People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

IBN KHALDOUN UNIVERSITY OF TIARET

Dissertation

Presented to:

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTEMENT OF COMPUTER SCIENCE

In order to obtain the degree of:

MASTER

Specialty: software engineer

Presented by:
Boubekeur Chaima

Belghitar Bouchra

On the theme:

A deep learning model for predicting greenhouse film degradation

Defended publicly on 15/ 06 /2023 in Tiaret in front the jury composed of:

Mr KHARROUBI Sahraoui	MCA Tiaret University	Chairman
Mr AID Lahcene	MCA Tiaret University	Supervisor
Mr Meghazi hadj madani	MAA Tiaret University	Examiner

2022-2023

Acknowledgements

First and foremost, we express our gratitude to the Almighty God for granting us the strength and perseverance to successfully complete this humble work. We extend our heartfelt greetings to the foremost educator, Prophet Mohammed (Peace and blessings be upon him).

We would like to extend our sincere appreciation to our supervisor, ***Mr. AID Lahcene****. We are immensely grateful for the honor you bestowed upon us by guiding this project. Without your valuable corrections and meticulous advice, this work would not have been possible. Your kindness, expertise, and outstanding professionalism serve as a constant source of inspiration for us. May this work stand as a testament to our sincere thanks and utmost respect.*

Our heartfelt thanks also go to the members of the jury, ***Mr. Kharroubi Sahraoui*** *and* ***Mr. Meghazi hadj madani.*** *It is a great honor and privilege to have you as jury members, entrusted with evaluating our modest work.*

Special gratitude is extended to ***Mr. Dehbi Abdelkader*** *for his invaluable assistance.*

His unwavering support played a crucial role in the completion of our project.

We would also like to express our appreciation to all those who have supported us, near or far, particularly the faculty staff. Their guidance and assistance have greatly contributed to the realization of our work.

Dedications

I dedicate this work to my dear parents, who have always been a source of encouragement, support and patience.

To my brothers and my sister

As well as to my colleagues and my friends, who have been there by my side throughout this journey.

B. Bouchra

Dedications

And if I write, the ink is over, and I fill the news papers, and I'm not done, any word expresses my longing for you. Here's my longing and nostalgia, you who left us and didn't leave our memory. I missed a father who would never come back and no one would come like him, if motherhood was tenderness, fatherhood was safety.

My most precious human being in existence, to those who gave me love, I knew its true meaning, to those who throughout her life preferred my comfort from her comfort to those who were our bond. No matter how much I keep writing, no matter how much my tongue pronounces, your right will not be fulfilled, my most precious blessing. God perpetuate you in my life, my beloved mother.

My dear sisters, Khadidja ,Kheira and Soumia, are my example, her generosity and her moral support who have been a source of courage and confidence for me, to my dear brothers, Tayeb, Ayoub and Ahmed the best people who were by my side in the bad times before good, for all our good memories together , and I especially thank Mima who really supported me and is still trying to find a way to destress me

All members of my large family for their encouragement and patience and especially My aunt Zohra, my sisters Wafaa, Matita,Kholoud,Israa,Wissal,Souhila ,My brothers Mohamed Amine,Lokmane and my sons Mohamed Ilyes ,Mohamed islam , Labib and Mohamed karim

To my dear Sarra, Imen and Kheira, with whom I had unforgettable memories during the five years of university studies . With them I shared the stress and pressure of studying, the flavor of success and a lot of crazy laughter.

To my dear Anouna ,Lina,Ikram,Hadjira,Nour el Houda and Fatima ,Thank you from the bottom of my heart for your enduring giving, your wonderful poses, words of praise don't give you your right, and all words of thanks don't describe how grateful I am to you my precious girlfriends. To my dear Mouloud Djenane, Thank you for your constant guidance.

To my dear partner Bouchera, No matter how much I have searched the dictionary for scattered words and expressions of thanks, I would not find words that would fulfill your right and your destiny.

All my close friends Sihem, Hadjer, Nessrine, Bouchera and Amel who supported me and who were unforgettable and wonderful people

B. Chaima

Abstract

The behavior of LDPE films has been studied experimentally, as temperatures; UV rays and precipitation are accumulated in these films, which are responsible for the deterioration of greenhouses. Multilayer films have been subjected to the aging process in different ways, whether it is artificial over a specific period of time (such as multilayer Agro-film) or as a result of external factors. Based on the cost of these experiments in terms of time and the complexity of the factors to consider, we aim in this work to use computer simulations as a tool to study the aging process in these films.

Deep learning-based AI has proven useful in a wide range of applications and research fields, as it can produce high-level graphical representations of large amounts of raw data, unlike traditional machine learning algorithms. Thus, it is an excellent solver for a wide range of linear problems and also shows high efficiency in solving nonlinear problems.

The purpose of this study is to propose a methodology for designing a deep learning model for nonlinear analysis, and subsequently to develop a CNN model capable of dealing with this problem. Then it will be applied to solve real-world problems such as life prediction and mechanical performance degradation of multilayer polyethylene films in greenhouses. To carry out this work, we created a deep learning hybrid model that is able to meet this challenge, and the training and validation phases of the model were successful, indicating the feasibility of using the D1 CNN model for nonlinear analysis.

Keywords: LDPE, Strain stress curves, Artificial intelligence, Machine learning, deep learning ,convolutional neural network.

Résumé

Le comportement des films LDPE a été étudié expérimentalement, comme les températures ; Les rayons UV et les précipitations s'accumulent dans ces films, responsables de la détérioration des serres. Les films multicouches ont été soumis au processus de vieillissement de différentes manières, qu'il soit artificiel sur une période de temps spécifique (comme le multicouche Agro-film) ou à la suite de facteurs externes. Compte tenu du coût de ces expériences en termes de temps et de la complexité des facteurs à prendre en compte, nous visons dans ce travail à utiliser des simulations informatiques comme outil pour étudier le processus de vieillissement de ces films.

L'IA basée sur l'apprentissage en profondeur s'est avérée utile dans un large éventail d'applications et de domaines de recherche, car elle peut produire des représentations graphiques de haut niveau de grandes quantités de données brutes, contrairement aux algorithmes d'apprentissage automatique traditionnels. Ainsi, c'est un excellent solveur pour un large éventail de problèmes linéaires et montre également une grande efficacité dans la résolution de problèmes non linéaires.

Le but de cette étude est de proposer une méthodologie pour concevoir un modèle d'apprentissage profond pour l'analyse non linéaire, et par la suite de développer un modèle CNN capable de traiter ce problème. Ensuite, il sera appliqué pour résoudre des problèmes du monde réel tels que la prédiction de la durée de vie et la dégradation des performances mécaniques des films de polyéthylène multicouches dans les serres. Pour mener à bien ce travail, nous avons créé un modèle hybride d'apprentissage en profondeur capable de relever ce défi, et les phases d'apprentissage et de validation du modèle ont été couronnées de succès, indiquant la faisabilité d'utiliser le modèle D1 CNN pour l'analyse non linéaire.

Mots clés : LDPE, courbes de contrainte de déformation, intelligence artificielle, apprentissage automatique, apprentissage en profondeur, réseau neuronal convolutif

ملخص

تمت دراسة سلوك أفلام البولي إثيلين المنخفض الكثافة تجريبياً ، مثل درجات الحرارة ؛ تتراكم الأشعة فوق البنفسجية وهطول الأمطار في هذه الأغشية ، وهي المسؤولة عن تدهور البيوت المحمية. تعرضت الأفلام متعددة الطبقات لعملية الشيخوخة بطرق مختلفة ، سواء كانت مصطنعة خلال فترة زمنية محددة (مثل أفلام Agro-film متعددة الطبقات) أو كنتيجة لعوامل خارجية. بناءً على تكلفة هذه التجارب من حيث الوقت وتعقيد العوامل التي يجب أخذها في الاعتبار ، نهدف في هذا العمل إلى استخدام المحاكاة الحاسوبية كأداة لدراسة عملية الشيخوخة في هذه الأفلام.

أثبت الذكاء الاصطناعي القائم على التعلم العميق أنه مفيد في مجموعة واسعة من التطبيقات ومجالات البحث ، حيث يمكنه إنتاج تمثيلات رسومية عالية المستوى لكميات كبيرة من البيانات الأولية ، على عكس خوارزميات التعلم الآلي التقليدية. وبالتالي ، فهو حل ممتاز لمجموعة واسعة من المشاكل الخطية ويظهر أيضًا كفاءة عالية في حل المشكلات غير الخطية.

الغرض من هذه الدراسة هو اقتراح منهجية لتصميم نموذج التعلم العميق للتحليل غير الخطي ، ومن ثم تطوير نموذج CNN قادر على التعامل مع هذه المشكلة. ثم سيتم تطبيقه لحل مشاكل العالم الحقيقي مثل التنبؤ بالحياة وتدهور الأداء الميكانيكي لأفلام البولي إيثيلين متعدد الطبقات في البيوت البلاستيكية. لتنفيذ هذا العمل ، أنشأنا نموذجًا هجينًا للتعلم العميق قادرًا على مواجهة هذا التحدي ، وكانت مراحل التدريب والتحقق من صحة النموذج ناجحة ، مما يشير إلى جدوى استخدام نموذج D1 CNN للتحليل غير الخطي.

الكلمات المفتاحية : منحنيات الإجهاد، الذكاء الاصطناعي، التعلم الآلي، التعلم العميق، الشبكة العصبية التلافيفية

Table of contents

General Conclusion

Table of Figures

List of tables

List Of Abbreviations

A

(ADAM) Adaptive Moment Estimation

(API) application programming interface

(AUC) Area under the Curve

(AI) Artificial Intelligence

(ANN) Artificial Neural Network

B

(BCE) Binary Cross- Entropy

C

(CPU) Central Processing Unit

(CLI) Command-line interface

(CCE) Categorical Cross-Entropy

(CNN) Convolutional Neural Network

D

(DL) Deep Learning

G

(GANs) Generative Adversarial Networks

(GPUs) Graphics Processing Unit

(GRU) Recurrent Gated Units

(GUI) Graphical User Interface

K

(KNN) K-Nearest Neighbors

(KL) Kullback Le

L

(LSTM) Long Short-Term Memory

M

(ML) Machine Learning

(MSE) Mean Squand Error

N

(NN) Neural Network

O

(ONEIROS) Open-ended Neuro Electronic Intelligent Robot Operating System)

P

(PE) Polyethylene

R

(ReLU) Rectified Linear Unit

(RNN) Recurrent Neural Network

S

(SCCE) Sparse Categorical Neural Network

(SGD) Stochastic Gradient Descent

(SVM) Support Vector Machine

U

(UV) Ultra-Violets

(UTS) Ultimate Tensile Strength

GENERALE INTRODUCTION

Greenhouse farming plays a critical role in modern agriculture by creating controlled environments that promote optimal plant growth and protect crops from external factors. An essential component of greenhouse systems is the greenhouse film, which serves as a protective cover and regulates environmental conditions such as temperature, humidity, and light intensity. However, over time, greenhouse films degrade due to exposure to factors such as UV radiation, temperature fluctuations, and chemical reactions. This degradation significantly affects the mechanical properties of the film and ultimately impacts its ability to maintain an ideal greenhouse environment.

Accurate prediction of greenhouse film degradation is crucial for farmers and greenhouse manufacturers. It allows them to anticipate the film's lifespan, plan for maintenance or replacement, and optimize resource allocation. Traditional degradation prediction methods rely on empirical models based on physical experiments, which are time-consuming, expensive, and often lack accuracy.

In recent years, deep learning models have emerged as powerful tools for various prediction tasks, leveraging the capabilities of artificial neural networks to learn complex patterns and relationships from large datasets. These models have demonstrated remarkable success in domains such as image recognition, natural language processing, and speech recognition. Applying deep learning techniques to greenhouse film degradation prediction offers a promising alternative to traditional methods, providing accurate and efficient predictions based on film characteristics and environmental conditions.

In this study, we propose a deep learning model specifically designed for predicting greenhouse film degradation. Our model leverages the capabilities of deep neural networks to capture intricate relationships between various factors influencing film degradation and accurately forecast its lifespan. By combining convolutional neural networks (CNNs) and support vector machines (SVMs), our model extracts relevant features from raw data and classifies the film's degradation status.

The main objective of this research is to develop a reliable and efficient tool for greenhouse farmers and manufacturers to predict greenhouse film degradation. Such a tool can optimize maintenance schedules, reduce downtime, and ensure optimal greenhouse conditions for crop growth. Furthermore, by providing accurate degradation predictions, the model contributes to sustainable farming practices by minimizing resource waste and environmental impact associated with premature film replacements.

GENERALE INTRODUCTION

In the following sections, we will discuss the methodology employed to develop the deep learning model, the dataset used for training and evaluation, and the experimental results obtained.

To finalize our project, we have structured our work as follows

CHAPTER I: Aging of greenhouse films

CHAPTER II: DEEP LEARNING

CHAPTER III: Implementation

CHAPTER I

Aging of greenhouse films

I.1 Introduction

In recent years, polymers have found many applications, especially in the field of packaging, where they are widely used. They are also used in mechanical systems such as bearings, pinions and seals due to their special mechanical properties and ease of use. In addition, they are commonly used for the manufacture of fluid transport tubes. However, the use of plastic materials can be limited by their aging, which is manifested by an irreversible degradation of their properties. In this chapter, we will present an overview of polymers, in particular low density polyethylene, which is relevant for our project. We will first describe this type of polymer, and then we will discuss the modes of polymer degradation.

I.2 Definitions

I.2.1 Polymer

The term "polymer" originates from the Greek words "poly," meaning many, and "mer," meaning part. From a chemical perspective, a polymer is a large organic molecule with a high molecular weight, composed of repeating units joined by covalent chemical bonds. Due to its size, this type of molecule is often referred to as a macromolecule. [01]

I.2.2 Monomers

Monomers are the basic compounds of polymers. Relevant to organic chemistry, they associate covalent bonds with a sequence of similar additional molecules or not (of carbon atoms and atoms of hydrogen, oxygen, nitrogen, etc.) under the appropriate polymer formation reaction conditions used for a process particular, each carbon atom, tetravalent, is linked to neighboring atoms by four covalent bonds, oriented in space towards the four vertices of a regular tetrahedron. [03]

I.3 Polymer polymerization

The term 'polymerization' refers to all the processes for forming chains from smaller molecules (monomers); each monomer is isolated, it is then combined with other monomers of the same nature or of a different nature during a chemical reaction called the Polymerization reaction. [04]

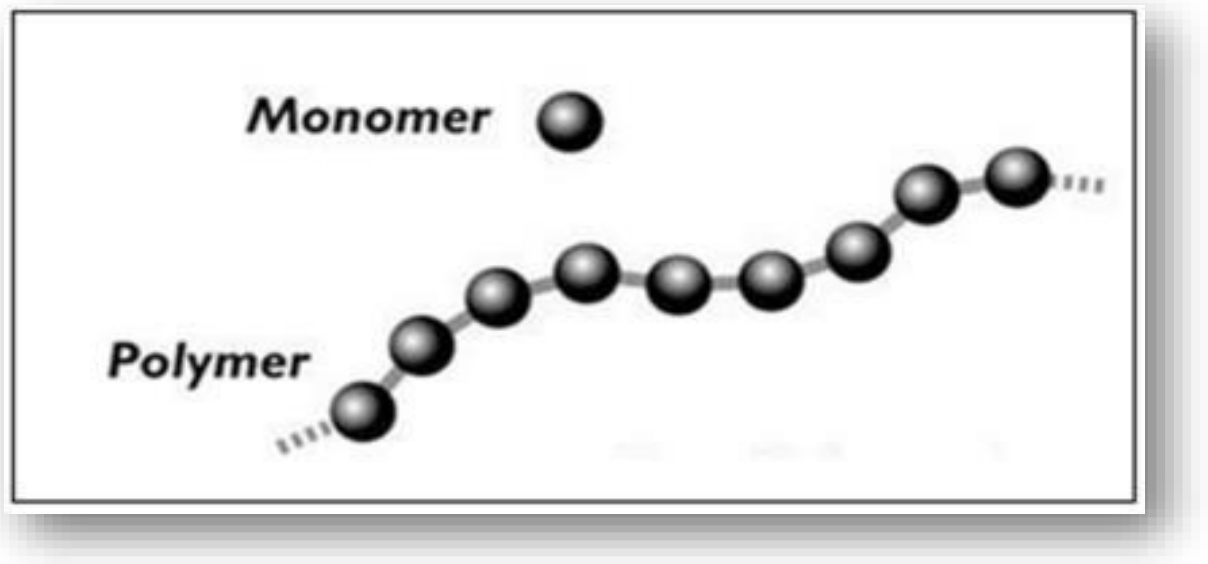


***Figure I.1*:** Polymerization.

I.4 Properties of polymers

The properties of polymer materials are much more sensitive than those of metals to external influences such as temperature, hardness, intensity of applied stress, UV radiation and chemical agents. Their modulus of elasticity is generally two orders of magnitude less than that of metals. Resistance characteristics can differ only by an order of magnitude. Apart from the specific parameters or materials (molar mass, branching rate, chain mobility, crosslinking rate, etc.), and the external conditions (humidity, agents chemicals, temperature, rate of stress, type and intensity of stresses applied). [03]

I.4.1 Physical properties

First of all, remember that there is a wide variety of plastic materials, just as there are a large number of metal alloys, one of the physical characteristics general polymers is: [5]

- **Density**

The density of plastics is low. The lightness of polymers is undoubtedly one of the qualities that has most widely contributed to their diffusion.

When it comes to the ratio (tensile strength / density), some polymers are actually far superior to metallic materials. The low density of plastics is due to the low atomic weight of the main atoms in their chains (mainly hydrogen and carbon). [5]

- **The amorphous state**

In the amorphous state, a chain unfolds in space to adopt a configuration in which no order can be distinguished. The absence of order gives the polymer a "frozen" liquid structure whose main characteristics are as follows: To. No TF melting point. b. Existence of a glass transition point TG marking the transition from the rubbery liquid state to the glassy state. Vs. Visible transparency. “Crystal” polystyrene or “crystal” poly (vinyl chloride) are transparent because they have an amorphous structure. [6]

- **The crystalline state**

The crystalline state is characterized by the existence of a long-range order. The chains having adopted a regular conformation in planar zigzag or in helix are bundled in a way neat and compact. We can then define a crystal lattice, which is repeated periodically in the three directions of space. The main characteristics of the state crystal are as follows:

- Compactness greater than that of the amorphous phase.
- Existence of a melting point TF, absence of glass transition.
- Refractive index higher than that of the amorphous phase (nc>na).

- Total impermeability to most gases and vapors or liquids.
- Stiffness greater than that of the amorphous phase. [6]

From the analysis of these two states, we can deduce the behavior of the polymer and generally classify polymers into different categories.

Linear polymers possess a certain cohesion thanks to physical bonds. So there are solvents for them. By heating them, we go through a plastic stage, then a liquid stage (physical bonds broken). They are thermoplastics. Solid thermoplastics are either semicrystalline (polyethylene, polypropylene, polyamide, etc.) or amorphous (polystyrene, polyvinyl chloride, poly (methyl methacrylate), etc.) and are fusible at High temperature. Thermoplastics are generally recyclable and easy to process in liquid form.

Network polymers have their cohesion due to chemical bonds. There is therefore no solvent or molten phase for them. If heated, the few ramifications of the network pendants cling. They are thermosets. They do not melt but decompose at high temperature. Thermosets are rigid, brittle and temperature-resistant materials. Thermosets are not recyclable and their automated implementation is more complex. On the other hand, they can be used as structural materials.

I.4.2. Thermal properties

The glass transition temperature "Tg" and the melting temperature "Tf" are the two fundamental temperatures necessary in the study of polymeric materials.The glass transition temperature is partly important for amorphous polymers, especially amorphous thermoplastics, for which there is no significant cohesive force other than entanglement. [7]

The characteristic temperatures of one and the same material can then be classified as follows:

- Glass transition temperature
- Crystallization temperature

- Melting temperature
- Thermal decomposition temperature. [7]

Depending on the temperature to which it is subjected, a polymer material can exhibit different mechanical behavior. This can occur for semi-crystalline thermoplastics in even a narrow temperature range. [7]

I.4.3 Mechanical properties

The success of polymers stems in part from the ease with which they can be formed into desired shapes; especially in the molten state they are very malleable (very plastic, hence their name). In fact, this plasticity varies in a wide range, from the most rigid, hard and brittle, to the most soft (pasta) or elastic (elastomers). [8]

I.5 Polyethylene (PE)

Today, polyethylene constitute a large family of products with very varied characteristics. The polyethylene used in the gas industry is a material thermoplastic made by chemical synthesis from ethylene. The finished product used is a rather complex mixture of various products such as carbon black, which gives its final colour, each providing enhancement and protection to the whole. In fact, the peculiarity that characterizes PE compared to other metallic materials is its evolution over time. In order to ensure its reliability, it is necessary to avoid any error at the level of manufacturing, storage – handling and implementation in particular.

I.5.1 Definition (EP)

A polyethylene molecule is nothing more than a long chain of carbon atoms, with two hydrogen atoms attached to each carbon atom. Its name comes because it is the polymer obtained by the polymerization of ethylene monomers (CH2 = CH2) into a complex structure with the generic formula: [9] − (CH2 − CH2) n –. It may be easier to draw it, simply with the chains of carbon atoms several thousand atoms long:

***Figure I.2*:** Representative diagram of polyethylene.

Polyethylene is translucent, easy to handle and cold resistant. Polyethylene is thermoplastic. This plastic material represents about a third of the total production synthetic materials and constitutes half of the plastic packaging. Polyethylene takes precedence over most other materials because it can be reused. [10]

I.5.2 Types of Polyethylene

PE (polyethylene) are part of the thermoplastics. There are different types of polyethylene classified according to their density, which depends on the number and length of the branches present on the molecular chains. [11]

- HDPE, high-density polyethylene synthesized under low pressure by the Ziegler process. It is in the form of a regular linear macromolecular chain. Some irregularities remain present (1%carbon) and appear in the form of short ramifications. Its density is the highest of all polyethylene has and varies between 0.955 and 0.970 g/cm3.
- LDPE, low-density polyethylene synthesized under high pressure by radical polymerization. Unlike HDPE, LDPE has a high number of relatively large branches (about 60% carbon) directly responsible for its low density (0.915 and 0.935 g/cm3). [12]

I.5.3 The properties of LDPE low-density polyethylene

In general, industrial methods do not lead to a perfectly regular architectural edifice, made up of indicated structural patterns. Certain chemical acts, which are integral parts of

the polymerization mechanism, are at the origin of irregular structure, introduced according to the case in the chain or at its ends. These irregularities can influence, directly or indirectly, the physical or chemical properties of the polymer. Indeed, the breaking strength is directly linked to the concentration at the end of the chains. They represent certain beginnings to the phenomenon of rupture. In addition, they oppose crystallization; Therefore, the ends of chains play a role fundamental in mechanical properties. [12]

I.6 Aging of polyethylene:

Aging describes a set of complex mechanisms that result in a slow and irreversible change in the intrinsic properties of materials. Thus, exposed to the weather the polyethylene films degrading quite quickly. The responsible factors are solar radiation, variations in temperature, humidity, wind, hail, sand, oxygen in the air and atmospheric pollution. [13]

I.6.1 General Aspects of aging

The different types of aging are generally classified according to the external cause Thus, the high temperature is at the origin of the thermal aging the UV or ionizing rays cause photochemical aging etc.… There are, however, a certain number of aspects common to all types of aging (chain breaks, cross-linking, etc.) whose study methods and consequences on the physical properties of polymers do not vary from one type of aging to another. [14]

- **Statistical cut-off reactions**

This is the most important process. A wide variety of mechanisms can lead to "statistical" cuts, i.e. randomly distributed over the chains. Macromolecular by mechanisms that contribute to this operation. The break in the continuity of the covalent bonds results in the deterioration of the mechanical properties, thus causing a reduction in the stress at break.

- **Depolymerisation reactions**

This is the reverse of polymerization reactions. They involve a sequential elimination of monomer molecules, from a particularly unstable site pre-existing (for example the end of the chain). In this case, the variation in molar mass of the polymer is much higher because each chemical act of depolymerisation leads to the release of a molecule of gaseous monomers.

- **Cross-linking reactions**

These reactions result in the creation of a two- or three-dimensional network by certain elastic and plastic properties of the polymer. It strengthens resistance to decomposition. An exclusively thermal route can carry out the crosslinking reactions. Although most of the time a chemical agent (oxygen for example) intervene.

I.6.2 Physical aging

Physical aging manifests itself in several different ways, which we will discuss:

- **Lamination**

Lamination Plasticization occurs when solvent molecules introduce themselves into the macromolecular network. This produces disorders that weaken, or even destroy, the bonds secondary between chains, responsible for the cohesion of the material. Thus, by destroying the secondary bonds of the polymer, water decreases the mechanical cohesion and increases the molecular mobility.

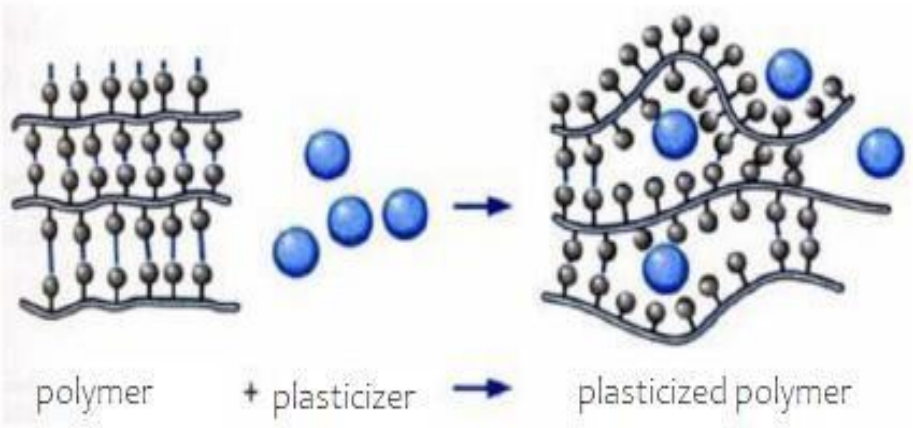


***Figure I.3*: *Schematizes the penetration of a solvent into a polymer*.**

Plasticization is characterized by a change in the mechanical properties of the material, which results in a drop in the glass transition temperature (Tg), transition between the rubber domain and the viscoelastic domain.

- **Swelling**

The penetration of solvent within the polymer material can induce a swelling likely to cause modifications of internal structure when there are heterogeneities which induce constraints between more or less swollen zones. These areas can be amorphous (relatively accessible), or crystalline (relatively impenetrable). This swelling can also occur when the diffusion kinetics of the solvent creates concentration gradients. Finally, it exists when the material undergoes successive cycles of aging.

- **The damage**

Stress damage can produce cracks or fissures in the material. Crazing relates to areas made up of voids and highly oriented fibrils in the polymer. Cracking is the result of cracking which can lead to the formation of a micro crack and then a crack.

I.6.3 Chemical aging

Chemical aging brings together several phenomena that can sometimes be complementary to each other, which considerably complicates the analysis. They translate mainly by phenomena of hydrolysis, oxidation, leaching or diffusion and can lead to an

alteration of the macromolecular skeleton according to three distinct mechanisms:

- Cuts in the backbone can induce a drop in the molar mass of the polymer, which leads to a drop in mechanical properties.
- One or more cross-linking reactions can occur, especially on polymers that have double bonds in the chain or reactive groups that have not yet reacted before.
- One or more reactions on the side groups can occur by substitution of one group on another, by elimination of the groups in favor of a double bond or a cyclization.

They are generally classified according to the external cause that provokes them: thermochemical, photochemical, biological aging

There is also another type of hydrolytic aging which mainly involves polymers whose chain contains hetero atoms. [15]

It should be noted that most aging phenomena are thermoactive, and that the presence of defects in the material accelerates these processes.

I.7 Stress-strain curves

Stress-strain curves are crucial graphical representations that measure the mechanical properties of materials, frequently encountered by students studying Mechanics of Materials. However, they possess certain intricacies, particularly when dealing with ductile materials that undergo significant geometric changes during testing. This module serves as an introductory discussion on various key points necessary for interpreting these curves. Additionally, it offers a preliminary overview of several aspects related to a material's mechanical properties. It is important to note that this module does not aim to cover the

extensive range of stress-strain curves exhibited by modern engineering materials. For a comprehensive survey on this topic, the atlas by Boyer cited in the References section can be consulted. Furthermore, topics such as yield and fracture mentioned here will be explored in greater details in subsequent modules. [16]

I.8 The engineering measures of stress and strain

The tensile test is widely considered as the most crucial examination of a material's mechanical response. This test involves clamping one end of a rod or wire specimen in a loading frame while subjecting the other end to a controlled displacement δ (refer to Figure I4). A transducer connected to the specimen measures the load P(δ) corresponding to the displacement. Alternatively, modern servo-controlled testing machines allow for loadcontrolled testing, where the displacement δ(P) is monitored as a function of the applied load. In this module, the engineering measures of stress (denoted as σe) and strain (denoted as ε) are determined by using the original cross-sectional area A0 and length L0 of the specimen. They are calculated as follows:

$$\sigma e = P/A0 , \quad \varepsilon = \delta/L0 \qquad (1).$$

By plotting the stress σe against the strain ε, we obtain an engineering stress-strain curve as shown in Figure I.5.

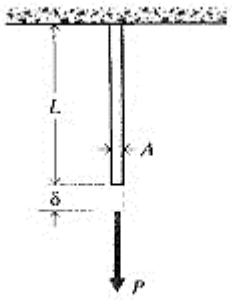


***Figure I.4*:** The tension test.

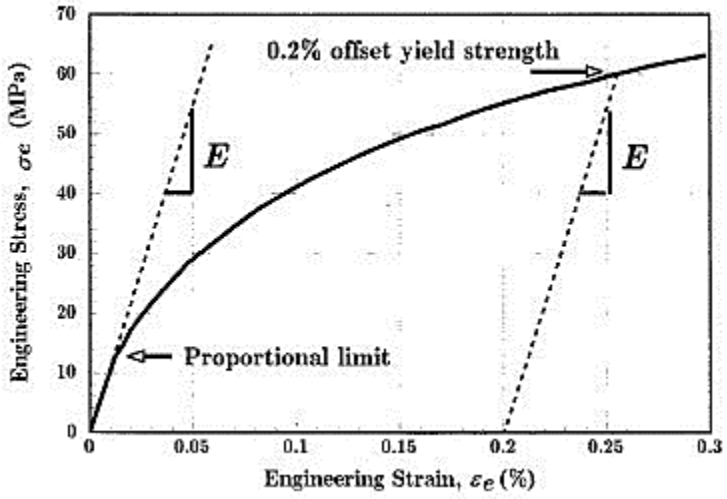


***Figure I.5*:** low-strain region of the engineering stress-strain curve for annealed polycrystalline copper.

Figure I.5illustrates the low-strain region of the engineering stress-strain curve for annealed polycrystalline copper, which is typical for many ductile metals. In the initial portion of the curve, often referred to as the low strain region, many materials approximately obey Hooke's law. According to Hooke's law, stress is directly proportional to strain, and the constant of proportionality is known as the modulus of elasticity or Young's modulus (denoted as E):

$$\sigma e = E\varepsilon \qquad (2).$$

As the strain increases, many materials deviate from this linear proportionality, which is usually observed at a point called the proportional limit. This nonlinearity is associated with stress-induced "plastic" flow in the specimen, indicating a rearrangement of the material's internal molecular or microscopic structure. During this process, atoms move to new equilibrium positions. In crystalline materials, plasticity occurs due to dislocation motion, which is discussed in detail in a later module. Materials lacking molecular mobility, such as those with internal microstructures that impede dislocation motion, tend to be brittle rather than ductile. The stress-strain curve for brittle materials typically remains linear over the entire range of strain, leading to fracture without significant plastic flow.

It is worth noting in Fig. 5 that in ductile materials, the stress required to increase the strain beyond the proportional limit continues to rise. The ma terial demands progressively higher stress to continue straining, a phenomenon referred to as strain hardening. The microstructural rearrangements associated with plastic flow are usually irreversible when the load is removed. As a result, the proportional limit often coincides with or is close to the material's elastic limit. Elasticity refers to the property of complete and immediate recovery from an imposed displacement upon load removal, and the elastic limit is the stress value at which the material experiences permanent residual strain that is not lost upon unloading.

Another related term is the yield stress (denoted as σY in this module), which represents the stress required to induce plastic deformation in the specimen. Since determining the exact stress at which plastic deformation initiates can be challenging, the yield stress is often defined as the stress necessary to induce a specified amount of permanent strain, typically 0.2%. Fig. 5 demonstrates a construction used to find the "offset yield stress," where a line with a slope of E is drawn from the strain axis at ε = 0.2%. This line represents the unloading behaviour that would result in the specified permanent strain. The stress at the point of intersection with the σe-ε curve corresponds to the offset yield stress.

In Figure I.6, we observe the engineering stress-strain curve for copper with an enlarged scale, depicting strains from zero up to specimen fracture. It appears that the rate of strain hardening diminishes up to a point labeled UTS (Ultimate Tensile Strength, denoted as σf in this module). Beyond this point, the material seems to undergo strain softening, where each additional increment of strain requires less stress.

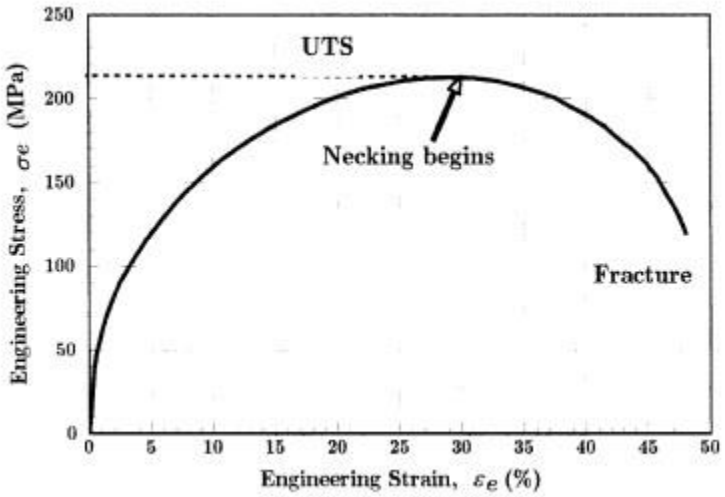


***Figure I.6*:** Full engineering stress-strain curve for annealed polycrystalline copper.

However, it is important to note that the apparent transition from strain hardening to strain softening and the maximum observed in the curve at the UTS are artifacts of the plotting procedure. Additionally, beyond the yield point, molecular flow causes a significant reduction in the specimen's cross-sectional area A, leading to the true stress $\sigma t = P/A$ being larger than the engineering stress computed from the original cross-sectional area ($\sigma e = P/A0$). As long as strain hardening increases σt sufficiently to compensate for the decreased area A, the load and engineering stress continue to rise with increasing strain. Eventually, the decrease in area due to flow surpasses the increase in true stress due to strain hardening, and the load begins to decrease. This geometric effect implies that if the true stress rather than the engineering stress were plotted, no maximum would be observed in the curve. At the UTS, the load P reaches a maximum, resulting in an analytical relationship between the true stress and the area at necking:

$$P = \sigma tA \rightarrow dP = 0 = \sigma tdA + Ad\sigma t \rightarrow -dA/A = d\sigma t/\sigma t \qquad (3)$$

The last expression indicates that the load and engineering stress reach a maximum as a function of strain when the fractional decrease in area equals the fractional increase in true stress. Although the UTS is commonly reported in tensile tests, it is not a direct measure of

the material's properties due to the influence of geometry, as discussed earlier. Thus, caution should be exercised when using the UTS as a design criterion. In designing with

ductile metals, the yield stress σY is usually preferred over the UTS. However, the UTS remains a valid design criterion for brittle materials that do not exhibit flow-induced reductions in cross-sectional area.

During the tensile test, the true stress is not uniformly distributed throughout the specimen, and there will always be a location, such as a nick or surface defect, where the local stress is maximum. Once the maximum point on the engineering curve is reached, the localized flow at that site cannot be compensated by further strain hardening, leading to a further reduction in the area. This intensifies the local stress, accelerating the flow even more. This localized and increasing flow eventually results in the formation of a "neck" in the gage length of the specimen, as shown in Figure I.7.

***Figure I.7*:** Necking in a tensile specimen.

After necking occurs, all subsequent deformation takes place within the necked region, while the rest of the specimen experiences limited deformation. The neck continues to decrease in size, causing an increase in local true stress until the specimen ultimately fails. This necking failure mode is common for most ductile metals. As the neck shrinks, the nonuniform geometry alters the stress state from uniaxial stress to a complex state involving shear components in addition to normal stresses. The final failure of the specimen often exhibits a "cup and cone" geometry, as depicted in Fig. 8, where the outer regions fail in shear, while the interior fails in tension.

Upon fracture, the engineering strain at break (denoted as εf) includes the deformation in both the necked and unnecked regions. Since the true strain in the neck is greater than that in the unnecked material, the value of εf depends on the fraction of the gage length that has necked. Therefore, εf is influenced by both the specimen's geometry and the material properties, making it only an approximate measure.

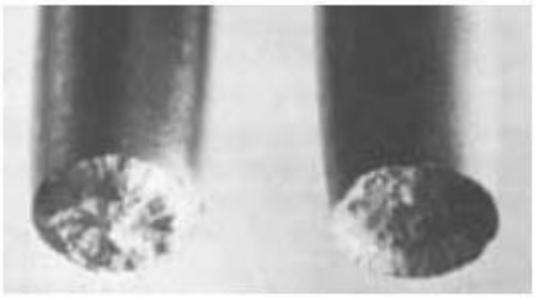


***Figure I.8*:** Cup-and-cone fracture in a ductile metal.

Figure I.9 presents the engineering stress-strain curve for a semi crystalline thermoplastic, which exhibits a similar response to copper as seen in Fig. 5. It shows a proportional limit followed by a maximum in the curve where necking occurs. In plastics, this maximum is commonly referred to as the yield stress, although plastic flow actually begins at earlier strains.

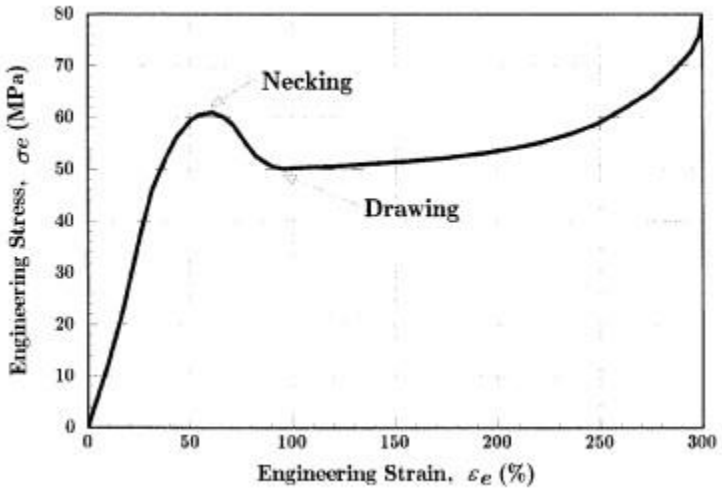


***Figure I.9*:** Stress-strain curve for polyamide (nylon) thermoplastic.

However, polymers differ significantly from copper in that the necking process does not continue until failure. Instead, the material in the neck stretches only up to a "natural draw ratio," which depends on temperature and specimen processing. Beyond this point, the material in the neck stops stretching, and new material at the neck shoulders down. This neck propagation continues until it spans the full gage length of the specimen, a process known as drawing. This drawing process can be observed without the need for a testing machine by stretching a polyethylene "six-pack holder," as shown in Figure I.10.

It is worth noting that not all polymers can sustain this drawing process, as it occurs when the necking process produces a strengthened microstructure with a breaking load greater than that needed to induce necking in the untransformed material just outside the neck.

***Figure I.10*:** Necking and drawing in a 6-pack holder.

I.9 True stress and strain measurements:

As discussed in the previous section, the engineering stress-strain curve must be interpreted with caution beyond the elastic limit, since the specimen dimensions experience substantial change from their original values, and obtaining a more accurate representation of material behaviour using true stress and strain measurements. We mention the following:

- To account for dimensional changes during plastic flow, true stress (σt) is preferable over engineering stress (σe), which doesn't consider these changes.
- True strain (εt) is calculated using the logarithmic strain measurement (dt = ln (L / L0)), where L and L0 represent current and original lengths.
- The volume constraint during plastic flow is expressed as AL = A0/L0, where A and A0 are current and original cross-sectional areas. The extension ratio (λ = L / L0) represents this constraint.

$$\sigma_t = \sigma_e (1 + \epsilon_e) = \sigma_e \lambda, \quad \epsilon_t = \ln (1 + \epsilon_e) = \ln \lambda$$

$$\frac{L}{L_0} = \frac{A}{A_0} \rightarrow \epsilon_t = \ln \frac{L}{L_0} = \ln \frac{A}{A_0}$$

- Equations connecting true and engineering stress and true strain facilitate the derivation of true stress-

strain curves from the engineering curves.

- Ductile metals often exhibit power-law behavior

$\sigma_t = A\epsilon_t^n \rightarrow \log \sigma_t = \log A + n \log \epsilon_t$ in the true stress-strain relationship, with the strain hardening parameter (n) indicating resistance to necking.

- The "Considère construction" graphical technique uses true stress (σt) and extension ratio (λ) to analyze yield characteristics, with different shapes of true stress-strain curves indicating distinct behaviours.

The three shapes observed are:

a) always concave upward, leading to fracture before yielding;
b) concave downward with one tangent, signifying a yield drop and necking; and
c) sigmoidal with two tangents, indicating initiation of yielding, necking, and drawing.

Semicrystalline polymers undergo microstructural transformations during strain, resulting in increased strain hardening rates, higher strengths, and stiffnesses, leading to a second tangent in the true stress-strain curve.

Overall, the paragraph emphasizes the importance of considering true stress and strain measurements and discusses their relevance in understanding material behaviour beyond the elastic limit and during plastic flow.

I.10 Area under stress-strain curve

The total mechanical energy per unit volume consumed by a material during straining up to a specific strain value is represented by the area under the engineering stress-strain curve. This can be demonstrated as follows:

$$U^* = \frac{1}{V}\int P\,dL = \int_0^L \frac{P}{A_0}\frac{dL}{L_0} = \int_0^\epsilon \sigma\,d\epsilon \quad (7)$$

In the absence of energy dissipation mechanisms like molecular slip; this mechanical energy is stored reversibly within the material as strain energy. When the stresses are within the elastic range, the strain energy is proportional to the triangular area under the stressstrain curve in Figure I.11.

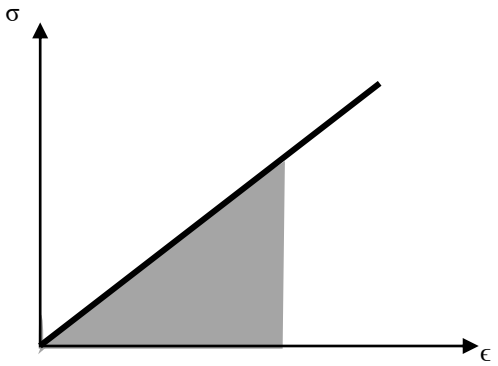


***Figure I.11*:** Strain energy = area under stress-strain curve.

It's important to note that the strain energy increases quadratically with stress or strain. As the strain increases, the energy stored per unit increment of strain grows as the square of the strain. This has practical implications, such as the design of archery bows. A real bow, when strung, stores substantial strain energy due to the bending of the initially straight piece of wood. This stored energy is then available to propel the arrow when the bow is further bent upon drawing. A curved piece of wood without the initial bending wouldn't store as much energy.

Figure I.12 schematically illustrates the amount of strain energy available for two equal increments of strain (Δ) applied at different existing strain levels. The area up to the yield point represents the modulus of resilience, while the total area up to fracture represents the modulus of toughness, as shown in Figure 13. The term "modulus" is used because strain energy per unit volume shares the same units as stress or modulus of elasticity (N-m/m3 or N/m^2)

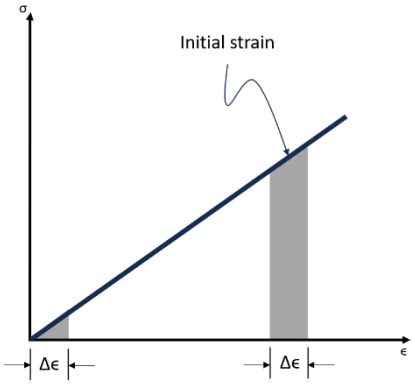


***Figure I.12*:** Energy associated with increments of strain

Material	Maximum Strain, %	Maximum Stress,MPa	Modulus of Toughness, MJ/m^3	Density kg/m^3	Max. Energy J/kg
Ancient Iron	**0.03**	**70**	**0.01**	**7,800**	**1.3**
Modern spring steel	**0.3**	**700**	**1.0**	**7,800**	**130**
Yew wood	**0.3**	**120**	**0.5**	**600**	**900**
Tendon	**8.0**	**70**	**2.8**	**1,100**	**2,500**
Rubber	**300**	**7**	**10.0**	**1,200**	**8,000**

***Table I.1*:** Energy absorption of various materials

The modulus of resilience refers to the material's ability to absorb energy without suffering damage until the point of yielding. The modulus of toughness represents the energy required to completely fracture the material. Materials with high moduli of toughness tend to exhibit good impact resistance.

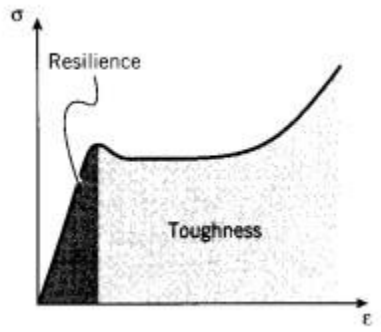


***Figure I.13*:** Moduli of resilience and toughness.

Table I.1 provides energy absorption values for various common materials. Natural and polymeric materials, in particular, can offer high-energy absorption per unit weight.

During loading, the area under the stress-strain curve represents the strain energy per unit volume absorbed by the material. Conversely, the area under the unloading curve represents the energy released by the material. In the elastic range, these areas are equal, resulting in no net energy absorption. However, if the material is loaded beyond the elastic range, as shown in Figure I.14, the absorbed energy exceeds the released energy, and the difference is dissipated as heat.

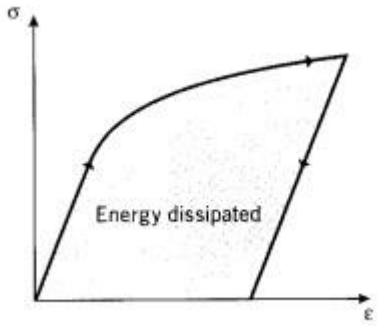


***Figure I.14*:** Energy loss = area under stress-strain loop.

I.11 Conclusion

Low-density polyethylene (LDPE) is extensively employed in plasticulture, particularly as agricultural greenhouse covers. It offers several advantages due to its ease of use and temperature resistance. Additionally, LDPE can be treated with various methods, such as anti-UV treatment, to enhance its properties. Its favorable attributes and cost-effectiveness make it a preferred choice in various applications. The present study aims to investigate the impact of different conditions on the structure and behavior of low-density polyethylenes and subsequently assess the long-term performance of this polymer.

CHAPTER II
Deep Learning

II.1 Introduction

AI stands for artificial intelligence, which refers to the development of computer systems capable of performing tasks that normally require human intelligence, such as visual perception, speech recognition, decision-making and language translation. Artificial intelligence is achieved through algorithms and statistical models designed to learn from data and make predictions or decisions based on that learning. This process is called machine learning, which involves training a computer system with large amounts of data so that it can learn from patterns and make predictions or decisions on its own. AI has a wide range of applications in various industries, including healthcare, finance, manufacturing, transportation and entertainment. Some examples of AI applications include self-driving cars, virtual personal assistants, facial recognition systems, and recommendation systems for e-commerce websites. Overall, AI has the potential to transform many industries and improve our lives in many ways, but it also raises important ethical and social issues that need to be addressed.

II.2 Artificial Intelligence

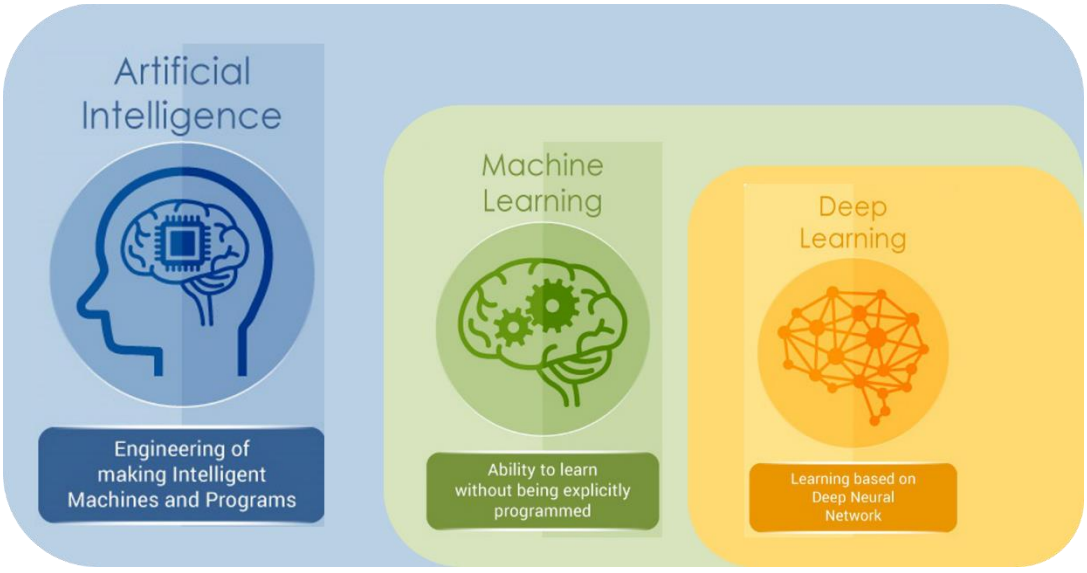


***Figure II.1*:** Deep Learning

"Artificial Intelligence or AI is the art of making computers think and behave in the same manner as human brain in order to solve more complex problems without the need of programmer's guide"

Artificial intelligence, commonly referred to as AI, is a field of computer science concerned with the development of intelligent machines capable of performing tasks that normally require human intelligence. These tasks range from understanding natural language to recognizing images and patterns to making decisions based on complex data. [18]

The history of artificial intelligence dates back to the mid-20th century, when computer scientists began exploring the concept of machine intelligence. In 1956, a group of researchers organized the Dartmouth Conference, where they floated the idea of creating a machine that could "think" like a human. The conference is widely considered the birthplace of artificial intelligence as a field of research.

Over the years, artificial intelligence has evolved from simple rule-based systems to more advanced algorithms that can learn and improve on their own. A major breakthrough came in the 1990s with the development of machine learning, a subfield of AI that focuses on creating algorithms that can learn from data.

Today, artificial intelligence is used in a variety of applications, from virtual assistants like Siri and Alexa, to self-driving cars, to medical diagnosis and treatment. As the technology continues to advance, there are concerns about its impact on society, particularly in terms of job losses and possible misuse of AI by governments and businesses

Despite these concerns, artificial intelligence has the potential to revolutionize many industries and improve our lives in countless ways. As the field continues to grow and evolve, it will be important for researchers, policymakers, and the public to work together to ensure that AI is used responsibly and ethically.

While symbolic AI has shown good improvements in solving well-defined logical problems, figuring out clear rules to solve complex problems like image classification or

language translation and speech recognition has been very difficult. This opens up a lot of room for a new approach called machine learning ML [19]

II.3 Distinction between AI, ML and DL

Some challenges experience exponential growth, such as processing time, which depends on the size of the given examples. In certain cases, the problem may be impossible to solve within a reasonable timeframe using human capabilities. To address this concern, various processing techniques have been developed to break down the problem into quantifiable sub-problems that can be solved.

There is often confusion in public discussions regarding the distinctions between "artificial intelligence," machine learning, and deep learning. However, these concepts are not equivalent but interconnected:

- Artificial intelligence is a broad term that encompasses machine learning, which, in turn, encompasses deep learning.
- Artificial intelligence can also encompass other types of software components, such as rule engines.[20]

II.4 Machine Learning

Learning is a concept to define the acquisition of knowledge and the reuse of this new knowledge. We learn from perceiving the environment with our five senses, from life experiences, and from repeating events with memory. For machines, since they are not endowed with senses or dynamic judgment, they obey instructions in a program with input data and an output or program response. [21]

Tom Mitchell, on the other hand, defined learning as a computer program that learns a specific task T from experiential data E, with a measure of performance P.

Machine Learning is currently a popular term that is widely discussed, and rightfully so, as it is one of the most captivating subfields within the realm of Computer Science.

Tasks related to machine learning focus on a practical definition rather than defining the field based on cognitive terms. This approach aligns with Alan Turing's proposition in his paper titled "Computing Machinery and Intelligence." In this paper, Turing replaces the question "Can machines think?" with the question "Can machines perform the tasks that we, as thinking entities, are capable of?"

By asking this question, a new programming paradigm emerges. In traditional programming, known as symbolic AI, humans provide rules (a program) and data to be processed according to these rules, resulting in answers. In contrast, machine learning involves humans providing both data and the expected answers derived from that data, resulting in the discovery of rules (a program). These rules can subsequently be applied to new data to generate original answers. [22]

II.4.1 Types of machine learning

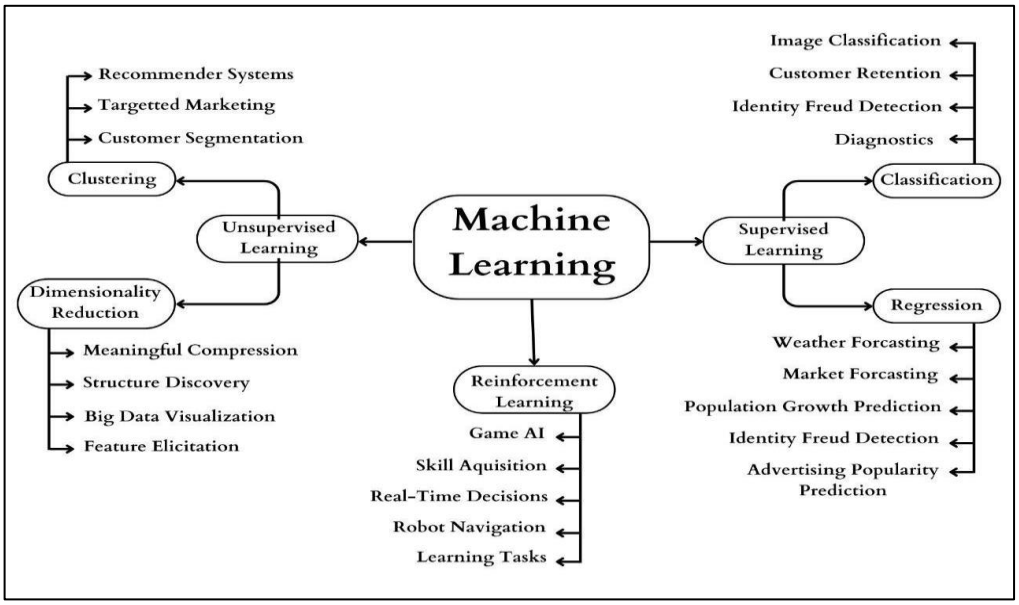


***Figure II.2*:** Machine Learning Approaches with Algorithm

There exist various types of machine learning algorithms, which include supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

A) Supervised learning

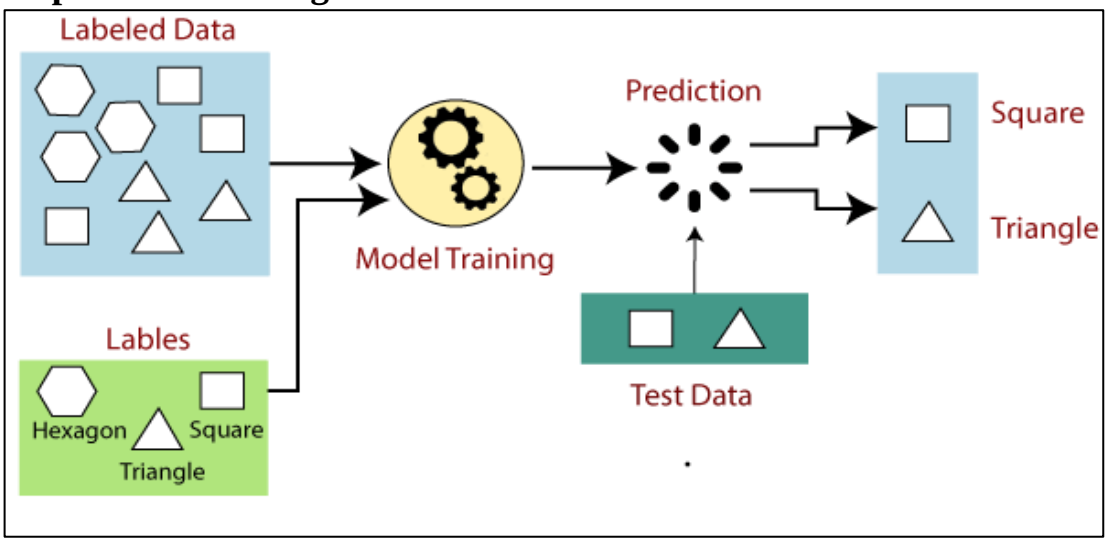


Figure II.3 : Supervised learning example

Supervised learning is synonymous with classification. Learning guidance is based on labeled examples from the educational dataset. For example, in the zip code recognition problem, a set of handwritten zip code images and their corresponding machine-readable translations are used as training examples to guide the learning of the classification model.
It can be grouped into two types:

- **Classification**

Classification is the main task of machine learning, which aims to classify input data into different classes or categories. This technique is widely used in many applications including image recognition, text classification, speech recognition and many other applications
The main purpose of a classification algorithm is to identify the class of a given data set, and these algorithms are mainly used to predict the output of categorical data.
Some of the most important classification models:

- Logistic regression: Logistic regression is a simple and widely used classification model. It estimates the probability of a binary outcome given the input characteristics using a logistic function. Despite its name, logistic regression is

actually a classification algorithm. 2. Naive Bayes: Naive Bayes is a probabilistic classifier that applies Bayes' theorem by assuming independence of features. It is known for its simplicity and efficiency, making it particularly useful for text classification tasks.

- Decision trees: Decision trees are versatile classifiers that recursively partition data based on various characteristics, creating a tree-like decision model. Each internal node represents an attribute or attribute, and each leaf node represents a class entry or decision.
- Support Vector Machines (SVM): SVM is a powerful classification algorithm that finds a hyperplane in a high-dimensional space to separate classes. This maximizes the margin between classes, which helps generalize to unseen data. SVM can also handle non-linear decision constraints using the kernel trick.
- k-Nearest Neighbors (k-NN): k-NN is a non-parametric classification algorithm that classifies new instances based on most of their k nearest neighbors in the feature space. It is a simple but powerful algorithm that does not require special training.

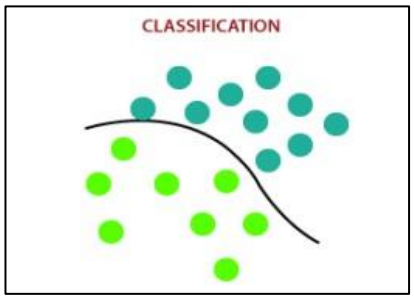


***Figure II.4*:** Example of Classification

❖ **Regression**

Regression models predict a continuous output value based on an input independent variable. This technique is used when the predicted output variable must be a continuous value, such as for weather forecasting or market trends. Different regression models exist and they vary depending on the relationship between the dependent and independent variables being considered and the number of independent variables used in the model.

Some of the most important regression models :

- Linear Regression: This is the simplest and most commonly used regression algorithm. It assumes a linear relationship between the dependent variable and the independent variables.
- Poisson regression: it is a statistical model used to analyze census data, where the outcome variable represents the frequency of occurrence of an event during a specified time period. This is particularly useful when the data follows a Poisson distribution characterized by a discrete set of non-negative values. In this regression framework, the goal is to estimate the relationship between the predictor variables and the expected value of the outcome variable, which is assumed to be linearly related to the predictors through a logarithmic link function.

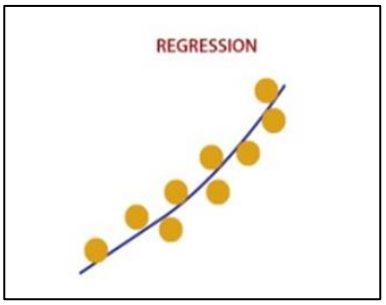


***Figure II.5*:** Example of Regression

B) Unsupervised Learning

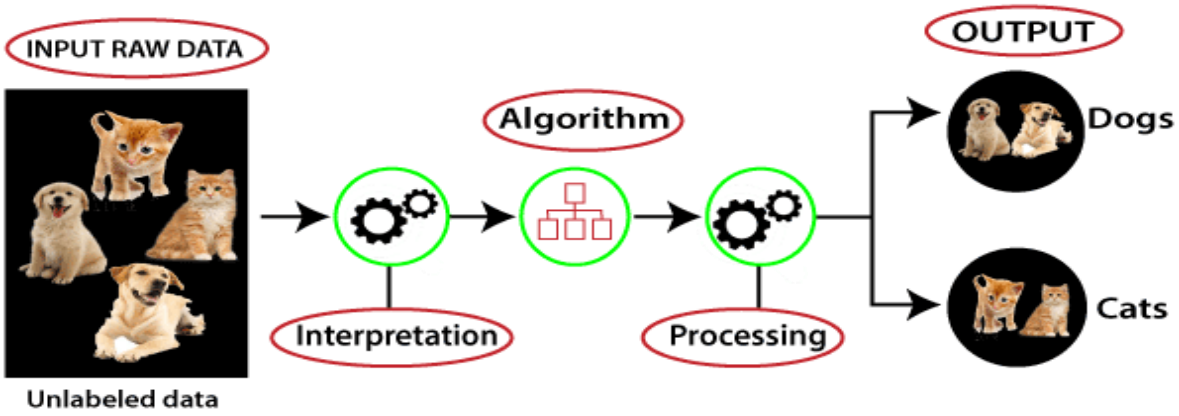


***Figure II.6*:** Unsupervised learning example

Unsupervised learning is a type of machine learning in which an algorithm learns patterns and relationships in data without explicit identifiers or instructions. The algorithm is given a dataset and tasked with finding structure or underlying patterns on its own, without any specific objective or goal. This can be useful for tasks such as clustering, dimensionality reduction, and anomaly detection.

- **Clustering (or data separation)**

This unsupervised classification method combines a set of learning algorithms that aim to group unlabeled data with similar properties. Thus, the separation of schemes or families also prepares the ground for the subsequent use of supervised learning algorithms (eg, KNN).

Clustering is especially used when data labeling is expensive. However, this is a mathematically ill-defined problem: different measures and/or different representations of the data lead to different groupings that are not necessarily better than others. Thus, the clustering method must be carefully chosen according to the expected result and the intended use of the data.

- **Association Rule Learning**

Association rule learning focuses on finding interesting relationships or associations between variables in large datasets. The purpose of these algorithms is to find co-occurrence patterns, dependencies or rules between objects. The Apriori algorithm is a well-known technique used for association rule learning.

- **Dimensionality Reduction**

Dimensionality reduction techniques aim to reduce the number of features or variables in the data while preserving the most significant information. These techniques help visualize high-dimensional data and extract meaningful representations. Common dimensionality reduction algorithms include Principal Component Analysis (PCA) and tdistributed Stochastic Neighbor Enrollment (t-SNE).

C) Semi-Supervised Learning

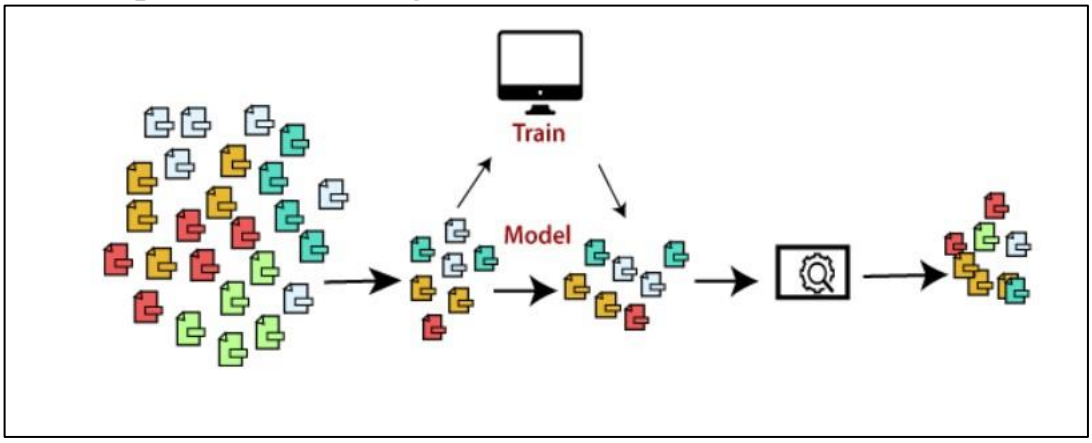


***Figure II.7*:** Semi-supervised learning example

Semi-supervised learning is a type of machine learning that combines labeled and unlabeled data to improve performance. The algorithm is trained on a small amount of labeled data and a large amount of unlabeled data. Labeled data is used to guide the learning process, while unlabeled data is used to discover hidden patterns and structures Some of the application areas where semi-supervised learning is used are machine translation, tag information, fraud detection and text classification

D) Reinforcement Learning

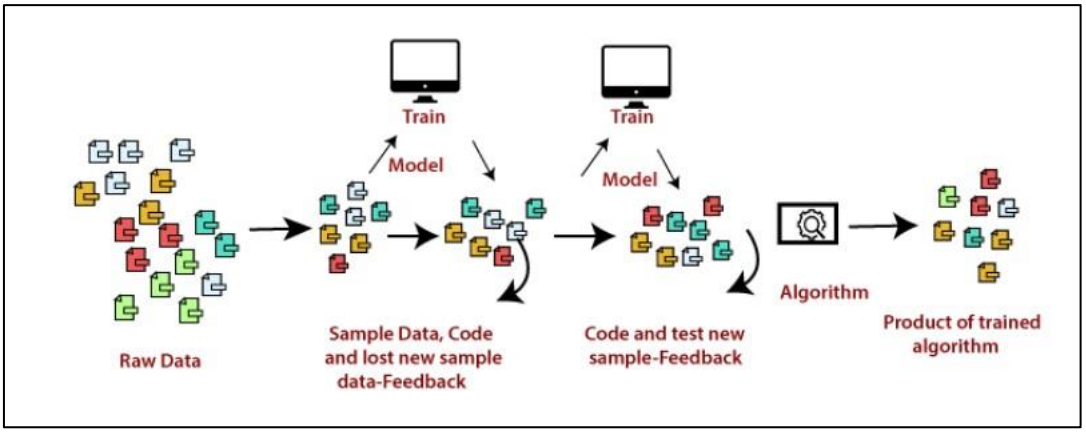


***Figure II.8*:** Reinforcement learning

Reinforcement learning is a problem faced by an agent that needs to learn behavior while interacting with a dynamic environment. The work described here has a strong

family resemblance to the psychological work of the same name, but it changes significantly in details and in the reinforcement of the word used.The agent acts in the environment and receives feedback in the form of rewards or punishments as a result of its actions. The goal of the agent is to learn a policy that maps states to actions so that the expected cumulative reward over time can be maximized

II.5 Real-world applications

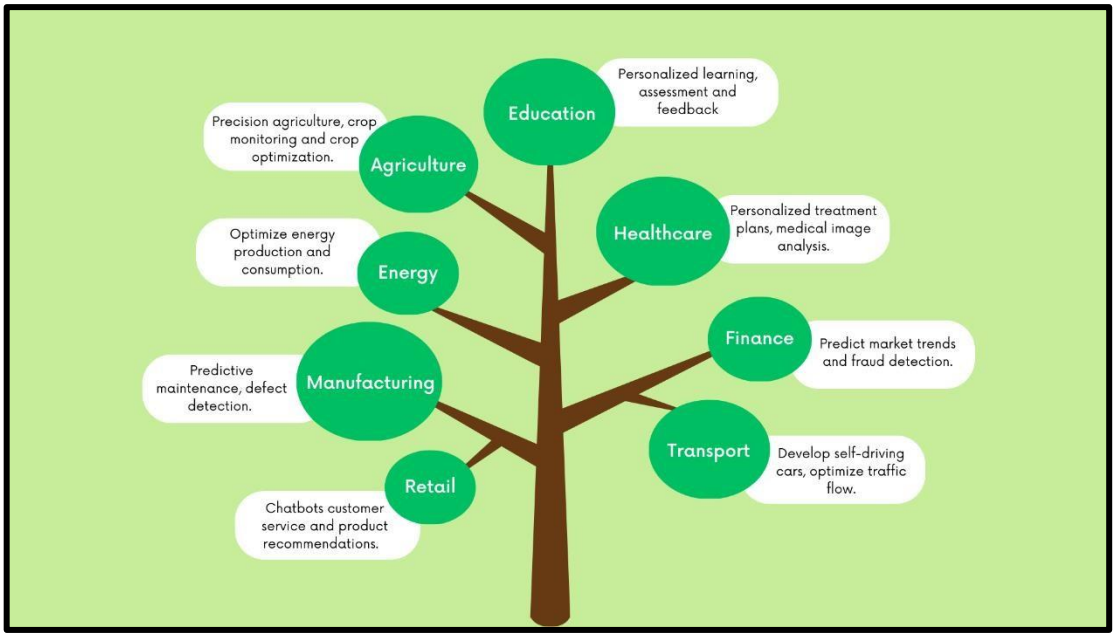


***Figure II.9*:** Real-world applications

Artificial intelligence has many real-world applications in many industries. Here are some examples:

- **Healthcare:** Artificial intelligence is used to develop personalized treatment plans for patients based on their medical history, genetics and other factors. It is also used for early detection and diagnosis of diseases, drug development and medical image analysis.

- **Finance :** Artificial intelligence is used to detect fraud, predict market trends and automate financial processes such as loan approval and risk assessment.

- **Transport :** Artificial intelligence is used to develop self-driving cars, optimize traffic flow and improve transport infrastructure.
- **Retail :** Artificial intelligence is used for product recommendations, inventory tracking and customer service through chatbots.
- **Manufacturing :** Artificial intelligence is used for predictive maintenance, defect detection and quality control in manufacturing processes.
- **Energy :** Artificial intelligence is used to optimize energy production and consumption, monitor and manage energy networks and reduce energy losses.
- **Agriculture :** Artificial intelligence is used in precision agriculture, crop monitoring and crop optimization.
- **Education :** AI is used for personalized learning, assessment and feedback

II.6 DEEP LEARNING

Deep learning has seen big success in latest time with applications like speech recognition, image processing, language translation, and listing is going on. Deep neural networks in general refer to neural networks with many layers and large number of neurons, frequently layered in a manner this is usually not area specific. Availability of compute power and large quantity of data has made those massive structures very effective in learning hidden capabilities along with facts patterns. [23]

II.6.1 Artificial Neural Network

Artificial Neural Networks (ANNs) take inspiration from the structure of biological neural networks. They are highly parallel computing systems consisting of a large number of interconnected processors. The objective of ANN models is to mimic the organizational principles found in the human brain. Specifically, one type of network views its nodes as 'artificial neurons', computational models that imitate natural neurons. ANNs primarily focus on information processing and find applications in various fields related to this domain.

There is a diverse range of ANNs used for different purposes. Some are designed to simulate real neural networks, allowing researchers to study animal and machine behavior and control. Others serve practical engineering tasks such as pattern recognition, forecasting, and data compression. These networks operate through the interaction of inputs, similar to synapses, which are multiplied by associated weights. These weights represent the flow of information between nodes and are computed using mathematical functions that determine neuron activation. Additionally, an output function, often involving a threshold, calculates the final output of the artificial neuron. In this network, neurons simply sum their inputs. Input neurons, having only one input, produce an output equal to the received input multiplied by a specific weight. [24]

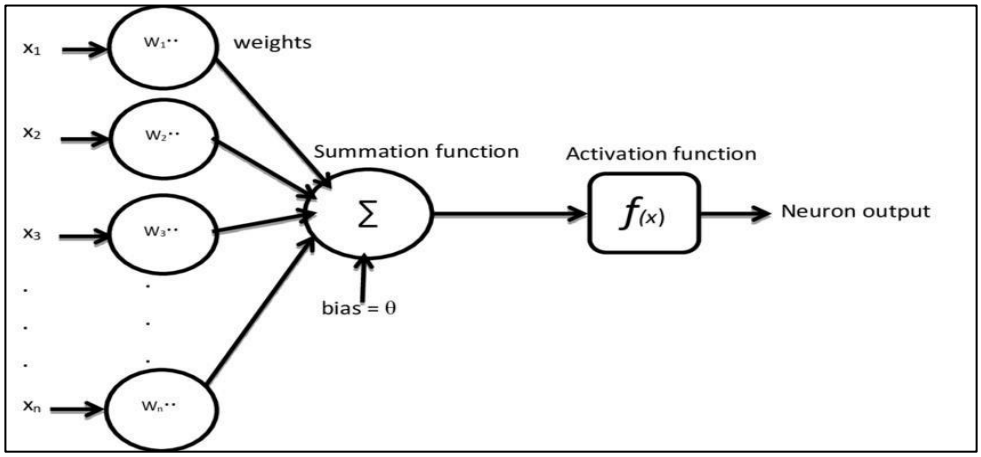
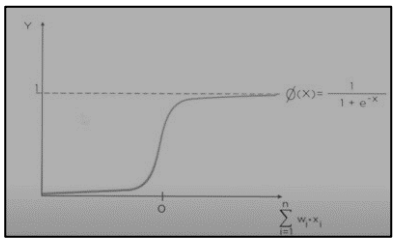
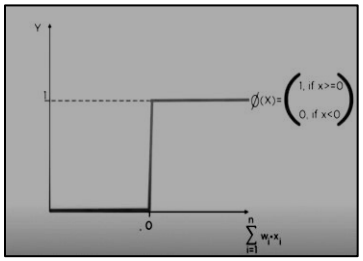
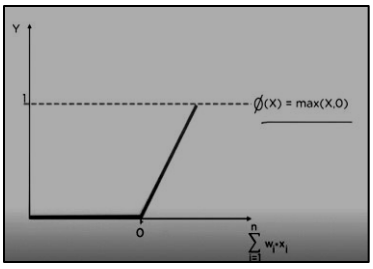


***Figure II.10*:** Artificial Neuron

When the weight is set to a higher value, it signifies a stronger input. The adjustment of weights in an artificial neuron allows us to achieve desired outputs for specific inputs. However, when dealing with ANNs comprising hundreds or thousands of neurons, manually determining all the required weights becomes extremely complex. Fortunately, there are algorithms available that can automatically adjust the weights of the ANN to achieve the desired output from the network. This process of weight adjustment is known as learning or training. During training, the weights are initially assigned random values, and the objective is to fine-tune them to minimize errors. [24]

II.6.2 Activation Function

An activation function is a mathematical function applied to the output of a neuron in a neural network. It introduces non-linearity into the model, allowing neural networks to approximate complex non-linear functions, which makes them powerful for various machine learning tasks. Activation functions play a crucial role in determining the output of a neuron and the overall performance of a neural network. Some common activation functions used in neural networks are:

Sigmoid Function	Used for models where we have to predict the probability as output It exists between 0 and 1	
Threshold Function	It is a threshold based activation function If Y value is greater than certain value, the function is activated and firef else not	
ReLU Function	It is the most widely used Activation function and gives an output of X If X is positive and 0 otherwise	

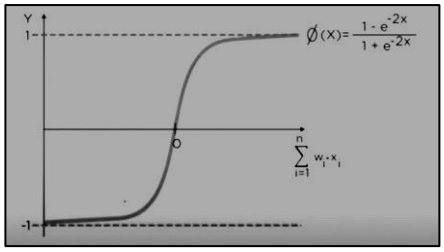
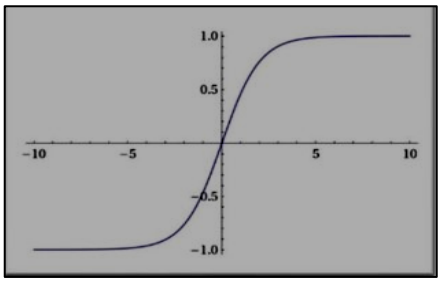
Hyperbolic Tangent Function	This function is similar to sigmoid and is bound to range (-1, 1)	
Softmax	The Softmax function is often used as an activation function in the output layer of a neural network in multi-class classification tasks. It maps the output values to a probability distribution between classes such that the sum of all class probabilities is equal to 1	

***Table II.1*:** Activation Function

II.6.3 LOSS FUNCTIONS

Neural network loss functions play a crucial role in training and optimization of neural networks. The loss function measures the discrepancy between the predicted output of a neural network and the actual target output. By minimizing this discrepancy, the neural network learns to make more accurate predictions.

	Formula	Application	Example
Mean Squared Error (MSE)	**MSE** = (1/n) * ∑(y_true - y_pred)^2	Regression problems where the magnitude of errors is important.	predicting housing prices
Binary CrossEntropy	**BCE** = -(y_true * log(y_pred) + (1 - y_true) * log(1 - y_pred))	Binary classification problems where each sample belongs to one of two classes.	email spam classification
Categorical Cross-Entropy	**CCE** = -∑(y_true * log(y_pred))	Multi-class classification problems where each sample belongs to one of more than two classes.	image recognition with multiple categories
Sparse Categorical Cross-Entropy	**SCCE** = -∑(y_true * log(y_pred))	Similar to categorical crossentropy, but suitable when the true class labels are integers instead of one-hot encoded vectors.	image recognition with multiple categories
Kullback-Leibler Divergence (KL Divergence)	**KL** = ∑(y_true * log(y_true / y_pred))	Used in scenarios where measuring the difference between two probability distributions is required.	variational autoencoders
Log-cosh Loss	**Log-cosh** = log(cosh(y_true - y_pred))	Similar to mean squared error, but with better handling of outliers due to the logarithmic and hyperbolic cosine function.	regression tasks

Table II.2 : Loss Functions

II.6.4 Deep Learning Technique

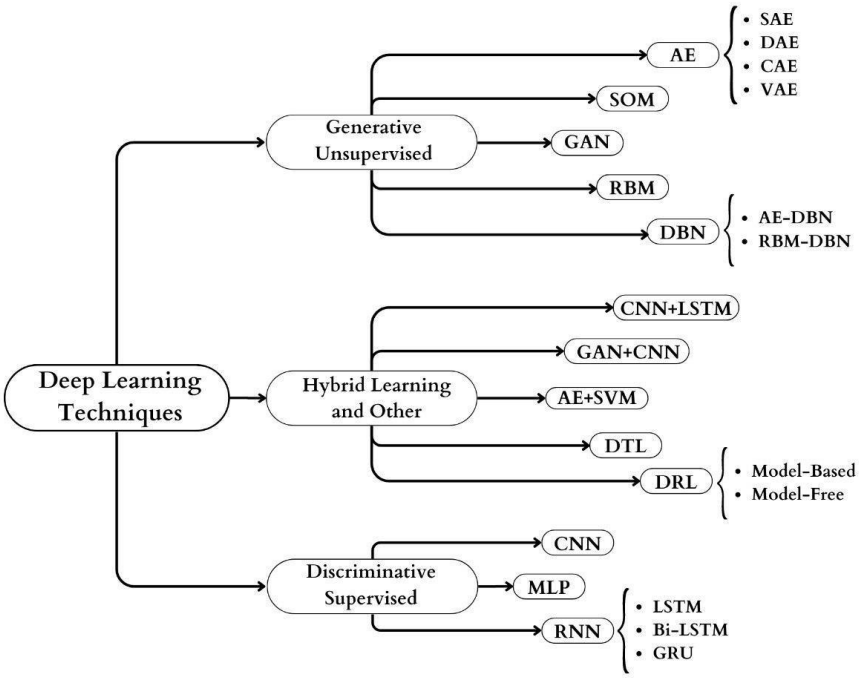


***Figure II.11*:** Deep learning technique

Deep learning techniques are a class of machine learning methods that aim to emulate the functionality of the human brain. These techniques involve constructing models that can perform various tasks, such as classification, using data from different domains like text, images, and sounds. The key building blocks of these models are neural networks, which consist of multiple layers, including hidden layers.

Neural networks are composed of interconnected nodes, or neurons, organized in layers. Each layer, from the left most to the rightmost, progressively extracts and learns increasingly abstract and high-level features from the input data. Initially, the lower layers capture low-level features such as edges, gradients, or textures. As information flows through the network, higher layers combine these low-level features to form more complex representations, eventually leading to accurate predictions or classifications.

Deep learning models leverage the hierarchical structure of neural networks to automatically learn relevant features directly from raw data. By iteratively adjusting the

weights and biases associated with each neuron, these models optimize their performance through a process known as training or learning.

This learning process involves minimizing a predefined loss function, which quantifies the discrepancy between the predicted outputs and the ground truth. The ability of deep learning models to autonomously discover and exploit intricate patterns in data has led to remarkable advancements in various domains, including computer vision, natural language processing, and speech recognition. These models have shown tremendous success in tasks such as image classification, object detection, sentiment analysis, language translation, and speech synthesis, among others.

II.6.5 Deep learning models

- **Recurrent Neural Networks (RNNs):** RNNs are designed to process data sequences with a temporal component, such as time series data or text. They have chains that allow information to persist over time, allowing them to capture sequential dependencies. Popular variants of RNNs include Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, which solve the vanishing problem associated with traditional RNNs.
- **Generative Adversarial Networks (GANs):** GANs consist of two neural networks, a generator and a discriminator, which are trained together in a process called adversarial training. A generator learns to create realistic data, such as images or text, while a discriminator learns to distinguish between real and generated data.

  GANs are used for tasks such as image synthesis, text generation, and video generation.
- **Autoencoders :** Autoencoders are a type of neural network used for unsupervised learning that aims to reconstruct the input data. They consist of an encoder that maps the input data to a lower-dimensional representation and a decoder that reconstructs the input data from the lower-dimensional representation.

Autoencoders are used for tasks such as image denoising, anomaly detection and data compression.

- **Reinforcement Learning Networks :** Reinforcement learning (RL) networks are used in machine learning, called reinforcement learning, where an agent learns to make decisions by acting on the environment to maximize a reward signal. RL networks typically consist of a policy network that selects actions and a value network that evaluates the value of various states or actions. RL networks are used for example in gaming, robotics and recommendation systems.
- **Convolutional Neural Networks (CNN):** which we will discuss in the next heading.

II.6.6 Convolutional Neural Network [25]

In recent years, Convolutional Neural Networks (CNNs) have become crucial and advanced deep learning algorithms. CNNs are structured as feedforward neural networks with multiple layers, including one or more convolutional layers, which are inspired by the organization of the visual cortex in animals. The name "convolutional" comes from the mathematical operation of convolution used in these networks, which is similar to matrix multiplication.

Unlike traditional fully connected networks, CNNs exhibit local connectivity, meaning that neurons in CNNs have sparse interactions and do not need to be connected to all neurons in the previous layer. This characteristic allows CNNs to efficiently process data in the form of arrays. For instance, 3D CNNs excel at handling 3D data like videos or volumetric images, while 2D CNNs are designed for 2D data such as images or audio spectrograms. Similarly, 1D CNNs are specifically tailored for 1D data, including signals and sequences.

The 1D CNNs mentioned here are newer versions of the well-known 2D CNNs, introduced only a few years ago. They have rapidly achieved state-of-the-art performance in various applications, including cardiac arrhythmia classification, electrical motor fault detection, wind prediction, and acoustic waste sorting.

Typically, a CNN structure consists of alternating convolution and pooling layers, with the final output provided by a fully connected layer. This arrangement is illustrated in figure

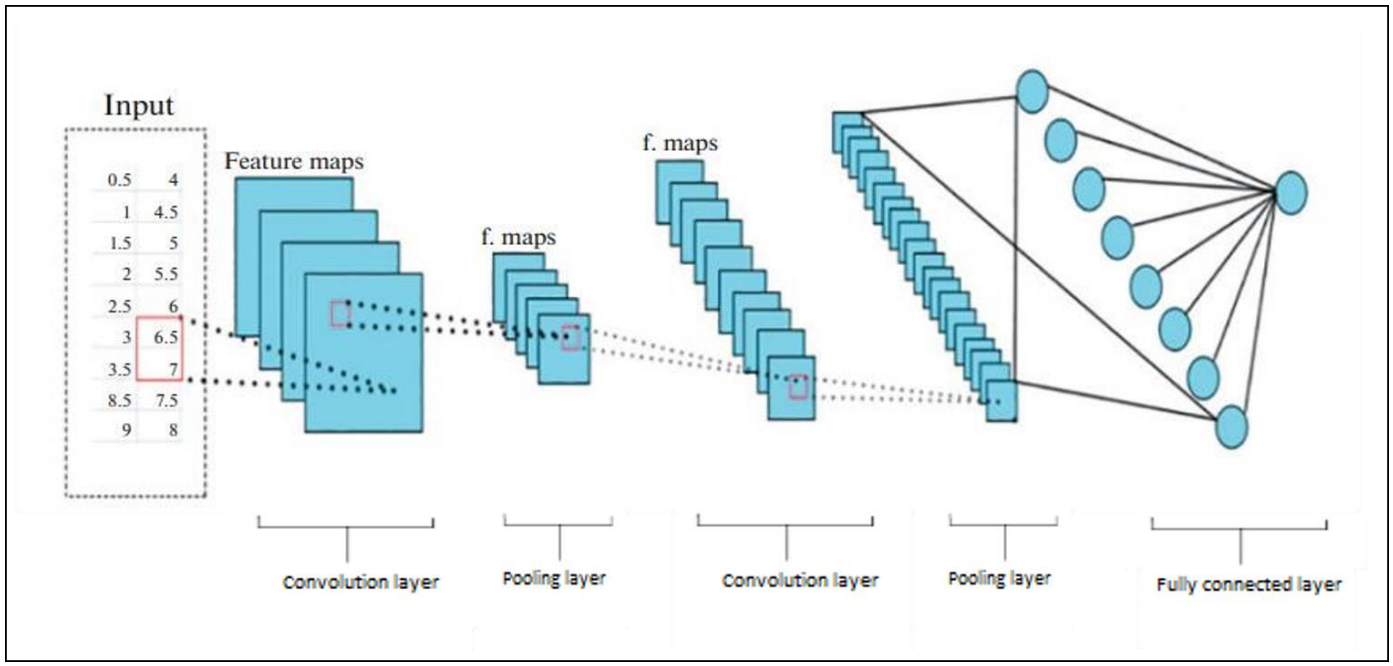


Figure II.12: An example of a convolutional neural . [25]

- **Convolution layer**

The convolution operation is a fundamental operation in convolutional neural networks (CNNs). It involves applying a small square matrix of numbers, called a kernel or filter, to an input matrix of numbers, known as a tensor. The kernel is systematically moved across the input, and at each position, a Hadamard product (element-wise multiplication) is performed between the kernel and the corresponding portion of the input. The resulting products are then summed to obtain the output value at the corresponding position of the target tensor. To illustrate this process, let's consider a two-dimensional input data, denoted as I, with dimensions p*q, a two-dimensional kernel, denoted as K, with dimensions n*n. The discrete convolution, often represented by an asterisk symbol, is calculated using the following formula:

$$V(i,j) = (I * K)(i,j) = \sum_{x=0}^{n}\sum_{y=0}^{n} I(i+x, j+y)K(x,y), (6)$$

Here, V(i, j) represents the output value at the corresponding position (i, j) in the target tensor. In summary, the convolution operation in a CNN involves applying a kernel to an input tensor, calculating the Hadamard product and summing the results at each position, resulting in an output value in the corresponding position of the target tensor.

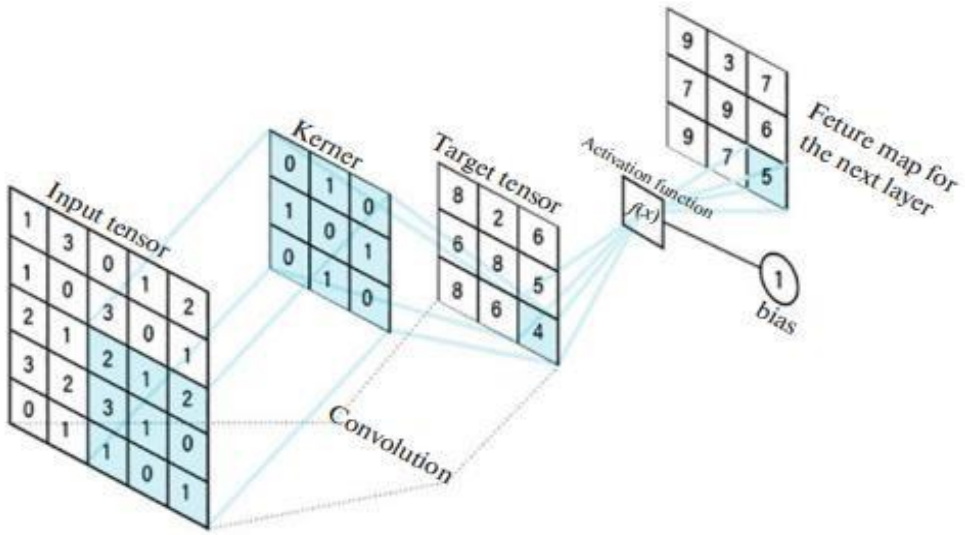


***Figure II.13*:** convolution layer neuron and the discrete convolution. [25]

A convolutional layer in a neural network consists of a set of kernels that are learned during training. Neurons within the same layer and sharing the same feature map also share the same kernel. Each neuron in the convolutional layer performs a discrete convolution between the input and the corresponding kernel to generate its input feature map. This input feature map is then passed through a nonlinear activation function (denoted as f) to produce the output feature map for that neuron. By stacking the feature maps of all kernels along the depth dimension, the output volume of the convolutional layer is obtained. The main objective of the convolutional layer is to identify local intersections of features from the previous layer. In addition, biases can be incorporated in

the forward operations of the convolutional layer, similar to other neural networks. Each unique kernel in the convolutional layer has its own associated bias, denoted as bij.

Therefore, for the jth feature map in the ith layer of the convolutional neural network, the value at the xth row and yth column, denoted as $V_{ij}^{x,y}$ is computed using the following formula:

$$V\_ij\hat{}(x,y) = f(\, b_{ij} \, + \sum_m \sum_{p=0}^{P_i} w_{ijm}^{p} \, v_{(i-1)m}^{x+p,y}) \qquad (7)$$

Here, m indexes over the set of feature maps in the (i-1)th layer that are connected to the current feature map. $wijmp$ represents the value at position p of the convolutional kernel, and Pi is the length of the convolutional kernel. The activation function f is applied to the sum of the convolutional operation and the corresponding bias value bij.

- **Pooling layer**

The pooling layer, also known as a subsampling layer, creates its own feature map by applying a pooling operator to aggregate information within small regions of the input feature maps and downsampling the results. The purpose of the pooling layer is to summarize the input information and reduce the spatial dimensions of the feature maps. It is important to note that pooling layers do not have any learnable parameters. One of the most widely used pooling operations is max pooling. In max pooling, the maximal values within each rectangular neighborhood of each point (i, j) are reported. The computation can be expressed using the following formula:

$$V_{ij}^{x,y} = \; max_{\,1 \le q \le Qi} (v_{(i-1)j}^{x+q,y}) \qquad (8)$$

Here, Qi represents the length of the pooling region, which determines the size of the neighborhood. The most common form of max pooling uses a stride of 2 and a pool size of 2. This corresponds to dividing the feature map into a regular grid of squares with a side length of 2 and taking the maximum value within each block for each input feature.

Overall, the pooling layer aggregates information through pooling operations and reduces the spatial dimensions of the feature maps, typically using max pooling with a stride of 2 and a pool size of 2.

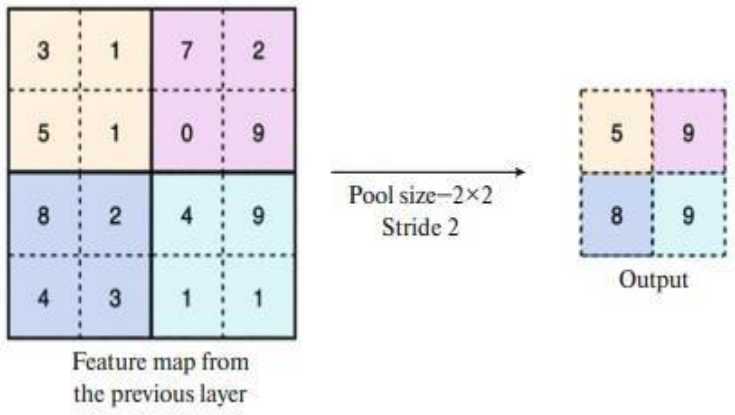


***Figure II.14*:** feature map and the output. [25]

- **Fully connected layer**

The fully connected layer, also known as the dense layer, is an essential component that takes the extracted information from previous layers and transforms it into the final output, typically used for tasks like classification. The structure of the dense layer resembles that of a regular multi-layer perceptron (MLP) layer found in traditional neural networks.

Before passing the data to the dense layer, a process called flattening is commonly applied. Flattening involves converting the output feature maps obtained from the preceding convolutional or pooling layers into a one-dimensional array of numbers, effectively creating a vector. This transformation allows the information to be represented in a sequential format, which can then serve as the input to the dense layer.

By flattening the output feature maps, the spatial structure and organization of the data are lost, but the important extracted features are retained. The resulting one-dimensional

vector can be easily connected to the neurons in the dense layer. Each element in the flattened vector is connected to every neuron in the dense layer, establishing a fully connected network structure.

In the dense layer, computations are performed on the input vector, utilizing weights and biases associated with each neuron. These computations produce the final output, which can be utilized for various purposes, such as classification tasks. In summary, the fully connected layer in a convolutional neural network (CNN) converts the extracted information from previous layers into the desired output. The process of flattening converts the output feature maps from convolutional or pooling layers into a one-dimensional vector, allowing seamless connectivity between the flattened input and the dense layer.

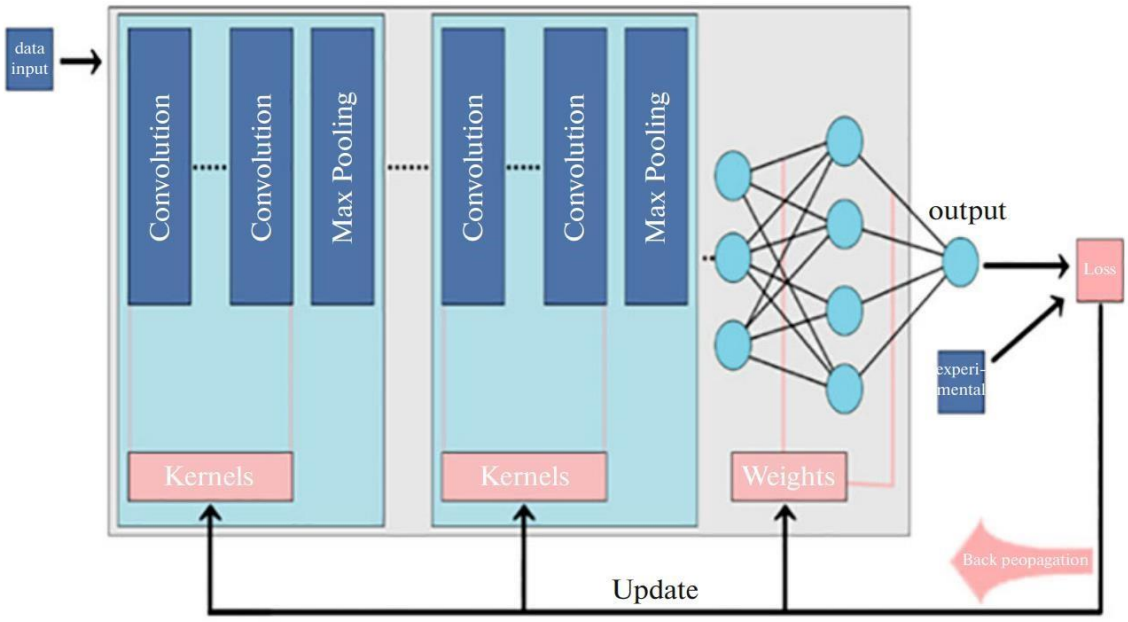


***Figure II.15*:** The fully connected layer. [25]

II.6.7 Training

- **Optimizers**

Optimizers are algorithms utilized in machine learning model training to minimize the loss or error function. They dictate how the model's parameters are updated throughout training to achieve optimal values. Commonly used optimizers include Stochastic Gradient Descent (SGD), Adam, RMSprop, and Adagrad. Each optimizer possesses its own advantages and can yield different performance outcomes depending on the specific problem.

- **Learning rate schedule**

The learning rate schedule pertains to the strategy employed to adjust the learning rate during the training process. The learning rate determines the rate at which the model's parameters are updated or the step size. A learning rate schedule can be either static, where the learning rate remains constant throughout training, or dynamic, where the learning rate is altered based on predefined rules. Typical techniques for learning rate scheduling include step decay, exponential decay, and cyclical learning rates.

- **Learning rate schedule**

Batch normalization is a technique used to normalize the activations of each layer in a neural network. It helps in stabilizing the learning process by reducing the internal covariate shift, which refers to the change in the distribution of layer inputs during training. By normalizing the inputs, batch normalization accelerates training, allows for higher learning rates, and reduces the dependence on initialization.

- **Batch size effects**

The choice of batch size in training has several effects on the training process. Batch size refers to the number of training examples used in each iteration of a training step. A larger batch size can result in faster convergence and better utilization of parallel processing capabilities, but it may require more memory. Smaller batch sizes provide a

more noisy estimate of the gradient but can lead to improved generalization and exploration of the parameter space.

- **Regularization**

Regularization is a technique implemented to prevent overfitting in machine learning models. It introduces a penalty term to the loss function to discourage the model from closely fitting the training data. The regularization term aids in controlling the model's complexity and reducing the impact of noisy or irrelevant features. Common regularization techniques include L1 regularization (Lasso), L2 regularization (Ridge), and dropout.

- **Multitask learning**

Multitask learning involves training a machine learning model to perform multiple tasks simultaneously. Instead of training separate models for each task, multitask learning leverages shared information among related tasks to enhance generalization and overall performance. By jointly optimizing multiple tasks, the model can learn more robust and transferable representations.

- **Transfer learning**

Transfer learning is a technique where knowledge acquired from training one model on a specific task is transferred or applied to a different but related task. Instead of starting the training process from scratch, transfer learning utilizes the learned features of a pre-trained model as a foundation for the new task. This approach is particularly valuable when the new task has limited labeled data, as it enables the model to benefit from the generalization learned on a larger, pre-existing dataset.

- **Learning rate schedule**

Curriculum learning is a training strategy inspired by human learning. Instead of randomly presenting training examples to the model, curriculum learning involves presenting the examples in a meaningful order or curriculum. The curriculum begins with easier examples and gradually increases the difficulty as the model learns. By providing a

curriculum, the model can build upon simpler concepts before tackling more complex ones, potentially improving learning efficiency and convergence.

II.8 Conclusion

In this chapter, we introduced the groundbreaking technique of deep learning, along with its architectures and methods. We initiated a conceptual study of the architecture of the convolutional neural networks we employed, exploring their various stages and highlighting their usefulness in regression.

CHAPTER III

Implementation

III.1 Introduction

In this chapter, we embark on the realization and implementation phase of our solution. We commence by introducing the software and external libraries utilized in our project. Subsequently, we provide a detailed description of the architecture of the deep learning model, specifically the convolutional neural networks, employed for predicting traction curves.

The objective of our work is to characterize and forecast the mechanical aging behavior of multilayer LDPE films using deep learning techniques, particularly CNN.

Before delving into the implementation details, we will present the essential tools that were employed in our study.

III.2 Technical choices

In order to carry out our project we used as development tools the Pycharm IDE to develop our application with the Python language.

III.2.1 Python

To develop our application, we opted for the programming language Python v3.9.64bit. Python is an interpreter, object-oriented, programming language high level with dynamic semantics. Is a simple language, easy to learn and allows a good reduction in the cost of code maintenance, Libraries (packages) python encourage code modularity and reusability. Contrary to many other languages, it does not use braces to delimit blocks, and semicolons after are optional. It has fewer syntactic exceptions and special cases than C or Pascal. [26]

III.2.2 Anaconda

Anaconda is a free and open source distribution of the programming languages. Python and R programming for scientific computing (data computing, machine learning and deep learning applications with scikit-learn, TensorFlow and Theano, large-scale data processing and performance with Dask, NumPy, pandas and Numba., predictive analysis, etc.), which aims to simplify the management and package deployment. Package versions are managed by the package management

system. Package conda. The Anaconda distribution 1400 popular computer data packages suitable for Windows, Linux and MacOS. [27]

III.2.3. Pycharm

It is an integrated development environment (IDE) used in computer programming, especially for the Python language. It provides code analysis, a graphical debugger, an integrated unit tester, and supports web development with Django as well as Data Science with Anaconda. PyCharm is cross-platform, with Windows, macOS, and Linux versions. Community Edition is distributed under the Apache License. [28]

In the next sections we will present the libraries and the main tools used in this development.

III.2.4.Additional Libraries

In order to achieve the goals of this project, we used other external libraries to perform some specific tasks. In this part, we will present the main libraries used. in addition to those provided by the Python standard library.

- **Scikit-Learn**

Scikit-learn [29] is a free open-source library for Python dedicated to machine learning, this tool is simple and efficient for data mining (Data Mining) and data analysis (Data Analysis). It notably features various classification, regression and clustering algorithms, including support vector machines (SVM: support vector machines), random decision forests classifier, gradient boosting classifier, k-means etc... and many other useful functions for analysis and data preprocessing. This Library is designed to harmonize with the numerical and scientific libraries of Python namely NumPy and SciPy.

We used to divide the dataset into two parts, one dedicated to training and the other to testing.

- **Keras**

Keras [30] is a high-level neural network API, written in Python and able to run on a TensorFlow backend to simplify the build process deep learning applications. It is designed to allow easy and fast

experimentation with deep neural networks, it focuses on modularity, scalability and user-friendly application design. Keras offers simple and consistent APIs, and minimizes the number of user actions required for cases of common use, and it provides clear and practical feedback on user errors. It was developed as part of the ONEIROS (Open-ended Neuro Electronic Intelligent Robot Operating System) project at Google. [31]

We used Keras to create the convolutional neural network, and train this model on an environment with CPU support.

- **Tensorflow**

TensorFlow [32] is a flexible open-source platform employed for diverse machine learning endeavors. It presents an extensive array of tools and features to oversee various facets of a machine learning system. While TensorFlow provides an expansive scope of capabilities, this course primarily highlights the utilization of a specific TensorFlow API for constructing and training machine learning models.

TensorFlow is an all-encompassing platform that empowers developers and researchers to address an extensive range of machine learning tasks. It streamlines the procedure of constructing and training machine learning models while offering adaptability, scalability, and ample resources to facilitate the complete machine learning workflow.

TensorFlow simplifies the implementation of machine learning models, abstracting complexities and offering high-level APIs such as Keras that facilitate the construction and training of neural networks. Additionally, TensorFlow excels in handling large datasets and efficiently performing computations across various hardware devices, including CPUs, GPUs, and specialized accelerators.

The platform further enhances the machine learning experience by providing tools for visualizing and monitoring training progress, optimizing model performance, and facilitating the deployment of models in real-world applications.

- **PyQt library**

PyQt5 [32] is a free module that links the Python language with the Qt library [33] distributed under two licenses, a free open-source license[35] and an other commercial as needed.. It allows to

create graphical interfaces in Python. We used this module to create the GUI (Graphical User Interface) for our application.

A certain amount of data is necessary to develop an efficient convolutional neural network model, including its architecture (Number of CNN layers, of grouping, the number of neurons of the fully connected layers), the learning functions, the learning algorithm and other network parameters.

The convolutional neural network development process involves the following four steps:

- Data preparation: we have data analysis and processing.
- Network training: this includes the choice of architecture, training functions and network parameters.
- Trained network test, to assess network performance.
- Using the trained network for prediction. The system inputs are strain, temperature and time, and the system output is stress. Tests Experiments were conducted to determine the tensile properties of an Agro-film:

III.3 Experimental Source of the database

III.3.1 Materials

This film is made up of three layers of polyethylene (PE). The various additives in the film are anti-condensation, anti-UV, antiparasite, anti-virus, anti-dust, anti-drop, etc. The exact nature of the additives is not provided by company. During extrusion, a film is formed with a thickness of 180 µm. This film is studied, when it is virgin, after having undergone artificial aging at temperature (10, 40 and 50°C) Precautions have also been taken to avoid stretching the film which could modify some of its mechanical and/or optical properties and to ensure good fixing. This last point is particularly important, poor fixing being the most frequent cause of tears. One of their main

weaknesses is the tearing of plastic films in general, in that after a cut, often accidental, has been made, the tear spreads slowly under the action of very weak forces. [34].

Data set preparation

We have used dataset of Engineering Laboratory, Materials Science Faculty, University of Tiaret , where the mechanical behavior of the unaged/virgin and naturally and artificially aged films was evaluated in different combined condition of temperatures (10, 40 and 50°C) water and UV-A radiations by conducting the tensile test:

- ✓ 10°C in 120 hours-550 hours -2040 hours-4000 hours. (4 strain-stress curves)
- ✓ 40°C in 10 hours-100 hours-500 hours-1000 hours-2480 hours-4000 hours-5480 hours. (7 strain-stress curves)
- ✓ 50°C in 10 hours-100 hours-500 hours-1000 hours-2480 hours-4000 hours. (6 strain-stress curves)

- **Virgin polyethylene**

The material used is a film supplied by the company AGROFILM to the industry of Setif (Multinational).

- **Aged polyethylene**

Samples of low density polyethylene and aged at 10 .40 and 50°C. [34]

III.4 Training and Parameterization of Models

After several runs, we ended up identifying the parameters that return the best accuracy for each convolutional network. In what follows, we will argue our choices in relation to the learning parameters:

III.4.1 Optimization Function

Adam "Adam" optimization algorithm is an extension of stochastic gradient descent.

We chose Adam because it is the most widely used optimization algorithm in deep neural network implementations. And for its remarkable advantages:

- Efficient calculation.
- Simple to implement.
- Not greedy in memory space.
- Appropriate for problems with a lot of noise.
- Well suited for problems that have a large number of data and parameters.

III.4.2 Error Function

The logcosh loss function is a smooth approximation of the logarithm of the hyperbolic cosine of the difference between the predicted and target values. It is often used in regression tasks and can handle outliers more effectively compared to other loss functions like mean squared error.

```
model.compile (optimizer='adam', loss=keras.losses.log_cosh)
```

III.4.3 Activation Function

Rectified Linear Unit (ReLU) (see section II) Relu is used in all network layers.

III.4.4 Number of Iterations (Epochs)

An iteration corresponds to a forward pass through the entire neural network plus a backward pass (Backpropagation) to modify the weights.

III.4.5 Force stopping "early stopping"

It is a form of regularization used to avoid overlearning when training a learner with an iterative method, such as gradient descent. These methods update the learner to better fit the training data with each iteration. To a certain extent, this improves the learner performance on data external to the training set. Beyond this point, however, improved fit of learner training data comes at the expense of increased generalization error. Stopping force rules provide guidance on how many iterations can be executed before the learner does begin to get oversized. [35]

```
Callback = tf.keras.callbacks.EarlyStopping (monitor='loss', patience=5)
```

- **Monitor :** quantity to monitor.
- **Patience :** number of epochs without improvement after which training will be stopped.
- **Mode :** one of {auto, min, max}. In min mode, the drive stops when the monitored amount has stopped decreasing ; In max mode, it stops when the controlled quantity has ceased to increase ; In auto mode, the direction is automatically inferred from the name of the quantity controlled

III.4.6 A callback

It is a function to be applied to given steps of the procedure of training. to get a view of the internal states and statistics of the model during the training we implemented the Callback method.

```
Callback = tf.keras.callbacks.EarlyStopping (monitor='loss', patience=5)
```

III.5 The suggested model

First, we developed a CNN model taking into account the three inputs simultaneously, which led to the following results:

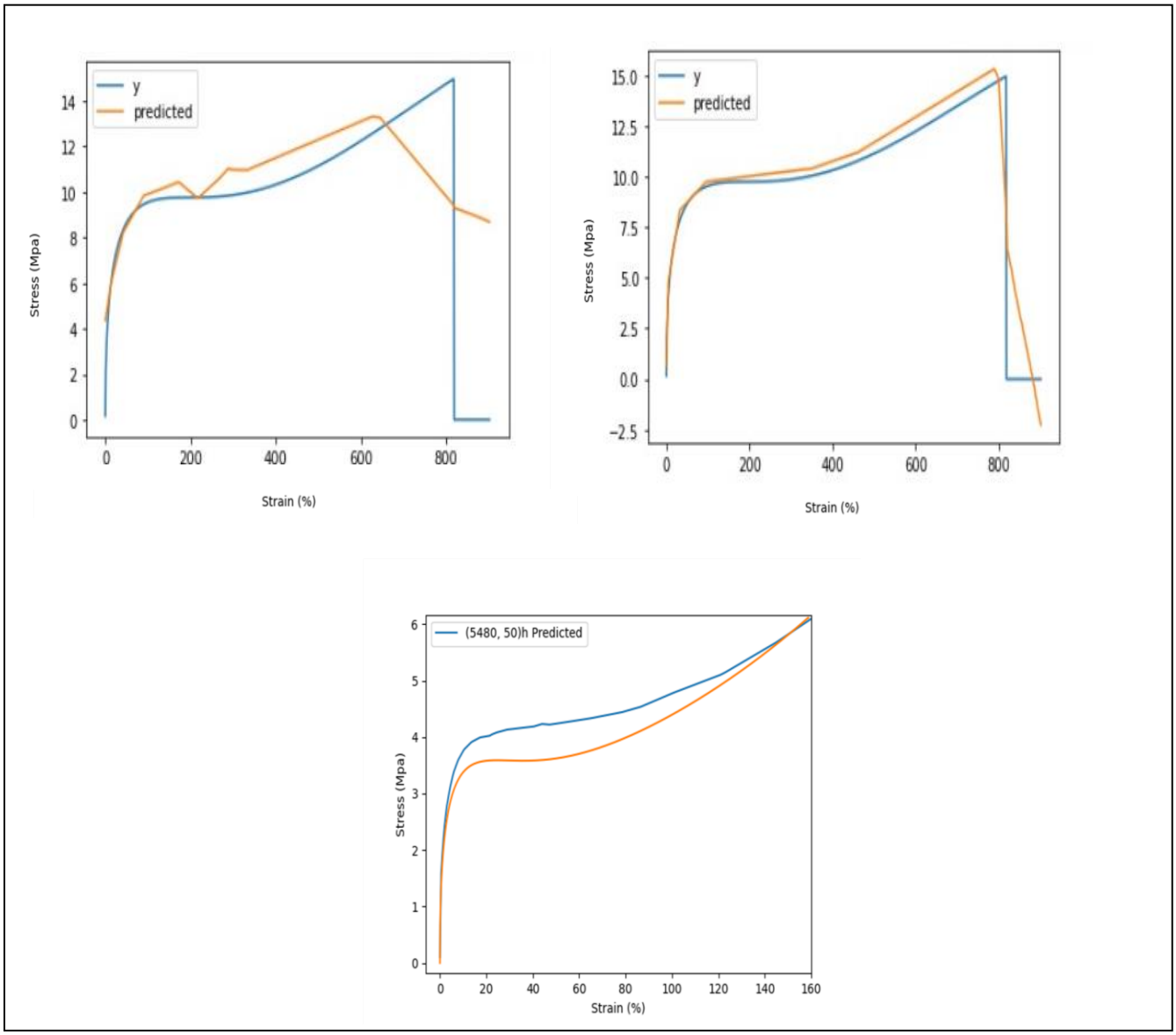


***Figure III.1*:** Graphical experimental

Then, a hybrid model called hybrid SVM-CNN is introduced by combining a deep Convolutional Neural Network (CNN) and a Support Vector Machine (SVM). This hybrid model aims to predict both the complete stress-strain curves and the material lifespan under specific usage conditions.

Predicting a stress-strain curve involves addressing two problems. The first problem is a classification task that involves separating each point on the curve based on strain values, essentially

classifying material fractures. The second problem is a regression task that entails predicting stress values.

Figure III.2 illustrates the hybrid SVM-CNN model, which comprises two submodels : the SVM submodel and the CNN submodel. These submodels are trained independently and subsequently combined for testing purposes.

In this proposed model, the SVM submodel acts as a classifier, while the CNN submodel functions as a feature extractor. The SVM component performs the classification task, whereas the CNN component focuses on extracting relevant features from the data.

By combining the strengths of SVM and CNN, the hybrid SVM-CNN model aims to leverage the classification capabilities of SVM and the feature extraction capabilities of CNN to enhance the accuracy and performance of stress-strain curve prediction.

This hybrid approach offers the potential to achieve more accurate and robust predictions by capitalizing on the unique strengths of both SVM and CNN.

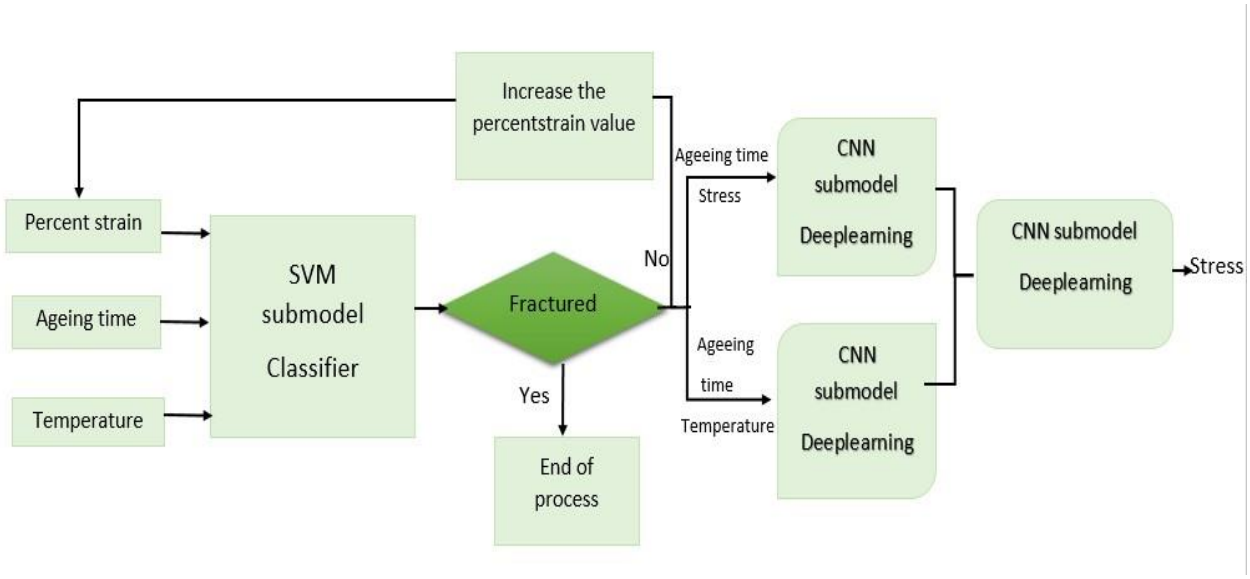


***Figure III.2*:** The proposed hybrid SVM-CNN model architecture

The convolutional neural network submodel

The configuration of a deep convolutional neural network (CNN) and the hyperparameters governing its learning process significantly impact its performance. Deep CNNs are commonly employed for feature extraction in image analysis, but there is a lack of guidance on constructing deep CNN models for regression problems. Designing an appropriate model for continuous input and output, which entails addressing nonlinear regression challenges, is a crucial task. The finalized architecture of the 1D CNN submodel used in this study is depicted in Figure III.3.

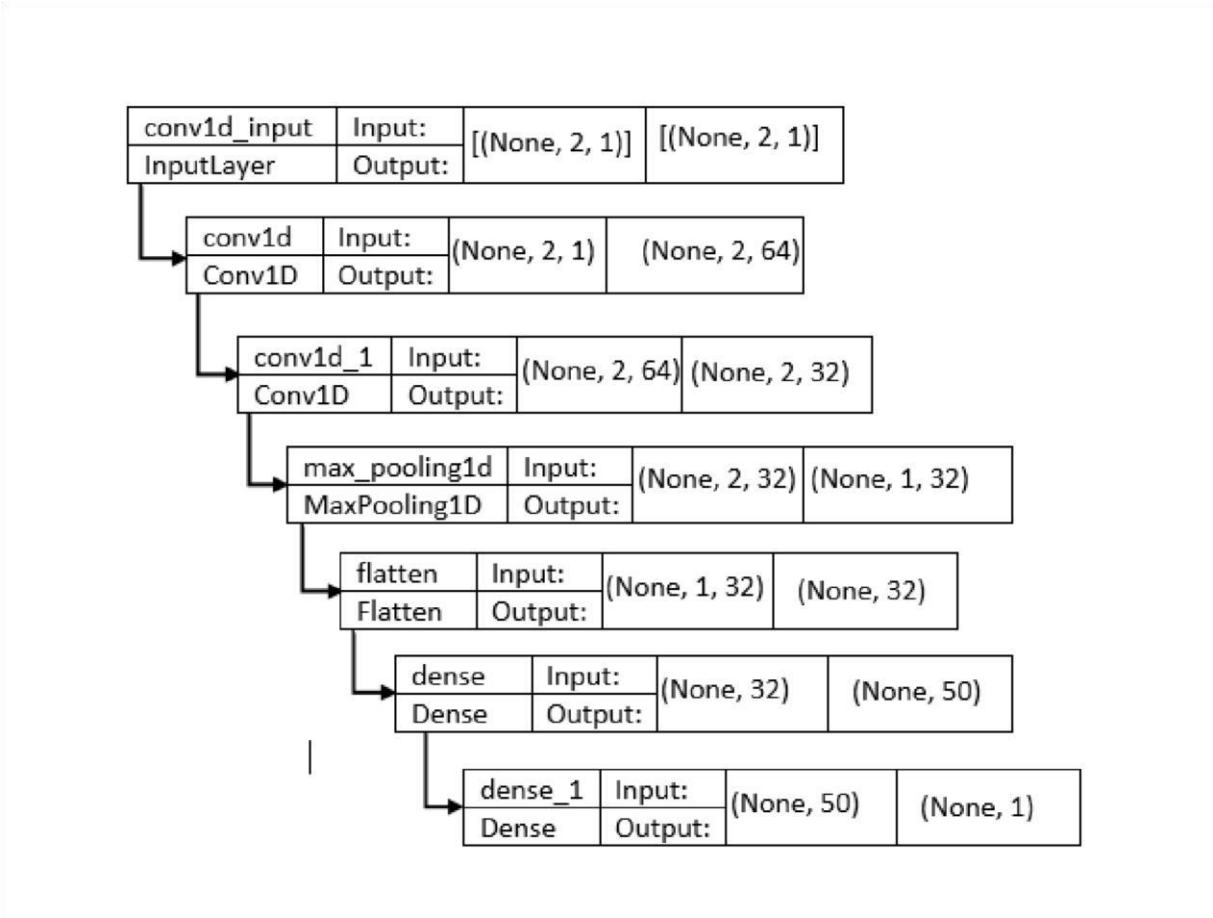


Figure III.3 : 1D CNN submodel architecture

Figure III.3 illustrates the proposed 1D CNN submodel, comprising an input layer, two convolutional layers, a pooling layer, a fully connected layer, and an output layer. The two

convolutional layers serve as feature extractors, while the subsequent layer consolidates semantically similar features, and the final layer maps the extracted features to the desired output. In this 1D CNN submodel, the relevant stress is estimated based on inputs such as percent strain, temperature, and ageing time. The hyperparameters, including filter sizes, stride, padding, pool size, and number of filters, were optimized through empirical experience to achieve an optimally designed 1D CNN.

Table III.1 presents the selected hyperparameters values.

	Hyperparameters
Convolution layer	Kernel size = 1, filters =64, strides =1, padding = valid, activation =relu
Convolution layer	Kernel size = 1, filters =64, strides =1, padding = valid, activation =relu
Pooling layer	Pooling method: maxpooling, pool size: 2, stride = 2, padding = valid
Fully connected layer	Neurons = 50,activation function = relu

***Table III.1*:** A list of hyperparameters in the 1D CNN submodel

The training of the model utilized the log hyperbolic cosine (log-cosh) loss function, which is commonly employed in regression-based problems to smoothen the curve [25].
The formula for the log-cosh loss is as follows:

$$L(X) = \frac{1}{N}\sum_{i=1}^{N} \log(\cosh(y_i - f_i\,(X))) \quad (8)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_{i^*})^2$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_{i^*})^2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - y^*{}_i\right|$$

(11)

Here, (X) represents the predicted values, y_i is the true values, and $coshcosh(x) = \frac{e^{x}+e^{-x}}{2}$. N denotes the number of samples.

To ensure a more stable error gradient, batch gradient descent was employed for training [37], leading to improved convergence. In batch gradient descent, the error is estimated for each example in the training dataset, but the model parameters are updated only after evaluating all training examples. This process continues until convergence is achieved.An epoch is a hyperparameter representing a single pass across the entire training dataset.

For the training of the 1D CNN submodel, (80% of the dataset) were used. The remaining curve (20% of the dataset) was kept as the test set, which the trained 1D CNN was tasked with predicting. The output of the network was then compared with the available experimental data to assess the submodel's validity.

During backpropagation, the Adam optimizer was employed to update the model's parameters. The training process consisted of up to 1750 epochs. To prevent overfitting, early stopping was applied using Callback functions. This approach ensures that the network exhibits high accuracy during training but also maintains accuracy when presented with new data during testing. At the end of the training process, the training error value was measured to be 0.0028.

We assessed the performance of the prediction model using three commonly used metrics: mean squared error (MSE), root-mean squared error (RMSE), and mean absolute error (MAE). These metrics are widely employed to evaluate the accuracy of regression models in predicting outcomes. The results obtained from the evaluation are presented in Table III.2. To calculate these evaluation metrics, we utilized the keras-metrics Python package [41]. Let y_i* denote the predicted values of

the stress variable for n prediction samples, and let y_i represent the observed values. The equations (9 to 11) represent the formulas for calculating MSE, RMSE, and MAE, respectively.

Activation functions in hidden layers	Evaluation metrics		
	MSE	RMSE	MAE
Tanh	1.467	1.207	0.71
Sigmoid	1.518	1.234	1.487
ReLU	0.00246	0.012	0.017

Table III.2: Evaluation metrics for 1D CNN submodel

To ensure the validity of the submodel, the trained network was tested by simulating a stress-strain curve for an ageing time and temperature of (5480hours, 50°). It is important to note that this specific curve was not included in the training data. The simulated curves closely align with the existing experimental data ; however, the predicted curve extends beyond the fracture point, even for the training curves, requiring manual intervention to stop the simulation. Consequently, an additional submodel was introduced to predict the fracture point using SVM. Once the fracture point is predicted, the simulation automatically halts without any manual intervention. This hybrid model employs automatic feature extraction from raw data and generates predictions accordingly.

The data initially passes through the SVM submodel, which categorizes the percent strain value into the appropriate class based on the ageing time, determining whether the material is fractured or not. If the material is broken, the process concludes. Otherwise, the data proceeds through the CNN submodel to predict the stress value, and the procedure repeats from the beginning, incrementing only the percent strain value.

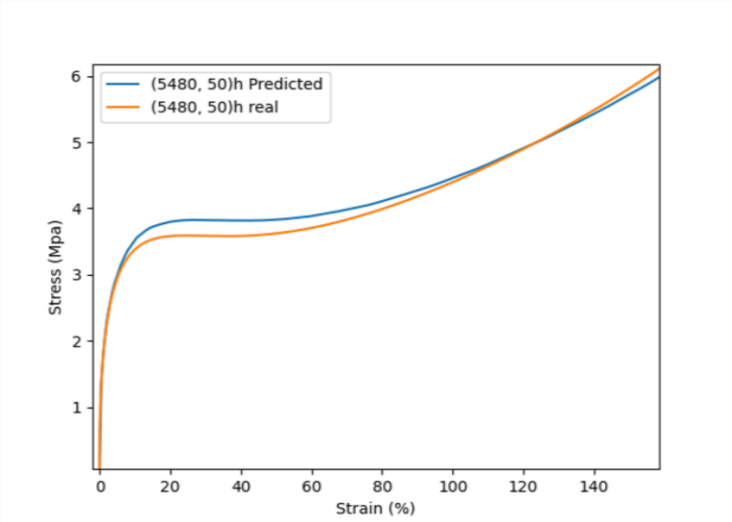


Figure III.4: Graphical comparison between predicted and unseen experimental stressstrain curve

▯ The Support Vector Machine (SVM) submodel

We used the same model developed by (aid&al 2022) which proved its effectiveness and we did the same process

Figure III.4 depicts the fracture point or fracture stress, which represents the ultimate point in the stress-strain curve. To predict the fracture stress of aged films, a support vector machine submodel was proposed. Each data point on the stress-strain curve was treated as a separate binary classification problem. The first class on the curve was assigned a label of '1', corresponding to strain values ranging from zero to the breaking strain. Beyond this threshold, the second class was labelled as '0'.

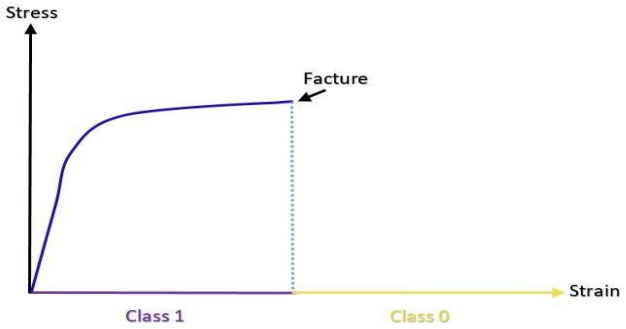


***Figure III.5*:** The two classes of stress-strain curve

The SVM submodel takes percent strain , temperature and ageing time as inputs. The desired output is a binary value of 1 or 0, representing "not fractured" or "fractured," respectively. The SVM submodel was trained using a dataset of 17 stress-strain curves, which were automatically labelled using a script.

To evaluate the performance of the SVM submodel, we employed two widely used metrics: accuracy and AUC (Area Under the Curve). Accuracy measures the proportion of correct predictions among all predictions made. If y_i* represents the predicted value of the it^h sample and yi represents the corresponding true value, the accuracy can be defined as follows:

$$accuracy(y, y*) = \frac{1}{N}\sum_{i=0}^{N-1} 1(y_i \ *= \ y_i) \qquad (12)$$

Here, 1(x) is the indicator function that returns 1 if x is true and 0 otherwise. AUC, on the other hand, is a combined measure of sensitivity and specificity. Given a dataset D = {(xi, yi)} with N examples, where yi belongs to {-1, +1} representing class labels and xi represents input vectors, let N_{pos} denote the number of positive examples (yi = +1) and N_{neg} denote the number of negative examples (N_{neg} = N - N_{pos}). The AUC of a predictor f is defined as: [41]

$$AU(f,D) = \frac{1}{N}\sum_{j=1}^{N^{pos}} 1\left[f(x_j) < f(x_k)\right] \qquad (13)$$

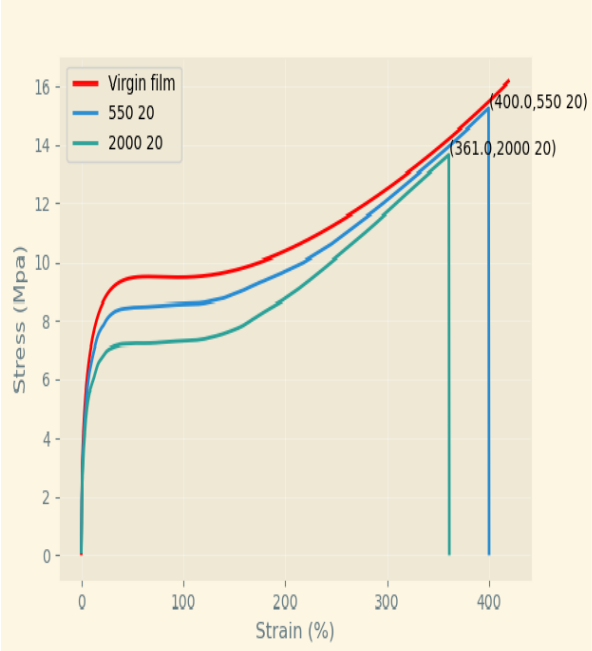
In this equation, 1(a) is the indicator function that returns 1 if a is true and 0 otherwise.

We calculated the AUC and accuracy using Python 3.9 with the Sklearn.metrics package [42]. The obtained values were 0.981 for AUC and 0.98 for accuracy, indicating that the submodel performs well in predicting whether the material is fractured or not.

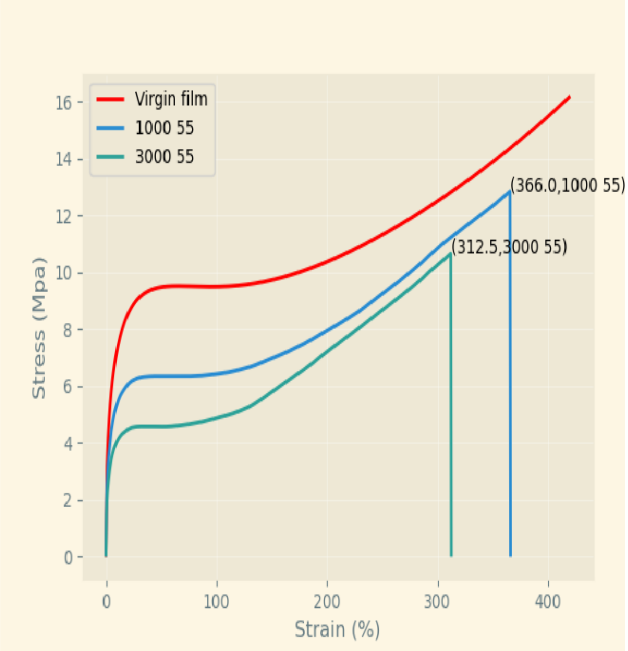
III.5.1 Findings and Analysis

The outcomes of the hybrid SVM-CNN model are presented in Figure III.6. All visualizations in this study were generated using Matplotlib [41]. For each specified ageing time (10 ,100 ,120 ,500 ,550 ,2040 and 4000h) and temperature (10°, 40° and 50°), the model was tasked with predicting the stress-strain curve. The predicted output was then compared with the available experimental data. It is evident that the curves align closely with each other, indicating the model's ability to accurately simulate the curve trends. The model demonstrates strong performance in predicting mechanical properties such as yield strength, fracture stress and strain, with accuracy approaching 100%. Moreover, the predicted stress-strain curves enable the calculation of other mechanical properties such as tensile strength, strain at break, percent elongation, Young's modulus, toughness, and the ratio of tensile strength to Young's modulus.

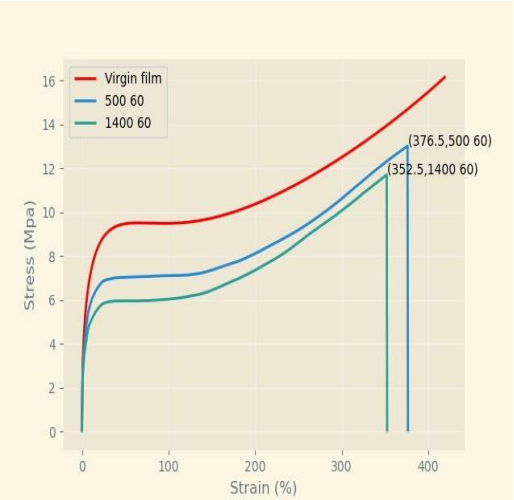
Predicted stress-strain curves for virgin film and for different ageing time and temperature



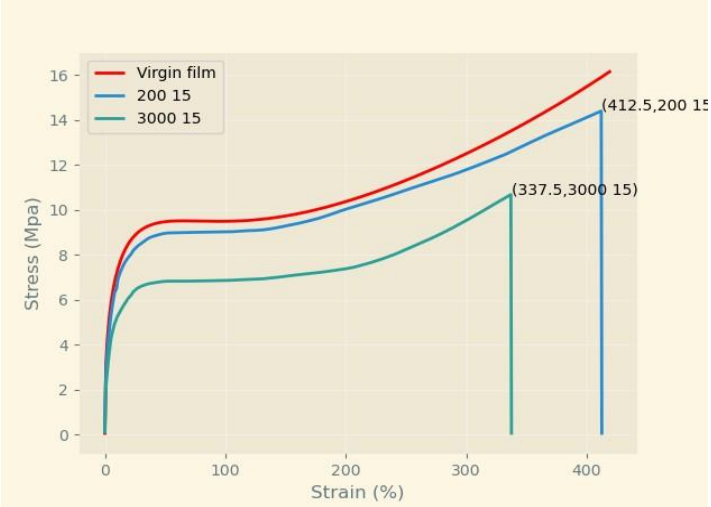
(550H/20°),(2000H/20°)



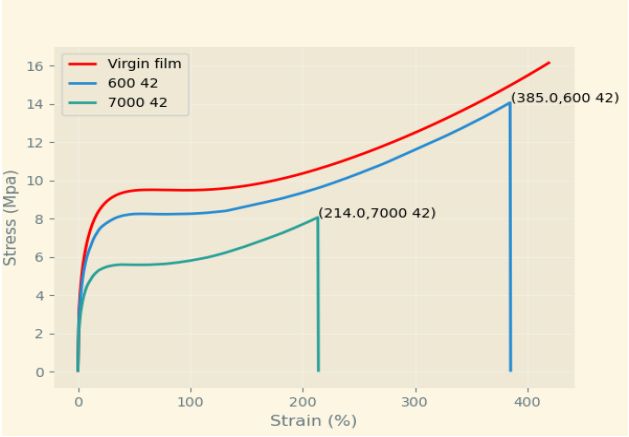
(1000H/55°),(3000H/55°)



(500H/60°),(1400H/60°)

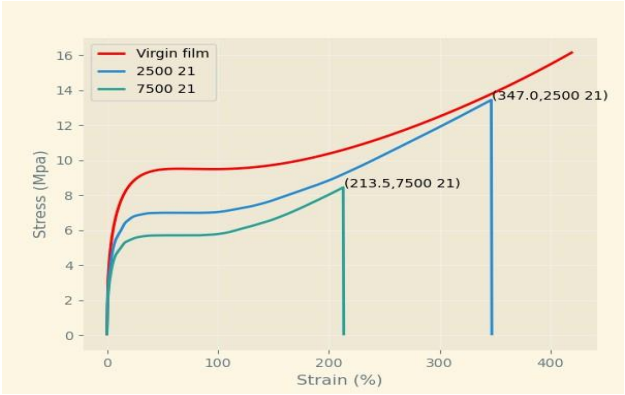


(200H/15°),(3000H/15°)

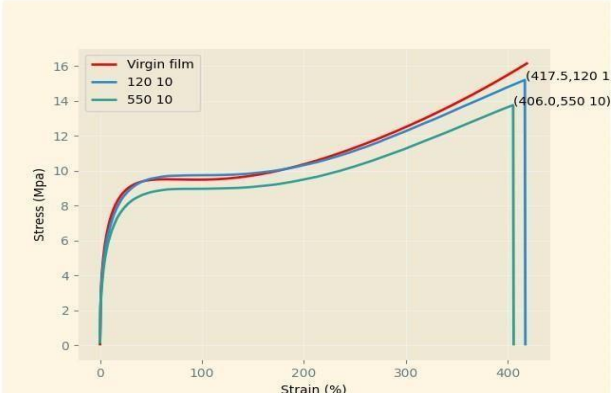


(600H/42°),(7000H/42°)

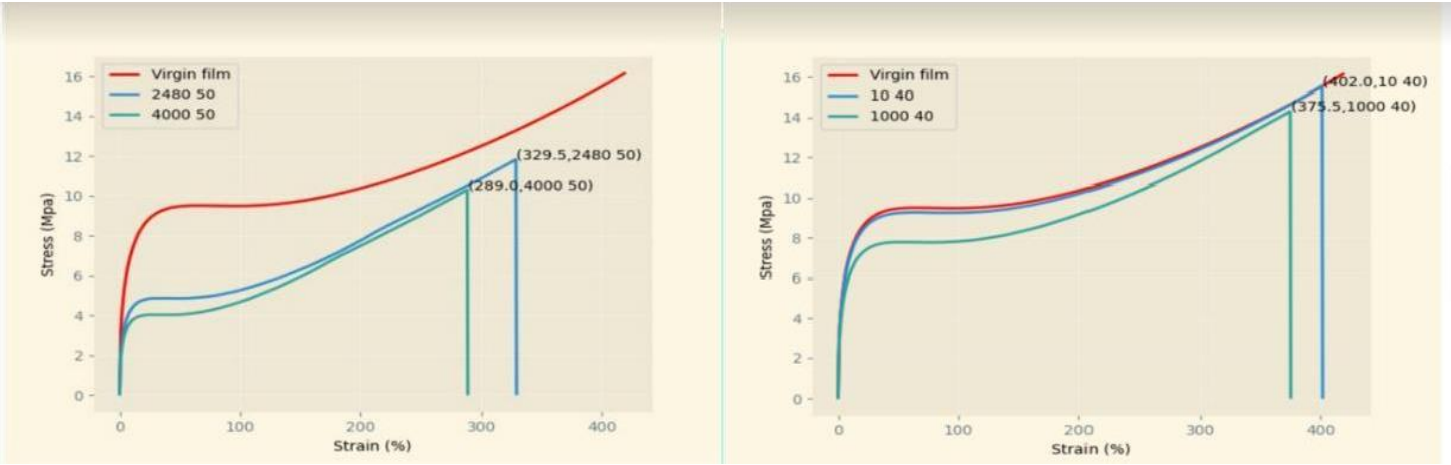
(300H/14°),(5500H/14°)



(2500H/21°),(7500H/21°)



(120H/10°),(500H/10°)



(2480H/50°), (4000H/50°) (10H/40°), (1000H/40°)

Figure III.6: Graphical comparison between experimental and predicted stress-straincurves

Our dataset consists of five ageing periods up to (5480 h,50°). However, the hybrid SVM-CNN model has the capability to predict stress-strain curves for any given ageing time, extending beyond the range of our dataset. Figure III.6 demonstrates this ability, as the model effortlessly predicts curves for (9000 h,25°). This showcases the model's powerful capacity to learn high-level feature representations through deep CNN.

Furthermore, the hybrid SVM-CNN model can be utilized to predict the maximum degradation time under specific usage conditions. By incrementing the ageing time up to (18000 h,25°) (2 years) using a Python loop, the model was instructed to predict the stress at each interval. The maximum degradation time observed was (14325 h,25°), beyond which the model indicates "material fractured," indicating complete destruction of the material. On the other hand, the predicted curve represents the stress-strain curve of the LDPE film during degradation at the selected periods. Consequently, both the predicted curve and the corresponding stress-strain curve for the unaged material can be plotted on the same graph, as demonstrated in our analysis. This provides valuable information regarding the limit of service time based on the criterion of a 50% loss in the original property [43].

In the subsequent sections, we will compare the performance of our model with the commonly used types of neural networks for supervised learning.

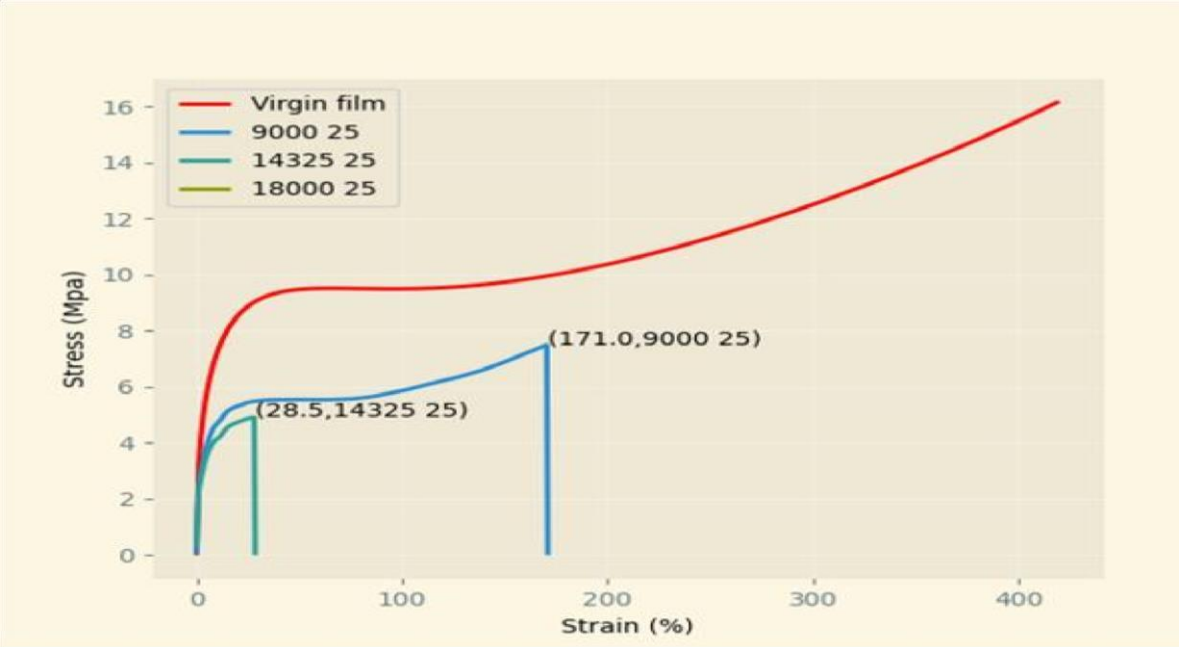


Figure III.7: Predicted stress-strain curves for maximum degradation time

III.6 Software

Once we achieve our objective of developing an efficient hybrid model capable of predicting the complete stress-strain curve and the material lifespan under real usage conditions, it is essential to provide users with an efficient and user-friendly software application.

III.6.1 User interaction and design

Recognizing that using a command-line interface (CLI) can be challenging and complex for many users, we have opted to utilize PySimpleGUI to develop a straightforward and user-friendly graphical user interface (GUI). PySimpleGUI is a Python package that simplifies the tkinter, Qt, and WxPython GUI frameworks, providing a more accessible interface, which aligns with our goal of simplicity.

The main window of the application will have the following appearance :

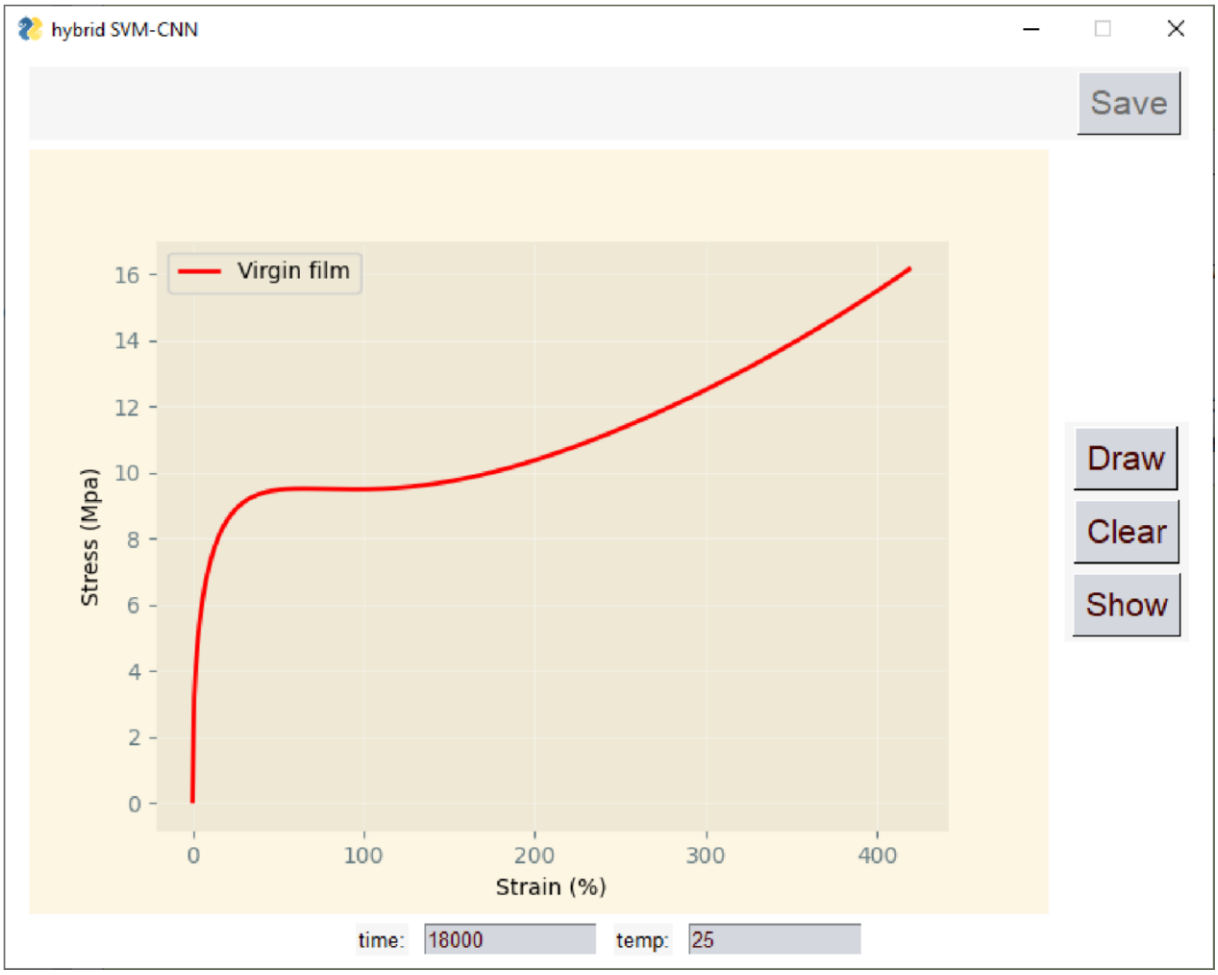


***Figure III.8*:** main window

Our application features a canvas that occupies the majority of the window, displaying an initial curve representing the virgin film for comparison with the stress-strain curves generated by the model. At the bottom of the window, there is an input field where users can specify the aging time for which they want to generate the curve. By entering the desired time and temperature values and clicking the "Draw" button, the model will predict and plot the corresponding stress-strain curve. This user-friendly process simplifies the task of generating and visualizing the curves.

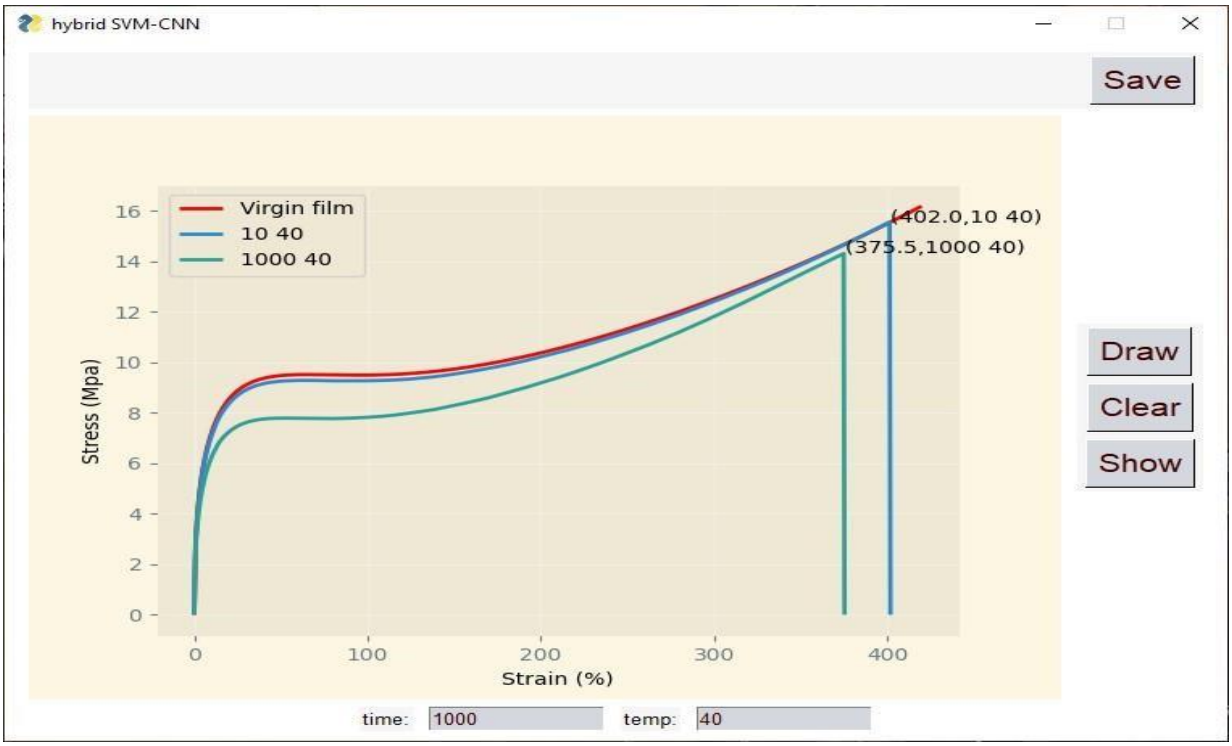


Figure III.9: plot the stress strain curve

Several stress-strain curves can also be depicted side by side.

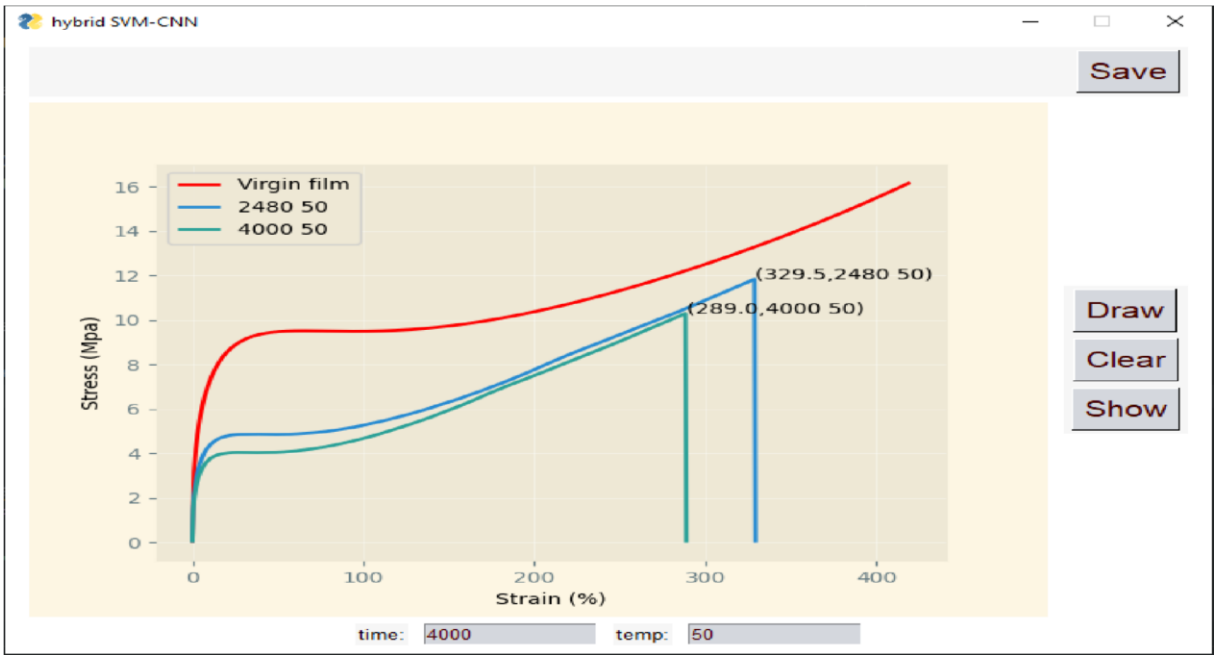


***Figure III.10*:** multiple stress strain curves adjacent to each other

To examine the chart in more detail, we can click the "Show" button, which allows us to zoom in and navigate through the chart.

Additionally, we have the option to save the chart as an image on our disk for future reference or sharing.

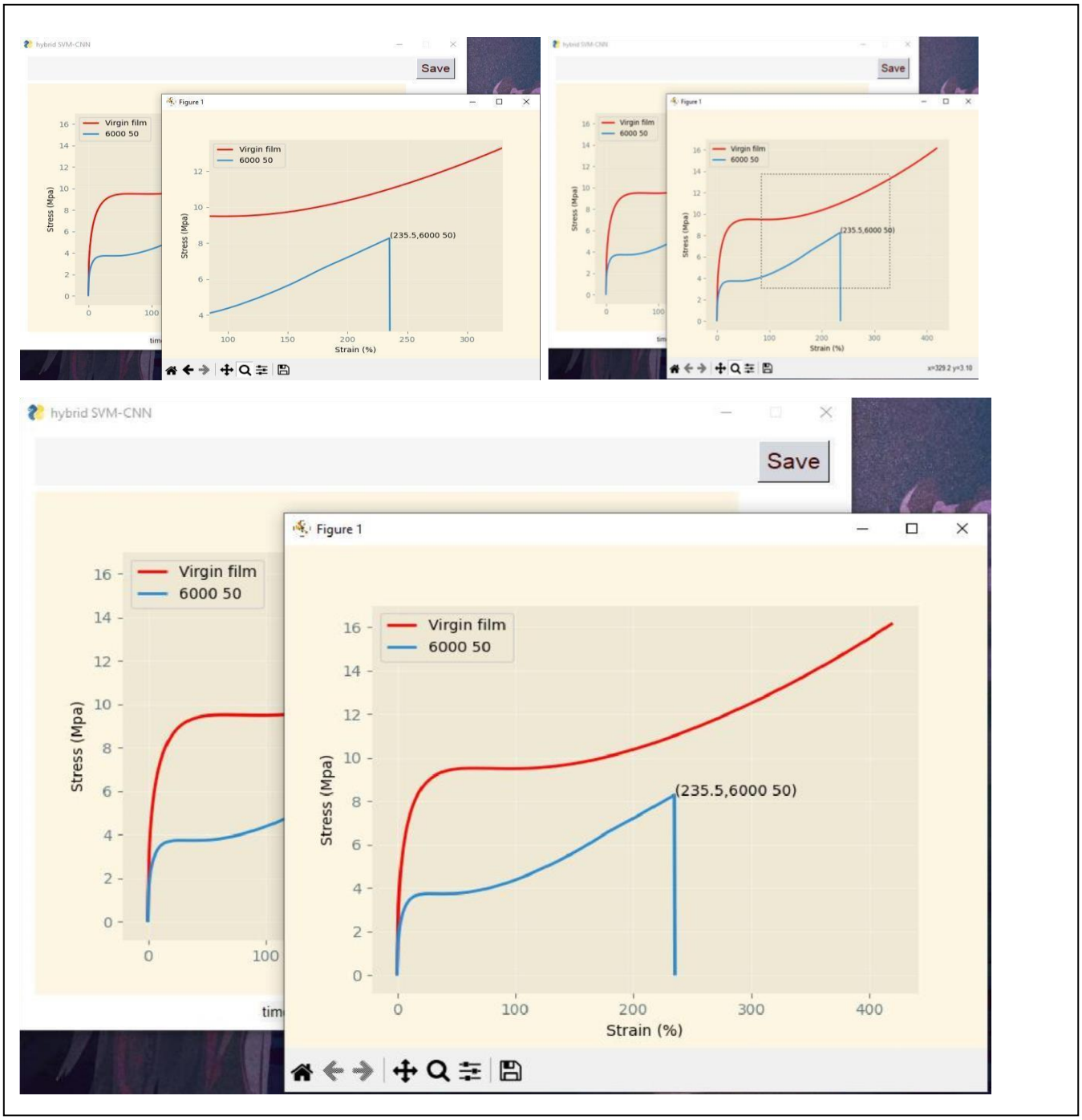


Figure III.11: detailed view at the chart

By clicking the "Save" button in the main window; we can easily store the predicted stress-strain curve generated by the model into an Excel file. This functionality allows us to conveniently save and access the data for further analysis or sharing purposes.

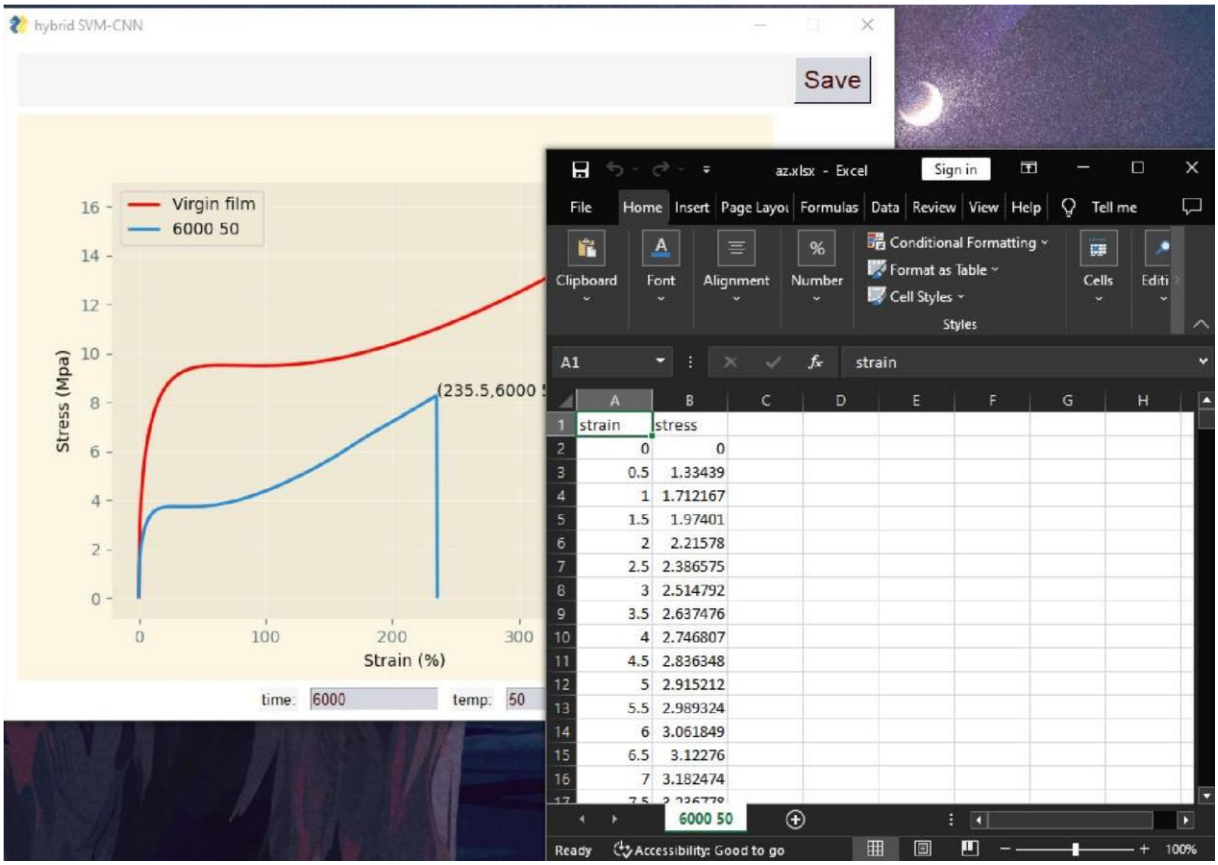


***Figure III.12*:** saving to excel

III.7 Conclusion

In this chapter, we initially introduced the various tools and programming languages employed for the implementation of our application. Subsequently, we developed a deep learning model to forecast the stress-strain curve of LDPE under changing time and temperature conditions.

We presented our application, elucidating its usage and providing insights into the testing and prediction results.

General conclusion

We have developed and validated a hybrid deep learning model to effectively predict the lifetime and mechanical performance degradation of tri-layer LDPE films used in greenhouses. The deep learning (DL) model demonstrated remarkable success during both the training and validation phases, showcasing its capability to accurately predict stress-strain curves for aged LDPE films at different time intervals. By combining SVM and CNN models, the hybrid SVM-CNN model offers a time-saving and cost-effective solution, providing a user-friendly tool to forecast both the mechanical properties and lifespan of greenhouse coverings under various usage conditions. This advancement will assist manufacturers in creating durable materials for long-term use and prove beneficial for engineering designers. The findings of this study strongly support the potential of DL as a promising model approach, inspiring further exploration and application within the field of polymer characterization.

References :

[01]: K. Hachour, « contribution à l'élaboration d'un critère de plasticité pour le Pehd », thèse de doctorat en génie mécanique, université mouloud mammeritizi ouzou,2010.

[02]: Nadine Lahoude '' Modélisation du vieillissement des isolants organiques sous contrainte électrique. Application à la fiabilité des matériaux.'' Thèse de Doctorat, 25 Mars 2009. Université de TOULOUSE.

[03]: BOSRI Rahil HARBOUCHE-Ibtissem Halima « Développement d'un modèle prédictif de durée de vie des polymères » /09 /07/ 2019 . Université ibn khaldoun de Tiaret .

[04]: P. Weiss, la chimie des polymères, université médicale virtuelle francophone, société francophone de biomatériaux dentaires, 2009-2010.

[05]: J.DorlotT, J. Massounave et al, "des matériaux", edition de l'ecole polytechnique des matériaux, Juillet, (1995).

[06]: B. Vieille, J. Aucher, L. Taleb « Carbon fiber fabrics reinforced PPS laminates: influence of environmental conditions on the behavior of notched laminates and bolted joints » soumis à Mechanics of Mat., 2009.

[07] : Etude du comportement du polyéthylène haute densité sous irradiation ultraviolette ou sollicitation m´mécanique par spectroscopie de fluorescence - Ludovic Douminge J. Martin et al, " traité des matériaux ", Tome 14, (2001).

[08]: Masakazu Matusugu, Katsuhiko Mori, Yusuke Mitari et Yuji Kaneda, « Subject independent facial expression recognition with robust face detection using a convolutional neural network », Neural Networks, vol. 16, n o 5, 2003, p. 555–559 .

[09]: «les techniques du polyéthylène » thème élaboré par le Chef de Subdivision en Enseignement Gaz Mr.DibNaouar.

[10]: THÈSE Doctorat. Mr. BENSAID Mohamed Ouassini. "Étude Des Polymères Par Dynamiques Moléculaire". 2015.

[11]: V, Chaffraix. Etude de l'extrusion du polyéthylène téréphtalate et de ses mélanges noncomptabilises avec le polyéthylène haute densité, mécanique [physics.med-ph], Ecole nationale supérieure des mine »s, Paris, 2002.

[12]: MBAREK BOUHARKAT. Prédiction des propriétés mécaniques de PEbd vieillis en utilisant les réseaux neurones. 2016.thése magister.

[13]: S. Fuzessery, Techniques de L'Ingénieur (1988).

[14]: A.Bélaid. mémoire de magister "caractérisation sur site du polyethylène tri-couches AFS3 utilise comme couverture de serre » Univ ORAN 2005.

[15]: VERDU J. - Vieillissement chimique des plastiques : aspects généraux, Techniques de l'ingénieur, traitéPlastiques et Compostes, AM 3 151, 14 pp.

[16]: STRESS-STRAIN CURVES David Roylance Department of Materials Science and Engineering Massachusetts Institute of Technology Cambridge, MA 02139 August 23, 2001.

[17] F. Chollet, "Deep learning with Python". Shelter Island, NY: Manning Publications Co,2018.

[18] Moreira, F. (2023, February 13). Who is Artificial Intelligence father? - Robot Artificial. Robot

Artificial. https://www.robotartificial.com/who-is-artificial-intelligence-father/

[19] System, I. (2019). L'intelligence artificielle : une solution pour améliorer la maintenance prédictive.

Informatique Industrielle & IIoT. https://blog.integral-system.fr/lintelligenceartificielle-une-solution-pour- ameliorer-la-maintenance-predictive/

[20] Wotton, D. (2023, March 9). Intelligence artificielle, machine learning, deep learning : kézako ?

L:EDigitalab. https://www.ledigitalab.com/2017/10/02/intelligence-artificielle-machine-learning-deep-learning-kezako/

[21] B. Benmammar, “Apprentissage automatique”, research report, p. 71

[22] ZERROUKI Sofiane, “ Proposition et Evaluation d’un Modèle Deep Learning pour la

Classification d’Images Basé sur les Espaces de Couleurs” ,univ Tiaret , 2019-2020.

[23] Gopinath Rebala · Ajay Ravi Sanjay Churiwala , “An Introduction to Machine Learning”,p.127

[24] N. Buduma, Fundamentals of Deep Learning, O’Reilly, 2017.

[25] Lahcene, A., Amine, D. M., & Abdelkader, D. (2021). A Hybrid Deep Learning Model for Predicting Lifetime and Mechanical Performance Degradation of Multilayer Greenhouse Polyethylene Films. Polymer Science, Series B, 63(6), 964-977.

[26] Python Language Copyright c 2001-2018 Python Software Foundation. All rights reserved. Licence : https://docs.python.org/3/license.html, Python documentation : http://docs.python.org/3/. https://wiki.python. org/moin/BeginnersGuide/Overview]

[27] https://www.anaconda.com/why-anaconda/

[28] https://www.jetbrains.com/pycharm/?fromMenu

[29] Scikit-learn Library : http://scikit-learn.org/stable/. Github : https: //github.com/scikitlearn/scikit-lear

[30] Keras Framework : https://keras.io/. Github : https://github.com/ keras-team/keras

[31] Gulli, A., Pal, S.: Deep Learning with Keras. PacktPublishing Ltd., Birmingham (2017).

[32] : Keras Licence : https://github.com/keras-team/keras/blob/master/ LICENSE

[33] : Bibliothèque PyQt5 v5.10.1 : http://pyqt.sourceforge.net/Docs/PyQt5/, Github : https://github.com/pyqt

[34] Entreprise Qt : https://www.qt.io, Github : https://github.com/qt

[35] PyQt Licence : https://www.riverbankcomputing.com/commercial/ license-faq

[36] : djakhdane khaled mémoire de magister(étude des vieillissements artificielles d'un film de polyethylene basse densite (PEbd)«température, eau et rayons uv »)univ tiaret 2011.

[37] Poort J, Omrani PS, Vecchia AL, Visser G, Janzen M, Koenes J. An automated diagnostic analytics workfow for the detection of production events-application to mature gas felds. In: Abu Dhabi international petroleum exhibition and conference. OnePetro, 2020.

[38] RAY, Susmita. A quick review of machine learning algorithms. In : 2019
International conference on machine learning, big data, cloud and parallel computing (COMITCon). IEEE, 2019. p. 35-39. [170] KINGMA, Diederik P. et BA, Jimmy. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014

[39] Brzezinski, D., &Stefanowski, J. (2017). Prequential AUC: properties of the area under the ROC curve for data streams with concept drift. Knowledge and Information Systems, 52(2), 531-562.

[40] Gulli, A., Pal, S.: Deep Learning with Keras. PacktPublishing Ltd., Birmingham (2017).

[41] Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. 12, 2825–2830 (2011)

[42] Calders, T., &Jaroszewicz, S. (2007, September). Efficient AUC optimization for classification. In European Conference on Principles of Data Mining and Knowledge Discovery (pp. 42-53). Springer, Berlin, Heidelberg.

[43] Briassoulis, D. (2006). Mechanical behaviour of biodegradable agricultural films under real field conditions. PolymerDegradation and Stability, 91(6), 1256-1272