ورقة بحثية مجلة مفتوحة

نافذة إلى الذكاء الصنعي
"ما بين الذكاء الصنعي وتعلم الآلة والتعلم العميق"

د. هلا نصار
(قسم الذكاء الصنعي ، جامعة تشرين
البريد الالكتروني : drhalanassar@gmail.com)

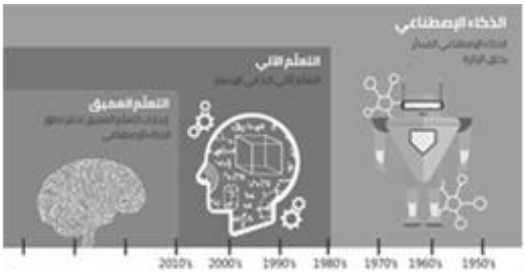
الملخص

تقدم هذه الوثيقة مقارنة بين الذكاء الصنعي وتعلم الآلة والتعلم العميق. انطلاقاً من الذكاء الصنعي وأنواع أنظمته، مروراً بتعلم الآلة ومراحله وصولاً إلى التعلم العميق، يتم تحديد أهم الميزات التي تخص كل منها، وتوضيحها من خلال أمثلة بسيطة تساعد على فهم الاختلاف.

كلمات مفتاحية – الذكاء الصنعي Artificial Intelligence، تعلم الآلة Machine Learning، التعلم العميق Deep Learning.

I. مقدمة

شهد مجال التكنولوجيا تطوراً متواصلاً لسنوات طويلة ومازال. ومع مرور الوقت أصبح العالم بمجالاته المختلفة، يتناول مصطلحات مثل الذكاء الصنعي وتعلم الآلة والتعلم العميق. على الرغم من استخدام هذه المصطلحات، إلا أن هناك مفاهيم خاطئة حول ما تعنيه هذه الكلمات. في الواقع كثير من الأحيان يتم الخلط بينها وبما يتعلق بمعنى هذه المصطلحات وتعريفاتها التي تبدو متشابهة، لكن لكل مصطلح تعريف يختلف عن الآخر.



الشكل 1 مخطط زمني لتسلسل ظهور الذكاء الصنعي، تعلم الآلة، التعلم العميق

حيث يوضح الشكل 1 المخطط الزمني لتسلسل ظهور هذه المفاهيم بدءاً من الذكاء الصنعي مروراً بتعلم الآلة وصولاً إلى التعلم العميق.

II. الذكاء الصنعي Artificial Intelligence

كانت ولادة حقل الذكاء الصنعي AI صيف 1956م، في مؤتمر أقيم في جامعة Dartmouth، وقد أصبح هؤلاء المشاركون في المؤتمر رواداً في أبحاث الذكاء الصنعي لعدة سنوات. حيث كان Allen Newell (CMU), Herbert Simon (CMU), John McCarthy (MIT), Marvin Minsky (MIT) and Arthur Samuel (IBM) أول من أنشأ مختبر ذكاء صنعي في MIT و Standford, CMU ، وكتبوا مع طلابهم العديد من البرامج الهامة في هذا المجال.

قام Defense Advanced Research –DARPA Projects Agency وهي وكالة بحث وتطوير تابعة لوزارة دفاع الولايات المتحدة، المسؤولة عن تطوير التقنيات الناشئة لاستخدامها من قبل الجيش في منتصف الستينات، بتمويل هذه الأبحاث، ولم يُكتب النجاح لجزء كبير من تلك المحاولات. في بداية الثمانينات عاد الاهتمام بهذا الحقل مع النجاح التجاري للأنظمة الخبيرة، وقد وصل تسويق برامج AI في عام 1985م إلى ما يقدر بمليار دولار. في التسعينات كان العصر الذهبي للذكاء الصنعي، حيث حقق نجاحات عظمى واعتُمد في مجالات واسعة وسُخّر في تقنيات الصناعة.

هذه النجاحات تعود إلى العديد من العوامل، منها القوة المدهشة للحواسيب، الاهتمام الكبير في حل المشكلات الجزئية المتعلقة بمجال معين، خلق روابط جديدة بين AI والحقول الأخرى التي تعمل على مشاكل مشابهة، وفضلاً عن هذا تعهد الباحثون الطرق الرياضية والمقاييس العلمية الدقيقة.

أ. النظم الخبيرة expert system:

تعد النظم الخبيرة واحدة من أوائل أشكال أنظمة الذكاء الصنعي، حيث تبنى على محركات استدلال آلية، تستند إلى شرط "If" لتكون نتيجة النظام بناءً على "Then". يمكن تقسيم تطبيقات الذكاء الصنعي إلى نوعين، من ناحية النتائج اعتماداً على قاعدة If–Then. الأولى تطبيقات تصنيف وتمييز، والثانية تطبيقات تحكم والقيام بأوامر معينة، بكل الأحوال تقوم المتحكمات بتصنيف الشروط قبل استنتاج الأحداث والأوامر، ومنه التصنيف يشكل الجزء الأساسي في معظم أنظمة الذكاء الصنعي.

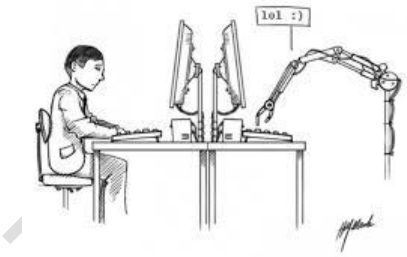
بدأ العمل على النظم الخبيرة في أواخر الخمسينات، وكانت الحصيلة تصميم برامج تستطيع حل المشاكل المعقدة التي يعتبر حلها سلوكاً ذكياً، إذا ما تم ذلك بواسطة الإنسان. في الستينات، حاول علماء الذكاء الصنعي محاكاة العملية المعقدة للتفكير لدى الإنسان، وذلك بإيجاد أساليب لحل مشكلة ذات نطاق واسع وكبير، ورغم تحقيق بعض التقدم الملحوظ، إلا أن هذه الاستراتيجية لم تؤدي إلى أي نتائج بارزة، فتصميم نظم ذات أهداف عامة كان صعباً للغاية، لذلك تم التركيز بدلاً من ذلك على تصميم نظم أكثر تحديداً وخصوصية.

خلال السبعينات استحوذ اهتمام العلماء على تقنيات كيفية تمثيل المشكلة، والبحث عن حل ناجح يوفر الذاكرة والوقت، ومرة أخرى لم تسفر هذه الاستراتيجية عن أي تقدم أو نجاح بارز. في أواخر السبعينات توصل علماء الذكاء الصنعي إلى أن قوة البرنامج وذكاءه في حل المسائل تأتي أولاً من المعرفة التي يحتويها، وليس من مجرد آلية التمثيل والبحث والاستدلال التي يستخدمها، وأدى ذلك إلى تحقيق نجاح كبير في الوصول إلى نظم تتصف بالذكاء، عن طريق تزويده بكمية من المعرفة في مجال محدد ومتخصص. وبدأ هذا المجال في الذكاء الصنعي ينتشر ويتطور بسرعة كبيرة في معظم مجالات الحياة العملية.

ومنه الذكاء الصنعي الذي شكل أحد العلوم الجديدة خلال الثمانينات يمكن القول بأنه العلم الذي يهتم بإنشاء برمجيات ومكونات مادية قادرة على محاكاة السلوك البشري. والتحدي الأكبر هو محاكاة الإنسان في عمليات الإدراك التي يقوم بها دون تعليم أو تفكير مثل النظر والسمع والكلام والحركة بطريقة متوازنة.

ب. تعريف الذكاء الصنعي

قام العديد من الباحثين في هذا المجال بتعريف الذكاء الصنعي، "الذكاء الصنعي هو العلم القادر على بناء آلات تؤدي مهاماً تتطلب قدراً من الذكاء البشري عندما يقوم الإنسان بها" Kurzweil 1991، كما في الشكل 2. "الذكاء الصنعي هو دراسة الطرق الحسابية التي تجعل من الممكن الادراك والمنطق والتأثير" Winston 1992.

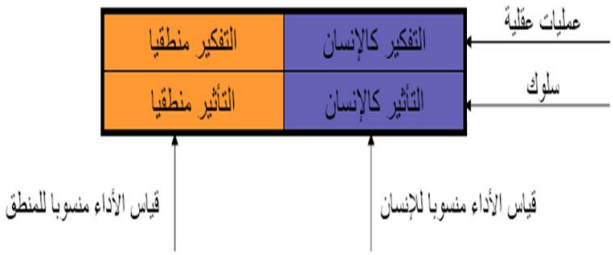


الشكل 2 آلة تقوم بمهام تحتاج للذكاء البشري

"دراسة كيفية توجيه الحاسب لأداء أشياء يؤديها الإنسان بطريقة أفضل" Rich and Knight 1991. "الذكاء الصنعي هو جهود مثيرة لجعل الآلات تفكر، أن يكون لها عقل بمعنى الكلمة" John Haugeland 1985. "الذكاء الصنعي يهتم بالسلوك الذكي عند التصنيع" Nilsson 1998.

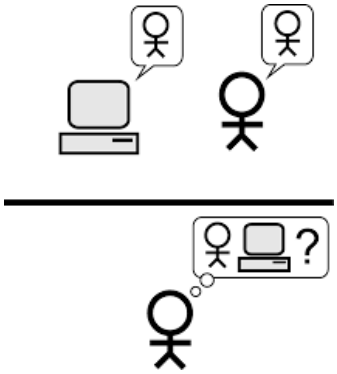
ت. أنظمة الذكاء الصنعي:

إن الذكاء الصنعي ينجح في حل المسائل التي تتميز بعدم وجود خوارزمية واضحة لحلها، أو تتطلب حدسية عالية، وعمليات حسابية هائلة. ومنه تم تصنيف أنظمة الذكاء الصنعي كما في الشكل 3 ضمن أربعة أنواع، اثنين منها تتعلق بالتفكير والأخرى تتعلق بالسلوك، وهي أنظمة تفكر منطقياً، أنظمة تفكر كإنسان، أنظمة تؤثر منطقياً، أنظمة تؤثر كإنسان.



الشكل 3 أنواع أنظمة الذكاء الصنعي

- أنظمة تفكر منطقياً: إن هذه الأنظمة تعتمد نظرية المنطق والاستدلال لبناء حلول تقوم على مصطلحات المنطق. ولكن ليس كل سلوك ذكي هو بالضرورة نتيجة تفكير منطقي، بالإضافة إلى صعوبة وضع صيغ منطقية لكل المعارف.
- أنظمة تفكر كإنسان: إن التفكير كإنسان يهتم به علم الإدراك، أي تحديد نظرية دقيقة لعمل المخ. ولكن إن أردنا القول بأن نظام ما يفكر كإنسان، يجب تحديد أولاً كيف يفكر الإنسان، ومنه الدخول لآلية عمل المخ البشري. يعتبر هذا غاية في الصعوبة فحتى يومنا هذا لم يستطع العلم التوصل لوضع آلية عن كيفية وآلية تفكير الانسان.
- أنظمة تؤثر كإنسان: من أشهر هذه الأنظمة اختبار تورينغ، الذي وضعه العالم Alan Turing عام 1950. حيث يقوم شخص معزول بكتابة سؤال، وخلف الجدار يوجد آلة وانسان عليهم الإجابة، وعلى صاحب السؤال اكتشاف هل المجيب آلة أم انسان كما في الشكل 4.



الشكل 4 اختبار تورينغ

وقد تنبأ انه في عام 2000 سيكون احتمال أن الآلة ستخدع الإنسان في سلوكها مدة 5 دقائق هو 30 بالمائة، وقد تم التوصل لذلك في العديد من الأنظمة.

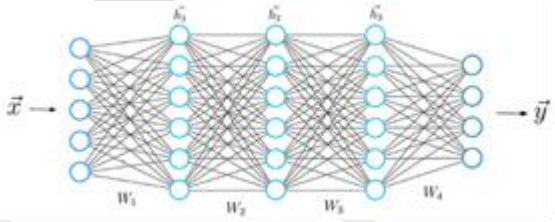
- أنظمة تؤثر منطقياً: إن السلوك المنطقي قد يكون نتيجة تحليل منطقي، أو قد يكون رد فعل لا يحتاج للكثير من التفكير. ومنه إن تنفيذ إجراء صحيح يعني أن يتم اختيار الإجراء الذي يجعل عملية الوصول إلى الهدف مثالياً مع أخذ المعلومات المتوفرة بعين الاعتبار، ومن هنا يأتي مفهوم العميل الذكي.

III. تعلم الآلة MACHINE LEARNING:

تعلم الآلة جزء مهم من الذكاء الصنعي، وهو أحد فروعه الذي يُعنى بجعل الحاسب قادراً على التعلم من تلقاء نفسه من الخبرات والتجارب السابقة، مما يجعله قادراً على التنبؤ واتخاذ القرار المناسب بصورة أسرع. تم استخدام مصطلح تعلم الآلة لأول مرة في عام 1959 من قبل Arthur Samuel، الأميركي الرائد في مجال الذكاء الصنعي وألعاب الكومبيوتر، عمل في شركة International Business Machines Corporation (IBM) في الستينات، حيث وصف تعلم الآلة بأنه "القدرة على التعلم بدون برمجة صريحة".

يمكن الحصول على الذكاء الصنعي بدون استخدام تعلم الآلة، لكن هذا يتطلب بناء ملايين الأسطر من المقاطع البرمجية بقواعد معقدة. لذا بدلًا من إجراء برامج تحوي معلومات محددة

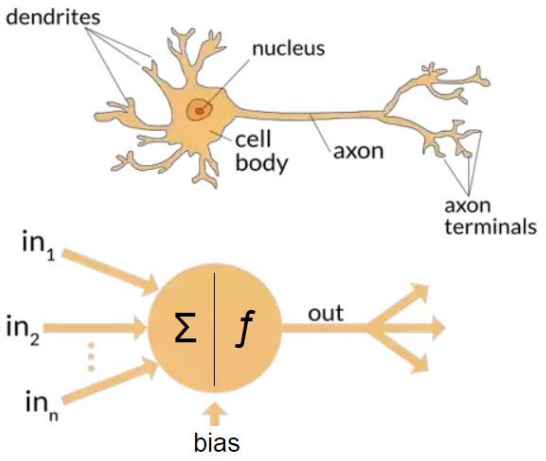
لإنجاز مهمة معينة، فإن تعلم الآلة هو طريقة تدريب الخوارزمية حتى تتمكن من التعلم. يشمل التدريب تغذية الخوارزمية بكميات كبيرة من البيانات، والسماح لها بضبط متحولاتها وتحسين نتائجها. أحد أهم طرقها استخدام الشبكات العصبونية الاصطناعية كما في الشكل 5 وهي تشبه آلية عمل العصبونات الحيوية في الدماغ البشري.



الشكل 5 الشبكة العصبونية الاصطناعية

تتألف الشبكة العصبونية من مجموعة خلايا عصبية (عصبونات) مترابطة مع بعضها، حيث تعرف الخلية العصبية الحيوية بأنها وحدة معالجة بسيطة لديها العديد من المدخلات تسمى الاستطالات وخرج يسمى المحور. إن المحور في نهايته يتفرع إلى الألاف من الفروع، كل منها يؤثر على خلايا عصبية أخرى في المشبك.

وبشكل مشابه تم تعريف الخلية العصبية الاصطناعية بشكل يحاكي الخلية العصبية الحيوية، حيث كل خلية عصبية تتلقى الإشارة من الخلايا العصبية المجاورة وبعد أجراء عمليات رياضية في نواة الخلية يكون خرج الخلية هو دخل لخلايا عصبية مجاورة أخرى كما في الشكل 6.



الشكل 6 الخلية العصبية الحيوية والخلية العصبية الاصطناعية

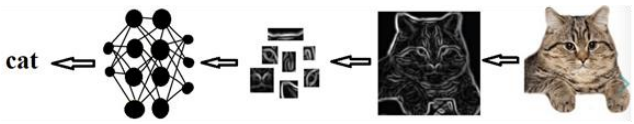
يمكن أن يقود تعلم الآلة إلى انجاز مجموعة متنوعة من التطبيقات الآلية، وفي مجالات مختلفة مثل التنبؤ بالطقس، التعرف على الوجوه، تشخيص الأمراض، وغيرها.. يتطلب تعلم الآلة الكثير من الرياضيات المعقدة وخاصة من عمليات الاشتقاق، وأيضاً تشمل العديد من الخوارزميات التقليدية خوارزميات العنقدة والتصنيف التي ليست بجديدة، البعض منها يعود للستينات مثل مصنف Naïve Bayes و Support Vector Machines. بالإضافة إلى خوارزميات تقليل أبعاد البيانات مثل خوارزميات principal component analysis، tSNE.

يمكن مقارنة النتائج التي تم الحصول عليها من النموذج المصمم مع النتائج الحقيقية في الواقع، وضبط المتحولات لتقليل قيمة الخطأ في المرة التالية. بحيث يصبح الفرق بين الهدف المرجو وخرج النموذج أصغر ما يمكن، ومنه يمكن القول إن تعلم الآلة يسعى لبناء خوارزميات قادرة على تقليل الخطأ في التصنيف والتنبؤ ومنه قادرة على التعلم.

أ.تطبيق عملي:

لنوضح هذا بمثال كما في الشكل 7، من خلال استخدام تعلم الآلة لإجراء تحسينات جذرية في مجال رؤية الحاسب. بداية يتم تحديد البيانات، حيث تُجمع آلاف الصور تحوي كائن ما، على سبيل المثال قطة، وأخرى صور تحوي كلب. لتأتي مرحلة هامة في بناء نموذج التصنيف، وهي معالجة الصورة واستخراج الصفات المميزة، في مثالنا هذا سيكون كل pixel في الصورة هو دخل للشبكة العصبونية التي تشبه الشبكات العصبية في الدماغ البشري. وبعد انتهاء مرحلة التدريب يمكن للنموذج تصنيف الصور وإعطاء نتيجة فيما إذا الصورة تحوي قطة أم كلب. عندما تصبح دقة التصنيف مرتفعة كفاية، يمكن القول إن الآلة "تعلّمت" كيف تميز الصور التي تحوي قطة من الصور

التي تحوي كلب، حتى لو أعطيت صوراً جديدة لأول مرة لم تستخدم في مرحلة التدريب.



الشكل 7 تصنيف صورة تحوي قطة باستخدام تعلم الآلة.

فيما يلي المقطع البرمجي بلغة python لبناء شبكة عصبونية وتدريبها مؤلفة من أربع طبقات وطبقة دخل حيث عدد عصبونات الدخل 3600 وعدد عصبونات الطبقة المخفية الأولى 20 عصبون، الطبقة المخفية الثانية 7 عصبونات والطبقة المخفية الثالثة 5 عصبونات وطبقة الخرج عصبون وحيد. تقوم بتصنيف صور فيما إذا كانت قطة أو كلب، حيث تتألف مجموعة التدريب من 11500 صورة لقطة، و11500 صورة لكلب، وكانت توابع التفعيل على التتالي linear, relue, linear, sigmoid.

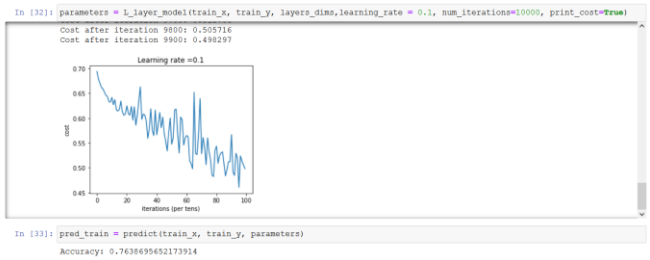
```
import pickle
import numpy as np
from dnn_app_utils import *
%load_ext autoreload
%autoreload 2
np.random.seed(1)
train_x_orig = pickle.load(open('X.pkl', 'rb'))
train_y = pickle.load(open('y.pkl', 'rb'))
train_y = train_y.reshape(1, len(train_y))
classes = ['Cat', 'Dog']
# Explore the dataset
m_train = train_x_orig.shape[0]
num_px = train_x_orig.shape[1]
print ("Number of training examples: " + str(m_train))
print ("Each image is of size: (" + str(num_px) + ", " +
str(num_px) + ", 3)")
print ("train_x_orig shape: " + str(train_x_orig.shape))
print ("train_y shape: " + str(train_y.shape))
# Reshape the training examples
train_x_flatten =
train_x_orig.reshape(train_x_orig.shape[0], -1).T
# Standardize data to have feature values between 0 and
1.
train_x = train_x_flatten / 255.
print ("train_x's shape: " + str(train_x.shape))
layers_dims = [3600, 20, 7, 5, 1] #  5-layer model
# GRADED FUNCTION: n_layer_model
def L_layer_model(X, Y, layers_dims,
learning_rate=0.075, num_iterations=3000,
print_cost=False): #lr was 0.009
    np.random.seed(1)
    costs = []
    # Parameters initialization.
    parameters = initialize_parameters_deep(layers_dims)
    # Loop (gradient descent)
    for i in range(0, num_iterations):
        # Forward propagation: [LINEAR -> RELU]*(L-1) -
> LINEAR -> SIGMOID.
        AL, caches = L_model_forward(X, parameters)
        # Compute cost.
        cost = compute_cost(AL, Y)
        # Backward propagation.
        grads = L_model_backward(AL, Y, caches)
        # Update parameters.
        parameters = update_parameters(parameters,
grads, learning_rate)
        # Print the cost every 100 training example
        if print_cost and i % 100 == 0:
            print ("Cost after iteration %i: %f" % (i, cost))
        if print_cost and i % 100 == 0:
            costs.append(cost)
    # plot the cost
    plt.plot(np.squeeze(costs))
    plt.ylabel('cost')
    plt.xlabel('iterations (per tens)')
    plt.title("Learning rate =" + str(learning_rate))
```

```
    plt.show()
    return parameters
# Run the training
parameters = L_layer_model(train_x, train_y,
layers_dims,learning_rate = 0.1, num_iterations=100,
print_cost=True)
## compute accuracy
pred_train = predict(train_x, train_y, parameters)
## write the parameters to csv file as result.
file = 'fifth.csv'
with open(file, 'w') as csv_file:
    writer = csv.writer(csv_file)
    for key, value in parameters.items():
        writer.writerow([key, value])
```

وبعد التدريب تم الحصول على جدول النتائج التالي:

Learning accuracy	Num iterations	accuracy
0.075	2500	0.6708
0.0075	5000	0.3691
0.02	5000	0.6764
0.2	5000	0.7237
0.1	10000	0.7639

الجدول 1 نتائج تدريب الشبكة العصبونية



الشكل 8 نتائج التدريب بعد 10000 تكرار ومعامل تعلم 0.1

IV. التعلم العميق DEEP LEARNING:

يعد التعلم العميق أحد أهم الحقول الفرعية لتعلم الآلة، تم استخدام مصطلح "التعلم العميق" من قبل Rina Dechter عام 1986. هذا العلم يعتمد على الشبكات العصبونية الاصطناعية العميقة، التي تتطلب كمية كبيرة من البيانات في عملية التعلم وحل المشكلات.

التعلم العميق مستوحى من بنية الدماغ ووظائفه، حيث يحاكي البنية البيولوجية للدماغ. في الشبكات العصبية العميقة، هناك خلايا عصبية مرتبة في طبقات منفصلة واتصالات مع خلايا عصبية أخرى. كل طبقة مسؤولة عن خاصية للتعلم، مثل المنحنيات والحواف وغيرها في تصنيف الصور. هذه الطبقات هي التي تعطي التعلم العميق اسمه، "العمق" ينشأ من استخدام طبقات متعددة المهام بدلًا من طبقات ارتباط .

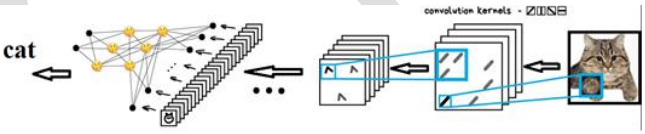
تتمتع الشبكات العصبية الاصطناعية العميقة بقدرات فريدة تمكّن نماذج التعلم العميق من حل المهام التي لا يمكن لنماذج التعلم الآلي حلها. جميع التطورات الحديثة في الذكاء تعود إلى التعلم العميق بدونه لن يكون لدينا سيارات ذاتية القيادة أو روبوتات محادثة أو مساعد افتراضي شخصي يمتاز بالذكاء، كما في الشكل 8 مثل Alexa و Siri.

الشكل 9 المساعد الافتراضي الذكي siri

في الواقع، يتضمن التعلم العميق مجموعة فرعية من أنظمة تعلم الآلة، التي تشكل قدرات التشغيل التي تملكها الآلات. وتشبه هذه التقنية نظام تعلم الآلة في بعض السياقات، ولكن يكمن الفرق في أن تعلم الآلة يحتاج إلى بعض الإجراءات التي يقوم بها المبرمج لأداء المهمة، في حين يستطيع التعلم العميق أداء المهمة دون تدخل المبرمج. وعند التعامل مع نظام تعلم الآلة، ينبغي على المبرمج تعديل الخوارزمية إذا كانت النتائج غير مناسبة، لكن بالنسبة لنماذج التعلم العميق، فإنها تتكفل بذلك بنفسها، تمامًا مثل العقل البشري.

أ.تطبيق عملي

لنعود إلى مثال تصنيف الصور التي تحوي قطة أو كلب. عند استخدام تعلم الآلة بعد معالجة الصورة واستخراج الصفات المميزة، يتم بناء نموذج شبيه بالشبكات العصبية في الدماغ البشري، يمكنه بعد التدريب تصنيف الصور بدقّة. ولكن باستخدام التعلم العميق، كما في الشكل 9، يتم استخدام الصورة مباشرة دون إجراء أي سلسلة عمليات، كدخل للشبكة العصبونية العميقة. وبعد سلسلة من عمليات التدريب تصبح الشبكة جاهزة لإعطاء الخرج المناسب.



الشكل 10 تصنيف صورة تحوي قطة باستخدام التعلم العميق

فيما يلي المقطع البرمجي بلغة python لبناء شبكة عصبونية عميقة cnn وتدريبها. المقطع البرمجي التالي لمعالجة الصور في مرحلة التدريب.

```
import numpy as np
import os
import cv2
import matplotlib.pyplot as plt
get_ipython().run_line_magic('matplotlib', 'inline')
DIRECTORY = r'C:\Users\balajiam\Documents\ML Data Analysis\Cats-Vs-Dog'
CATEGORIES = ['cat', 'dog']
data = []
for category in CATEGORIES:
path = os.path.join(DIRECTORY, category)
for img in os.listdir(path):
img_path = os.path.join(path, img)
label = CATEGORIES.index(category)
arr = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE)
new_arr = cv2.resize(arr, (60, 60))
data.append([new_arr, label])
```

```
import random
random.shuffle(data)
X = []
y = []
for features, label in data:
X.append(features)
y.append(label)
X = np.array(X)
y = np.array(y)
import pickle
pickle.dump(X, open('X.pkl', 'wb'))
pickle.dump(y, open('y.pkl', 'wb'))
```

المقطع البرمجي التالي لتدريب الشبكة العصبونية العميقة وقد حققت دقة 0.8936:

```
import pickle
X = pickle.load(open('X.pkl', 'rb'))
y = pickle.load(open('y.pkl', 'rb'))
X = X/255
X = X.reshape(-1, 60, 60, 1)
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Dense, Flatten
model = Sequential()
model.add(Conv2D(64, (3,3), activation = 'relu'))
model.add(MaxPooling2D((2,2)))
model.add(Conv2D(64, (3,3), activation = 'relu'))
model.add(MaxPooling2D((2,2)))
model.add(Flatten())
model.add(Dense(128, input_shape = X.shape[1:], activation = 'relu'))
model.add(Dense(2, activation = 'softmax'))
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
model.fit(X, y, epochs=5, validation_split=0.1)
```

المقطع البرمجي التالي للتنبؤ بأن صورة ما هي قطة أم كلب.

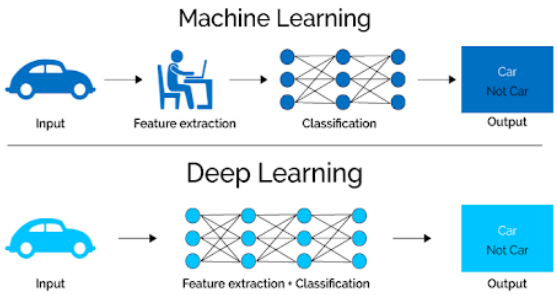
```
import cv2
```

```
from tensorflow import keras
import numpy as np
CATEGORIES = ['Cat', 'dog']
def image(path):
img = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
new_arr = cv2.resize(img, (60, 60))
new_arr = np.array(new_arr)
new_arr = new_arr.reshape(-1, 60, 60, 1)
return new_arr
model = keras.models.load_model('3x3x64-
catvsdog.model')
prediction = model.predict([image('test_dog.jpg')])
print(CATEGORIES[prediction.argmax()])
```

الخلاصة

إن تعلم الآلة والتعلم العميق مفاهيم مرتبطة بالذكاء الصنعي ولكن لابد من التمييز بينها عند استخدامها، ويمكن تلخيص الفرق بشكل واضح من خلال مثال آخر "تحديد فيما إذا صورة تحوي سيارة أم لا"، كما في الشكل 10.

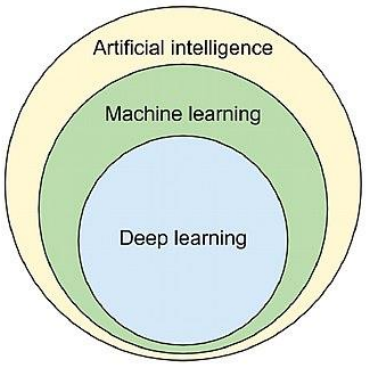
عند استخدام تعلم الآلة نحتاج أولاً إلى معالجة الصورة وتحديد الصفات المميزة والفريدة للسيارة من الشكل والحجم والنوافذ والعجلات... واستخراجها، لتكون بيانات دخل لنموذج التصنيف الذي هو عبارة عن شبكة عصبونية متعددة الطبقات. بينما في التعلم العميق يكفي أن تكون الصورة كدخل للشبكة العصبية العميقة، لنحصل على النتيجة المرجوة بعد عدد من مرات التدريب.



الشكل 11 تصنيف صورة تحوي سيارة باستخدام تعلم الآلة والتعلم العميق

ومنه أدى تعلم الآلة والتعلم العميق إلى قفزات هائلة في مجال الذكاء الاصطناعي في السنوات الأخيرة، وساهم النظامين بتوسيع آفاق الذكاء الاصطناعي في المستقبل. حيث يتطلب تعلم الآلة والتعلم العميق كميات هائلة من البيانات، التي يمكن جمعها من خلال مليارات من أجهزة الاستشعار المتصلة عبر الإنترنت. بالمقابل نجد أن تعلم الآلة يتطلب مراحل لاستخلاص الميزات بتدخل من المبرمج بينما استطاع التعلم العميق الاستغناء والتعامل مباشرة مع الصور من خلال شبكات عصبونية عميقة.

أخيراً يمكن القول بأن علاقة الذكاء الصنعي وتعلم الآلة والتعلم العميق علاقة احتواء، من حيث زمن ظهورها والمفاهيم المتعلقة بها كما في الشكل 11.



الشكل 12 العلاقة بين الذكاء الصنعي وتعلم الآلة والتعلم العميق

المراجع:

[1]. Oswald Campesato, Artificial Intelligence Machine Learning and Deep Learning, MERCURY LEARNING AND INFORMATION, Dulles, Virginia Boston, Massachusetts, New Delhi, 2020.
[2]. Stuart J. Russell and Peter Norvig, Artificial Intelligence A Modern Approach, 3nd ed., Upper Saddle River, New Jersey, 2010.

[3]. Wolfgang Ertel, Introduction to Artificial Intelligence, University of Applied Sciences, Weingarten, Germany: Springer-Verlag, 2011.
[4]. M. Tim Jones, Artificial Intelligence A Systems Approach, INFINITY SCIENCE PRESS LLC, Hingham, Massachusetts,New Delhi,2008.
[5]. Ben Coppin, Artificial Intelligence Illuminated, 1st ed, Jones and Bartlett Publishers,2004.
[6]. Rudolph Russell, machine learning, step-by-step guide to implement machine learning algorithms with python, 2018.
[7]. Manohar Swamynathan, Mastering Machine Learning with Python in Six Steps, Bangalore, Karnataka, India,2017.
[8]. Navin Kumar Manaswi, Deep Learning with Applications Using Python, Bangalore, Karnataka, India,2018.
[9]. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, United States of America,2017.
[10]. Himanshu Singh,Practical Machine Learning and Image Processing, Allahabad, Uttar Pradesh, Indi,2019

منشورات المؤلف:

[1]. Abusaleh, N., Saloum, k. and Nassar, H. (2015). Face Expression Classification Using Neuro-Fuzzy Controller. *Journal of the Baath University*, Volume 37.
[2]. Abusaleh, N., and Saloum, k., and Nassar, H. (2015). Face Expression Classification Using Fuzzy Controller. *Journal of the Baath University*, Volume 37.
[3]. Abusaleh, N., and Saloum, k., and Nassar, H. (2014). Face Expression Classification Using Neural Network and PCA algorithm. *Journal of the Baath University*, Volume 36.
[4]. Abusaleh, N., and Saloum, k., and Nassar, H. (2014). Face Expression Classification Using Neural Network. *Journal of the Baath University*, Volume 36.
[5]. Abusaleh, N., and Farah, H., and Nassar, H. (2009). Face Classification Using Neuro-Fuzzy System. *Journal of the Baath University*.