Towards a Domain-Specific Framework for Predictive Analytics in Manufacturing

David Lechevalier, Anantha Narayanan, Sudarsan Rachuri

National Institute of Standards and Technology, Gaithersburg, USA
{david.lechevalier,anantha.narayanan,sudarsan.rachuri}@nist.gov

Abstract. Data analytics is proving to be very useful for achieving productivity gains in manufacturing. Predictive analytics (using advanced machine learning) is particularly valuable in manufacturing, as it leads to production improvement with respect to the cost, quantity, quality and sustainability of manufactured products by anticipating changes to the manufacturing system states. Many small and medium manufacturers do not have the infrastructure, technical capability or financial means to take advantage of predictive analytics. A domain-specific language and framework for performing predictive analytics for manufacturing and production frameworks can counter this deficiency. In this paper, we survey some of the applications of predictive analytics in manufacturing and we discuss the challenges that need to be addressed. Then, we propose a core set of abstractions and a domain-specific framework for applying predictive analytics on manufacturing applications. Such a framework will allow manufacturers to take advantage of predictive analytics to improve their production.

Keywords: domain-specific modeling, predictive analytics, machine learning, manufacturing

1 Introduction

Manufacturers today face the unique challenge of improving productivity in industries that are already quite efficient [1]. They look to achieve productivity gains by improving efficiency in design and production, but are pushed to squeeze more and more out of already streamlined processes. One method that promises to deliver significant productivity gains is the application of data analytics. Manufacturing is one of the largest data generators today [2]. The use of data analytics in manufacturing has steadily increased over the past several years [3]; however, we are still far away from taking full advantage of manufacturing data.

Predictive analytics uses statistical techniques, machine learning, and data mining to discover facts in order to make predictions about unknown future events. Some of the applications of predictive analytics for manufacturing data include fault detection and failure prediction, forecasting product demand, cost modeling for product pricing, analytics for predicting warranty and product maintenance, etc. Many small and medium manufacturing enterprises lack the infrastructure and technical know-how to collect, store, process, and analyze their data, and translate them to productivity gains.

In this paper, we propose a domain-specific modeling framework for predictive analytics of manufacturing data. Our goal for this framework is to integrate the tools and techniques for predictive analytics and data visualization with a domain-specific modeling environment that makes problem specification easier for manufacturing domain experts. We first provide a survey of some of the applications of predictive analytics in manufacturing. We then describe the framework, which includes a domain-specific modeling environment and a set of data analytics and visualization components. The goal of this framework is to make powerful analytics and prediction methods more accessible to manufacturing domain experts. We also present some initial thoughts on the core concepts for a meta-model that the domain-specific modeling environment could instantiate.

The paper is organized as follows: Section 2 provides some background material on predictive analytics methods; Section 3 presents some of the applications of predictive analytics techniques in manufacturing; Section 4 discusses our proposed domain-specific framework for performing predictive analytics for manufacturing applications; Section 5 provides some of the core concepts that must be included in the meta-model that defines a domain-specific language for specifying manufacturing application problems for predictive analytics; Section 7 discusses some of the related work; and finally, Section 8 provides a summary and concluding remarks.

2 Background

In this section, we introduce important techniques involved in predictive analytics. *Data mining* [4] is the process of discovering patterns in large data sets. Data mining is a combination of various techniques from machine learning, artificial intelligence and statistics. The process is usually divided into three steps: pre-processing, discovery and validation. Pre-processing is used to clean the data by removing noise or resolving missing data. The discovery step involves structuring the data using various techniques. In this step, the relationships between the data are studied to classify the data, summarize them, and provide a more compact structure. Finally the validation step uses new data sets to verify that the structure is compliant with all potential data sets.

Machine learning is a branch of artificial intelligence. It involves building systems that can learn from data to make inferences and predictions about the future. There are two main classifications of machine learning algorithms: supervised learning algorithms and unsupervised learning algorithms. In supervised learning, the training sets are composed of known input data and response values. The training sets feed the machine learning system that tries to generalize the function involved in mapping new data sets. In unsupervised learning, the output is unknown and the objective is to discover a classification for the data. Machine learning involves various approaches such as Bayesian networks and artificial neural networks. Each approach has its advantages and disadvantages in terms of accuracy and speed. An important concept in machine learning is the cost function, a function indicating the penalty for an incorrect prediction. Machine learning involves optimizing the cost function over the input data in order to determine the model coefficients.

An *artificial neural network* (ANN) [5] is a machine learning approach that mimics the functioning of a nervous system. It is usually represented as a network of nodes called "neurons". A weight is assigned to the connections between the neurons to represent the strength of a connection when the neurons are activated. A neural network is composed of several layers or neurons. The number of layers depends on the best fit for a model of the problem studied. When the neural network is fed with data, propagation algorithms (backward or forward) run to update the connection strength and minimize the cost function. ANNs are used for solving problems that are not easily solved by classical logic programming. They are mainly used in speech recognition or object recognition.

A *Bayesian network* (BN) [6] is a graphical representation of a joint probability distribution over a set of variables. It is a directed acyclic graph where nodes are the variables involved in the problem and edges represent the conditional dependencies between the variables. A probabilistic table is assigned to each node. This table represents the conditional probability of the variable represented by the node for each combination of its parent nodes. Bayesian networks may be constructed by learning from data sets, or by modeling expert domain knowledge. Once the network has been constructed, the probability value of a node can be set to one if the corresponding fact is observed in the scenario. Then, computations on the network will update the probability values of the other nodes depending on the new known fact.

There are other approaches involved in machine learning such as Support Vector Machine [7], linear regression, etc.

3 Applications of Predictive Analytics in Manufacturing

In this section, we survey a number of articles in the literature about the application of predictive analytics in manufacturing. We have grouped them into the major problem areas within manufacturing where predictive analytics may be applied. The goal of this section is to illustrate the usefulness of these methods, and to highlight the effort involved in developing dedicated solutions to specific problems in the domain. In Section 4, we will discuss the idea of generalizing these methods in order to apply them to a wider range of manufacturing related problems.

3.1 Applications in Manufacturing System Control

Predictive analytics techniques have been applied to improve manufacturing system control. Several techniques can be applied such as statistical techniques or machine learning techniques to better regulate the manufacturing system by controlling current and future states of the system. In [8], Bayesian networks have been used to identify the major process variables that have an impact on the rolling process and lead to defects. Rolling is a deformation process that reduces the thickness or changes the cross-section of a long work piece. The authors studied seam defects, which are cracks aligned parallel to the metal surface. They identified twelve process variables involved in this defect and built a causal network that establishes the relationships between these variables. Finally they identified the variables that are likely to cause the defect, by training the causal network with records collected from 100 000 rolled bars. The study suggests that monitoring these variables carefully can increase the efficiency of the system.

Other investigations on process control have been done, such as [9] where the authors have developed a tool to run diagnosis and prognosis against source terms (the amount of radio-active material released in an accident) in nuclear power plants. The tool uses the NETICA [10] application. NETICA is a software application that enables users to build Bayesian networks, to assign the probabilities associated with the variables, and run algorithms for performing Bayesian inference.

3.2 Applications in Manufacturing Quality Control

Predictive techniques have also been used to address challenges in product quality. In [11], a Bayesian network approach has been developed to predict and avoid defects in castings called “micro-shrinkage” or “secondary contraction”. Micro-shrinkage is a result of tiny pores that develop as the casting cools. The predictive technique is composed of five steps. First, relevant variables are identified and divided into those that are metal-related and those that are mold-related. Metal-related variables are subdivided into composition variables, nucleation potential and melt quality variables, and pouring variables. Mold-related variables are subdivided into sand variables and molding variables. In addition, the authors add several variables to control the dimension and the geometry of the casting and the configuration of the machines as well. In total, about fifty variables are defined. Next, the model is trained with data collected over a year. Once these two steps have been done, the authors start their third step, which involves structural learning to refine their model. An algorithm called PC-Algorithm described in [12] is used to discover the causal structure. Some other algorithms to discover causal structure are also described in [12]. The fourth step consists in parametrical learning that leads to recalculating the probability based on new data and the structure defined from the third step. At this stage, 60 % of the variables can be eliminated without reducing the network accuracy. Finally, the inference capability of the Bayesian network helps to calculate the values of the variables. In addition to the Bayesian network, the authors developed a module that records the different variables’ values to trace the impact of each value on the appearance of micro-shrinkage. Related works on predicting defects for quality control in manufacturing have been done such as predicting ferrite number in austenitic stainless steel welds with Bayesian Neural Networks[1] [13] or predicting surface roughness using regression and neural networks [14].

In [15], the authors introduce an automated data mining system for quality control. Knowledge discovery in database (KDD) techniques are used to extract characteristics of a production process that are not directly accessible by reading the data. The KDD process is composed of two steps called “discovery” and “verification”. The data mining system is created to automatically execute the two KDD steps and to allow users to

---

[1] Bayesian Neural Network applies artificial neural network techniques in a Bayesian framework

find the cause of production process problems. First, a preprocessing step transforms the data and categorizes them. Then to discover the data and extract the rules that represent the relationships between the data, the authors run an algorithm called CHRIS [16]. This algorithm extracts rules in the form of “if A then B”. The algorithm looks for the feature A that is the most present in the instance that populates B. This first step is executed without human intervention. By using a process of rules ranking, the authors are able to quickly find the cause of a production problem that has been recorded previously in the factory. The data mining system extracts the rules related to the fault and these rules get a higher rank close to the time that the fault occurred.

3.3 Applications in Manufacturing in Fault Diagnosis

Identifying faults early and preventing faults from happening can provide significant savings for manufacturing enterprises. Predictive techniques have been used for fault diagnosis in manufacturing applications. In [17], the authors present a system called “Wisdom”. This system has been developed to enhance fault diagnosis on a Base Transceiver Station (BTS) in a Motorola factory. The system makes a probabilistic diagnosis of the cause of failure and suggests remedies. The authors use a Bayesian network and identify the variables involved in the system such as the state of a cable used to run a fan test or whether the hardware is in service. The fan test is a test to ensure the alarm monitor will generate a correct message under normal working conditions. They conduct interviews in the factory to extract expert knowledge from factory employees. Based on these interviews, they define the procedure of the fan test that is the subject of the diagnosis in their example. Each step of the procedure is a node in the Bayesian network. Probabilities of failure are assigned to each node depending on the collected data in the factory and the previous fault diagnoses. The authors use the inference capabilities to infer the probability of each node when a failure occurs. The system also provides a visualization of the results by using an intelligent user interface (IUI). When a test fails, the IUI loads the Bayesian network built from the collected data and looks for the node with the highest probability of failure. Based on these results, the IUI displays advice to fix the problem. The Wisdom system runs tests on fans, and on test equipment, system, module, interconnection, test cable, antenna matrix, transceiver, alarm module, software, test solution and power supply. If a test fails, the system suggests advice until the test passes. Once the test passes, the system starts testing the next fault type. Results show that the system decreases the time to correct the fault, especially complex faults.

In [18], the authors develop a system based on artificial neural networks for fault diagnosis of power transformers. The transformer station under study is composed of two sides (primary and secondary) that can lead to a fault. The system is designed to detect faults, estimating the faulty side, classifying the fault type and identifying the faulted phase. The system consists of three hierarchical levels. The first level is a preprocessing level to clean the data. The second level is an ANN to detect the faulted side. The third level is composed of two parallel side diagnosis systems (SDS). Each SDS is assigned to a side and is used to detect the fault type and the faulted phase. Each SDS is composed of one ANN in series with four ANNs in parallel. Each ANN has a specific task to do to identify the fault type or the faulted phase. First, the system cleans the data in its first level. Once it is done, the ANN at the second level defines whether the situation is normal, failed on the primary side, or failed on the secondary side. To do so, the ANN has three levels of output that are low for the normal situation, medium for a fault on the secondary side, and high for a fault on the primary side. In the case of fault detection, the third level and the SDS in charge of the faulted side runs the evaluation to define the fault type and the faulted phase. The first ANN of the SDS is in charge of the fault type, and the four parallel ANNs are in charge of finding the faulted phase. This system does not work on real time data. To train their ANNs and define the ANNs structure, they use generated data from the electromagnetic transient program (EMTP). Based on case studies that the authors define, they build their ANNs structure with a number of nodes defined using a training set and observing the results for different structures. They keep the structure that works with the best results for the training sets that they submit to the ANNs. This work is developed in MATLAB using its Neural Network Toolbox [19] to design the ANNs. Observed results prove that the system is fast and quite accurate.

3.4 Applications in Manufacturing Maintenance

Maintenance is a critical area for manufacturing enterprises. Enterprises can take advantage of predictive modeling to plan for maintenance and achieve significant cost savings during maintenance. In [20], the authors define a Bayesian network to predict machine maintenance needs. The equipment has two condition monitoring values called CM1 and CM2, and can have six different states (good, wear 1, wear 2, wear 3, failure mode 1 and failure mode 2) which represent the "True Condition" of the equipment. CM1 gives the True Condition while CM2 gives an indication of the machine vibration, which can be low, medium, or high. The load of the machine leads to a change of the "True Condition" of the machine. This change depends on the current "True Condition" and on whether the load is normal or abnormal. The authors define a Bayesian network to model the machine condition and the transition probabilities of the condition modification. To simulate the machine states, they use a software called GeNIe [21] that enables them to construct a dynamic Bayesian network (DBN) [22], which consists of multiple copies over time of a static Bayesian network (the DBN expands the Bayesian network to reflect temporal changes to system variables). They implement two different scenarios. In the first one, the machine does not get any maintenance. They observe that the probability that the machine is in failure condition increases quickly and goes above 0.5 after 277 iterations. In the second scenario, they permit maintenance based on the two indicators CM1 and CM2. Maintenance can be *reset* that makes the machine return to its previous wear state, or maintenance can be *replace* that makes the machine return to good condition. A policy is defined for the maintenance. As an example, if CM1 is "Good" or "Wear 1" and CM2 reports "High Vibration", then the equipment is "Reset". They observe that the system maintains a steady state where the probabilities of being in failed state are very low. Future work could lead to modeling the maintenance as the decision taken depending on the probability of the other characteristics involved in the Bayesian network.

Artificial Neural Networks have also been applied on manufacturing maintenance, as in [23]. The authors develop a system to support predictive maintenance of rotating equipment. The system is composed of three components: a degradation database, an artificial neural network model, and cost matrix and probabilistic replacement model. The objective of this system is to optimize the expected cost of the maintenance per unit time. The degradation database is built from collected data during tests from the point of installation until bearing failure (bearing failure is considered as a machine failure in this study). They observe two phases in the degradation data. Phase I represents operation with no defect. Phase II represents operations with defects that lead to failure state at a failure time. The authors divide their database into two parts: training set and data set. An ANN is modeled to predict the life percentage of each bearing. To train their ANN, they use the training set part of their database. Next, they validate the model by using the validation data set part of their database. Based on the ANN and the real-time data collected from monitoring thirteen bearings, the authors develop a residual life distribution. The last component is the cost matrix, developed by using the residual life distribution and defining two types of cost. The first type of cost is the corrective maintenance cost after a failure occurs. The second type of cost is the planned replacement cost before failure. Based on this matrix and the data from monitoring, the system can suggest the machine maintenance with the minimum cost. If the value of planned replacement is too high compared to the cost of the corrective maintenance, the strategy would be to wait for failure. Alternatively, support vector machine (SVM) can also help in this area as shown in [24]. In this work, the authors develop an intelligent tool breakage detection system to recognize process abnormalities during a manufacturing process, specifically in a milling process. The system can initiate corrections on the process to fix the detected problem.

4 Predictive Analytics Framework for Manufacturing

In the previous section, we surveyed a number of papers on the use of machine learning and predictive analytics for manufacturing applications. Each of these applications has been addressed with a point solution, with specific models built using specific tools. We believe that the underlying techniques are very powerful, and are very valuable if they can be generalized to be applicable to a wide range of manufacturing related problems. In this section, we propose an idea for a framework towards achieving this generalization. Our vision for this framework is illustrated in **Fig. 1**. The intended functions, requirements, and interrelationships of the components of this framework are described below.

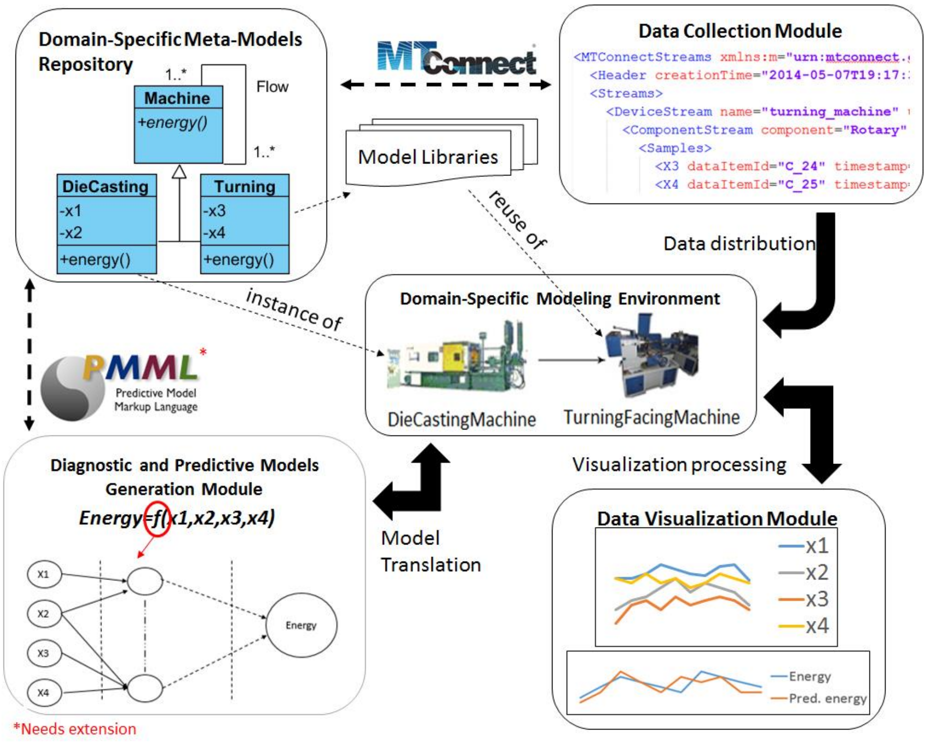

Fig. 1. Overview of the domain-specific framework for predictive analytics in manufacturing

4.1 Domain-Specific Modeling Environment

The goal of this framework is to make it easier for manufacturing domain experts to build models and run diagnostic and predictive analytics. With the framework, these tasks should not require extensive knowledge of machine learning techniques. We believe that this can be addressed through an intuitive interface that allows manufacturers to specify their production systems, and automatically generate the necessary analytical models from the system specification. This interface is the central module of our framework, the Domain-Specific Modeling Environment (DSME). It should provide tools to model problems that the framework users want to solve. An intuitive graphical domain-specific language (DSL) should be available to allow users to easily design manufacturing system specifications. The DSL should allow users to instantiate concepts from the meta-models or reuse existing models from existing libraries. From this system description, it must be possible to automatically generate the predictive models and execute algorithms necessary to make predictions. The components of the domain-specific language must be easy to understand and to use by manufacturing domain experts.

4.2 Meta-Model Repository

The meta-model repository consists of meta-models that define the various concepts that will be used to specify various aspects of the manufacturing analytics problem. This will include abstractions to represent various manufacturing scenarios such as machine and process abstractions. Abstractions for failures, maintenance states, or any characteristics that the users would like to predict also need to be represented. The meta-model must also incorporate abstractions to represent the data formats and operations that the framework must support. These abstractions will be discussed in Section 5.

4.3 Data Collection Module

The data collection module contains the tools needed to collect and pre-process data for performing predictive analytics. As we have observed during our survey, data in manufacturing can come from many different sources and have differing formats. This data can be archived data collected from previous operations, simulated data (if the real data are not available), or real time data collected from machine monitoring. The data module should be able to collect the data from a wide variety of sources and formats. Moreover the data can be collected from a data stream and the module must support the tools needed to handle data flows of high volume and velocity. Data collected from sensors in manufacturing operations are generally structured data. However, their structure is not always usable for processing with the available technologies. The data collection module must be able to understand different data formats, such as MTConnect [25], to pre-process and transform the data into a workable form, to make it understandable by the other modules.

4.4 Diagnostic and Predictive Model Generation Module

The most important functionality provided by this framework is the automatic generation of a predictive analytics model from the system specification created using the DSME. The diagnostic and predictive model to be defined depends on the variables that the user wants to observe and predict, and the characteristics of the systems being defined. Manufacturers should be able to define their objectives by using abstractions of the meta-model. The meta-model must also provide abstractions for analytical models to support automatic generation of the predictive model from the system specification. Standards such as Predictive Model Markup Language [26] could help to achieve this. In addition, data are needed to make the model accurate through training and validation. There must be a communication between the diagnostic and predictive model generation module and the data collection module to support training and validation.

4.5 Data Visualization Module

The data visualization module will enable framework users to understand the results of the study. The visualization module needs to display the collected data and the information inferred from the data. The visualization module must closely interact with the

DSME to provide the appropriate data visualization for corresponding elements in the system specification model. In addition, users need to understand trends and other information that they cannot extract by reading the data. The visualization module will need to transform the results from the diagnostic and predictive model generation module into understandable information for the system users. This could be done by defining rules to classify the results of the diagnostic and predictive model, for example by using color codes to describe different fault criticality levels.

5 Building a Meta-model for Manufacturing Analytics

Building a meta-model to provide the high-level abstract concepts that will encompass the wide range of objects involved in manufacturing systems is a challenge. To simplify the problem in the manufacturing context, we discuss the concepts by dividing them into three main categories: objects, flows and metrics. Objects represent the machines or mechanical processes involved in the system. Flows are the resources that are going through the systems such as energy or material. Metrics are the data and indicators that allow manufacturers to evaluate their systems. The meta-models need to abstract these different concepts while managing the information that they represent.

5.1 Objects

One of the main components of a discrete manufacturing system is the machine tool. In our meta-model, the machine tool needs to be represented in a way that allows users to model the machine, to interconnect several machines, and to model the parameters of the machine. In addition, many machine tools can run a variety of different processes. Classifications such as [27] show that classification will differ based on the criteria used to classify. The meta-model must allow handling the different characteristics and parameters of a process. The meta-model must provide abstractions to represent processes as they appear in the manufacturing scenario, and also as they are represented in the predictive model. There must be a mapping between these two aspects of the process, to support the generation of predictive models from the system specification. To allow predictive modeling for identifying faulty products or evaluating product quality, the meta-model must provide abstractions that represent fault characteristics and failures. In addition, product characteristics such as design, geometry or feature should also be representable. The meta-model must provide relationships to allow the specification of any combination of these features as a prediction objective for the predictive model.

5.2 Flows

Material, information and energy are the main flows in the manufacturing system. Representation of material properties [28] is a crucial aspect of specifying a manufacturing system. The meta-model must support the specification and management of a wide range of material properties and material interactions. In addition, these properties evolve through the manufacturing system. For instance, a material can take different shapes depending on the process that uses this material. Energy used by the processes or the machines can have many forms. The meta-model must support monitoring the energy input to the system, and the energy lost due to heat and waste. The meta-model should provide an easy way for users to model and follow the evolution of the materials and the energy in the system.

5.3 Metrics

By metrics, we refer to the data and indicators associated with machines and manufacturing processes that allow manufacturers to evaluate their production systems. Machine tools are data producers. The kind of data generated depends on the machine and the sensors that collect the data, and the data may be collected in differing formats. An emerging standard to facilitate the exchange of data between shop floor equipment and software applications used for monitoring and data analysis is the MTConnect [29] standard. Supervisory systems have been developed for manufacturing systems by using MTConnect that allow communications through an entire system. The meta-model must understand these kinds of data formats. The meta-model must also support other abstractions to represent data formats from machines that do not support MTConnect. Performance metrics [30, 31] allow manufacturing experts to evaluate their processes or their systems. The meta-model must support the representation of these metrics to allow the evaluation of the system. Finally, the meta-model must provide abstractions to represent the maintenance of the machine in order to enable maintenance planning from the predictive model.

6 Example: Domain-Specific Model for a Production System

In this section, we will introduce a simple example to highlight the potential capabilities of the described domain-specific framework. Let's suppose that a manufacturer wants to model a factory production chain as shown in **Fig. 2**. The production chain consists of a series of machines. Parts and raw materials come in from three different flows (starting from the green points on the left) and merge into a fastening machine. In this production chain, parts from the die casting machine are distributed into three different turning machines running in parallel (the machines may have different performance parameters, and therefore different energy use and throughput). A possible requirement for the manufacturer would be to optimize energy or time by controlling how the parts are distributed among three turning machines. Our goal is to provide a DSME that will allow manufacturers to specify systems like these, and perform analyses to predict metrics such as energy use and throughput.

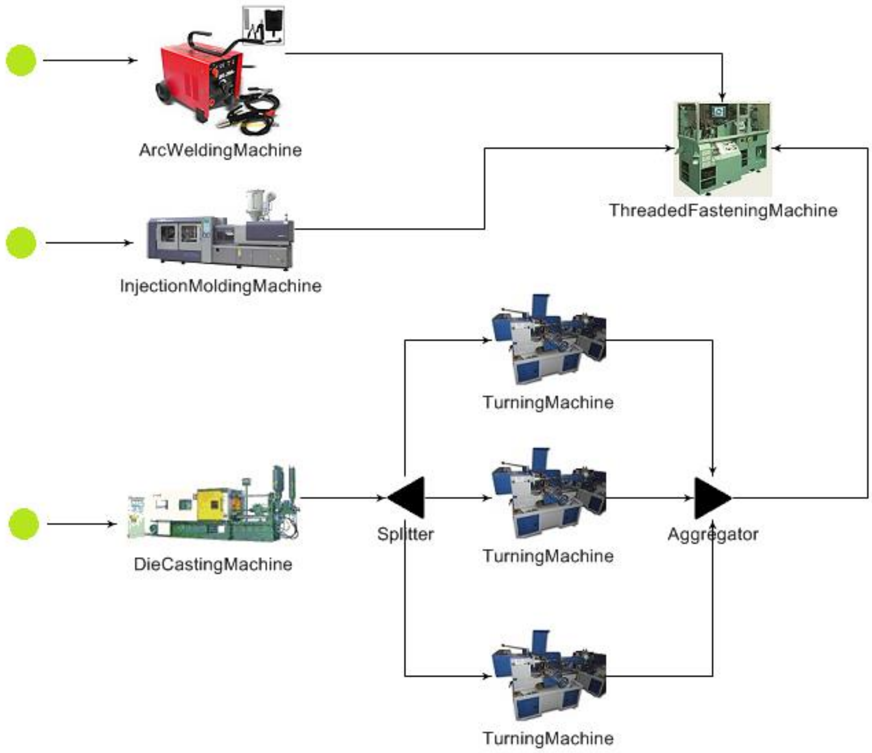

Fig. 2. Example of a production chain

In the example, the manufacturer needs to represent the machines that are involved in the production plan. A simple example meta-model that could be built by domain experts is shown in **Fig. 3**. For this example, we used the Generic Modeling Environment tool [32]. In this meta-model, we create the meta-concepts that allow the manufacturer to represent the production plan. We modeled concepts to represent the machines described in the scenario above. The meta-model could be extended if a manufacturer wishes to add a sub-concept for a new type of machine or take advantage of existing classification as mentioned in [27]. One of the challenges in the design of a meta-model for the domain-specific environment for manufacturing will be leveraging existing classification schemes in the literature, while managing the large scale of machines and processes that need to be defined. In addition to creating meta-concepts, developing a meta-model also enables us to define rules for interpreting the models.

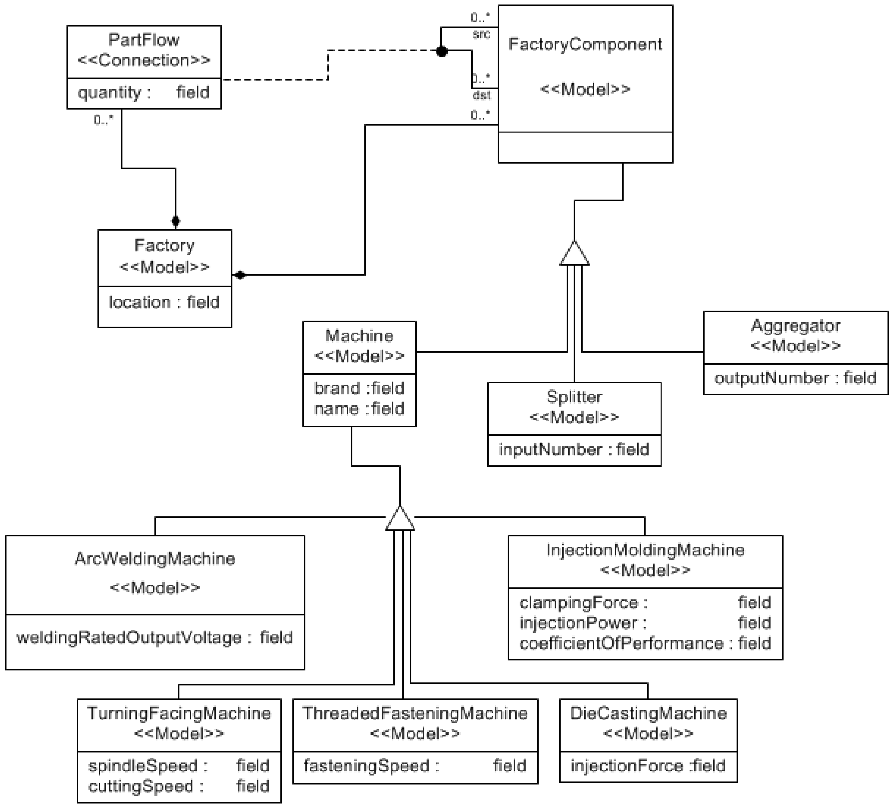

Fig. 3. Example meta-model for manufacturing production

Manufacturers can use the meta-model through the domain-specific environment to model the production plan. Libraries that partially instantiate the meta-concepts can be defined for the DSME. These libraries would define commonly used objects in the domain. **Fig. 4** shows an example of a machine library. A production plan designed in the DSME is the basis for analysis. For example, we can compute the energy consumed in the system from attribute values of the machines and the processes in the model, or generate an optimization model for optimizing energy use or throughput. In our example, if the manufacturer wants to know how to split the flow between the three turning machines to improve throughput, the model must be transformed to a model that an optimization software will be able to understand. The optimization software would then compute the best way to split the flow to optimize the throughput, and this information can be used to reconfigure the model in the DSME. An example of generating an optimization model is in [33], where the authors introduce the Sustainable Process Analytical Formalism (SPAF) for modeling process component, flow and metrics for sustainability related optimization. SPAF has been applied to a use case of energy optimization [34] where SPAF models have been translated to mathematical models that an optimization software can manage.

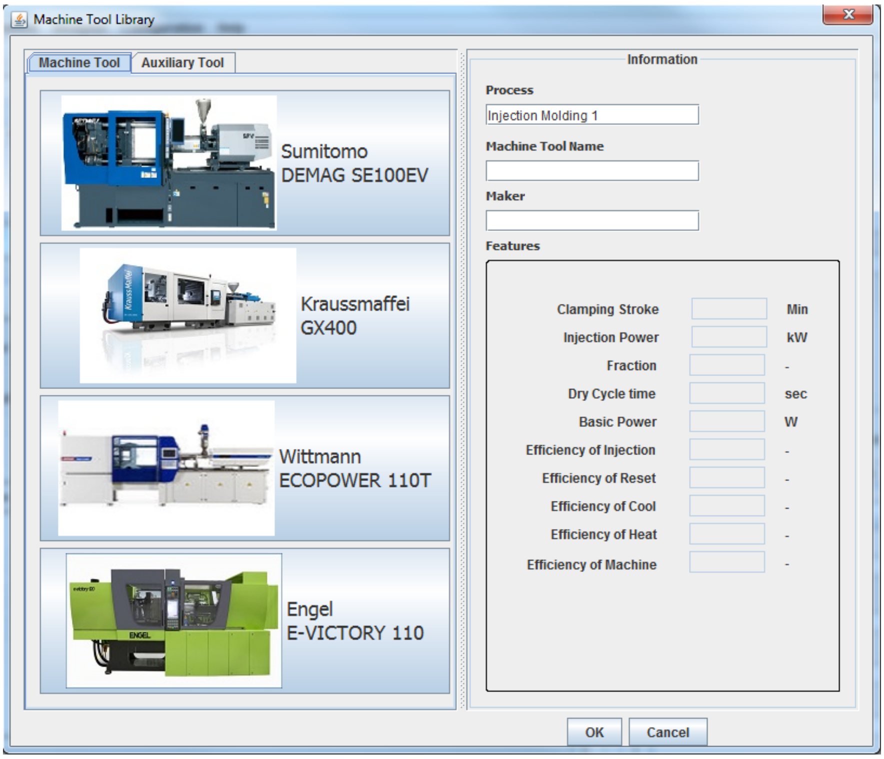

Fig. 4. Example of machine library

7 Related work

Model representations for discrete manufacturing have generally focused on two aspects: representing shapes and features of manufactured products, and representing manufacturing capabilities of enterprises. In [35], a Core Product Model (CPM) is presented as an open, generic and extendable product model. CPM captures product information using key concepts called Artifact and Feature. The Open Assembly Model [36] was developed as an extension to CPM to represent assembly operations. The STEP series of standards [37] provide more granular representations to model various aspects of manufactured products, especially product geometry and tolerances. [38] presents a manufacturing information model to support design for manufacturing in virtual enterprises. It uses the EXPRESS language which is also defined in ISO 10303 [37], and provides information models to facilitate design for manufacturing.

Manufacturing capability models are used to represent the manufacturing capability of an enterprise. They typically provide concepts to represent manufacturing resources such as machine tools, and the types of manufacturing processes supported by the enterprise. In [39], the authors provide a model representation for a flexible manufacturing facility. They present a four-level model to represent the functionality of a manufacturing facility, which involves modeling at the factory, shop floor, cell, and station level. [40] presents an ontology based manufacturing service capability (MSC) by analyzing several use cases and supplier capability descriptions. MSC models are used to model product and process requirements between OEMs (Original Equipment Manufacturers) and suppliers.

Apart from the above models, we also discussed classification systems for manufacturing processes and resources in the previous sections. We are not aware of any work on generic model representations that relate manufacturing operations to predictive analytical models. With the increased capabilities of data processing and machine learning techniques available today, we believe that it is an important area for model development.

8 Summary

Predictive analytics is a very valuable tool for improving productivity in a wide range of manufacturing applications. In this paper, we surveyed a number of uses of predictive analytics for various manufacturing scenarios, to highlight the importance and value of these techniques. The papers surveyed generally offer point solutions applying specific predictive modeling techniques for specific application scenarios. In this paper, we proposed a framework to make predictive analytics techniques available for a generic range of manufacturing problems. We discussed the main components of the framework, and provided some initial thoughts on their development. We also provided some initial thoughts on the meta-model for specifying the DSME.

This paper only provides an overview and initial thoughts on the structure and capabilities of the predictive analytics framework for manufacturing. Our future work will involve a detailed study of the abstractions necessary to specify manufacturing systems for the purposes of predictive analytics. We are also studying ways to generate predictive models such as Bayesian networks automatically or semi-automatically from the system description models specified in the DSME. We are also investigating tools for creating and managing domain-specific models. We believe that this framework will eventually have a great impact on the manufacturing industry, as it will make the power of predictive analytics available to a larger number of manufacturers, and help them achieve greater efficiency in their operations.

Disclaimer

Certain commercial products may have been identified in this paper. These products were used only for demonstration purposes. This use does not imply approval or endorsement by NIST, nor does it imply that these products are necessarily the best for the purpose.

Acknowledgment

The work represented here was partially funded through a cooperative agreement #70NANB13H159 between the University of Maryland and NIST.

References

1. Manyika, James, et al. "Big data: The next frontier for innovation, competition and productivity." *Technical report*, McKinsey Global Institute, 2011.
2. Young, M., and Pollard, D. What businesses can learn from big data and high performance analytics in the manufacturing industry" Big Data Insight Group, 2012.
3. Harding, J. A., A. Kusiak, and M. Shahbaz. "Data mining in manufacturing: a review." *Journal of Manufacturing Science and Engineering* 128.4 (2006): 969-976.
4. Han, Jiawei, Micheline Kamber, and Jian Pei. *Data mining: concepts and techniques*. Morgan Kaufmann, 2006.
5. Haykin, Simon, and Neural Network. "A comprehensive foundation." *Neural Networks* 2.2004 (2004).
6. Pearl, Judea. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.
7. Steinwart, Ingo, and Andreas Christmann. *Support vector machines*. Springer, 2008.
8. Li, Jing, and Jianjun Shi. "Knowledge discovery from observational data for process control using causal Bayesian networks." *IIE Transactions* 39.6 (2007): 681-690.
9. Grindon, E., and C. G. Kinniburgh. "SPRINT: a tool for probabilistic source term prediction for use with decision support systems." *Radiation protection dosimetry* (2004): 35-39.
10. Netica application, Norsys Software Corporation. Retrieved July 3, 2014 from http://www.norsys.com/netica.html
11. Penya, Yoseba K., P. Garcıa Bringas, and Argoitz Zabala. "Advanced fault prediction in high-precision foundry production." *Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference on*. IEEE, 2008.
12. Spirtes, Peter, Clark N. Glymour, and Richard Scheines. *Causation, prediction, and search*. Vol. 81. MIT press, 2000.
13. Vasudevan, M., M. Murugananth, and A. K. Bhaduri. "Application of bayesian neural network for modelling and prediction of ferrite number in austenitic stainless steel welds." *BOOK-INSTITUTE OF MATERIALS* 784 (2002): 1079-1100.
14. Özel, Tuğrul, and Yiğit Karpat. "Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks." *International Journal of Machine Tools and Manufacture* 45.4 (2005): 467-479.
15. Maki, Hideyuki, and Yuko Teranishi. "Development of automated data mining system for quality control in manufacturing." *Data Warehousing and Knowledge Discovery*. Springer Berlin Heidelberg, 2001. 93-100.
16. Maeda, Akira, Hideyuki Maki, and Hiroyuki Akimori. "Characteristic rule induction algorithm for data mining." *Research and Development in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 1998. 399-400.
17. Chan, A., and K. R. McNaught. "Using Bayesian networks to improve fault diagnosis during manufacturing tests of mobile telephone infrastructure."*Journal of the Operational Research Society* 59.4 (2008): 423-430.
18. Mohamed, E. A., A. Y. Abdelaziz, and A. S. Mostafa. "A neural network-based scheme for fault diagnosis of power transformers." *Electric Power Systems Research* 75.1(2005):29-39.
19. Neural Network Toolbox, MatWorks Retrieved July 3, 2014 from http://www.mathworks.com/products/neural-network/
20. McNaught, K. R., and A. Zagorecki. "Using dynamic Bayesian networks for prognostic modelling to inform maintenance decision making." *Industrial Engineering and Engineering Management, 2009. IEEM 2009. IEEE International Conference on*. IEEE, 2009.

21. GeNIe. Decision Systems Laboratory, University of Pittsburgh. Retrieved July 3, 2014 from http://genie.sis.pitt.edu/
22. Murphy, Kevin Patrick. *Dynamic bayesian networks: representation, inference and learning*. Diss. University of California, Berkeley, 2002.
23. Wu, Sze-jung, et al. "A neural network integrated decision support system for condition-based optimal predictive maintenance policy." *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 37.2 (2007): 226-236.
24. Cho, Sohyung, et al. "Tool breakage detection using support vector machine learning in a milling process." *Intl. Journal of Machine Tools and Manufacture* 45.3 (2005): 241-249.
25. Vijayaraghavan, A., et al. "Improving Machine Tool Interoperability Using Standardized Interface Protocols: MTConnect." *International Symposium on Flexible Automation*, 2008.
26. Wettschereck, Dietrich, and Stefan Muller. "Exchanging data mining models with the predictive modelling markup language." *International Workshop on Integration and Collaboration Aspects of Data Mining, Decision Support and Meta-Learning*. 2001.
27. Todd, Robert H., Dell K. Allen, and Leo Alting. *Manufacturing processes reference guide*. Industrial Press Inc., 1994.
28. Ashby, Michael F., and Kara Johnson. *Materials and design: the art and science of material selection in product design.* Butterworth-Heinemann, 2013.
29. MTConnect. "Part 1-Overview and protocol, Version 1.2.0" *MTConnect Institute* (2014).
30. K.K.B. Hon, "Performance and Evaluation of Manufacturing Systems" *CIRP Annals - Manufacturing Technology*, Volume 54, Issue 2, 2005, Pages 139-154, ISSN 0007-8506
31. A. Gunasekaran, C. Patel, E. Tirtiroglu, (2001) "Performance measures and metrics in a supply chain environment." *IJOPM*, Vol. 21 Iss: 1/2, pp.71 – 87
32. Ledeczi, Akos, et al. "The generic modeling environment." *Workshop on Intelligent Signal Processing*, Budapest, Hungary. Vol. 17. 2001.
33. Brodsky, A., Guodong, S., and Riddick, F. "Process analytics formalism for decision guidance in sustainable manufacturing." *Journal of Intelligent Manufacturing* (2013): 1-20.
34. Kim, Duck Bong, et al. "Sustainable Process Analytics Formalism: A Case Study of Book Binding System for Energy Optimization." *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2013.
35. Fenves S, et al. "CPM2: a revised core product model for representing design information." *National Institute of Standards and Technology, NISTIR 7185*. 2004.
36. Sudarsan R, et al. "A model for capturing product assembly information." *Journal of Computing and Information Science in Engineering* 2006;6(1):11–21.
37. International Organization for Standardization. "ISO 10303-11: 1994. Industrial automation systems and integration – product data representation and exchange – part 1: overview and fundamental principles." 1994.
38. Giachetty, R. E., "A standard manufacturing information model to support design for manufacturing in virtual enterprises." *Journal of Intelligent Manufacturing*, March 1999, Volume 10, Issue 1, pp 49-60.
39. A Molina and R Bell, "A Manufacturing Model Representation of a Flexible Manufacturing Facility." *Proceedings of the Institution of Mechanical Engineers, Part B: journal of Engineering Manufacture*, 1999, 213:225
40. Jun H. Shin; Boonserm Kulvatunyou; Yunsu Lee; Nenad Ivezic; "Concept Analysis to Enrich Manufacturing Service Capability Models." *Conference on Systems Engineering Research*, March 2013, Atlanta, GA.