University of Saskatchewan

Course No: ME886

Course Name: Engineering Design Methodology

Project

Engineering Design Approach to Generalized Classification Models of Machine Learning Algorithm

Date of Submission: 31st December 2018

Mohammad Habibullah; NSID: moh220
Rakibul Islam Chowdhury; NSID: ric797
Mohammad Abdul Moin Oninda; NSID: mmo323
Department: Electrical and Computer Engineering

Abstract

Engineering design methodology consists of a series of steps used by engineers to solve problems. Researchers have been working on obtaining solutions to problems using various techniques – one of the most widely used technique is the implementation of machine learning algorithms. Machine learning can be expressed as the capability of computers to adapt and respond, without being explicitly programmed and gives a relationship from input, X to output, Y. This aim of this project is to propose a generalized characterization of machine learning algorithms using established engineering design methodologies such as Function(F), Context (C), Behavior (B), Principal (P), State (S) and Structure (S), FCBPSS framework, Axiom 1 (i.e. to prove that the implementation of machine learning algorithm can be classified as either decoupled or uncoupled system), Axiom 2 (i.e. when multiple designs satisfy Axiom 1, the process of selecting the best design is based on Axiom 2 which consists of the independence and the information axioms) thereby achieving a more systematic way of approaching and eventually solving a problem.

Introduction

Design means to create a thing that is not existed. We, human beings design everything either for the functionality of the artefact or because they make us feel comfortable or help to change our emotions. System means the object which is engineered or modified. Much research has been conducted on what may be included as the components of a system or generic model [1]. Rodenacker first proposed the concept of FBS – Function, Behavior, Structure in his book, "*Mehodisches Konstruieren*" in the early seventies. As many different views of the FBS model have been developed and researched [2], Zhang et al. proposed the unified FCBPSS model in [1]. The FCBPSS framework modified and extended the FBS framework to have more layers of concepts. The FCBPSS model uses the few key definitions namely – Function (F), Context (C), Behavior (B), Principle (P), State (S) and Structure (S).

FCBPSS Framework

The FCBPSS framework layouts these concepts along the two dimensions: the engineering dimension (state/structure-behavior-function) and the governing dimension (the context and the principle) **Error! Reference source not found.** [1].

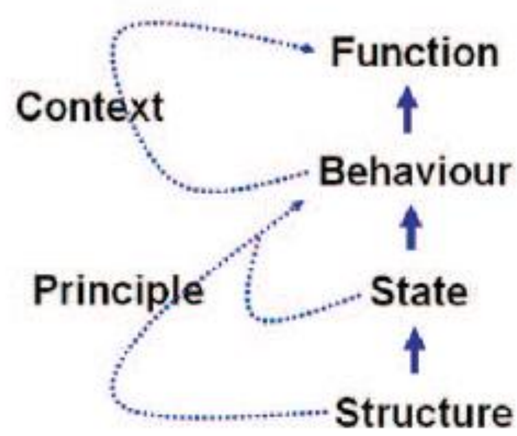

Fig 1. The FCBPSS Framework

The FBPSS model uses the following definitions -

Structure & State:

A system has a structure which is a set of entities connected in a meaningful way. These entities are perceived in the form of their states when the system is in operation. States of the entities are thus quantities (numerical or categorical) of either physical or chemical domains. The states change with respect to time which implies the dynamics of an underlying system. In order to express the dynamics of the system in a formal way, the states are expressed by the variables called state variables. The state variable can further be divided into two categories. If it receives the information, energy or material from the outside of the system then it is called independent state variable. On the contrary, if it offers information energy or material to the outside of the system then it is called dependent stable variable. Different structures can have same state variable.

Behavior:

The behavior of a system is about the response of the system when it receives a stimuli. So, basically, it is the relationship between the indepenedent state variable and the dependent state variable or simply relationship between the input and the output.

Principle:

The principle is the fundamental law with which one can develop a quantitative relation for the state variable. Principle governs the behavior.

Function & Context:

Function implies the usefulness of the system and is context-sensitive. Function can play generic or specialized role. Specialized role depends on what and how the system deals with its evironment. Environment entity includes energy, material and information.

In recent year, machine learning and artificial intelligence have become buzzwords. Many complicated problems are being solved easily with the help of these algorithms. There are mainly three different applications of these algorithms in data processing. They are – regression, classification and clustering. In every arena stated, there exists different unique algorithms to improve speed or accuracy or robustness.

In this project, we have chosen a very famous and widely used classification algorithm namely "*K Neareset Neighbor (KNN)*" as an example to model with FCBPSS framework layout.

<u>Function and Context:</u>

The general function of any classifier is to classify an unknown data i.e. to fit the unknown data into any predefined classes. But how do they do that? The class of an unknown data is estimated using labeled training dataset. The result of any classifier may largely vary based on the training dataset. So, the training dataset is the imposed context of the classifier. Also, there are different specific functions of different classifier algorithms. For example for KNN, easy adaptibility to new training dataset is a special function.

<u>Behavior:</u>

Different classification algorithms behaves differently when stimulated by input. Here, to explain the behavior of a classifier, we have taken KNN as an example. When an unclassified data is required to classify, the algorithm first determines the distance of all training points from the unclassified data points and sorts them in ascending order. Then depending on the majority of classes of first k (an integer) training point, the test data point is classified. In this case the behavior largely depends on both the value of k and training data set. This can be depicted by Fig 2.

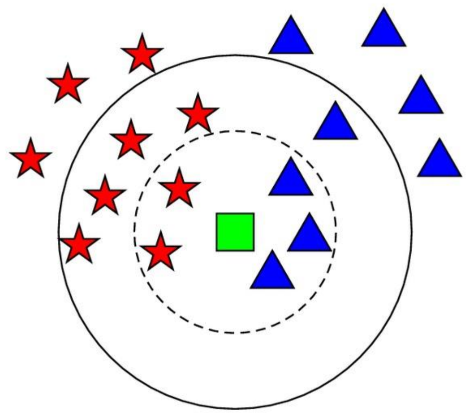

Fig 2: Behavior of KNN algorithm

In Fig 2, the green rectangle is our test point to be classified into either the blue triangle or red star. Now, if we take the value of k as 5, the dotted circle represents the 5 nearest neighbor of the test point. In this arena, the blue triangle is the majority. So, the test point (green rectangle) should be classified as blue triangle. On the contrary, if we double the previous value of k and set it to 10, the undotted circle will represent the 10 nearest neighbours of the test point. This time majority of the neighbors are red star. So the classifiers identifies the test point as the red star.

Thus based on the training set and the value of k, the output varies largely.

Principle:

The mathematical base of the KNN algorithm mainly depends on how we measure the distance of two points. There are several principles of finding the right distance. Few includes but not limited to Eucledian method, Manhattan method, Minkowsky method and so on. If we consider two points in a 2D plane (X_1,Y_1) and (X_2,Y_2) the distance between them according to different principles are listed in Table 1.

Table 1: Different Methods to determine distance between two points in 2D

Name of the Method	Distance
Eucledian Method	$\sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2}$
Manhattan Method	$\lvert(X_1 - X_2)\rvert + \lvert(Y_1 - Y_2)\rvert$
Minkowsky Method	$(\lvert(X_1 - X_2)\rvert^r + \lvert(Y_1 - Y_2)\rvert^r)^{1/r}$

State and Structure:

From the discussion above, it is obvious that, the training set, the value of k and the test data are independent states of the KNN algorithm. On the other hand, only the class of test point is dependent state which in fact depends on all independent states.

The structure of the classifier is the algorithm which can be represented as the form of flow chart. For example, the structure of KNN algorithm has been represented in Fig 3.

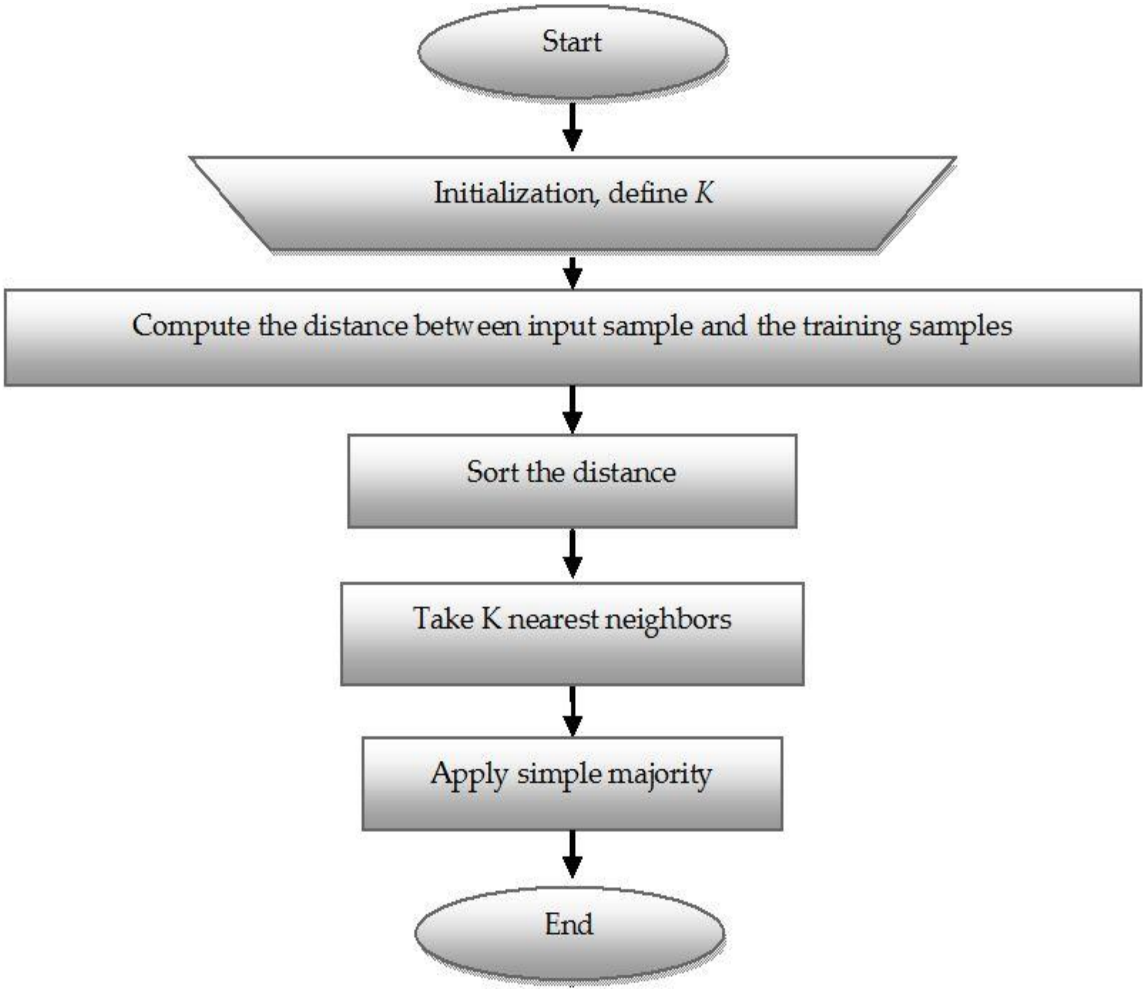

Fig 3: Flowchart of KNN algorithm

Axiomatic design of classification algorithms

The design world consists of four domains. Design involves interplay between "*what we want to achieve*" and "*how we choose to satisfy the need (i.e., the what)*". The design process can be thought of as four different kinds of activities that ensures an important foundation of axiomatic design. The world of design is made up of four domains: the customer domain, the functional domain, the physical domain, and the process domain [3]. The domain structure is illustrated schematically in Fig.4. The domain on the left relative to the domain on the right represents "*what we want to achieve*" whereas the domain on the right represents the design solution represents "*how we propose to satisfy the requirements specified in the left domain*".

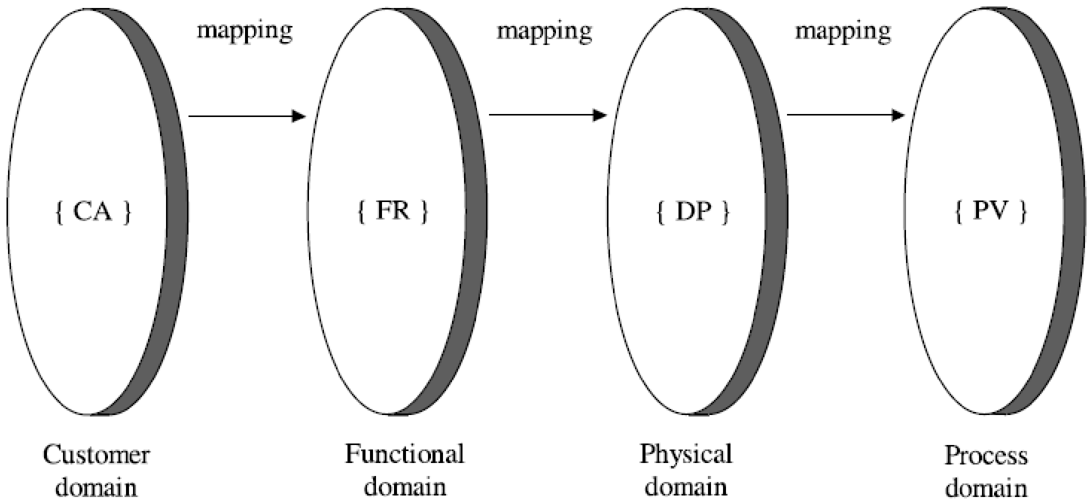

Fig 4. Four domains of the design world [3]

The customer domain is characterized by the attributes that the customer is looking for in a product or process or system or materials. In the functional domain, the customer needs are specified in terms of functional requirements (FRs) and constraint requirement (CRs). To satisfy the specified FRs, we conceive design parameters (DPs) in the physical domain. Finally, to produce the product specified in terms of DPs, we develop a process that is characterized by process variables (PVs) in the process domain.

In case of designing a machine learning algorithm to satisfy a customer need, all designers go through similar design stages regardless of the specific goal of their effort. In this section, it will be shown that how a machine learning classification algorithm is designed with axiomatic design approach to satisfy a customers need.

Customer Domain:

The customer domain is characterized by the needs (or attributes) that the customer is looking for in a product or process or system or materials. Let’s assume that an Agri-based based company say ‘X’ asks a developer to design an algorithm that can readily predict the concentration levels of three major nutrients of plants - Nitrogen (N), Phosphorus (P) of plant leaves with high accuracy. For convenience, it is considered that the company ‘X’ can provide spectral reflectance dataset in both visible and near-infrared regions of N and P deficient plant leaves.

<u>Functional domain:</u>

In the functional domain, the customer needs are specified in terms of functional requirements (FRs) and constraints. So, to satisfy the customer requirements, the functional requirements are:

FR1: Predict Nitrogen (N) levels of plant leaves.
FR2: Predict Phosphorus (P) levels of plant leaves.
FR3: Execution time of the algorithm should be as less as possible.

CR (Global): The accuracy of the algorithm should be more than 90%.

<u>Physical domain:</u>

To satisfy the specified FRs, we conceive design parameters (DPs) in the physical domain. Design parameters are the solutions that characterize the FRs. In this domain, conceptual design is performed to solve the FRs.

The first functional requirement is predicting nitrogen levels of plant leaves. Visible and NIR reflectance dataset are available. Now, the question is how nitrogen contents in plants are related to the dataset. From [4], it evident that nitrogen contents can be estimated by spectral reflectance at red (671 nm) and NIR (780 nm). So, suitable classification algorithm, say Support Vector Machine (SVM), can be used to train the model with the dataset and make prediction. So, the first design parameter for FR1 is:

DP1: Train a classification algorithm with red (671 nm) and NIR (780 nm) dataset with proper ‘class’ labels.

The second functional requirement is predicting phosphorus levels of plant leaves. According to [5], reflectance in the blue region (440 nm, 445 nm) and NIR (730 nm) can be considered as features for phosphorus level classification in plants. So, the second parameter for FR2 is:

DP2: Train a classification algorithm with blue (440nm, 445 nm) and NIR(730 nm) dataset with proper 'class' labels.

The third functional requirement of the design is making the execution time of the algorithm as less as possible. So, to make the algorithm work fast, dimensionality, that is, number of features can be reduced by using suitable feature extraction algorithms, say Random Forest. So, the third design parameter for FR3 is:

DP3: Reduce dimensionality of the dataset by using feature extraction algorithm.

The mapping process between the domains can be expressed mathematically in terms of the characteristic vectors that define the design goals and design solutions. At a given level of the design hierarchy, the set of FRs that defines the specific design goals constitutes the FR vector in the functional domain. Similarly, the set of design parameters in the physical domain that has been chosen to satisfy the FRs constitutes the DP vector. The relationship between these two vectors can be written as:

$$\{FR\} = [A]\{DP\}$$

where, [A] is called the design matrix that characterizes the product design. The design matrix is of the following form for a design that has three FRs and three DPs:

$$[A] = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

According to the above discussion, FR1 depends only on DP1 and FR2 depends only DP2 as, DP1 and DP2 are the reflectance from different wavelengths. But, FR3 depends on DP1, DP2 and DP3. The reason is that reducing dimensionality depends on both DP1 and DP2.

So, in the design matrix, A_{12}, A_{13}, A_{23} and A_{21} are zero. So,

$$\begin{bmatrix} FR1 \\ FR2 \\ FR3 \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} DP1 \\ DP2 \\ DP3 \end{bmatrix}$$

As, [A], the design matrix is triangular matrix the design is a decoupled design. So, this design satisfies the Axiom 1 that is - The independent Axiom.

The above stated problem can be solved in various ways. When more than one design satisfies Axiom 1, the problem of selecting the best design can be chosen based on Axiom 2. The best design has minimum information content and thereby has the highest probability of success. The information axiom provides a quantitative way to select the optimum form of design solution. Information content, [3] can be calculated by –

$$I = \log_2\left(\frac{System\ range}{Common\ range}\right)$$

Conclusion

From the discussion above, it is understandable that, we can safely assume a machine learning classifier algorithm like a product design and apply existing product design rules. We can model a classifier using product design modeling layout e.g. FCBPSS. We can also apply Axiom 1 for a better classifier design. In case of multiple possible solutions to a specific classification problem, we can choose the best classifier using Axiom 2. This will help to solve any classification problem in an efficient way.

References

1. W. Zhang, Y. Lin and N. Sinha, "On the Function-Behavior-Structure Model for Design," in *2nd CDEN Conference*, Alberta, Canada, 2015.
2. W. Rodenacker, Methodisches Konstruieren, Berlin: Springer-Verlag, 1971
3. Lee, D. G., & Suh, N. P. (2005). Axiomatic design and fabrication of composite structures-applications in robots, machine tools, and automobiles. *Axiomatic Design and Fabrication of Composite Structures-Applications in Robots, Machine Tools, and Automobiles, by Dai Gil Lee and Nam Pyo Suh, pp. 732. Foreword by Dai Gil Lee and Nam Pyo Suh. Oxford University Press, Nov 2005. ISBN-10: 0195178777. ISBN-13: 9780195178777*, 732.
4. Stone, M. L., Solie, J. B., Raun, W. R., Whitney, R. W., Taylor, S. L., & Ringer, J. D. (1996). Use of spectral radiance for correcting in-season fertilizer nitrogen deficiencies in winter wheat. *Transactions of the ASAE*, *39*(5), 1623-1631.
5. Osborne, S. L., Schepers, J. S., Francis, D. D., & Schlemmer, M. R. (2002). Detection of phosphorus and nitrogen deficiencies in corn using spectral radiance measurements. *Agronomy journal*, *94*(6), 1215-1221.