Superimposition: Augmenting Machine Learning Outputs with Conceptual Models for Explainable AI

Roman Lukyanenko[1], Arturo Castellanos[2], Veda C. Storey[3], Alfred Castillo[4], Monica Chiarini Tremblay[5], and Jeffrey Parsons[6]

[1] HEC Montreal, Montreal, QC, Canada
[2] Baruch College, CUNY, New York, NY United States
[3] Georgia State University, Altanta, GA United States
[4] CalPoly, San Luis Obispo, CA United States
[5] William and Mary, Williamsburg, VA United States
[6] Memorial University of Newfoundland, St. John's, NL, Canada
roman.lukyanenko@hec.ca, arturo.castellanos@baruch.cuny.edu, vstorey@gsu.edu, acasti63@calpoly.edu, monica.tremblay@mason.wm.edu, jeffreyp@mun.ca

Abstract. Machine learning has become almost synonymous with Artificial Intelligence (AI). However, it has many challenges with one of the most important being explainable AI; that is, providing human-understandable accounts of why a machine learning model produces specific outputs. To address this challenge, we propose *superimposition* as a concept which uses conceptual models to improve explainability by mapping the features that are important to a machine learning model's decision outcomes to a conceptual model of an application domain. Superimposition is a design method for supplementing machine learning models with structural elements that are used by humans to reason about reality and generate explanations. To illustrate the potential of superimposition, we present the method and apply it to a churn prediction problem.

Keywords: Artificial Intelligence, Machine Learning, Superimposition, Conceptual Modeling, Explainable AI, Human Categorization

1 Introduction

Machine learning (ML), which is now almost synonymous with Artificial intelligence (AI), has become a key driver of innovation and change in organizational and daily life [1]. Machine learning consists of methods that use data and algorithms to build models that make inferences on new data and perform specific tasks without being explicitly programmed [2–4]. Growing numbers of organizations are turning to machine learning as part of their drive to make data-driven decisions and seek new efficiencies [5–7]. However, decision makers and the public remain skeptical of relying on ML for their

In: 1st International Workshop on Conceptual Modeling Meets Artificial Intelligence and Data-Driven Decision Making (CMAI 2020) in Conjunction with ER 2020

decisions and actions [8–12]. Given its focus on data and algorithms, an important challenge in using ML is being able to understand how and why models make their decisions – a challenge known as Explainable AI (XAI) [13, 14].

Explainable AI refers to “systems that can explain their rationale to a human user, characterize their strengths and weaknesses, and convey an understanding of how they will behave in the future” [15]. The problem of machine learning explainability is urgent as societal reliance on machine learning grows. At the same time, the tendency in machine learning practice is to employ more powerful and sophisticated algorithms that are increasingly opaque and difficult to explain. Many are considered “black box” models, such as deep learning networks, that are difficult to understand. The inability to understand why machine learning models make certain decisions makes it difficult to detect and mitigate biases, and prevent discriminatory practices embedded inside machine learning models, thus limiting adoption by organizations, especially in highly regulated fields. The rush to open the black box of AI is further fueled by calls from the public and policy makers to treat the ¨right to explanation¨ as a new basic human right [16].

There are many approaches to XAI. Most rely on calculating importance weights to reflect the contributions of features to the decision made by a ML model. None appear to consider using domain knowledge to contextualize the importance of features with respect to the entities or objects they describe.

Conceptual models (CM) are (semi-)formal diagrammatic representations of domain knowledge developed to support information systems development [17–20]. The conceptual modeling community has a rich research tradition of using conceptual modeling to improve various aspects of information systems development. Typical uses include database design, process reengineering, and software development. However, conceptual modeling has only recently been considered within the context of machine learning [21, 22], and has not been applied to the problem of explainable AI. We propose using conceptual models to improve explainability by superimposing the features used in the development of machine learning models to the conceptual models of the domain. We illustrate the use of this *superimposition* method by applying it to predicting customer churn and discuss the implications of doing so.

2 Background: The problem of Explainable AI

Historically, artificial intelligence focused on symbolic representation using logical formalisms [23]. For example, some approaches developed an AI application by first engineering requisite rules in the domain (e.g., by using ontologies or semantic networks typically created manually). The resulting models were, thus, relatively easy to understand.

With the increased availability of data and advances in computing and hardware, the AI field shifted its focus from developing rule-based domain models to computationally intensive data-driven (machine learning) approaches [3, 24]. The power of modern machine learning rests on its ability to make thousands, if not millions, of iterations over the training data to detect complex relationships among the input variables (i.e., feature engineering) and the target variable. These approaches are generally not easily understood by humans, leading to the need for work on XAI.

XAI research includes methods that weight the importance of input features in contributing to a model's decision. Such techniques include local interpretable model-agnostic explanations (LIME) [25], game theoretic approaches to compute explanations of model predictions (SHAP) [26] and use of counterfactuals to understand how removing features changes a decision [27]. These approaches focus on specific features and fail to abstract to higher-level concepts.

In this research, we propose a new approach to explainable AI based on concepts from conceptual modeling. We focus on ML-model agnostic approaches that contribute to explainability of any ML model. Popular techniques include explanations by simplification (e.g., creating a decision tree in addition to a neural network) [28]. Others seek to reduce model complexity by grouping features, making it easier to follow the logic of the model [29]. Work also considers making the marginal contribution of each feature more explicit [26]. However, there does not appear to be research that considers superimposing the features onto domain models. Such an approach can complement existing approaches by combining the logic derived from a machine learning model with knowledge about the application domain. It can provide cognitive benefits that facilitate explanation and understanding.

3 Superimposition Method

Superimposition compensates for the absence of structural semantic information about the real-world domain in a dataset used for machine learning, which, we argue, impedes explainability. Although this information is absent in current ML practice, it is routinely employed by humans to understand their day-to-day experiences. The design idea of superimposition has theoretical roots in cognitive psychology, which argues that humans are continuously subjected to diverse sensory experience. To cope with the sensory input, humans actively employ conceptual structures to filter, interpret, and assimilate the information they receive [30–32]. Such structures are category-based and relational in nature, as we discuss below.

First, a fundamental tenet of modern psychology is that much of sensory and mental experience of humans are organized into categories or concepts. The categories group related (typically similar) individual objects or events, such as trees and birds. Grouping sensory and mental experiences into categories provide many benefits (e.g., cognitive economy, ability to draw inferences, communicative efficiency), leading to nearly automatic imposition of categories onto sensory input [30]. Categories are fundamental units of attention, perception and thought. Human understanding and explanation of phenomena invariably utilizes categories. A set of categories and relationships among them can be viewed as a theory of a domain [33].

Second, human knowledge organization and the mechanisms used to understand and interpret phenomena are also relational. To cope with the large number of acquired categories, humans organize them into higher order structures, such as hierarchies, taxonomies, or networks [34, 35]. These structures are based on some type of *relationships* among the categories (e.g., type of, part of, similar to).

Taken together, categories and relationships provide the fundamental structuring that facilitates reasoning, understanding and explanation. However, these elements are either absent from the typical output of machine learning models or inaccessible to the naked eye. For example, a model built using a deep learning algorithm is comprised of features, path coefficients, bias, and activation functions [2]. The categories and higher order categorical structures (e.g., hierarchies) are absent, whereas the relational elements, such as path coefficients between features are opaque and difficult to understand, especially in large models. Even a relatively simple decision tree, while containing interpretable relationships among features, lacks categories. Considering the preponderance of categories and relationships for human interpretation and explanation, we assume the lack of such mechanisms undermines explainability in machine learning models.

We propose *superimposition* as a design method for supplementing existing machine learning models with conceptual models. Specifically, we observe that any machine learning model is a model of some domain (e.g., credit card fraud, image classification, online auctions). The model itself is a set of rules for estimating a value of interest or discriminating among the cases of interest based on previously provided domain examples. Most commonly, these rules, through a series of mathematical transformations, describe patterns of relationships among variables of the domain and a target.

Based on the arguments above, we reason that, to support the understanding of a machine learning model in a domain, we can leverage the knowledge about the categories and the relationships within that domain. Such knowledge can be obtained from conceptual data models [36].

Major conceptual modeling grammars, such as the Entity-Relationship Diagrams or Class Diagrams in UML, rely on entity types or classes (i.e., categories) to represent domains. Classes distill essential features of objects for storage and use in an information system [37, 38]. Identifying classes has traditionally been viewed as one of the most important steps in systems development [39]. Likewise, relationships are also seen as fundamental to modeling, because they capture structural connections among the classes [40]. Research on conceptual modeling, has focused on facilitating accurate (and complete from the point of view of a predefined purpose) representation of domains using classes and relationships [20].

Superimposition maps the output of machine learning models (i.e., the features, rules and transformation functions) onto a conceptual model of the domain. First, the method assumes a conceptual model of the domain needs to be available or prepared in the form of an Extended Entity Relationship (EER) diagram. We assume the availability of a typical EER diagram containing entity types and their corresponding sets of attributes, which are the fields for the variables used in the machine learning. The entity types are connected through the relationship types.

Second, once a machine learning model from the same domain is developed, its output needs to be mapped to the related constructs of the conceptual model. The execution of this step depends on the type of the machine learning model. In all cases, a machine learning model includes features that are related to attributes in a conceptual model. These variables can be mapped to attributes in the conceptual model. However, as it is common to conduct feature engineering and transform variables (e.g., by merging them, or engineering new variables from the existing ones), this step may not be straightforward in all applications. The method is intended to provide traceability between the final variables used and the original source attributes in the conceptual model. This can be done, for example, by using graphical elements and comments inside the conceptual model to show transformations from the original features to their final form [25].

Third, the method suggests indicating inside the conceptual model information about the rules of the machine learning models. This step depends on the type of machine learning model. For example, if a regression model is used, these rules can be represented as feature weights or feature coefficients. These coefficients can be appended to the attributes in the conceptual model, or the attributes can be highlighted differently to indicate the different relative importance of each attribute.

The final step of the superimposition method involves analyzing the resulting conceptual model to gain a clearer understanding of the underlying rules machine learning models use to makes its decisions, and to identify opportunities to improve the machine learning model further.

4 Illustration: Superimposition using EERD

We illustrate superimposition using a churn model on a publicly available dataset[1], Telco Customer Churn. The dataset includes information about customers who left (churned) within the last month. Each row represents a different customer. A customer may have signed up for different services such as a phone, multiple lines, internet, online security, online backup, device protection, tech support, or streaming service. In addition, the data contains demographic information (e.g., sex, age, gender) and information about the customer's account (e.g., tenure, contracts, payment method, monthly charges, and total charges).

For simplicity, we assume that the conceptual model already exists and is available to the analyst (see Figure 1). Entity types represent the categories of interest in a domain, such as CUSTOMER, CONTRACT, PHONE, INTERNET, or SENIORCITIZEN. Relationships (e.g., isBilled, subscribes) in a conceptual model represent associations among entity types. Relationships also capture some constraints on the interactions among entities of different types. For example, a CUSTOMER isBilled through *multiple* CONTRACTS. In a typical ML process, information about relationships among entities is not explicit, and must be learned, requiring sufficient training data.

[1] https://www.kaggle.com/blastchar/telco-customer-churn

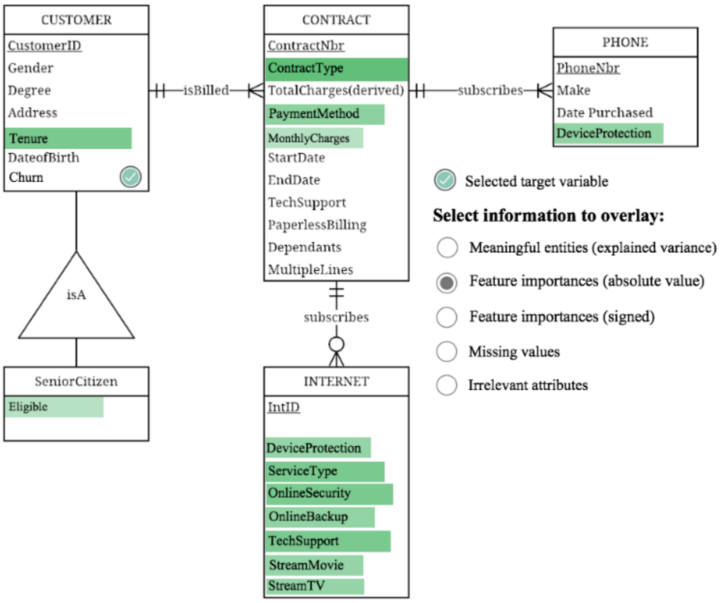
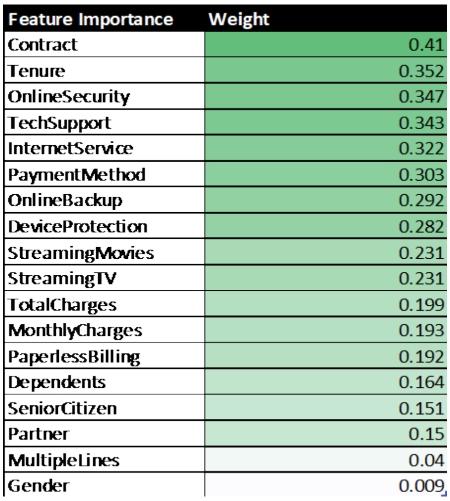


Figure 1. Superimposed EER Model for the Telecom. Churn Dataset

In this illustration the machine learning task is a classification task (i.e., the target variable is the binary variable *Churn*). The goal is to develop a predictive model that maps the input features to this target variable. Each customer can subscribe to many services such as phone, internet, online security, online backup, device protection, tech support, and streaming through different contracts. The contracts have information such as payment method, paperless billing, monthly charges, and total charges. Finally, there is information about customers, such as gender and age.



Feature Importance	Weight
Contract	0.41
Tenure	0.352
OnlineSecurity	0.347
TechSupport	0.343
InternetService	0.322
PaymentMethod	0.303
OnlineBackup	0.292
DeviceProtection	0.282
StreamingMovies	0.231
StreamingTV	0.231
TotalCharges	0.199
MonthlyCharges	0.193
PaperlessBilling	0.192
Dependents	0.164
SeniorCitizen	0.151
Partner	0.15
MultipleLines	0.04
Gender	0.009

Figure 2. Feature importance Random Forest (surrogate model)

We construct a machine learning model using a general-purpose CPU compute Amazon AWS cloud instance with an Intel Xeon E5-2676 v3 (Haswell) processor, 16 Gb RAM, and h2oai-driverless-ai-1.8.5 AMI (Amazon Linux). To build a machine learning model, we used random forest – a popular machine learning model. In Figure 2, we retrieve the feature importance from the random forest model by weight in descending order. We then followed the steps of the superimposition method. We color-code these weights which are then overlaid in the ER diagram in Figure 1.

Compared to the traditional features shown in Figure 2, following the superimposition (in Figure 1) provide some insightful patterns for interpreting model results. For instance, Customers with month-to-month contracts (ContractType) had a higher chance of churning or, after 18 months of having the service (CustomerTenure), the likelihood of churn decreases. The sharp increase in the likelihood of churning occurs for customers who pay more than $65.86 a month. Considering that internet subscriptions has all features detected as important, there also may be opportunity for strategic bundling of internet service features in order to better serve existing customer needs. Note that we focus on the feature importance (absolute value). However, we can generate different representations in Figure 1 by choosing different layers: Meaningful entities (e.g., aggregating the explained variance of all the features within an entity), missing values (e.g., potentially identify any structural issues in the data collection process), and irrelevant attributes (i.e., not relevant for our purpose) can help in feature selection. In each case, we provide more information to the decision makers to explain what the machine learning model is doing.

5 Discussion and Future Work

Our work contributes a method called *superimposition* to improve explainability of AI by using conceptual modeling. Although this is work-in-progress, it has potential to contribute to both conceptual modeling and machine leaning research and practice. The ML context expands the scope of conceptual modeling beyond traditional information systems development, process modeling, and database design [41]. The application of conceptual modeling to ML can create a bridge between the conceptual modeling and ML communities, foster interdisciplinary connections, and underscore the continued importance and value of conceptual modeling research [41].

The superimposition method can help increase ML explainability. The method makes it possible to indicate which entities contribute to an ML model's predictions and how these entities are related. It also allows the expression of the relationships between predictors and the target as the relationship between entities and the target. Such information is helpful for humans to make sense of phenomena; its absence from current XAI approaches inhibits their effectiveness. While the method cannot provide an explanation or justification why the model makes a certain prediction, it might aid humans in reasoning about the logic behind an ML model.

To better support our method, grammar extensions or new modeling grammars might be needed. For example, as complex ML models require translation of decision rules and path coefficients into conceptual modeling grammars, new conceptual modeling constructs may be needed to accommodate this. As we illustrate, grammars could allow color-coding of attributes included in the ML process as inputs and, perhaps, use one color to indicate a target attribute and a different color for attributes that cannot be used in a predictive model due to compliance to regulations (e.g., gender or race). Furthermore, the method can be applied to other representational artifacts (not just EER, as we showed here), and, for example, it could superimpose onto domain ontologies or semantic networks. We thus call on research to extend the method in response to the need to improve XAI.

We plan to experimentally evaluate the superimposition method by comparing it with current approaches to XAI based on feature weights as well as other approaches to explainability. We will expand the method by superimposing the outputs of more opaque models such as neural networks. Future work should study how to interpret abstract and complex engineered features using conceptual modeling, particularly when the underlying features are not from related or adjacent entities. Moreover, future work should extend the concept of superimposition beyond EER to more general ontologies.

References

1. Marr, B.: The top 10 AI and machine learning use cases everyone should know about. Forbes. (2016).
2. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge, Mass. (2016).
3. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature. 521, 436–444 (2015).
4. Maass, W., Parsons, J., Purao, S., Storey, V.C., Woo, C.: Data-driven meets theory-driven research in the era of big data: opportunities and challenges for information systems research. Journal of the Association for Information Systems. 19, 1253–1273 (2018).
5. Chen, H., Chiang, R.H., Storey, V.C.: Business Intelligence and Analytics: From Big Data to Big Impact. MIS Quarterly. 36, 1165–1188 (2012).
6. Davenport, T., Harris, J.: Competing on Analytics: Updated, with a New Introduction: The New Science of Winning. Harvard Business Press, Cambridge, Mass (2017).
7. Khatri, V., Samuel, B.: Analytics for Managerial Work. Communications of the ACM. 62, 100–108 (2019).
8. Akbilgic, O., Davis, R.L.: The Promise of Machine Learning: When Will it be Delivered? Journal of Cardiac Failure. 25, 484–485 (2019).
9. Bailetti, T., Gad, M., Shah, A.: Intrusion learning: An overview of an emergent discipline. Technology Innovation Management Review. 6, (2016).
10. Holzinger, A., Kieseberg, P., Weippl, E., Tjoa, A.M.: Current advances, trends and challenges of machine learning and knowledge extraction: From machine learning to explainable ai. In: International Cross-Domain Conference for Machine Learning and Knowledge Extraction. pp. 1–8. Springer (2018).

11. Ransbotham, S., Kiron, D., Prentice, P.K.: Beyond the hype: the hard work behind analytics success. MIT Sloan Management Review. 57, 3–15 (2016).
12. Sun, T.Q., Medaglia, R.: Mapping the challenges of Artificial Intelligence in the public sector: Evidence from public healthcare. Government Information Quarterly. 36, 368–383 (2019).
13. Castelvecchi, D.: Can we open the black box of AI? Nature News. 538, 20 (2016).
14. Gunning, D.: Explainable artificial intelligence (XAI). Defense Advanced Research Projects agency. Defense Advanced Research Projects Agency (DARPA), nd Web, 2. (2016).
15. Gunning, D., Aha, D.W.: DARPA's explainable artificial intelligence program. AI Magazine. 40, 44–58 (2019).
16. Wachter, S., Mittelstadt, B., Floridi, L.: Why a right to explanation of automated decision-making does not exist in the general data protection regulation. International Data Privacy Law. 7, 76–99 (2017).
17. Bubenko, J.A.: On the Role of'Understanding Models' in conceptual schema design. Presented at the Fifth International Conference on Very Large Data Bases, 1979. (1979).
18. Mylopoulos, J.: Information modeling in the time of the revolution. Information Systems. 23, 127–155 (1998).
19. Pastor, O.: Conceptual modeling of life: beyond the homo sapiens. In: International Conference on Conceptual Modeling. pp. 18–31. Springer, Gifu, Japan (2016).
20. Wand, Y., Weber, R.: Research commentary: Information systems and conceptual modeling - A research agenda. Information Systems Research. 13, 363–376 (2002).
21. Lukyanenko, R., Castellanos, A., Parsons, J., Chiarini Tremblay, M., Storey, V.C.: Using Conceptual Modeling to Support Machine Learning. In: Cappiello, C. and Ruiz, M. (eds.) Information Systems Engineering in Responsible Information Systems. pp. 170–181. Springer International Publishing, Cham (2019).
22. Nalchigar, S., Yu, E.: Conceptual modeling for business analytics: a framework and potential benefits. Presented at the 2017 IEEE 19th Conference on Business Informatics (CBI) (2017).
23. Crevier, D.: Ai: The Tumultuous History of the Search for Artificial Intelligence. Basic Books (1993).
24. Cerf, V.G.: AI is not an excuse! Communications of the ACM. 62, 7–7 (2019).
25. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you?: Explaining the predictions of any classifier. Presented at the Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (2016).
26. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Advances in neural information processing systems. pp. 4765–4774 (2017).
27. Martens, D., Provost, F.: Explaining data-driven document classifications. Mis Quarterly. 38, 73–100 (2014).
28. Rai, A.: Explainable AI: From black box to glass box. Journal of the Academy of Marketing Science. 48, 137–141 (2020).

29. Henelius, A., Puolamäki, K., Boström, H., Asker, L., Papapetrou, P.: A peek into the black box: exploring classifiers by randomization. Data mining and knowledge discovery. 28, 1503–1529 (2014).
30. Harnad, S.: To Cognize is to Categorize: Cognition is Categorization. Presented at the , Amsterdam (2005).
31. Murphy, G.: The big book of concepts. MIT Press, Cambridge, MA (2004).
32. Palmeri, T.J., Blalock, C.: The role of background knowledge in speeded perceptual categorization. Cognition. 77, B45–B57 (2000).
33. Parsons, J., Wand, Y.: Extending Classification Principles from Information Modeling to Other Disciplines. Journal of the Association for Information Systems. 14, 2 (2012).
34. Collins, A.M., Quillian, M.R.: Retrieval time from semantic memory. Journal of verbal learning and verbal behavior. 8, 240–247 (1969).
35. Hutchinson, J., Lockhead, G.: Similarity as distance: A structural principle for semantic memory. Journal of Experimental Psychology: Human Learning and Memory. 3, 660 (1977).
36. Burton-Jones, A., Weber, R.: Building conceptual modeling on the foundation of ontology. In: Computing handbook: information systems and information technology. p. 15.1-15.24. , Boca Raton, FL, United States (2014).
37. Borgida, A.: Features of languages for the development of information systems at the conceptual level. IEEE Software. 2, 63 (1985).
38. Parsons, J., Wand, Y.: Choosing classes in conceptual modeling. Communications of the ACM. 40, 63–69 (1997).
39. Sowa, J.F.: Top-level ontological categories. International journal of human-computer studies. 43, 669–685 (1995).
40. Chen, P.: The entity-relationship model - toward a unified view of data. ACM Transactions on Database Systems. 1, 9–36 (1976).
41. 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).