Improving Machine Learning Performance Using Conceptual Modeling

Arturo Castellanos[a], Alfred Castillo[b], Monica Chiarini Tremblay[c], Roman Lukyanenko[d], Jeffrey Parsons[e], and Veda C. Storey[f] *

[a] *Baruch College (CUNY), New York, NY, USA*
[b] *Cal Poly, San Luis Obispo, CA, USA*
[c] *William & Mary, Williamsburg, VA, USA*
[d] *HEC Montréal, Montreal, Québec, Canada*
[e] *Memorial University, Newfoundland, Canada*
[f] *Georgia State University, Atlanta, GA, USA*
*authors listed in alphabetical order—contributed equally.

Abstract
Advances in machine learning (ML) make it possible to extract useful information from large and diverse datasets. ML methods aim to identify patterns in a dataset based on the values of features and their combinations. Recent research has proposed combining conceptual modeling, specifically data models, with artificial intelligence. In this paper, we employ conceptual modeling principles to develop a method for data preparation, which is comprised of six guidelines. We illustrate the method by applying it to a business case from a foster care organization; namely, predicting the length of stay of a child in the foster care system. The results show how conceptual modeling can improve ML model performance by imbuing explicit domain knowledge, instead of relying solely on data-driven rules.
Keywords:
Artificial Intelligence, Machine Learning, Conceptual Modeling, Model Performance

1. Introduction

Although artificial intelligence research has traditionally focused on logic-based, model-driven learning, abstraction, and inference methods [1], the ubiquity of large-scale, heterogenous data and computing power has shifted the focus towards machine learning (ML) [2]. Machine learning consists of methods that use data and algorithms to build models that make inferences from the provided data examples [3]. Both the opportunities and limitations of machine learning are rooted in its reliance on building models from data and, therefore, on the quality of the data used to train and test these models [4]. As reliance on machine learning grows, it is crucial to ensure these models exhibit good performance and are interpretable and transparent. This tradeoff is often augmented by opaque transformations in the input data (i.e., feature engineering), which makes it challenging to assess the effectiveness of the input data on the outcome [5]. The objective of this research is to improve model performance in supervised machine learning in a repeatable and transparent manner by using domain knowledge represented in conceptual models. The contribution is a set of guidelines.

2. Machine Learning Performance and Conceptual Modeling

A learning machine is a computer program that can improve its performance with experience for some class of tasks and performance measures [5]. For supervised ML, performance is the ability of a ML model to "reproduce known knowledge" [6]. Supervised learning guides the learner in acquiring knowledge in a domain through examples, so new cases can be handled in a manner most appropriate

In A. Martin, K. Hinkelmann, H.-G. Fill, A. Gerber, D. Lenat, R. Stolle, F. van Harmelen (Eds.), Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021) - Stanford University, Palo Alto, California, USA, March 22-24, 2021.

based on the knowledge learned from similar cases. Performance can be assessed as the accuracy of a program predicting values of interest for new cases.

Machine learning performance can be improved using complex methods (e.g., deep learning neural networks) [7], improving the quality of data used to train and test algorithms [8], transforming training data into a form more amenable to learning [8], and/or increasing the quantity of the training data [9]. There are also efforts to augment data-driven ML processes with domain knowledge [10].

Conceptual modeling formally describes "some aspects of the physical and social world around us for the purposes of understanding and communication" [11, p. 2]. Conceptual data models are widely used to represent data requirements commonly conceptualizing a domain in terms of entities that belong to entity types, which possess attributes, and participate in relationships with other entities [12].

Recent research has proposed combining conceptual modeling with artificial intelligence [13] [14] [15] [16] [17]. Conceptual modeling can affect ML model performance in several ways. ML methods identify patterns in a dataset based on the values of features and their combinations. With enough data, they might be able to extract the knowledge expressed in a conceptual model; however, there is often insufficient data to do so. Conceptual models represent real world domain knowledge for different uses [12] and, thereby, can augment a purely data-driven approach to ML with a knowledge-driven one [10]. This can provide reliable rules about the domain without depending on extracting them from the data. Moreover, a conceptual model explicitly represents domain knowledge to select and prepare a dataset before using it in ML. Thus, we propose that conceptual model-driven guidelines can improve ML performance by using explicit domain knowledge to prepare and filter data before input to ML algorithms, instead of relying solely on algorithms to find such rules in the data.

2.1. Method

The method is based on the three main constructs of the Extended Entity-Relationship (EER) model: entities, relationships, and attributes (see Table 1). We assume a conceptual model of the domain is available in the form of an EER diagram. The method iterates over the constructs of the EER model to preserve the domain knowledge expressed in the EER diagram during transformation of the dataset. Applying the method results in changes to the input dataset to create a modified dataset, augmented with domain knowledge from the EER diagram.

Table 1. Constructs of EER for Machine Learning Guidelines

Element	Definition	Example
Entity (G1 & G2)	Class or category of entities (e.g., people, places, events)	CHILD, PLACEMENT, CASE
Relationship (G3 & G4)	Association between entities (e.g., buys, writes, owns) that represent the relationship between entities	- A child is assigned to a case - A case has one or many children assigned to it
Attribute (G5 & G6)	Characteristic shared by entities of the same type (e.g., age, social security number)	*Regular attribute:* predictor variables (e.g., Child_Name) *Target attribute:* what the ML model aims to predict (e.g., Length of stay in foster care)

Appending descriptive information to features is common in ML practice. We suggest doing so systematically; specifically, by appending the name of the entity to the attribute names. Furthermore, it recovers information about the entity in a domain, which are otherwise lost in ML models. For example, the attribute age can refer to an attribute from the Child entity or the CaseWorker entity type (not pictured in Figure 1). By appending the entity, we clarify the object to which it belongs.

Guideline 1 (G1): *Preserve information about entity types by appending their names*

If a consistent naming convention is followed, it can be used to guide automated feature engineering algorithms to perform feature transformations by considering the entity (e.g., conducting dimensionality reduction within each entity type).

Guideline 2 (G2): *Perform feature engineering by transforming features based on entity types*

Relationships (e.g., sells, supervises, inspects) in a conceptual model represent associations among entities. The intent of guidelines for handling relationships is to make this information explicit for machine learning purposes and suggest rules for handling certain relationship patterns. These patterns carry different implications for preparing data for ML. An entity-relationship diagram contains additional information about relationships in the form of participation constraints.

Guideline 3 (G3): *If the optional side contains the target variable, remove all records with missing instances of the target-bearing entity from the training dataset.*

In machine learning, partial participation is concerning because it might result in missing values for records where an entity of one type does not have a corresponding record for the entity of another type. This reduces the amount of data used for training and might negatively affect model performance. Evaluate whether missing values are manifestation of subtypes. If the optional attribute is determined to be missing and not a subtype, use imputation techniques.

Guideline 4 (G4): *If the target-bearing entity is found to contain groups that differ with respect to the optional entity, assess the performance of the different subtypes. If the target-bearing entity does not otherwise differ with respect to the optional entity, consider imputing values.*

The method highlights a potential issue when dealing with optional attributes. In a dataset used as input to ML, a missing value can mean that either the value is not applicable to all members of the corresponding entity type, or the value is not available or unknown. In a conceptual model, the corresponding construct is an optional attribute, which represents an attribute that is not applicable to all members of the entity. The method proposes that a machine learning algorithm should not impute missing values if they are not applicable to an entity and impute them if they are truly missing.

Guideline 5 (Optional attributes): *Impute missing values of optional attributes in a data set only if they can be interpreted as meaning that the value is presently unknown, but potentially knowable.*

Attributes represent properties of entities. In machine learning, attributes are called features or variables of the training, validation and scoring datasets [32]. The distinction between simple and composite attributes is important for machine learning because composite attributes might contain components that are individually predictive. However, unless they are decomposed into distinct variables, it might be difficult for the machine learning models to use composite attributes and extract their predictive components (e.g., seasonality of products across a multi-year span).

Guideline 6 (Composite attribute): *For each composite attribute, replace the composite attribute with the individual attributes that comprise it. Label each attribute to capture as closely as possible the semantics of the application domain.*

3. Illustration and Conclusion

To illustrate the application of the method, we use data from a US-based foster care organization. A foster care system is a temporary arrangement to care for a child or children whose birthparents are unable to care for them. We worked closely with the foster care agency to develop ML models to predict the length of stay of a child in the foster system – an important consideration for proper allocation of resources (see Figure 1 for a snippet of the conceptual model).

The predictor attributes in this case include the attributes of the CHILD, PLACEMENT, and CASE. The objective is to predict the length of stay of a child (an attribute that is derived from the EntryDate and ExitDate). For G1, the method suggests preserving the entity type (e.g., Child_Name, Placement_EntryDate, Case_PlanGoal). For G2, the method suggests to group attributes of each entity type (e.g, CHILD, PLACEMENT, CASE) into one or more higher-level dimensions (e.g., Child_Aggr_i, Case_Aggr_i). For G3, we remove all records with missing instances unless there are groups that differ with respect to the optional entity (G4), then we would develop separate models. For G5, Child_Address can be broken down into Child_StreetName, Child_City, and Child_ZipCode. For G6, a missing value can mean that either the value is not applicable to all members of the entity type, or the value is not available or unknown. In a conceptual model, the corresponding construct is an optional attribute, which represents an attribute that is not applicable to all members of the entity (i.e., ExitDate). Our dataset has over 25,000 records on almost 10,000 children. We created two datasets – one following our method (labeled DS_1 in Table 2) and the other was prepared using basic data preparation steps in ML (labeled as DS_0). We used the same ML algorithms with identical hyperparameters to predict length of stay (in days).

Table 2 below provides the results of the comparison, based on consecutive applications of the guidelines. As seen from the results, application of our method shows ML performance improvements –as shown by the decrease in the RMSE of the target variable.

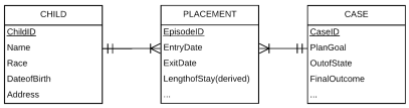

Figure 1: Foster Care Child Placement Conceptual Model

Table 2. Model Comparisons (RMSE, NRMSE, and R2)

Element	Deep Learning	Random Forest	GBM	Light GBM
DS_0	356.31	307.45	316.57	320.38
	.23 (0.26)	.20 (0.45)	.20 (0.42)	.21 (0.40)
DS_1	**218.06**	**196.87**	**208.44**	**208.22**
G2, G6	**.14 (0.59)**	**.13 (0.67)**	**.13 (0.63)**	**.13 (0.63)**
DS_1	359.58	327.58	325.32	325.21
G1, G2	.23 (0.25)	.21 (0.38)	.21 (0.39)	.21 (0.39)
DS_1	**322.06**	**290.52**	**286.08**	***305.45***
G4, G5	**.21 (0.29)**	**.19 (0.42)**	**.18 (0.44)**	***.20 (0.36)***

DS_0: original dataset; DS_1: dataset after applying our method

This research-in-progress proposes that conceptual models can be used effectively to increase ML model performance. A conceptual model represents agreed-upon domain knowledge. Our method can be used to improve ML performance and should be especially effective for situations where there is insufficient data to extract all relevant domain knowledge in a data-driven manner. Future work is needed to expand the current set of guidelines and evaluate them through the application to real-world problems. In our example, not all guidelines applied. Additional research is needed to identify the conditions under which each of the guidelines affect performance.

4. References

[1] Crevier, D.: AI: the tumultuous history of the search for artificial intelligence. Basic Books (1993).
[2] Cerf, V.G.: AI is not an excuse! Communications of the ACM. 62, 7–7 (2019).
[3] McCorduck, P.: Machines who think. A. K. Peters, Ltd, Natick, MA (2004).
[4] Sheng, V.S., Provost, F., Ipeirotis, P.G.: Get another label? improving data quality and data mining using multiple, noisy labelers. In: ACM SIG KDD. pp. 614–622 (2008).
[5] Mitchell, T.M.: Machine learning. 1997. Burr Ridge, IL: McGraw Hill. 45, 870–877 (1997).
[6] Thrun, M.C., Ultsch, A.: Swarm intelligence for self-organized clustering. Art. Intell. 1–54 (2020).
[7] Tremblay, M.C., Dutta, K., Vandermeer, D.: Using data mining techniques to discover bias patterns in missing data. Journal of Data and Information Quality (JDIQ). 2, 2 (2010).
[8] Duboue, P.: The Art of Feature Engineering: Essentials for Machine Learning. Cambridge, UK (2020).
[9] Halevy, A., Norvig, P., Pereira, F.: The unreasonable effectiveness of data. IEEE Intell. Sys. 24, 8–12 (2009).
[10] Heaven, D.: Why deep-learning AIs are so easy to fool. Nature. 574, 163 (2019).
[11] Mylopoulos, J.: Conceptual modelling and Telos. Conceptual Modelling, Databases, and CASE: an Integrated View of Information System Development, New York: John Wiley & Sons. 49–68 (1992).
[12] Recker, J., Lukyanenko, R., Jabbari, M.A., Samuel, B.M., Castellanos, A.: From Representation to Mediation: A New Agenda for Conceptual Modeling Research in A Digital World. MIS Quarterly. (2021).
[13] Reimer, U., Bork, D., Fettke, P., Tropmann-Frick, M.: Preface of the First Workshop Models in AI. Presented at the Modellierung (Companion) (2020).
[14] Fettke, P.: Conceptual Modelling and Artificial Intelligence: Overview and research challenges from the perspective of predictive business process management. Presented at the Modellierung (Companion) (2020).
[15] Bork, D., Garmendia, A., Wimmer, M.: Towards a Multi-Objective Modularization Approach for Entity-Relationship Models. Presented at the ER Forum, Demo and Posters 2020 (2020).
[16] Lukyanenko, R., Castellanos, A., Storey, V.C., Castillo, A., Tremblay, M.C., Parsons, J.: Superimposition: Augmenting Machine Learning Outputs with Conceptual Models for Explainable AI. In: Conceptual Modeling Meets Artificial Intelligence and Data-Driven Decision Making. pp. 1–12. Vienna (2020).
[17] Lukyanenko, R., Castellanos, A., Parsons, J., Tremblay, M. C., & Storey, V. C. Using conceptual modeling to support machine learning. In CAiSE Forum, pp. 170-181. Springer (2019).