams & KSQL, Spark Streaming and python scikit-learn [8]. DFB can be used to perform supervised (classification and regression with algorithms such as RandomForest or neural networks) and unsupervised Machine Learning algorithms (e.g Clustering with Kmeans).
- **DFB Core** is responsible for providing business logic and managing all the data flows. It is a custom REST API (based on Java Spring). It exposes a configurable set of web services for providing Decision Support to external systems and managing/monitoring the whole platform.

1.1.4 Preliminary Analysis of Data and Dataset

Advanced data analysis will be applied in a dataset to detect critical parts (using a binary classification algorithm that return “1” when a part is critical and “0” when it is not), then assess and recommend countermeasures again based on calculations from input data, and finally perform decision making and take into consideration the final decision of the user for continuous

improvement. There is also consideration into extending the classification of parts into three classes: non-critical, critical and very critical part.

A preliminary analysis of that dataset to explore possible correlations among the various fields and the suitability of different machine learning algorithms has been performed. While this dataset is considered too small for reliable outcomes, some initial results and the methodology used is presented below.

1.1.4.1 Data Pre-processing and Visualisation

Data Understanding using descriptive statistics: Quantitative summary of raw data received as input using measures of central tendency and measures of variability. This process allows the identification of distinct values for each field and the distribution for the numeric values.

Handling missing values: If missing values are not handled properly an inaccurate inference about data might be drawn. Columns which had no values were removed.

Feature selection: Processing of input variables to select features with optimal contribution to the target variable. Removing redundant data helps in reducing data noise and improves model accuracy. This step is achieved with visualisation tools aiming at the detection of highly correlated variables.

Continuous vs categorical feature detection: Automatically identify which features are categorical and convert original values to category indices. This process improves the efficiency of the machine learning algorithms.

Categorical feature encoding: Transforming categorical variables to numbers by mapping each category to a binary vector denoting the presence or absence of the feature. This process also improves the efficiency of the machine learning algorithms.

1.1.4.2 Classification Models

Four different machine learning algorithms (Multilayer Perceptron Neural Network, Random Forest, Gradient Boosted Tree and Decision Tree) were used on the sample data. In every case 70% of the dataset was used for the training of the algorithm and 30% for testing. The model using the **multilayer perceptron neural network** had the higher accuracy. Other algorithms can be used in the future if needed as well. The results are presented below [9].

PerceptronTrainer

Confusion Matrix

	Non Critical	Critical	
Actual Value 0	91	6	Sum row 97
Actual Value 1	3	75	Sum row 78
	Sum column 94	Sum Column 81	

Figure 1.1.3 Multilayer perceptron neural network results.

RandomForestTrainer

Confusion Matrix

	Non Critical	Critical	
Actual Value 0	88	11	Sum row 99
Actual Value 1	8	63	Sum row 71
	Sum column 96	Sum column 74	

Figure 1.1.4 Random forest results.

Multilayer Perceptron Neural Network achieved an accuracy score of over 90% based on cross-validation results (Figure 1.1.3 and Figure 1.1.4). The confusion matrix that summarizes the proportion of correct vs incorrect classifications is as follows:

GBTrainer

Confusion Matrix

	Non Critical	Critical	
Actual Value 0	81	9	Sum row 90
Actual Value 1	11	59	Sum row 70
	Sum column 92	Sum column 68	

Figure 1.1.5 Gradient boosted tree results.

DecisionTreeTrainer

Confusion Matrix

	Non Critical	Critical	
Actual Value 0	80	17	Sum row 97
Actual Value 1	9	51	Sum row 60
	Sum column 89	Sum column 68	

Figure 1.1.6 Decision tree results.

Gradient Boosted Tree achieved an accuracy score based on cross-validation results (Figure 1.1.5). The confusion matrix that summarizes the proportion of correct vs incorrect classifications is as follows:

Decision Tree achieved an accuracy score based on cross-validation results (Figure 1.1.6). The confusion matrix that summarizes the proportion of correct vs incorrect classifications is as follows:

1.1.5 Conclusion

The inbound supply process in the automotive industry is a complex structure of the availability of required material, calculable risks and unpredictable events which have a direct influence on the entire value added in the production of the vehicles but also on the supporting value creation processes. Dealing with these events and making the right decisions poses major challenges for every automotive manufacturer and supplier, especially since the supply chains in an international network of manufacturing units are very volatile. In this big data environment, artificial intelligence offers excellent technology to make this complexity manageable and to make the right decisions in critical areas. With the MPDSS system and the underlying architecture, the first major progress can be achieved at an early stage of implementation. The greatest challenge here is the integration of the right data with all the requirements described as well as the labelling and integration of human experience-based knowledge for alternative courses of action.

Acknowledgements

This work is conducted under the framework of the ECSEL AI4DI “Artificial Intelligence for Digitising Industry” project. The project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 826060. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Germany, Austria, Czech Republic, Italy, Latvia, Belgium, Lithuania, France, Greece, Finland, Norway.

References

- [1] AI4DI project, Deliverable D2.4: Report on hybrid intelligent system and sub-system level modelling and simulation, Public Report.
- [2] Data Fusion Bus, Available online at: <https://itml.gr/data-fusion-platform>
- [3] AI4DI project, Deliverable D2.3: Report on HW/SW partitioning and sub-system level key architecture designs, Public Report.
- [4] APACHE KAFKA, Available online at: <https://kafka.apache.org/>
- [5] APACHE hadoop, Available online at: <https://hadoop.apache.org/>
- [6] Elastic, Available online at: <https://www.elastic.co/>
- [7] APACHE Spark, Available online at: <https://spark.apache.org/>
- [8] Scikit-learn, Available online at: <https://scikit-learn.org/stable/>
- [9] AI4DI project, Deliverable D2.5: Report on IIoT/AI enabled system level platforms, Public Report.

