CLASSICAL MODELS FOR MACHINE LEARNING

Anita Wasilewska
Computer Science Department, Stony Brook University
Stony Brook, NY, USA

TALK OUTLINE

Machine Learning as a Process

- Syntax, Semantics, Model

General Model

Semantic Model

- Machine Learning as Generalization

Descriptive Model

General Model revisited

- Descriptive Language, Satisfaction and Truth

Future Directions:

- **Topological Models** for Machine Learning:
  Building topological Syntax and Semantics

INTRODUCTION

Machine Learning - ML - is a **process** that includes the following phases:

creating the **target data**,

data **preprocessing**,

learning proper,

patterns **evaluation**,

knowledge presentation

INTRODUCTION

Syntax

Syntax, or **syntactical concepts** refer to simple relations among symbols and expressions of formal, symbolic **languages**

The expressions of formal languages, even if created with a specific meaning in mind, **do not** carry themselves any meaning

The meaning is being assigned to them by establishing a proper **semantics**

INTRODUCTION

Semantics

Semantics for as given symbolic language $\mathcal{L}$ **assigns** a specific interpretation in some domain to all **symbols** and **expressions** of the language

It also involves related ideas such as **truth** and **model**
They are called **semantical concepts** to distinguish them from the syntactical concepts

INTRODUCTION

Model

The word model is used in many situations

and has many meanings but they all reflect some parts,

if not all, of its following formal meaning

A structure M is a **model** for a set $\mathcal{E}_0$ of expressions of the formal language $\mathcal{L}$ if every expression $E \in \mathcal{E}_0$ is **true** in M

INTRODUCTION

We present here three abstract models for ML:

Semantic, Descriptive and General

All of them are abstract **structures** that allow us to

formalize main properties of the ML Process

We want to stress that they **do not cover** all of the Machine Learning field **nor** of all of its existing algorithms and methods

General Model: Syntax and Semantics for ML

The models we consider are defined in order to address formally the

semantics-syntax duality inherent to the

Machine Learning Process

We usually **view** Machine Learning results and

present them to the user in their descriptive,

i.e. **syntactic form** which is the most natural form of

communication

But the ML process is deeply **semantical** in its nature

We hence build the **General Model** on two levels:

syntactic and semantic

General Model: Syntax and Semantics for ML

The **syntactic level** is represented by a Descriptive Model

The **semantic level** is described by the Semantic Model

The **semantics-syntax duality** of the ML process is expressed in the General Model by the **satisfiability** relation

General Model: Syntax and Semantics for ML

We also use the General Model to provide a formal **definition** of the ML process as a process of information **generalization**

In the model the data preprocessing and learning algorithms are defined as certain **operators** that act on data

Data are represented in a form of Knowledge Systems that have granularity associated with them
The **operators** change, or not, their granularity

General Model: Syntax and Semantics for ML

General Model is a system

$$\mathbf{GM} \;=\; (\; \mathcal{S}\mathbf{M},\; \mathcal{D}\mathbf{M},\; \models \;)$$

where

$\mathcal{S}\mathbf{M}$ is a **Semantic Model**;

$\mathcal{D}\mathbf{M}$ is a **Descriptive Model**;

$\models \subseteq \mathcal{P}(U) \times \mathcal{E}$ is called a **satisfaction relation**,

U is the **universe** of $\mathcal{S}\mathbf{M}$,

$\mathcal{E}$ is the set of **descriptions** defined by the $\mathcal{D}\mathbf{M}$

Semantic Model

Semantic Model is the most important component of the **General Model**

When we perform a learning algorithm the first **step** is to remove the key attribute

This step allows us to introduce similarities in the database as now records do not have their unique identification

The **input** into the learning algorithm is hence a a data table obtained from the target data by removal of the key attribute

We call it a **target data table**.

Semantic Model

Machine Learning, as it is commonly said, is a process of **generalization**

In order to model this process we have first to define what does it mean from semantical point of view that one stage of the process is **more general** then the other

The idea behind is very simple

It is the same as saying that a formula for example

$(a + b)^2 = a^2 + 2ab + b^2$

is **more general** then the formula

$(2 + 3)^2 = 2^2 + 2 \cdot 2 \cdot 3 + 3^2$

From **semantical** point of view it means that ML process consists of putting objects (records) in **sets of objects**

Semantic Model

From **syntactical** point of view the ML process consists of building **descriptions** in terms of pairs (attribute, values of attribute) of these **sets of objects**, with some extra parameters, if needed

To model this idea we generalize Pawlak's model of Information System

$$I = (U, A, V_A, f)$$

where $U \neq \emptyset$ is called a set of **objects**;

$A \neq \emptyset$, $V_A \neq \emptyset$ are called the set of **attributes** and **values of of attributes**, respectively;

$f: \; U \times A \longrightarrow V_A$ is called an **information** function

Knowledge System

Knowledge System based on an Information System

$I = (U, A, V_A, f)$ is a system

$K = (K(U), A, E, V_A, V_E, g)$

where $K(U) \subseteq \mathcal{P}(U)$;

E is a finite set of **knowledge attributes** (k-attributes) such that $A \cap E = \emptyset$;

V_E is a finite set of **values of k- attributes**;

g is a partial function called **a knowledge function** (k-function);

$g : \mathcal{P} \times (A \cup E) \longrightarrow (V_A \cup V_E)$ is such that:

$g \mid (\bigcup_{x \in U}\{x\} \times A) = f$;

$\forall S \in \mathcal{P},\ \forall a \in A\ ((S, a) \in dom(g)\ \Rightarrow\ g(S, a) \in V_A)$;

$\forall S \in \mathcal{P},\ \forall e \in E\ ((S, e) \in dom(g)\ \Rightarrow\ g(S, e) \in V_E)$

Semantic Model

We view ML algorithms as certain **operators** and define the model as follows

Semantic Model is a system

$$\mathcal{S}\mathbf{M} = (\mathcal{P}(U),\ \mathcal{K},\ \mathcal{G})$$

where

$U \neq \emptyset$ is the **universe**;

$\mathcal{K} \neq \emptyset$ is a set of knowledge systems, called learning process **states**;

$\mathcal{G} \neq \emptyset$ is the set of **operators**;

Each operator $p \in \mathcal{G}$ is a partial function on the set of all ML process states, i.e.

$$p :\ \mathcal{K} \longrightarrow \mathcal{K}$$

Machine Learning Operators

In machine learning the preprocessing and ML proper stages are inclusive/exclusive categories

The preprocessing is an integral and very important stage of the ML process and needs as careful **analysis** and the ML proper stage

We distinguish **two** disjoint classes of **operators**:

the **preprocessing** operators $\mathcal{G}_{prep}$ and

machine **learning proper** operators $\mathcal{G}_{ml}$

We put

$$\mathcal{G} = \mathcal{G}_{prep} \cup \mathcal{G}_{ml}$$

Semantic Model; ML Operators

Machine Learning Operators

The main idea behind the concept of an ML operator is to capture not only the fact that ML techniques generalize the data but also to categorize existing methods and algorithms

We want to make sure that our categorization **distinguishes**, as it should, for example clustering from classification , or from association analysis

Semantic Model; ML Operators

Machine Learning Operators

We want make sure that all classification algorithms fall into one category defined by **classification operators** and

all clustering algorithms would fall into a category defined by **clustering operators**

The third category we consider is the association analysis described in our framework by **association operators**

Semantic Model; ML Operators

We don't include in our analysis purely statistical methods like regression and others

This gives us only three classes of operators to consider:

classification operators $\mathcal{G}_{class}$

clustering operators $\mathcal{G}_{clust}$

association operators $\mathcal{G}_{assoc}$

Semantic Model; ML Operators

We prove the followiNg

Theorem

Let $\mathcal{G}_{class}$, $\mathcal{G}_{clust}$ and $\mathcal{G}_{assoc}$ be the sets of all classification, clustering, and association operators, respectively

The following conditions hold

(1) $\mathcal{G}_{class} \neq \mathcal{G}_{clust} \neq \mathcal{G}_{assoc}$

(2) $\mathcal{G}_{clust} \cap \mathcal{G}_{class} = \emptyset$

(3) $\mathcal{G}_{assoc} \cap \mathcal{G}_{clust} = \emptyset$

Machine Learning Process

We model here preprocessing and learning proper stages of **learning process** and adopt the following definition

Definition

Any sequence $K_1, K_2, ...K_n$ $(n \geq 1)$ of learning states is called a **data preprocessing** process, if there is a preprocessing operator $G \in \mathcal{G}_{prep}$, such that

$$G(K_i) = K_{i+1}, \ i = 1, 2, ...n - 1$$

The sequence $K_1, K_2, ...K_n$ is called a **learning proper** process if there is a learning operator $G \in \mathcal{G}_{ml}$, such that

$$G(K_i) = K_{i+1}, \ i = 1, 2, ...n - 1$$

Descriptive Model

Given a Semantic Model $\mathcal{S}\mathbf{M} = (\mathcal{P}(U), \mathcal{K}, \mathcal{G})$

We associate with it its **descriptive counterpart** and we define it as follows

Descriptive Model is a system

$$\mathcal{D}\mathbf{M} = (\mathcal{L}, \mathcal{E}, \mathcal{DK})$$

where:

$\mathcal{L} = (\mathcal{A}, \mathcal{E})$ is called a descriptive **language**;

$\mathcal{A}$ is a countably infinite set called an **alphabet**;

$\mathcal{E} \neq \emptyset$ and $\mathcal{E} \subseteq \mathcal{A}^*$ is the set of **expressions** of $\mathcal{L}$;

$\mathcal{DK} \neq \emptyset$ and $\mathcal{DK} \subseteq \mathcal{P}(\mathcal{E})$ is a set of **descriptions** of knowledge states

Decscriptive Model

As in a case of semantic model, we build the descriptive model for a given application

We assume however, that whatever is the application, the **descriptions** are always build in terms of attributes and attributes values, some logical connectives, and some parameters, if needed

For example, a whole neural network with its nodes and weights can be seen as a formal **description** and the knowledge states could represent changes in parameters during the training

Descriptive Learning

When we build here a model for a **descriptive learning** we assume that the **descriptions** are build from attributes and attributes values and two logical connectives: conjunction and implication

We use them to model different kind of **rules** that are being learned by **descriptive** ML algorithms:

discriminant and **characteristic** rules in classification analysis;

association rules in association analysis;

or other rules obtained by hybrid systems

General Model Revisited: Satisfaction and Truth

General Model is a system

$$\mathbf{GM} = (\mathcal{S}\mathbf{M}, \mathcal{D}\mathbf{M}, \models)$$

where

$\mathcal{S}\mathbf{M}$ is a **Semantic Model**;

$\mathcal{D}\mathbf{M}$ is a **Descriptive Model**;

$\models \subseteq \mathcal{P}(U) \times \mathcal{E}$ is called a **satisfaction relation**,

U is the **universe** of $\mathcal{S}\mathbf{M}$,

$\mathcal{E}$ is the set of **descriptions** defined by the $\mathcal{D}\mathbf{M}$

All the components of **GM** except the **satisfaction relation** have already been defined

General Model Revisited: Satisfaction and Truth

As the last step we define the **satisfaction relation** $\models$ and the notion of **truth** in **GM** as follows

Satisfaction Definition

For any $S \in \mathcal{P}(U)$ and for any $F \in \mathcal{E}$,

$$S \models F \quad \text{if and only if} \quad \exists K \in \mathcal{K}(S \models_K F)$$

Truth Definition

We define $F \in \mathcal{E}$ is **true** (we write it symbolically $\models F$) in **GM** as follows

$$\models F \quad \text{if and only if} \quad \exists K \in \mathcal{K}(\models_K F)$$

FUTURE DIRECTIONS

Topological Models for Machine Learning

Building **Topological Machine Learning** Syntax and Semantics