الجمهورية العربية السورية

المعهد العالي للعلوم التطبيقية والتكنولوجيا

قسم المعلوميّات

العام الدراسي 2022/2023

توليد الصور من التوصيف النصي بالاعتماد على تقنيات التعلم العميق

إعداد

اميرة منذر القطلبي

إشراف

د. وسيم صافي

9/9/2023

الإهداء

إلى رفاق الخطوة الأولى والخطوة الأخيرة..

إلى والديّ وإخوتي..

الشكر

أتقدم بجزيل الشكر إلى كل من وقف بجانبي، وساهم في نجاح هذا العمل.
أخص بالشكر الدكتور المشرف وسيم صافي لتوجيهاته ونصائحه طيلة فترة المشروع.
أخيراً، أشكر من كان سنداً ودعماً لي في كل اللحظات...أهلي وأصدقائي.

ملخص

تعد تقنية توليد الصور من التوصيفات النصية إحدى الابتكارات المذهلة في مجال الذكاء الصنعي وتعلم الآلة. تتيح هذه التقنية إمكانية تحويل وصف نصي دقيق لمشهد أو كائن إلى صورة واقعية تماثل المحتوى المذكور في الوصف. تتجلى أهمية هذا المجال في تقديم نهج جديد لتوليد المحتوى البصري، حيث يمكنها أن تجمع بين القدرات الإبداعية للإنسان في إنشاء الوصف النصي وبين قوة ودقة التعلم العميق والذكاء الاصطناعي في إنتاج صور واقعية. تمتد التطبيقات المتنوعة لهذا المجال من تحسين تجارب الألعاب والتصميم إلى تسهيل التعليم وتوضيح المفاهيم وإثراء التفاعل البشري مع التكنولوجيا. قدم هذا المشروع نظرة على أهم التقنيات المستخدمة في هذا المجال، كذلك تم في هذا المشروع بناء نموذج لتوليد الصور من التوصيف النصي وتدريبه من الصفر على عدد من مجموعات البيانات. تم أيضاً عرض النتائج التي تم الحصول عليها بشكل مفصل، بالإضافة إلى محاولة تفسير هذه النتائج والاستفادة منها في تحسين التدريب بحيث يتم الوصول لأفضل نتيجة يمكن الحصول عليها ضمن الموارد التقنية والعتادية المتاحة وأيضاً ضمن الزمن المخصص لتنفيذ المشروع. اعترض تنفيذ المشروع الكثير من التحديات منها الحاجة لوحدة معالجة بيانية متطورة وموارد حاسوبية ضخمة متاحة لأزمنة تدريب طويلة. في محاولة التغلب على هذه التحديات جرى حفظ النموذج خلال التدريب بشكل دوري كنقاط تدقيق مؤقتة checkpoints وذلك لاستئناف التدريب عليها في حال انقطاع التدريب. تم في النهاية الوصول إلى نتائج مقبولة نوعاً ضمن الموارد التقنية والعتادية المتاحة من حيث الربط بين التوصيف والصورة وكذلك من حيث وضوح الصورة. كانت نتيجة التدريب السابق ثلاثة نماذج جرى وضعها في الاستخدام من خلال مكاملتها مع موقع وب يتيح للمستخدم توليد الصورة الموافقة للتوصيف النصي الذي يدخله، كذلك يتيح له تحميلها أيضاً في حال أراد ذلك.

Abstract

The technology of text-to-image generation is one of the amazing innovations in the field of artificial intelligence and machine learning. This technology makes it possible to transform an accurate textual description of a scene or object into a realistic image that matches the content mentioned in the description. The importance of this field is evident in providing a new approach to generating visual content, as it can combine the creative capabilities of humans in creating textual descriptions with the power and accuracy of deep learning and artificial intelligence in producing realistic images. The diverse applications of this field extend from improving gaming and design experiences to facilitating education, clarifying concepts, and enriching human interaction with technology. This research provided an overview of the most important techniques used in this field. In this research, a model for generating images from text descriptions was built and trained from scratch on a number of datasets. The results obtained were also mentioned in detail, in addition to an attempt to interpret these results and benefit from them in improving training so that the best result that can be obtained is achieved within the available technical and hardware resources and also within the time allocated to implement the project. The implementation of the project faced many challenges, including the need for an advanced graphic processing unit and huge computer resources available for long training periods. In an attempt to overcome these challenges, the model was saved periodically during training as checkpoints in order to resume training on it in the case of training interruption. In the end, somewhat acceptable results were achieved within the

available technical and hardware resources in terms of linking description and image, as well as in terms of image clarity. The result of the previous training were three models that were put into use by integrating them with a website that allows the user to generate the image corresponding to the textual description he enters, and also allows him to download it if he wants to do so.

جدول المحتويات

قائمة الأشكال

قائمة الجداول

جدول الاختصارات

الاختصارات	اللغة العربية	اللغة الانجليزية
AI	ذكاء صنعي	Artificial Intelligence
ML	تعلم الآلة	Machine Learning
DL	التعلم العميق	Deep Learning
NN	الشبكات العصبونية الصنعية	Neural Network
SGD	الانحدار التدريجي العشوائي	Stochastic Gradient Descent
CNN	الشبكات العصبونية التلافيفيّة	Convolutional Neural Network
GAN	شبكات الخصومة التوليدية	Generative Adversarial Net
cGAN	شبكات الخصومة التوليدية الشرطية	Conditional Generative Adversarial Net
RNN	شبكة عصبونية عوديّة	Recurrent Neural Network
GRU	الوحدة العوديّة المبوّبة	Gated Recurrent Unit
DDPM	نماذج انتشار تقليل الضجيج الاحتمالية	Denoising Diffusion Probabilistic Model
ReLU	تابع الوحدة الخطية المصحَّحة	Rectified Linear Unit
Leaky ReLU	تابع الوحدة الخطية المصحَّحة المتسرّبة	Leaky Rectified Linear Unit
MSE	تابع متوسط الخسارة المربّعة	Mean Squared Error
MAE	تابع متوسط الخسارة المطلقة	Mean Absolute Error
Adam	خوارزمية آدم (تقدير اللحظة التكيفية)	Adaptive Moment Estimation
COCO	الكائنات المشتركة في السياق / مجموعة بيانات	Common Objects in Context
GPU	وحدة معالجة الرسومات	Graphics processing unit

جدول المصطلحات

اللغة العربية	اللغة الانجليزية
شبكة ذات تغذية أمامية	Deep Feedforward Network
انتشار تقدّمي / أمامي	Forward Propagation
انتشار تراجعي / عكسي	Backward Propagation
معدل التعلم	Learning rate
تدرّج	Gradient
إنتروبيا متقاطعة	Cross Entropy
تابع الخسارة	Loss function
تابع التنشيط	Activation function
فرط الملائمة	Overfitting
الرؤية الحاسوبية	Computer Vision
التعلم الخاضع للإشراف	Supervised Learning
طبقة تجميع	Pooling Layer
التعلم غير الخاضع للإشراف	Unsupervised Learning
مرشّح	Filter
خطوة	Stride
خريطة واصفات	Features Map
حشو	Padding
التعلم شبه الخاضع للإشراف	Semi-supervised Learning
سلسلة ذات اعتمادية طويلة	Long term dependency sequence
التعلم المعزز	Reinforcement Learning
عملية الانتشار الأمامي	Forward diffusion process
عملية تقليل الضجيج	Reverse diffusion process
تضمين (ترميز كأشعّة)	Embedding
تجميع بأخذ القيمة المتوسطة	Global Average Pooling
عملية التقطيع لرموز	Tokenization
تلاشي التدرجات	Vanishing Gradients

Multi-Head Self-Attention	الانتباه الذاتي متعدد الطبقات
Token	رمز

الفصل 1. مقدمة عن المشروع

نستعرض في هذا الفصل تمهيد يوضّح فكرة المشروع وأهدافه وأهميته.

1.1. تمهيد

يشهد العالم مؤخراً تطوراً ملحوظاً في تقنيات الذكاء الصنعي، حيث انطلقت مجالات جديدة ومبهرة تحقق مزيجاً مذهلاً من الإبداع البشري والتقنيات الحديثة. من هذه المجالات الرائجة التي تثير الدهشة هي توليد الصور من التوصيف النصي، الذي يقوم على تحويل نص يصف مشهداً أو فكرة معينة إلى صورة واقعية تجسد تلك الفكرة تماماً باستخدام تقنيات الذكاء الصنعي والتعلم العميق. تساهم تقنية توليد الصور هذه في إثراء تجارب الألعاب من خلال خلق عوالم غامرة بالتفاصيل والإثارة، تسهيل عمليات التصميم والإبداع، إضفاء واقعية وجمالية على البيئات الافتراضية وتحسين تجربة المستخدم، وصولاً إلى تعزيز عملية التعليم و توضيح المفاهيم التعليمية من خلال صور توضيحية دقيقة وواضحة وغيرها الكثير من الاستخدامات الأخرى. يلقي هذا العمل نظرة على أهم الأوراق البحثية التي تسلط الضوء على كيفية عمل هذه التقنيات والأسس العلمية وراءها، هذا ويقدم أيضاً نبذة عن التحديات التي تعترض العمل في هذا المجال، مثل تقديم التفاصيل البصرية الدقيقة وضمان تطابقها مع النص المصدر والكثير من المشاكل الأخرى سيجري ذكرها لاحقاً.

2.1. أهداف المشروع

بناءً على ما سبق ذكره في التمهيد، سنهدف في مشروعنا إلى القيام بما يلي:

1- بناء وتدريب نماذج تعلم عميق قادرة على توليد الصور انطلاقاً من التوصيف النصي، وذلك بناءً على مجموعات المعطيات المتوافرة وضمن الموارد التقنية والعتادية Hardware المتاحة.

2- الاستفادة من عملية تحليل نتائج نموذج معين في تطوير نموذج جديد أفضل وذلك بغرض الوصول لأفضل نتيجة ممكنة.

3- تفسير الأداء الذي تقدّمه هذه النماذج والمقارنة بينها.

4- بناء موقع وب نوظف فيه النماذج التي أعطت أفضل النتائج.

الفصل 2. الدراسة النظرية

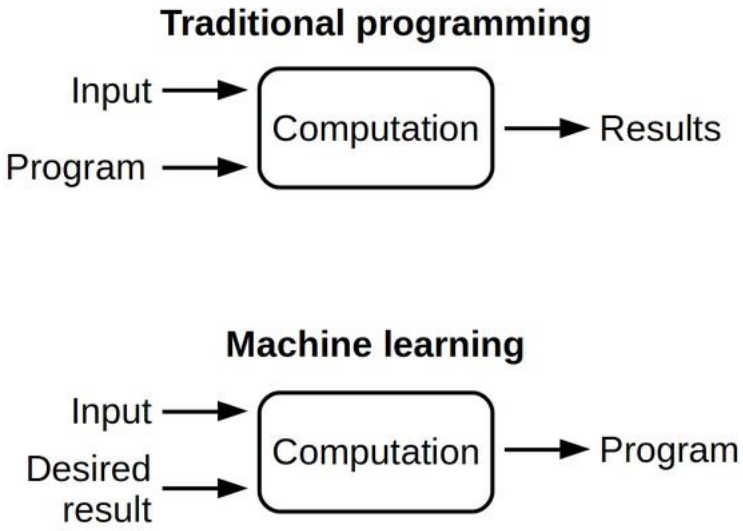
نستعرض في هذا الفصل مفهوم الذكاء الصنعي، مروراً بمفهوميّ تعلم الآلة والتعلم العميق، يلي ذلك شرح نظري عن الشبكات العصبونية الصنعية وبعض أنواعها المستخدمة في مجال توليد الصور من التوصيفات النصيّة. ونختم بالحديث عن كيفية استخدام هذه الشبكات في عملية التعلم.

1.2. الذكاء الصنعي Artificial Intelligence

الذكاء الصنعي هو مجال من مجالات العلوم الحاسوبية يهدف إلى إنشاء نظم وبرمجيات قادرة على محاكاة الذكاء البشري والقيام بالأنشطة التي تتطلب تفكيراً ذكياً [1]. يتناول هذا المجال الاستفادة من القدرات العقلية للإنسان وتطبيقها في النظم الحاسوبية. تتمحور فكرة الذكاء الصنعي حول بناء نظم تكنولوجية تتمتع بالقدرة على تحليل البيانات والمعلومات بطرق مشابهة لتلك التي يتبعها الإنسان، واتخاذ قرارات مستنيرة بناءً على تلك التحليلات. يعتمد الذكاء الصنعي بشكل أساسي على مفهوم تعلم الآلة Machine Learning، وهو فرع من الذكاء الاصطناعي يسمح للنظم بتعديل سلوكها وأدائها بناءً على البيانات التي تتعامل معها، بدلاً من الاعتماد على برمجة صرفة تماماً [1]. في الوقت الحالي، يعتبر الذكاء الصنعي من التقنيات الحيوية والمتطورة التي لها تأثير كبير على العديد من المجالات والصناعات مثل التجارة الإلكترونية والرعاية الصحية والتصنيع والنقل والموارد البشرية والألعاب والتسويق الرقمي وغيرها، حيث يتم تطوير تطبيقات الذكاء الصناعي لتحسين الكفاءة واتخاذ قرارات أفضل وتسهيل المهام التي قد تكون صعبة على البشر. ولكن برغم التقدم العلمي والتقني الضخم الذي يشهده العالم لا تزال تعترض هذا المجال عوائق وتحديات عديدة من أبرزها التحديات التقنية مثل قوة المعالجة والتخزين والتحسين المستمر للأداء.

1.1.2. تعلم الآلة Machine Learning

تعلم الآلة هو فرع من الذكاء الاصطناعي يتعلق بإنشاء نماذج ونظم قادرة على تعلم الأنماط والقوانين من البيانات واستخدام هذا التعلم لاتخاذ القرارات أو التنبؤ بالنتائج [1]. يُستخدم تعلم الآلة في مجموعة واسعة من التطبيقات مثل تمييز البريد الإلكتروني المزعج أو الضار، توصيات الأفلام على منصات الترفيه، التعرف على الصوت والنص، القيادة الذاتية للمركبات، والعديد من المجالات الأخرى. يوضح الشكل 1 الفرق بين خوارزميات تعلم الآلة والخوارزميات التقليدية.



الشكل 1. الفرق بين خوارزميات تعلم الآلة والخوارزميات التقليدية

بالنسبة للأقسام التي يتكون منها مجال تعلم الآلة يمكن تقسيمها إلى أربعة أقسام رئيسية:

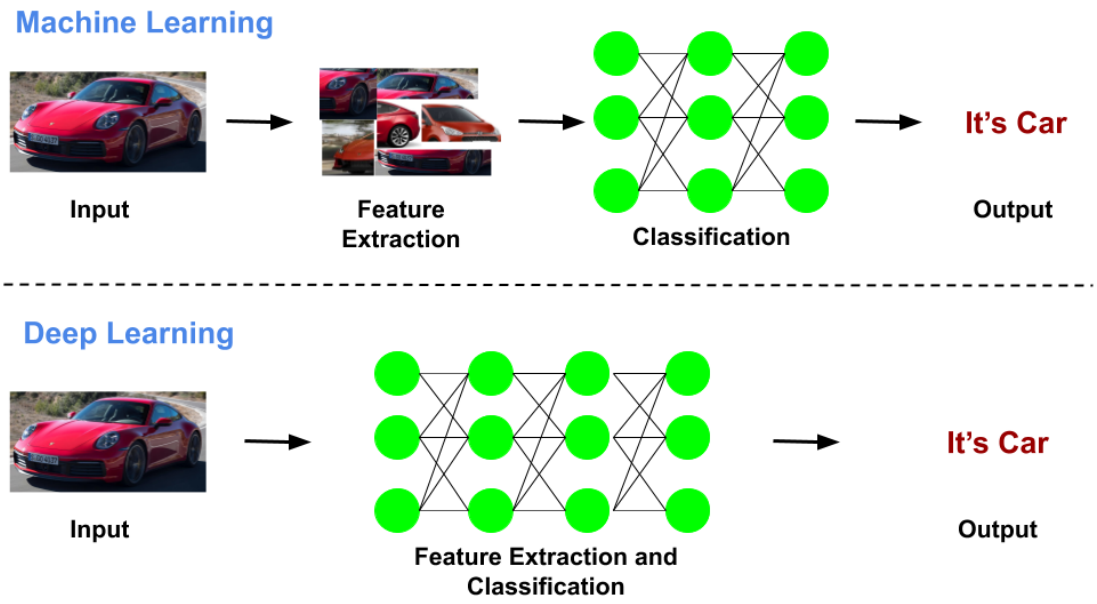
- التعلم الخاضع للإشراف Supervised Learning: يتم تدريب النماذج باستخدام أزواج من البيانات المكونة من المدخلات والمخرجات المتوقعة، حيث يحاول النموذج التعرف على السياق وإيجاد الربط بين الدخل والخرج للتنبؤ بالمخرجات المناسبة للمدخلات الجديدة.
- التعلم غير الخاضع للإشراف Unsupervised Learning: على النقيض من التعلم الخاضع للإشراف، يتم تدريب النماذج على البيانات دون وجود مخرجات متوقعة، حيث يحاول النموذج اكتشاف الترتيبات والأنماط الكامنة في بيانات الدخل عن طريق خوارزميات معينة مثل التجميع (تجميع نقاط البيانات المتشابهة معاً) وتقليل الأبعاد (ضغط البيانات عالية الأبعاد في تمثيل منخفض الأبعاد). يستخدم هذا النوع من التعلم عادة في مهمة الكشف عن الهيكل الأساسي للبيانات.
- التعلم شبه الخاضع للإشراف Semi-supervised Learning: يجمع بين التعلم الخاضع للإشراف والتعلم غير الخاضع للإشراف. في هذا النوع من التعلم، تتوفر مجموعة من البيانات تحتوي على مدخلات مرتبطة بتصنيفات (مخرجات متوقعة)، ومجموعة من البيانات غير مصنفة. عادةً، يكون هدف التعلم شبه الخاضع للإشراف هو استخدام البيانات غير المصنفة مع البيانات المصنفة لتحسين أداء النموذج الذي يقوم بتصنيف البيانات. واحدة من

الطرق الشائعة للقيام بذلك هي استخدام البيانات المصنفة لتدريب النموذج، ثم استخدام البيانات غير المصنفة لتحسين النموذج عن طريق تعديل الأوزان. يتميز التعلم شبه الخاضع للإشراف بقدرته على تحسين أداء النموذج بشكل كبير عندما تكون البيانات المصنفة محدودة وتكون البيانات غير المصنفة بكميات هائلة، وهذا هو السبب في أنه يُستخدم في العديد من التطبيقات مثل التصنيف النصي والتعرف على الكلام والتحسينات في تطبيقات الرؤية الحاسوبية والمزيد.

- التعلم المعزز Reinforcement Learning: يتم في هذا النوع من التعلم تدريب النماذج باستخدام نظام مكافآت وعقوبات، حيث يتعلم النموذج من خلال التفاعل مع بيئته واتخاذ الإجراءات التي تزيد من المكافآت وتقلل من العقوبات. الهدف هو أن يتعلم النموذج تحسين سلوكه عن طريق تحسين الاستراتيجية التي يتبعها بناءً على التجارب السابقة والمكافآت والعقوبات المتلقاة.

2.1.2. التعلم العميق Deep Learning

التعلم العميق هو مجموعة من تقنيات تعلم الآلة تقوم على الشبكات العصبونية الصنعية متعددة الطبقات. تمثل الشبكات العصبونية نموذجاً مستوحى من بنية الشبكة العصبية في الدماغ البشري، والتي تتكون من طبقات متتالية من العقد (الوحدات الحسابية) المرتبطة بالمدخلات والمخرجات. يتم تدريب هذه الشبكات باستخدام مجموعة ضخمة من البيانات، حيث تقوم النماذج باكتشاف السياقات والأنماط في البيانات تلقائياً دون الحاجة لتحديد الخصائص بشكل يدوي [1]. تتعرف نماذج التعلم العميق على الأنماط المعقدة في الصور والنصوص والأصوات والبيانات الأخرى لإنتاج رؤى وتنبؤات دقيقة. تُستخدَم أساليب التعلم العميق في أتمتة المهام التي تتطلب عادةً ذكاءً بشرياً، مثل وصف الصور أو تفريغ ملف صوتي إلى نص وما إلى ذلك. يوضح الشكل 2 الفرق بين تعلم الآلة والتعلم العميق.



الشكل 2. الفرق بين تعلم الآلة والتعلم العميق

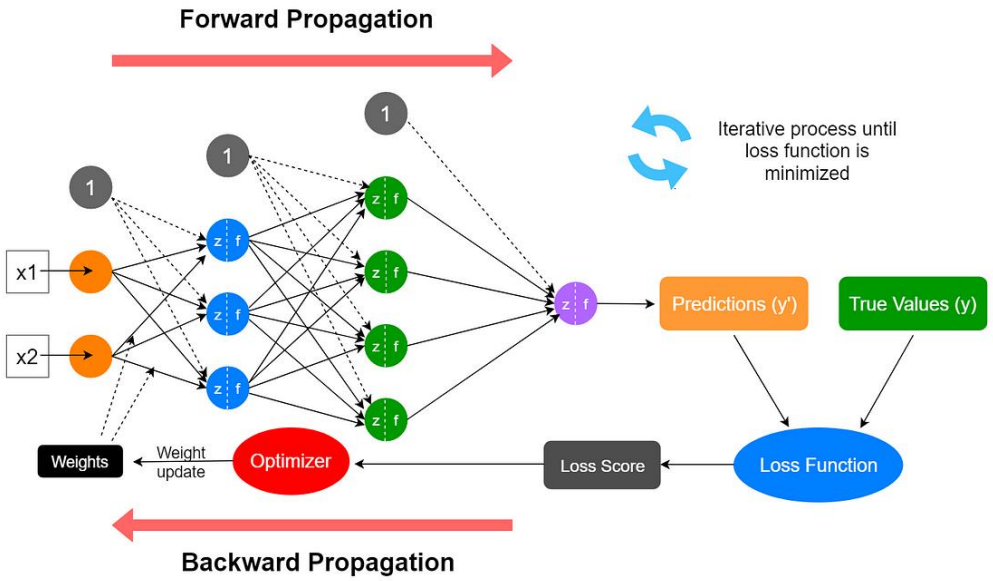
2.2. الشبكات العصبونية الصنعية Neural Networks (NN)

الشبكات العصبونية الصنعية هي نماذج حاسوبية تحاكي الجهاز العصبي البيولوجي. تتضمن الشبكة العصبونية الصنعية وحدات حسابية تعرف باسم عصبونات Perceptrons، يتم تنظيم هذه الوحدات في طبقات متتالية [2]. تتألف الشبكة العصبونية عموماً من ثلاثة أنواع من الطبقات:

1. طبقة الدخل Input Layer: هي الطبقة الأولى في الشبكة وتقوم بتلقي المدخلات من البيانات وتمريرها إلى الطبقات التالية.
2. الطبقات الخفية Hidden Layers: عبارة عن طبقات موجودة بين طبقة الدخل وطبقة الخرج، وهي المكان الذي يحدث فيه التعلم واستخراج المميزات من البيانات.
3. طبقة الخرج Output Layer: هي الطبقة الأخيرة في الشبكة التي تقوم بإخراج النتائج أو التصنيفات المتوقعة بناءً على المدخلات والمعالجات الحسابية التي تمت في الطبقات الخفية.

يرتبط الدخل مع العصبونات في الطبقة الأولى عن طريق وصلات connections لكل منها وزن weight، وتتصل كذلك العصبونات مع بعضها عن طرق وصلات موزونة. يتم حساب الخرج عن طريق عملية تدعى الانتشار الأمامي Forward Propagation بحيث يكون دخل كل عصبون هو عبارة عن مجموع الخرج من الوصلات التي تدخل إليه

مثقلاً كل منها بوزن معين بالإضافة لمعامل ثابت يدعى الانحياز Bias، ومن ثم يخضع الدخل لعملية معالجة من قبل العصبون عن طريق تطبيق تابع تنشيط Activation Function عليه (سوف نتحدث عن آلية اختيار توابع التنشيط في فقرة لاحقة)، وبعد ذلك يتم تمرير الخرج لعصبون آخر وهكذا وصولاً إلى طبقة الخرج Output Layer. يحدث التعلم عن طريق تغيير أوزان الوصلات التي تصل العصبونات بالاعتماد على أمثلة معطيات التدريب Training Data التي تتضمن أزواج الدخل-خرج للتابع الذي يجب على الشبكة العصبونية تعلمه. تعطي معطيات التدريب تغذية راجعة Back Propagation حيال صحة الأوزان التي تم اختيارها بالاعتماد على مقدار صحة الخرج الذي توقعته الشبكة العصبونية من أجل دخل معين بالمقارنة مع القيمة الحقيقة للخرج الموافقة لنفس هذا الدخل في معطيات التدريب [2]. يوضح الشكل 3 بنية الشبكة العصبونية وآلية التعلم المستخدمة فيها.



الشكل 3. بنية الشبكة العصبونية وآلية التعلم المستخدمة فيها

1.2.2. الشبكات العصبونية كاملة الارتباط Fully Connected Neural Networks

تُعرف أيضاً باسم الشبكات العصبونية الكثيفة Dense Neural Networks. يعتبر هذا النموذج هو الأكثر بساطة وشيوعاً في مجال تعلم الآلة. في هذا النوع من الشبكات يكون كل عصبون متوضع ضمن طبقة معينة مرتبطة مع كل عصبونات الطبقة السابقة [2]. تُعد الشبكات العصبونية الكاملة الارتباط قوية للعديد من المشاكل والتطبيقات، ولكنها تتطلب عدداً كبيراً من الموسطات parameters عند تزايد عمق الشبكة، مما يزيد من الحمل الحسابي بشكل كبير. لهذا

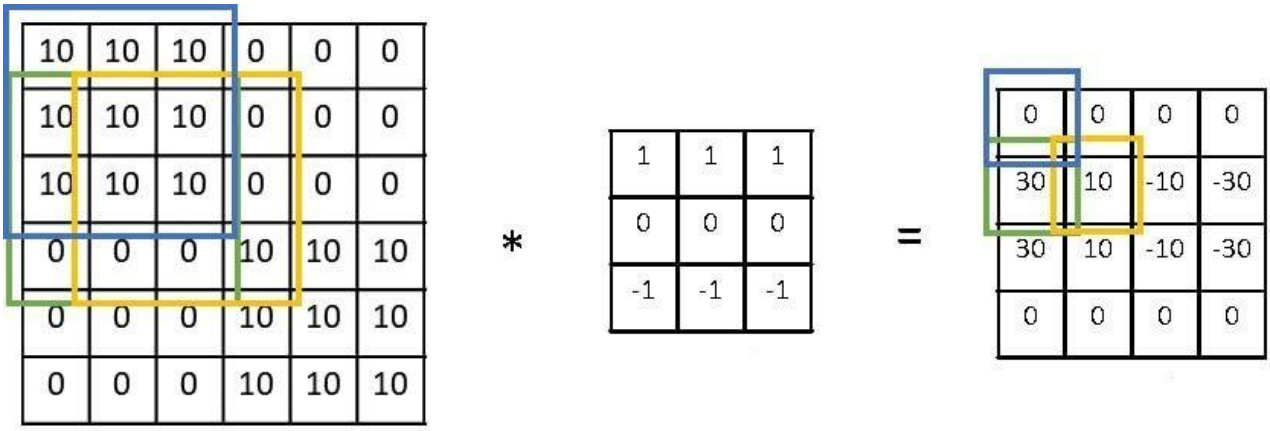
السبب، تُستخدَم هذه الشبكات في العديد من التطبيقات التي تتطلب معالجة بيانات ذات أبعاد منخفضة مثل مسائل التصنيف البسيطة.

2.2.2. الشبكات العصبونية التلافيفيّة Convolutional Neural Networks (CNN)

عندما نشأ مجال الرؤية الحاسوبية Computer Vision وفكرة التعامل مع المعطيات البصرية، لم تعط الشبكات العصبونية كاملة الارتباط فائدة مرجوة على الإطلاق. كما رأينا في الفقرة السابقة، فإن هذا النوع من الشبكات يتألف من عدة طبقات كل طبقة فيها عدة عصبونات وكل عصبون متصل بكل العصبونات في الطبقة السابقة. فمثلاً عندما يكون الدخل صورة ذات حجم 200x200 بكسل مثلاً (يعتبر هذا الحجم للصورة عادي وليس كبير)، فإن عدد بارامترات الدخل سيكون 200x200x3 (بسبب وجود ثلاث قنوات لونية R,G,B)، وهو عدد هائل جداً لكي يتم التعامل معه واستخراج المعلومات منه، وحتى في حال استطاع النموذج التعامل معه فإنه غالباً سوف يعاني من مشكلة فرط الملائمة Overfitting، والتي تحدث عندما يلائم النموذج معطيات التدريب بشكل جيد جداً بحيث ينخفض أداؤه على معطيات التحقق، أي المعطيات التي لم يراها من قبل. لذلك نلاحظ أن هذا النوع من الشبكات غير قادر على التعامل مع المعطيات البصرية ومن هنا جاءت فكرة الشبكات العصبونية التلافيفية التي استطاعت حل هذه المشكلة وتقديم طريقة قادرة على معالجة هذا النوع من المعطيات عن طريق الاعتماد على عملية جداء التلاف.

1.2.2.2. طبقة جداء التلاف Convolutional Layer

تعمل هذه الطبقات كمرشحات تقوم بمعالجة الصور وتمريرها إلى الطبقات اللاحقة، تحتوي هذه الطبقات بشكل رئيسي على مرشحات بأحجام مختلفة، لكل منها مجموعة موسطات يتم تحديدها في مرحلة التعلم، كل من هذه المرشحات يتم تطبيقه على الصورة أو على خرج طبقة جداء التلاف السابقة، حيث ينزلق هذا المرشح على الصورة ويتم حساب جداء التلاف بينه وبين المساحة التي تتم تغطيتها من هذه الصورة في كل مرة، ويمكن أن يكون الانزلاق بخطوة واحدة أو أكثر، ويمكن أن يختلف ما بين البعدين الطول والعرض [3].



الشكل 4. مرشّح كشف الحواف الأفقية

لأخذ فكرة عن آلية عمل هذه الطبقات يمكن ملاحظة أحد أبسط هذه المرشحات وأكثرها انتشاراً وهو مرشح تحديد الحواف detection edge (الشكل 4)، حيث يكون عبارة عن مصفوفة 3x3 سطرها الأول جميع عناصره هي الواحد، سطرها الثاني صفري، وسطرها الثالث مكون من العنصر 1-. عند تمرير هذا المرشح على الصورة، نكون في إحدى الحالتين، إما جميع عناصر الصورة التي يغطيها هذا المرشح من لون واحد وبالتالي لا يوجد حافة، وسيكون خرج العملية هو العنصر صفر، وإما القيم اللونية للعناصر مختلفة باختلاف السطر، وعندها ستنتج العملية قيمة مغايرة للصفر تشير إلى وجود حافة أفقية في الصورة.

2.2.2.2. الطبقة التجميعية Pooling Layer

مبدأ عملها مشابه لمبدأ طبقات جداء التلاف، فمن ناحية البنية هي عبارة عن مرشحات يتم تطبيقها على مصفوفة ثنائية البعد ومن ثم يتم زلق هذه المرشحات على كامل المصفوفة بخطوة معينة، ولكن الفرق الجوهري هنا أن موسطات parameters هذه المرشحات ثابتة ولا تتغير خلال مرحلة التعليم، ولهذا السبب لهذه الطبقات أنواع معروفة مثل:

- طبقات التجميع الأعظمي Pooling Max: حيث يكون خرج تطبيق الفلتر على منطقة معينة من المصفوفة ثنائية الأبعاد هو العدد الأكبر من هذه المصفوفة، وبالمثل يوجد أيضاً Pooling Min.
- طبقات التجميع بالمتوسط Pooling Average: يكون هنا خرج تطبيق الفلتر على منطقة معينة من المصفوفة ثنائية الأبعاد هو متوسط العناصر الذي يغطيها هذا الفلتر في المصفوفة.

يتم تطبيق هذه الطبقات بشكل رئيسي على خرج طبقات جداء التلاف، حيث تعمل على تلخيص خرج هذه الطبقات، وبالتالي التقليل من التعقيد الحسابي في الشبكة ككل من جهة، ومن جهة أخرى تساعد في حل مشكلة فرط الملائمة Overfitting [3].

3.2. عملية التعلم Learning في الشبكات العصبونية

تحدث عملية التعلم من خلال معالجة معينة للمعطيات المدخلة (من خلال توابع التنشيط)، ومن ثم تقييم أداء النموذج بعد هذه المعالجة (من خلال تابع الخسارة)، ومن ثم ضبط الأوزان بناء على هذا التقييم (من خلال خوارزمية أمثلة).

1.3.2. توابع التنشيط Activation Functions

وحدات صنع القرار للشبكات العصبية الأساسية هي توابع التنشيط إذ أنها تقوم بتقييم خرج الخلية في الشبكات العصبية وتعديل العلاقات بين البيانات للوصول لنتيجة أقرب للمرجوة. وبالتالي فإنها ضرورية لأداء الشبكة بأكملها. لذلك من المهم اختيار تابع التنشيط الذي سيعطي النتيجة الأفضل لخرج الخلية العصبية ويساهم في تحسين الخرج أثناء عملية التدريب. سنناقش أشهر توابع التنشيط غير الخطية التي أظهرت فعاليتها في مجال التعلم العميق.

1.1.3.2. تابع سيغمويد sigmoid function

هو نوع من دوال التنشيط المستخدمة في الشبكات العصبونية. تعطى صيغته بالعلاقة (حيث x هي الإشارة الواردة إلى التابع):

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

ميزة هذا التابع هي أنه يمكن استخدامه لتحويل أي قيمة إلى نطاق محدد بين 0 و1، مما يسمح بتفسير الخرج كاحتمال. يمكن استخدامه في الطبقات الخفية وكذلك يمكن استخدامه في طبقة الخرج لنماذج الشبكة العصبونية في مسائل التصنيف الثنائية. تكمن المشكلة الأساسية في استخدام هذا التابع في الطبقات الخفية هي أنّه يزيد من احتمالية خلق مشكلة تلاشي التدرجات Vanishing Gradient، وذلك لأن تدرجاته تتناقص مع تزايد القيمة أو تناقصها. أي أنه مثلاً عندما تصبح القيمة قريبة جداً من 1 أو قريبة جداً من 0 فإن مشتق هذا التابع عندها سوف يصبح تقريباً معدوم، وهذا ما سيبطّئ

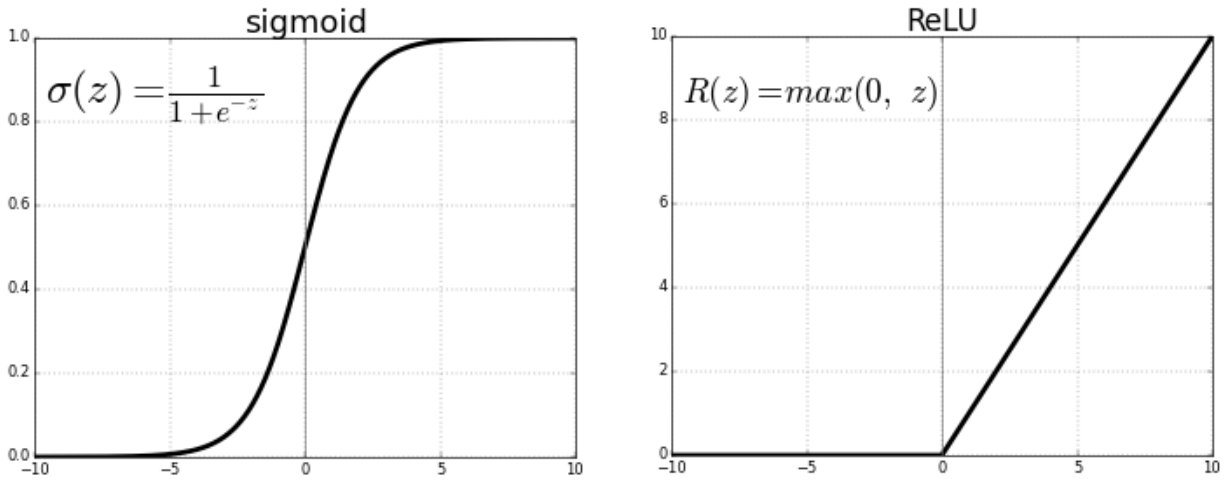
عملية التعلم بشكل كبير في حال استخدام خوارزمية أمثلة تعتمد على التدرج لضبط الأوزان بل وربما سيجعلها غير ممكنة في الشبكات المعقدة [2].

2.1.3.2. تابع الوحدة الخطية المصحَّحة Rectified Linear Unit (ReLU)

من خلال ما سبق، نلاحظ الحاجة لتابع تنشيط للوحدات في الطبقات الخفية يقلّل من احتمالية حدوث مشكلة تلاشي التدرجات، ولكن غير خطي لكي تستفيد الشبكة العصبونية من زيادة عدد الطبقات فيها، هذا ما يحققه تابع الوحدة الخطية المصحَّحة. يكون هذا التابع خطياً من أجل القيم الموجبة ومعدوماً من أجل القيم السالبة. لذلك يعتبر من توابع التنشيط الأكثر انتشاراً في الشبكات العصبونية وله الصيغة التالية:

$$ReLU(x) = max(0, x)$$

أثبت هذا التابع مدى كفاءته في العديد من نماذج الشبكات العصبونية التي تستخدمه بسبب قدرة هذه الشبكات على تعلم أنماط معقدة دون كلفة حسابية عالية ومع السيطرة على مشكلة تلاشي التدرجات [2]. العديد من الأوراق البحثية التي حقّقت أفضل النتائج أشارت إلى أهمية استعمال هذا التابع في الطبقات الخفية مثل [9] حيث تمت الإشارة فيها إلى أن تدريب الشبكات العصبونية التلافيفيّة باستعمال ReLU يتم أسرع بأضعاف من استخدام غيره من توابع التنشيط. يوضح الشكل 5 الفرق بين التابعين sigmoid وReLU.



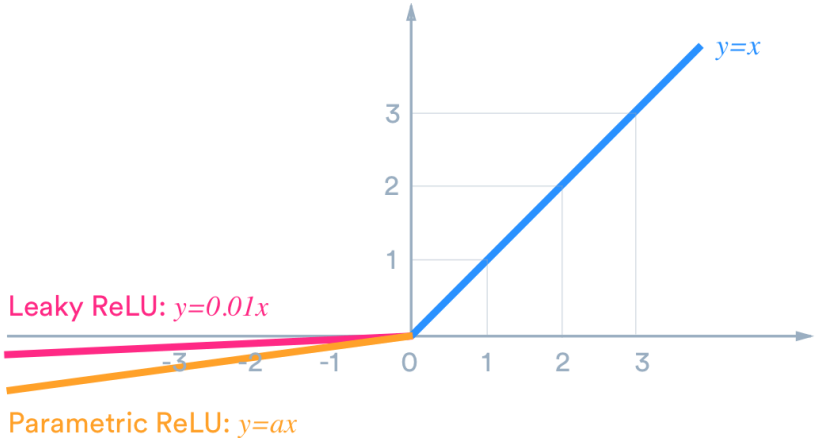
الشكل 5. الفرق بين تابعي التنشيط sigmoid وReLU

3.1.3.2. تابع الوحدة الخطية المصحَّحة المتسرِّبة Leaky ReLU

جاء هذا التابع كنسخة مطورة من التابع السابق ليحل مشكلة تدهور ReLU (Dying ReLU) التي ظهرت بسبب انعدام مشتق ReLU للقيم السالبة مما يؤدي لعدم تحديد قيم الأوزان أثناء عملية الانتشار الرجعي أو الخلفي وقد تنتج خلايا ميتة تماماً لا يتم تحديد قيم الأوزان الداخلة لها أبداً مما يجعلها غير نشطة أبداً. حلّ Leaky ReLU هذه المشكلة بإعطاء مجال سماحية للقيم السالبة كما هو موضح في الشكل 6، وله الصيغة التالية:

$$Leaky\ ReLU(x) = max(x, ax)$$

حيث a ثابت ما.



الشكل 6. تابع التنشيط Leaky ReLU

2.3.2. توابع الخسارة Loss Functions

تابع الخسارة أو تابع الكلفة هو تابع يستخدم لتقييم أداء الخوارزمية (مجموعة الأوزان التي قامت بحسابها) أثناء عملية التعلم. يتعلق اختياره بطبيعة المسألة وخرج الطبقة الأخيرة من الشبكة العصبونية. فيما يلي نستعرض بشكل عام أشهر أنواع المسائل وتوابع الخسارة المستخدمة فيها.

1.2.3.2. مسائل الانحدار Regression problems

هي المسائل التي يكون الهدف منها هو توقع قيمة حقيقية أو مستمرة.

1.1.2.3.2. تابع متوسط الخسارة المربّعة Mean Squared Error

يقيس هذا التابع متوسط مربع الفروقات (الخطأ) بين القيمة الحقيقية والقيمة التي تم تقديرها باستخدام خوارزمية التعلم [5]:

$$\mathcal{L}_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(\mathbf{x}_i))^2$$

يعتبر هذا التابع من أكثر التوابع المستخدمة في مسائل الانحدار. تدرج (مشتق) هذا التابع متغيّر مع قيمة الخسارة (بسبب وجود التربيع في صيغته)، حيث يكون هذا التدرج كبيراً من أجل العينات ذات الخسارة الكبيرة، بينما يتناقص عندما تتقارب قيمة الخسارة من الصفر. هذه الخصائص مفيدة من أجل تقارب سريع ودقة عالية للنموذج. ولكن هذا التابع حسّاس للقيم المتطرفة لأنه يربع الخطأ، حيث أنه عندما يصبح الخطأ أكبر، تزيد الخسارة بشكل أسرع. هذا ما يجعل الخسارة من أجل القيم المتطرفة كبيرة جداً ومنه يجعل النموذج يراعي انتباه أكبر لهذه القيم.

2.1.2.3.2. تابع متوسط الخسارة المطلقة Mean Absolute Error

يقيس هذا التابع متوسط القيمة المطلقة للفروقات (الخطأ) بين القيمة الحقيقية والقيمة التي تم تقديرها باستخدام خوارزمية التعلم [5]:

$$\mathcal{L}_{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - f(\mathbf{x}_i)|$$

يعتبر هذا التابع من التوابع الشهيرة في مسائل الانحدار. هذا التابع لا يتزايد بسرعة مع تزايد الخطأ، ولذلك هو أكثر صلابة من تابع الخسارة المربّعة في حال وجود قيم متطرفة في معطيات التدريب. ولكن هذا التابع أقل استخداميةً من تابع متوسط الخسارة المربّعة نظراً لعدم اشتقاقيّته عند الصفر. بالإضافة لذلك، تدرج (مشتق) هذا التابع ثابت ولا يتعلق بقيمة الخسارة، وهذا ما سيؤثر على كفاءة النموذج.

2.2.3.2. مسائل التصنيف Classification Problems

هي المسائل التي يكون الهدف منها هو توقع قيمة صحيحة (تعبّر عن صنف معيّن).

1.2.2.3.2. تابع الصفر أو الواحد Zero–One loss

هذا التابع يعطي قيمة خسارة تساوي الواحد في حال عدم تطابق القيمة المتوقعة مع القيمة الحقيقة وإلا يعطي صفر [5] بالشكل الآتي:

$$L_{\text{ZeroOne}}(f(\mathbf{x}), y) = \begin{cases} 1 & \text{if } f(\mathbf{x}) \cdot y < 0 \\ 0 & \text{otherwise} \end{cases}$$

نلاحظ أن هذا التابع لا يراعي درجة الخطأ، وبالإضافة لذلك هو ليس تابع محدب ولذلك غالباً لا يتم استخدامه أثناء عملية التدريب.

2.2.2.3.2. تابع الإنتروبيا المتقاطعة Cross Entropy

يقيس هذا التابع المسافة بين التوزع الاحتمالي للخرج الفعلي والتوزع الاحتمالي للخرج المتوقع، وكلما كانت قيمته أصغر يكون التوزعان أكثر تقارباً [5]. يتم استخدامه في الشبكات العصبونية لتفادي معدل التعلم البطيء في عصبونات طبقة الخرج. تعطى علاقته بالمعادلة:

$$H(q, f_{\Theta}) = -\sum_{i=1}^{N} q(\mathbf{x}_i) \log(f_{\Theta}(\mathbf{x}_i))$$

حيث N هو عدد عينات مجموعة المعطيات، وq تمثل التوزيع الاحتمالي الفعلي لمجموعة البيانات، وf_{Θ} هو تابع التنشيط المستخدم في طبقة الخرج .

يوجد أيضاً تابع خسارة يدعى تابع الإنتروبيا المتقاطعة التصنيفي categorical cross entropy له نفس شكل معادلة تابع خسارة الإنتروبيا المتقاطعة ولكن ترميز القيم الفعلية فيه يكون وفق one hot encoding method. ويوجد أيضاً تابع يدعى تابع الإنتروبيا المتقاطعة المتناثر sparse categorical cross entropy، له أيضاً نفس شكل معادلة تابع الإنتروبيا المتقاطعة ولكن ترميز القيم الفعلية فيه يكون على شكل أعداد صحيحة.

3.3.2. خوارزميات أمثلة عملية التعلم Learning Optimization Algorithms

بعد الحديث عن أشهر توابع الخسارة في الفقرة السابقة، سنتحدث الآن عن أشهر الخوارزميات المستخدمة في أمثلة هذا التابع (تقليل قيمته قدر الإمكان).

1.3.3.2. خوارزمية انخفاض التدرج وفق كل الدفعة Batch Gradient Descent

تعتبر خوارزميات انخفاض التدرج بشكل عام من أكثر خوارزميات الأمثلة الشائعة في مجال التعلم العميق. تندرج خوارزمية انخفاض التدرج وفق كل الدفعة تحت هذا النوع من الخوارزميات بحيث أنها تعتمد على استخدام كل معطيات التدريب من أجل عملية تحديث الأوزان. ترتكز فكرة هذه الخوارزمية في البداية على إسناد قيم عشوائية للأوزان، ومن ثم تحديث هذه

الأوزان مع كل خطوة بحيث تصل في النهاية إلى قيمة محلية صغرى أو الحد الأدنى لتابع الخسارة (بحسب تحدّب تابع الخسارة). للتوضيح بشكل أكبر، تعتمد هذه الطريقة على حساب تدرج gradient تابع الخسارة بالنسبة للأوزان من أجل كل معطيات التدريب في كل خطوة، وبناءً عليه تقوم بسلك اتجاه معين ضمن فضاء الأوزان (اختيار جديد للأوزان) بحيث تتقارب من الحل. تستخدم الخوارزمية أيضاً في هذه العملية معامل يسهم في تحديد شدة الخطوة التي ستسلكها الخوارزمية في كل مرّة ويدعى هذا المعامل معدل التعلم Learning rate. وفق هذه الخوارزمية، يتم تحديث الأوزان من أجل كل الدفعة على الشكل الآتي [10]:

$$W = W - \eta \cdot \nabla_W L(W)$$

بحيث: L هو تابع الخسارة من أجل كل معطيات التدريب، W هي مصفوفة الأوزان و η هو معدل التعلم.

تتقارب هذه الخوارزمية حتماً نحو القيمة الصغرى العظمى في حال كان تابع الخسارة محدب Convex، وإلّا فإنّها حتماً ستتقارب نحو إحدى القيم المحلية الصغرى للتابع. يعتبر تحديد معدل التعلم في هذه الخوارزمية هو أهم مسألة، لأنّه في حال كان هذا المعامل صغير فإن التقارب سيكون بطيء، وفي حال كان كبير فإنّه من الممكن ألّا تتقارب الخوارزمية من الحل (إما تتباعد أو تسلك طريق متأرجح حول الحل). وأخيراً، نظراً لأن هذه الخوارزمية تعتمد على كل معطيات التدريب من أجل القيام بعملية تحديث واحدة للأوزان، تعتبر بطيئة وغير مناسبة لمجموعات المعطيات التي لا تتسع في الذاكرة [10].

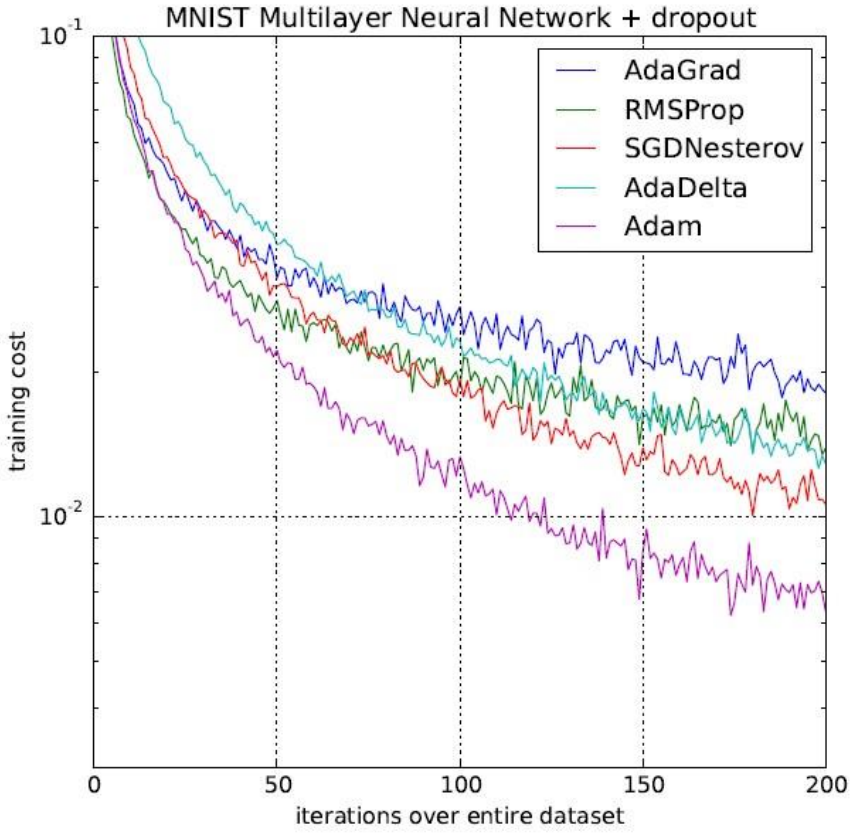
2.3.3.2. خوارزمية انخفاض التدرج العشوائي وفق دفعة صغيرة Mini Batch Stochastic Gradient Descent

تعتمد هذه الخوارزمية نفس مبدأ خوارزمية انخفاض التدرج ولكن تقوم بتحديث الأوزان بناءً على دفعة batch عشوائية من معطيات التدريب وليس على كل المعطيات مثل الخوارزمية السابقة، حيث يكون تابع الخسارة محسوباً من أجل هذه الدفعة فقط. عندما يكون حجم الدفعة مساوياً للواحد (عينة واحدة فقط)، يطلق على هذه الخوارزمية اسم انخفاض التدرج العشوائي Stochastic Gradient Descent. تعتبر هذه الخوارزمية أسرع من الخوارزمية السابقة لأنها تقوم بالتحديث بعد رؤية عينة واحدة من معطيات التدريب، ولكن نظراً لذلك فإنها قد تتسبب أيضاً بتباين بين تحديثات الأوزان [10]. خوارزمية انخفاض التدرج العشوائي وفق دفعة صغيرة تحلّ هذه المشكلة وتقود لتقارب أكثر استقراراً.

3.3.3.2. خوارزمية آدم Adaptive Moment Estimation (Adam)

Adam هي خوارزمية تندرج أيضاً تحت خوارزميات الأمثلة التي تعتمد على التدرج (المشتق) من الدرجة الأولى لتابع الخسارة ولا تحتاج إلى متطلبات ذاكرة كبيرة. في الخوارزميات السابقة يبقى معدل التعلم ثابت خلال عملية التدريب، أمّا في هذه

الخوارزمية يتم حفظ معدل تعلم لكل وزن ضمن الشبكة وتكييف كل معدل من هذه المعدلات بشكل مستقل خلال عملية التعلم. تظهر النتائج التجريبية أنها تعمل بشكل ممتاز عملياً على مسائل التعلم العميق والمعطيات الكبيرة وبشكل أفضل من طرق الأمثلة العشوائية الأخرى [11]. يظهر الشكل 7 مقارنة أجراها [11] بين استخدام خوارزمية آدم واستخدام مجموعة خوارزميات أمثلة أخرى لتدريب شبكة عصبونية متعددة الطبقات على مجموعة المعطيات MNIST.



الشكل 7. مقارنة بين خوارزمية آدم وخوارزميات أمثلة أخرى على مجموعة المعطيات MNIST

4.2. أهم البنى المستخدمة في مجال توليد الصور من التوصيف النصي

نورد في هذه الفقرة أهم بنى الشبكات العصبونية المستخدمة في مجال توليد الصور من التوصيف النصي.

1.4.2. شبكات الخصومة التوليدية Generative adversarial Networks (GAN)

نوع من الشبكات العصبونية المستخدمة في تعلم الآلة، تُستخدَم بشكل أساسي بهدف التدرب على إنشاء بيانات مزيفة مشابهة للبيانات الحقيقية، يصعب على مراقب بشري أو آلي التفريق بينهما [4]. تتعلّم هذه الشبكة إنشاء بيانات جديدة تتبع نفس التوزيع الخاص ببيانات التدريب. على سبيل المثال، يمكن لـ GAN المدرَّبة على الصور الفوتوغرافية إنشاء صور جديدة تبدو حقيقية للمراقبين البشريين، ولها العديد من الخصائص الواقعية.

فكرة هذه الشبكات مستوحاة من نظرية الألعاب Theory Game، تتكوّن بشكل أساسي من شبكتين عصبونيتين تتنافسان فيما بينهما وهما شبكة التوليد Generator، وشبكة التمييز Discriminator. تتولّى شبكة التوليد مهمّة التقاط التوزيع الحقيقي للبيانات، ومن ثمّ محاولة إنشاء بيانات جديدة تتبع ذات التوزيع، بينما تقوم شبكة التمييز بتقييم عمل شبكة التوليد وذلك من خلال تقدير احتماليّة كون العينة قد جاءت من البيانات الحقيقية (بيانات التدريب) بدلاً من البيانات المزيفة التي تنتجها شبكة التوليد، لذلك غالباً ما تكون شبكة التمييز عبارة عن شبكة تصنيف ثنائية binary classifier.

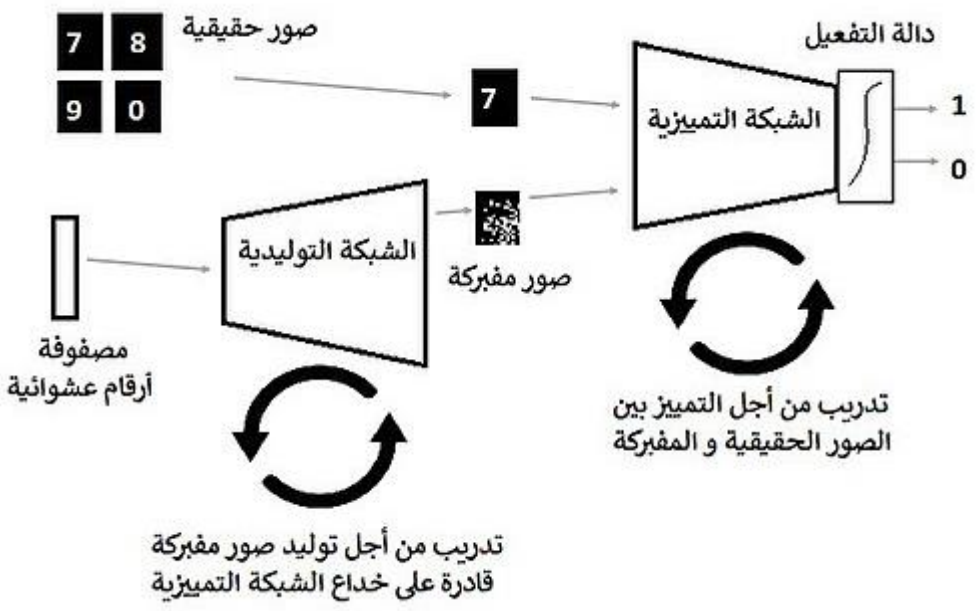
يتم تدريب هاتين الشبكتين على التوازي في آنٍ معاً، ويمكن أن تتبع كلّ منهما أي نوع شبكات معروف مثل Fully Connected أو CNN. يتم التدريب بطريقة minimax، وذلك إلى حين الوصول إلى ما يسمى بتوازن ناش Nash Equilibrium، وهي الحالة التي يكون فيها رد كل من الشبكتين على الأخرى أمثلياً.

انطلاقاً من مصفوفة من الأرقام العشوائية (إشارة ضجيج)، تتولى الشبكة التوليدية G إنشاء بيانات جديدة. بينما يكون دور الشبكة التمييزية D، هو التمييز بين البيانات الحقيقية و المزيفة التي تنتجها الشبكة التوليدية. يتم تدريب هذه الشبكات على الشكل الآتي:

- أولاً، يتم إنشاء حزمة من البيانات المزيفة من طرف الشبكة التوليدية.
- بعد ذلك، يتم إضافة هذه البيانات إلى عدد من البيانات الحقيقية وعرضها على الشبكة التمييزية للتدريب بهدف تقوية قدراتها على التفريق بين البيانات الحقيقية و المزيفة. يجب أن تميّز هذه الشبكة بين النوعين، وبالتالي يمكن التعامل مع هذه المسألة على أنها مسألة تصنيف ثنائية يتم فيها تدريب الشبكة (المميز) لإعطاء القيمة 1 في حال كانت البيانات حقيقية، و0 في حالة البيانات المزيفة القادمة من الشبكة التوليدية.
- في الخطوة الثالثة، يتم تدريب الشبكة التوليدية ذاتها على تحسين إنتاج بيانات مزيفة وذلك تبعاً لخرج الشبكة التمييزية، بحيث تصبح أكثر قدرة على خداعها.

- يتم بعد ذلك تكرار هذه الخطوات الثلاث عدداً كبيراً من المرّات بحيث تتحسن في كل مرّة قدرة الشبكة التمييزية على التفريق بين البيانات الحقيقية و المزيفة، وفي نفس الوقت تتحسن قدرة الشبكة التوليدية على إنتاج بيانات مشابهة بشكل أكبر للبيانات الحقيقية تستطيع بها خداع الشبكة التمييزية.
- يستمر التدريب إلى أن يصبح من الصعب على أي مراقب تفريق خرج الشبكة التوليدية عن البيانات الحقيقية.

أي أن شبكة التمييز تحاول تعظيم فرصها في تحديد الصف الصحيح الذي ينتمي إليه الدخل، بينما تقوم شبكة التوليد بمحاولة خداع شبكة التمييز، وذلك من خلال تقليل فرص فوز المميز، يطلق على هذه الطريقة في التعلم اسم Minmax Game. يجب الانتباه إلى أنه يجب الحفاظ على الاتساق بين شبكتي التوليد والتمييز أثناء التدريب، أي أنه يجب مزامنة D بشكل جيد مع G أثناء التدريب (على وجه الخصوص، يجب عدم تدريب G كثيراً دون تحديث D) [4]. يوضح الشكل 8 آلية تدريب هذه الشبكات.



الشكل 8. آلية تدريب شبكات الخصومة التوليدية

تابع الخسارة المستخدم في عملية التدريب:

يمثل تابع الخسارة المستخدم في تدريب هذا النوع من الشبكات بشكل مباشر الانتروبيا المتقاطعة بين توزيعات البيانات الحقيقية والبيانات المولدة من قبل شبكة التوليد. يطلق عليه اسم Minimax loss [5] ويعطى بالعلاقة التالية:

$$L_{minimax}(D,G) = E_{x\sim Pdata(x)}[\log D(x)] + E_{z\sim Pz(z)}[\log(1 - D(G(z)))]$$

حيث:

x: عينات مأخوذة من التوزع الحقيقي p_{data}.

z: عينات مأخوذة من توزع الضجيج المطبق على دخل شبكة التوليد p_z، والذي قد يكون توزع منتظم أو غوصي أو أي توزع آخر.

G(z): هو ناتج شبكة التوليد عند إدخال الشعاع العشوائي z.

D(x): هو ناتج شبكة التمييز عند إدخال الشعاع x.

$E_{x\sim Pdata(x)}$: هي القيمة المتوقعة على جميع عينات البيانات الحقيقية.

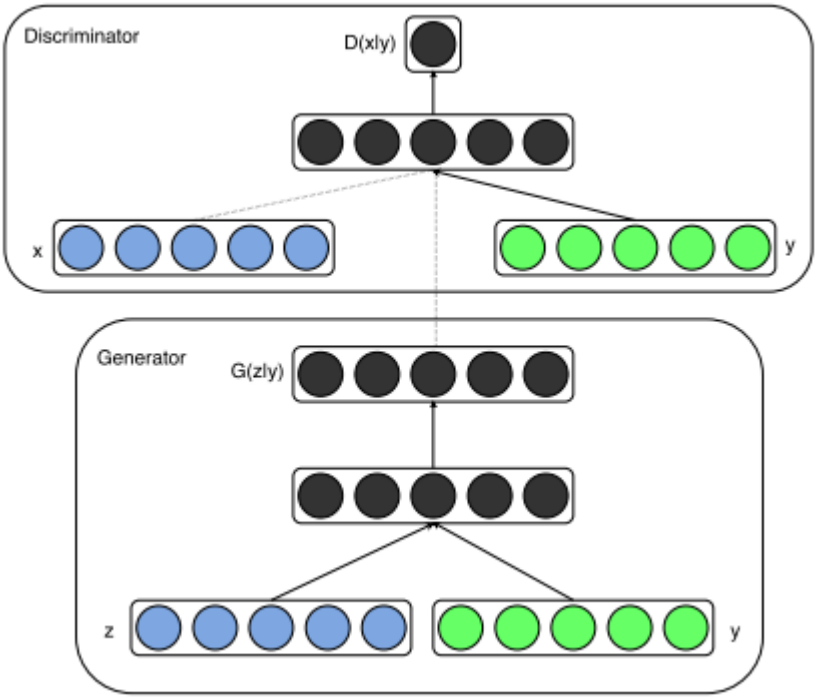
$E_{z\sim Pz(z)}$: هي القيمة المتوقعة على جميع عينات البيانات المفبركة التي أنتجها المولّد.

يحاول المولد تقليل التابع السابق، بينما يحاول المميّز تعظيمه. وبذلك تصبح عملية الأمثلة للشبكة الكلية تتبع الصيغة التالية المستمدة من خوارزمية Minimax من نظرية الألعاب [5]:

$$\min_G \max_D \{ E_{x\sim Pdata(x)} [log\, D(x)] + E_{z\sim Pz(z)} [log(1 - D(G(z)))] \}$$

2.4.2. شبكات الخصومة التوليدية الشرطية Conditional GANs

يمكن تمديد شبكات الخصومة التوليدية إلى نموذج شرطي إذا كان كل من المولد والمميز مشروطاً ببعض المعلومات الإضافية ولتكن y. يمكن أن تكون هذه المعلومات الإضافية y أي نوع من المعلومات المساعدة، مثل اسم الصنف class label أو أي شرط على الغرض المنشأ [6]. يمكن تنفيذ هذه الشرطية عن طريق تغذية y في كل من المولد والمميز كطبقة إدخل إضافية، كما في الشكل 9.



الشكل 9. شبكة خصومة توليدية شرطية

يتم تدريب هذه الشبكات بشكل مشابه للشبكات السابقة، ولكن يستبدل تابع الخسارة المستخدم في شبكات الخصومة التوليدية بعد إضافة الشرطية بالتابع التالي [6]:

$$L_{minimax}(D,G) = E_{x\sim Pdata(x)}[\log D(x|y)] + E_{z\sim Pz(z)}[\log(1 - D(G(z|y)))]$$

3.4.2. نماذج انتشار تقليل الضجيج الاحتمالية Denoising Diffusion Probabilistic Models

عبارة عن نماذج توليدية احتمالية تعتمد التعلم العميق لتوليد عينات البيانات. خلال التدريب يتعلم النموذج الانحلال المنهجي للبيانات بسبب إضافة ضجيج غوصي طفيف بشكل تكراري حتى تفقد هذه البيانات خصائصها المميزة (تعرف هذه العملية باسم عملية الانتشار الأمامي Forward diffusion process)، ثم عكس هذه العملية (فيما يعرف باسم تقليل الضجيج أو عملية الانتشار العكسي Reverse diffusion process) لاستعادة البيانات المتحلّلة من الضجيج مرة أخرى. يتعلم النموذج تدريجياً إزالة الضجيج بالاعتماد على سلاسل ماركوف [7]، بحيث يصبح قادراً على توليد صور جديدة عالية الجودة من صور مضجّجة عشوائية، كما هو موضح في الشكل 10.

الشكل 10. نموذج انتشار احتمالي لتقليل الضجيج

4.4.2. المحوّلات Transformers

هيكلية جديدة أحدثت ثورة في عالم معالجة اللغات الطبيعية. رغم النجاح الكبير لشبكات مثل RNN وGRU، إلّا أنّها لم تكن قادرة على القيام بعملية معالجة على التوازي، إضافة لعدم قدرتها على التعامل مع التبعيّات طويلة المدى Long term dependency sequence. جاءت المحوّلات كحل لهاتين المشكلتين نتيجة اعتمادها على ما يسمى الانتباه الذاتي Self Attention دون السلاسل، وهو ما يسمح للنموذج بأن يركز على أقسام معينة من سلسلة الدخل بينما في نفس الوقت يتنبأ بالخرج. تتكون بنية المحوّلات بشكل أساسي من مكدّسات Encoder-Decoder:

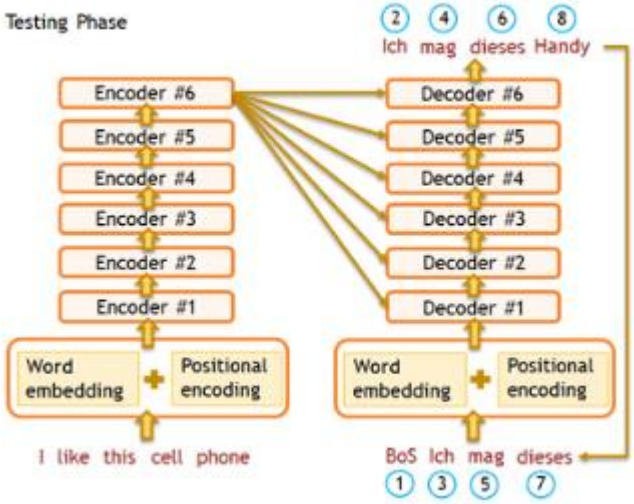
يقوم المرمّز Encoder بعمل ربط mapping بين رموز الدخل x ورموز جديدة z، كل مرمّز يحوي طبقتين فرعيتين:

- Multi-head self attention mechanism على أشعة الدخل أي آليات انتباه ذاتي تعمل سويّة عند الدخل.
- Fully connected feed-forward network للمعالجة اللاحقة.

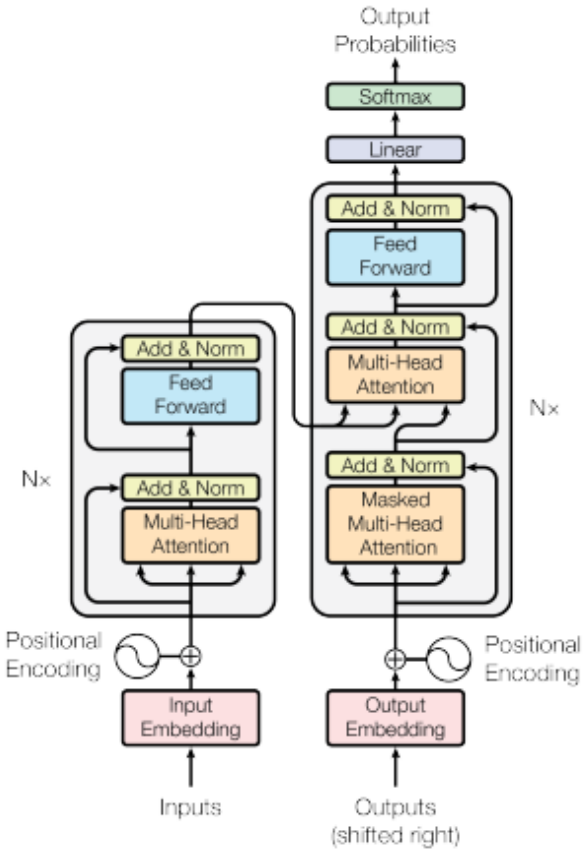
وتتكرر العملية السابقة عدة مرّات من رمّاز إلى آخر حتى الوصول إلى الرمّاز الأخير في المكدس ليُمرَّر بعدها إلى فاك الرمّاز Decoder. يعالج فاك الرمّاز الخرج z ليولد سلسلة جديدة من الرموز y، رمز واحد في كل مرّة. لكل فاك رمّاز ثلاث طبقات فرعية:

- Masked multi-head self attention mechanism على أشعة الخرج z للدورة السابقة.
- Multi-head self attention mechanism على خرج المرمّز z وأيضاً خرج الطبقة السابقة Masked multi-head self attention mechanism.
- Fully connected feed-forward network للمعالجة اللاحقة.

والخرج في كل الحالات يجب أن يكون شعاع بنفس البعد [8].



الشكل 11. مكدسات الرمّاز وفاك الرماز في المحوّل



الشكل 12. بنية المحول

الفصل 3. الدراسة المرجعية

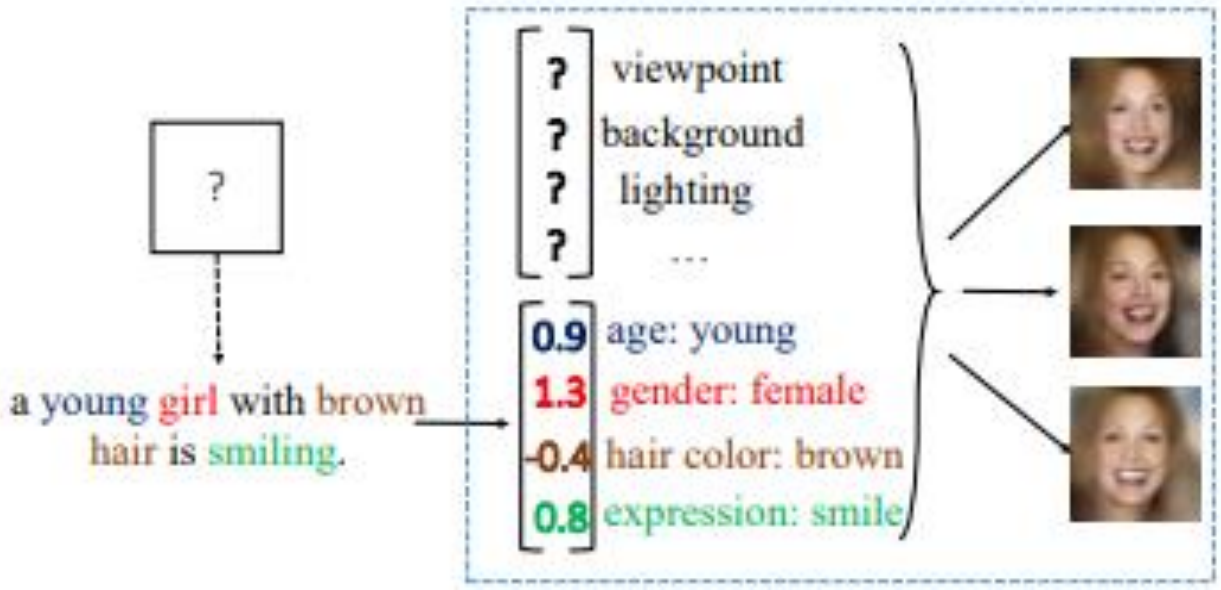
سنبدأ هذا الفصل باستعراض تقنيات التعلم العميق المستخدمة في توليد الصور من التوصيفات النصية بشكل عام، ومن ثم سنستعرض الأعمال التي تم طرحها في هذا المجال. في النهاية سنتحدث عن تقنية نقل التعلم التي تم استخدامها في كثير من هذه الأعمال.

1.3. استخدام التعلم العميق في توليد الصور من التوصيف النصي

قبل استخدام التعلم العميق في هذا المجال، غالباً ما كانت الطرائق التقليدية لتوليد الصورة من النص تعتمد على الأساليب القائمة على القواعد والخصائص المستخرجة بشكل يدوي، وخوارزميات التعلم الآلي التقليدية لاكتشاف وحدات النص والبحث عن أجزاء الصورة التي تصل هذه الوحدات بهدف تحسين الشكل العام للصورة. فيما يلي بعض التقنيات التي تم استخدامها للقيام بذلك:

- الأساليب القائمة على القوالب: تستخدم هذه الأساليب قوالب أو مكونات رسومية محددة مسبقاً لإنشاء صور بناءً على أوصاف النص. يمكن أن تتكون القوالب من أشكال أو كائنات أو مشاهد، وسيوجه النص اختيار هذه المكونات وموضعها لإنشاء صورة [21].
- معالجة السمات: يتضمن هذا الأسلوب تمثيل الصور وأوصاف النص باستخدام سمات محددة مسبقاً أو ناقلات الميزات. تم بعد ذلك استخدام خوارزميات التعلم الآلي لتعلم التعيين بين سمات النص والصورة (الشكل 13)، مما يسمح بمعالجة الصور وإنشاءها بناءً على التعديلات على مستوى السمات [22].

غالباً ما واجهت الطرق التقليدية في تحويل النص إلى صورة قيوداً من حيث إنتاج صور واقعية ومتنوعة. فقد اعتمدوا على ميزات مستخرجة يدوياً وأنظمة واضحة قائمة على القواعد، بينما افتقروا إلى القدرة على التقاط التفاصيل الدقيقة. ومع ذلك، فقد وضعوا الأساس للتطورات اللاحقة في المناهج القائمة على التعلم العميق، والتي حققت تقدماً كبيراً في إنشاء صور أكثر واقعية وجاذبية بصرية من الأوصاف النصية.



الشكل 13 .الطرائق التقليدية لاكتشاف وحدات النص والبحث عن أجزاء الصورة التي تصل هذه الوحدات

أحدث التعلم العميق ثورة في مجال تحويل النص إلى صورة من خلال الاستفادة من بنيات الشبكات العصبية وبيانات التدريب واسعة النطاق [23]. تتفوق نماذج التعلم العميق في تعلم الأنماط والتمثيلات المعقدة من البيانات وذلك بسبب قدرتها على تحليل واستخلاص أنماط وخصائص البيانات بشكل تلقائي، مما يمكنها من محاكاة الأنماط الفنية أو إعادة إنشاء مشاهد من العالم الحقيقي. تمتد إمكانات توليد الصور المدعومة بالذكاء الصنعي إلى ما هو أبعد من مجرد النسخ، فقد أصبحت هذه النماذج قادرة على إنشاء صور خيالية تماماً.

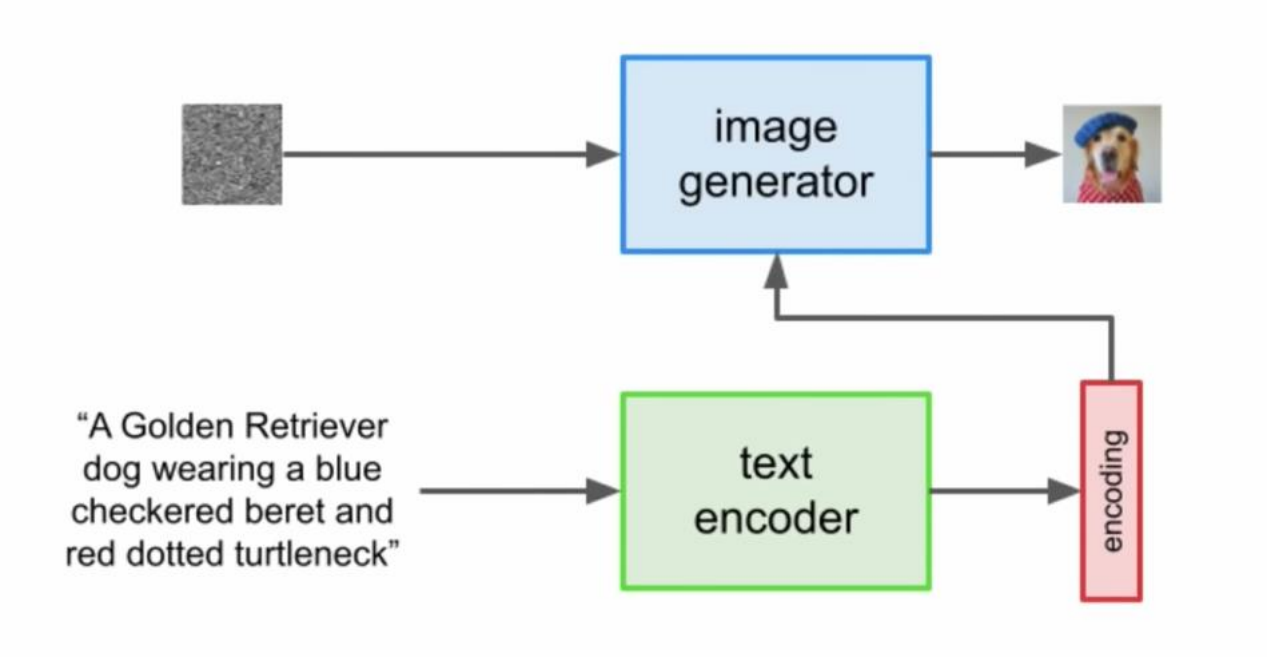
بشكل عام، تعتمد نماذج تحويل التسلسل إلى تسلسل sequence transduction models السائدة على شبكات عصبية متكررة أو تلافيفية معقدة في هيئة جهاز فك تشفير encoder-decoder [8]. فيما يلي نظرة عامة عالية المستوى على بنية شبكة تحويل النص إلى صورة (الشكل 14)، حيث تتكون هذه الشبكة من العناصر التالية وغالباً ما تكون بالترتيب المذكور:

- مرمّز أو مشفّر النص Text Encoder:

يقوم برنامج ترميز النص بمعالجة وصف نص الإدخال وتحويله إلى تمثيل ذي معنى أو تضمين. يلتقط هذا الترميز المعلومات الدلالية للنص ويعمل كدليل أو موجه لإنشاء الصور.

- شبكة التوليد أو مولّد الصور Image Generator:

يأخذ مولد الصور النص المضمَّن كدخل ويقوم بإنشاء تمثيل أولي للصورة. غالباً ما يكون هذا التمثيل منخفض الدقة أو خشناً، تستعمل عندئذٍ نماذج أخرى إضافية تسمى وحدات التحسين Refinement Modules وذلك لتحسين تمثيل الصورة الأولي عن طريق إضافة المزيد من التفاصيل وتعزيز التماسك البصري.



الشكل 14 . البنية العامة لنموذج توليد الصور من التوصيف النصي

يمكن تنفيذ خطوة ترميز النص باستخدام شبكة عصبية عودية RNN مثل شبكة الذاكرة طويلة المدى (LSTM)، ولكن بشكل عام، يمكن القول أن نماذج المحولات transformers أصبحت الخيار الأكثر شيوعاً. أمّا بالنسبة لخطوة توليد الصور، تم استخدام شبكات الخصومة التوليدية الشرطية بشكل شائع كما في النماذج المقترحة في [12] و[13]، ولكن على الرغم من تقديم نتائج مثيرة للإعجاب، فإن شرطية GAN على شعاع تضمين الجملة الكلي يفتقر إلى معلومات دقيقة مهمة على مستوى الكلمة، ويمنع إنشاء صور عالية الجودة. الأمر الذي أسهم في ظهور شبكات الخصومة التوليدية المقودة بالانتباه ذات القدرة على توليد صور تتضمن أدق التفاصيل الواردة في التوصيف النصي، كما في النموذج المقترح في [14]. كذلك أصبحت نماذج الانتشار أيضاً خياراً مطروقاً بشكل أكبر في السنوات الأخيرة وذلك نظراً لكونها ذات أداء وفعالية تدريب أفضل وصور واقعية أكثر، تم استخدام نماذج الانتشار في الكثير من الأوراق البحثية التي تم طرحها في السنوات الأخيرة منها [15] و[16] و[17].

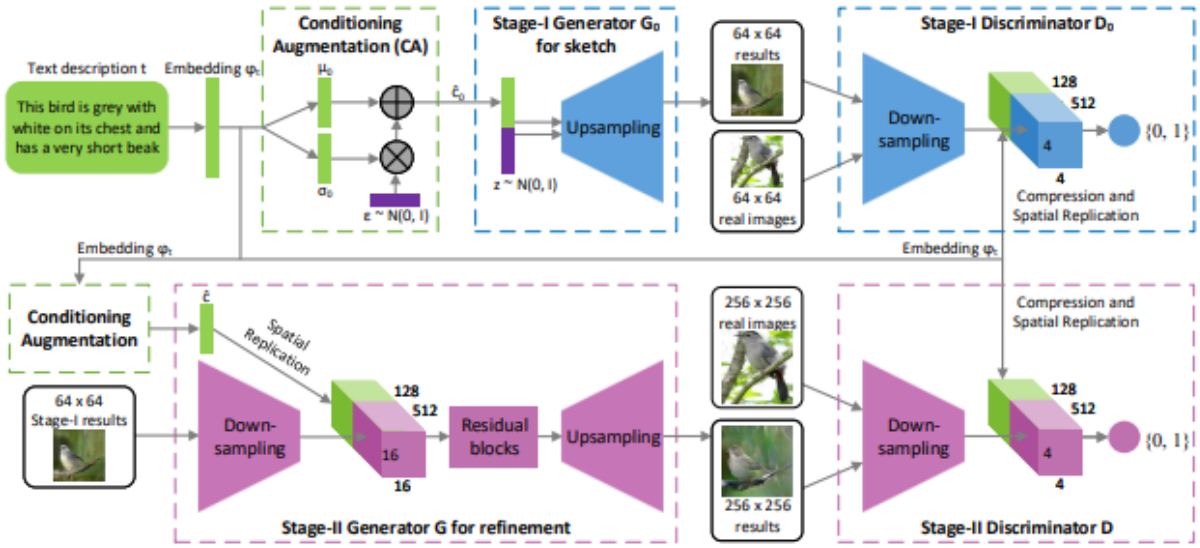
2.3. الأعمال المنجزة في مجال توليد الصور من التوصيفات النصية

- تطرح [13] نموذجاً استعمل فيه مرمز للنصوص على مستوى الحرف قائم على الشبكات العصبونية التلافيفية CNN والمتكررة LSTM العميقة للحصول على شعاع ترميز النص، يليه شبكة خصومة توليدية لتوليد الصور، تم تدريبه على ثلاث مجموعات بيانات مختلفة CUB[1], Oxford[2], COCO، حقق بحسب معيار Inception score[3] النتائج التالية على مجموعات البيانات السابقة على الترتيب 7.88 ,2.66 ,2.88، وبحسب معيار التقييم البشري Human rank الأرقام التالية 1.89 ,1.87 ,2.81.
- استعملت [12] كدسة مكونة من شبكتي خصومة توليديتين كل منهما مشروطة بترميز النص القادم من المرمز (الشكل 15)، الشبكة الأولى مسؤولة عن رسم الشكل البدائي والألوان الأساسية للكائن المشروط بوصف النص المحدد ورسم تخطيط الخلفية من شعاع ضجيج عشوائي ينتج عنه صورة منخفضة الدقة، تقوم الشبكة الثانية بتصحيح العيوب في الصورة ذات الدقة المنخفضة من المرحلة الأولى واستكمال تفاصيل الكائن من خلال الشرطية على الوصف النصي مرة أخرى، مما يؤدي إلى إنتاج صورة واقعية عالية الدقة، تم تدريبه على ثلاث مجموعات بيانات مختلفة CUB, Oxford, COCO، حقق هذا النموذج بحسب معيار Inception score النتائج التالية على مجموعات البيانات السابقة على الترتيب 8.45 ,3.2 ,3.7، وبحسب معيار التقييم البشري Human rank الأرقام التالية 1.11 ,1.13 ,1.37.

[1] http://www.vision.caltech.edu/datasets/cub_200_2011/

[2] https://www.robots.ox.ac.uk/~vgg/data/flowers/102/index.html

[3] (IS) خوارزمية تستخدم لتقييم جودة الصور التي تم إنشاؤها بواسطة نموذج صورة توليدي مثل شبكات الخصومة التوليدية.

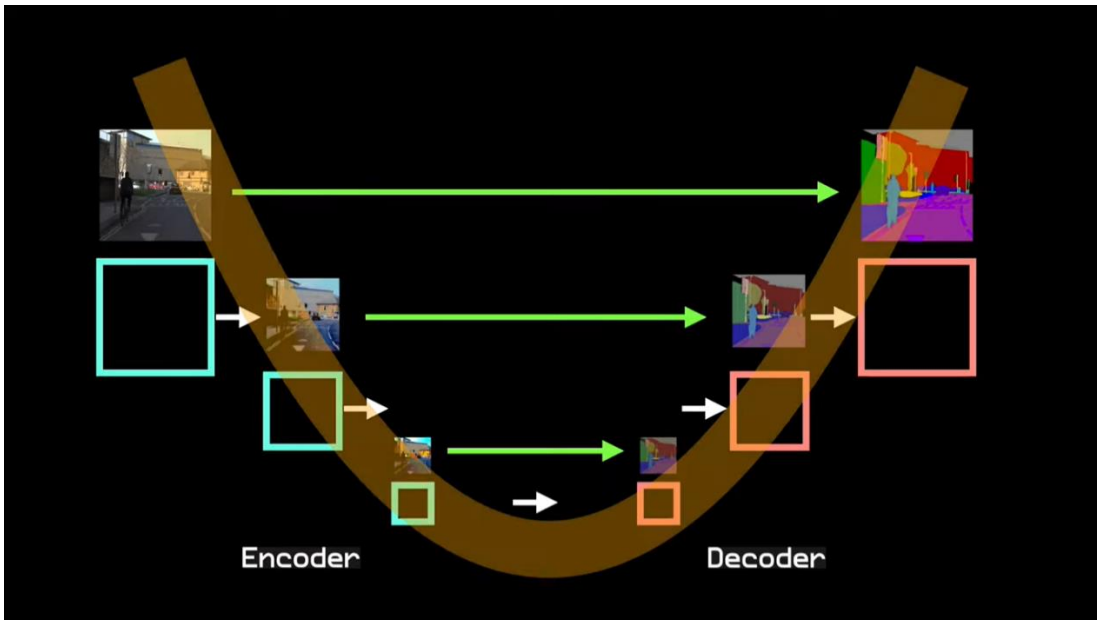


الشكل 15. نموذج يستخدم كدسة منش شبكات الخصومة التوليدية StackGAN

- استخدم النموذج المقترح في [14] الانتباه من خلال مكونين، المكون الأول هو شبكة توليدية انتباهية، حيث يتم تطوير آلية انتباه للمولد لرسم مناطق فرعية مختلفة من الصورة من خلال التركيز على الكلمات الأكثر صلة بالمنطقة الفرعية التي يتم رسمها. وبشكل أكثر تحديداً، إلى جانب ترميز الجملة الكلية في شعاع، يتم أيضاً ترميز كل كلمة في الجملة إلى شعاع آخر. تستخدم الشبكة التوليدية شعاع الجملة الكلية لإنشاء صورة منخفضة الدقة في المرحلة الأولى. في المراحل التالية، يتم ربط شعاع الصورة في كل منطقة فرعية للاستعلام مع أشعة ترميز الكلمات باستخدام طبقة الانتباه لتشكيل أشعة ترميز سياق الكلمة. المكون الثاني هو نموذج التشابه العميق متعدد الوسائط (DAMSM). باستخدام آلية الانتباه، يستطيع DAMSM حساب التشابه بين الصورة التي تم إنشاؤها والجملة باستخدام معلومات مستوى الجملة الشاملة ومعلومات مستوى الكلمة الدقيقة. وبالتالي، يوفر DAMSM خسارة إضافية لمطابقة نص الصورة الدقيقة لتدريب المولد.
- استخدم النموذج المقترح في [16] نماذج الانتشار لتوليد الصور، يتم توجيه عملية تدريب نموذج الانتشار المستخدم باستخدام نموذج CLIP (Contrastive Language-Image Pretraining)، عبارة عن شبكة عصبونية جرى تدريبها على أزواج (صورة، نص) توفر درجة لمدى قرب الصورة من التوصيف النصي. يضيف CLIP خسارة إضافية إلى تابع الخسارة الأساسي المستخدم في التدريب تسهم في توليد صور أكثر واقعية. حقق هذا النموذج حسب المعيار FID[4] نتيجة 12.24 على مجموعة البيانات MS-COCO.

[4] مقياس يستخدم لتقييم جودة الصور التي تم إنشاؤها بواسطة نموذج توليدي، مثل شبكة الخصومة التوليدية (GAN). على عكس (IS)، التي تقيم فقط توزيع الصور المولدة، يقارن FID توزيع الصور المولدة مع توزيع مجموعة من الصور الحقيقية.

- تقترح [15] نموذج يتكون من مرمز للنص يقوم بتحويل النص إلى سلسلة من التضمينات، تليه سلسلة من نماذج الانتشار المشروط التي تقوم بتحويل هذه التضمينات إلى صور ذات دقة متزايدة.
  - مرمز النص المستخدم عبارة عن أحد نماذج اللغات الكبيرة مسبقة التدريب T5 وهو فعال بشكل مدهش في ترميز النص المستخدم لتركيب الصور، زيادة حجم نموذج اللغة يعزز دقة العينة ومحاذاة نص الصورة أكثر بكثير من زيادة حجم نموذج الانتشار المستخدم لتوليد الصورة.
  - تعتمد نماذج الانتشار المستخدمة بنية U-Net (الشكل 16)، وهي شبكة تعتمد في أساسها على شبكات encoder-decoder، مكونة من طبقات جداء تلاف متتالية، تسمح للأشعة المتناظرة من جزء التشفير بالعبور وإضافتها إلى جزء فك التشفير. الشبكة مشروطة بشعاع ترميز النص عبر ناقل تضمين الناتج من المرمز، تتم إضافته إلى تضمين خطوة الانتشار الزمنية على غرار طريقة الشرطية على النص المستخدمة في [18] و[19]. يتم الشرط على أشعة تضمين النص عن طريق إضافة الانتباه المتقاطع [20] على تضمينات النص بدقة متعددة.



الشكل 16. بنية شبكة U-Net

فيما يلي شرح للآلية التي يعمل بها هذا النموذج:

1- أولاً، يتم إدخال التوصيف النصي في برنامج ترميز النص. يقوم المرمز بتحويل التوصيف النصي إلى تمثيل رقمي (شعاع تضمين) يلخص المعلومات الدلالية داخل النص.

2- بعد ذلك، يقوم نموذج توليد الصورة بإنشاء صورة انطلاقاً من الضجيج، وتحويلها ببطء إلى صورة تصل التوصيف النصي المطلوب. لتوجيه هذه العملية، يتلقى نموذج توليد الصورة ترميز النص كدخل، والذي له تأثير في إخبار النموذج بما هو موجود في التوصيف النصي حتى يتمكن من إنشاء صورة تقابله. الخرج عبارة عن صورة صغيرة (64x64 بكسل) تعكس بشكل مرئي التوصيف النصي الذي تم إدخاله إلى المرمز.

3- يتم بعد ذلك تمرير الصورة الصغيرة إلى نموذج زيادة الدقة، يقوم بزيادة دقة الصورة إلى (256x256 بكسل). يأخذ هذا النموذج أيضاً ترميز النص كدخل، مما يساعد النموذج على تحديد كيفية التصرف لأنه يملأ الفجوات الناتجة من المعلومات المفقودة التي تنشأ بالضرورة من مضاعفة حجم الصورة أربع مرات. لتكون نتيجة هذه المرحلة هي صورة متوسطة الحجم.

4- أخيراً، يتم تمرير هذه الصورة متوسطة الحجم إلى نموذج آخر لزيادة الدقة، والذي يعمل بشكل شبه مطابق للنموذج السابق، إلا أنه هذه المرة يأخذ الصورة متوسطة الحجم ويحولها إلى صورة عالية الدقة. والنتيجة هي صورة بحجم (1024x1024 بكسل) تعكس بصرياً الدلالات الموجودة في التوصيف النصي.

تم شرح النموذج الأخير بشكل مفصل، حيث سنقوم في الجزء العملي بالاعتماد بشكل كبير على المفاهيم الواردة في [15] لكونها تعتبر من أفضل الأوراق البحثية من حيث النتائج والتي كانت 7.27 حسب FID وذلك على مجموعة التحقق الخاصة ب COCO، وذلك رغم كون النموذج المقترح فيها غير مدرب على COCO في الأساس.

الفصل 4. البيئات والمكاتب المستخدمة

سوف نشرح في هذا القسم عن الأدوات البرمجية المستخدمة، مع إيضاح المساهمة التي قدمتها كل أداة في تنفيذ القسم العملي.

- **لغة البرمجة python[5]**: لغة Python هي لغة تفسيرية عالية المستوى، تستخدم أسلوب البرمجة غرضية التوجه. تتميز لغة بسهولة تعلمها وبساطة نحوها syntax، ولذلك يتم استخدامها في تطبيقات كثيرة. اعتمدنا هذه اللغة لأنها توفر مكتبات عديدة مفتوحة المصدر تساعد على بناء نماذج التعلم العميق ومعالجة المعطيات الكبيرة.
- **بيئة التطوير Google Colaboratory[6]**: استخدمت بشكل أساسي في عملية التدريب نظراً لإتاحتها طاقات حاسوبية قوية وموارد كبيرة مقارنة بالحواسب الأخرى المتوافرة.
- **محرر النصوص Visual Studio Code[7]**: تم استخدام هذه البيئة في كتابة النصوص البرمجية script وتنفيذها عند التدريب على مخدم وحدة المعالجة البيانية الخاص بالمعهد العالي، كذلك أثناء بناء موقع الوب.
- **منصة Weights & Biases[8]**: تساعد مطوري الذكاء الصنعي وتعلم الآلة على بناء نماذج أفضل بشكل أسرع. تسمح بتتبع التجارب بسرعة، وإصدار مجموعات البيانات وتكرارها، وتقييم أداء النموذج، وإعادة إنتاج النماذج، وإدارة سير عمل تعلم الآلة من البداية إلى النهاية.
- **wandb[9]**: واجهة سطر الأوامر (CLI) ومكتبة للتفاعل مع Weights & Biases API.
- **مكتبة PyTorch[10]**: تتيح تجربة سريعة ومرنة وإنتاجاً فعالاً من خلال دعم التدريب موزع والتعامل مع الكثير من الأدوات والمكتبات، إضافة إلى كونها تدعم استخدام وحدة المعالجة البيانية GPU في التدريب.
- **منصة تطوير الوب Django[11]**: إطار عمل framework عالي المستوى يستخدم في تطبيقات الوب، ويشجع فكرة التطوير السريع والتصميم النظيف حيث يعتمد النمط التصميمي MVT. يوفر هذا الإطار

[5] https://www.python.org/
[6] https://colab.research.google.com
[7] https://code.visualstudio.com/
[8] https://wandb.ai/site
[9] https://pypi.org/project/wandb/
[10] https://pytorch.org/

الكثير من متاعب تطوير الوب بحيث يتم التركيز على كتابة التطبيق فقط. استخدمنا Django من أجل بناء صفحة وب لتوليد الصور من التوصيفات النصية.

- **مكتبة imagen-pytorch[12]**: توفر هذه المكتبة تنجيزاً لنموذج توليد الصور الخاص ب Google والذي يسمى imagen باستعمال مكتبة PyTorch.
- **مخدم وحدة المعالجة البيانية GPU الموجود على سحابة المعهد العالي**: تم استعماله بشكل عام بسبب الأزمنة الطويلة التي استغرقها التدريب والزمن المحدود لجلسات colab.

[11] https://www.djangoproject.com

[12] https://github.com/lucidrains/imagen-pytorch

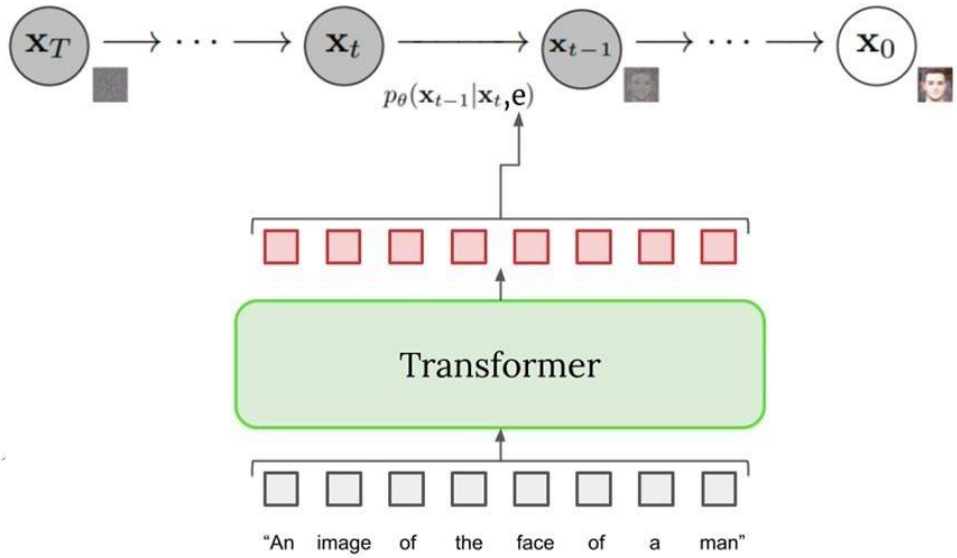
الفصل 5. القسم العملي

يتم فيما يلي شرح الخطوات الأساسية في تنفيذ المشروع، انطلاقاً من بناء النماذج وتدريبها ووصولاً إلى بناء واجهة التخاطب مع المستخدم.

1.5. بناء وتدريب النماذج

سوف نشرح في هذا القسم النماذج التي قمنا بتدريبها لتوليد الصور انطلاقاً من توصيفات نصية مكتوبة، وسنستعرض الدقة التي حصلنا عليها بعد تدريب كل منها مع إجراء عملية تفسير للنتائج التي حصلنا عليها والمحاولات التي قمنا بها لتحسين النتائج.

بشكل عام يتكون نموذج توليد الصور الذي قمنا باستعماله من محوّل Transformer أو بما معناه نموذج لترميز النص يقوم بتحويل النص إلى شعاع تضمين Embedding (سلسلة من الأرقام كترميز للنص)، يليه نموذج انتشار Text-to-Image Diffusion Model يقوم بالربط بين تضمين النص السابق والصور ليقوم بناءً على ذلك بتوليد الصورة المناسبة. للقيام بذلك استعملنا مكتبة imagen-pytorch[13] التي توفر تنجيز لنموذج Google لتوليد الصور من التوصيف النصي Imagen باستخدام Pytorch.



الشكل 17. آلية عمل نموذج التعلم العميق المستخدم

[13] https://github.com/lucidrains/imagen-pytorch

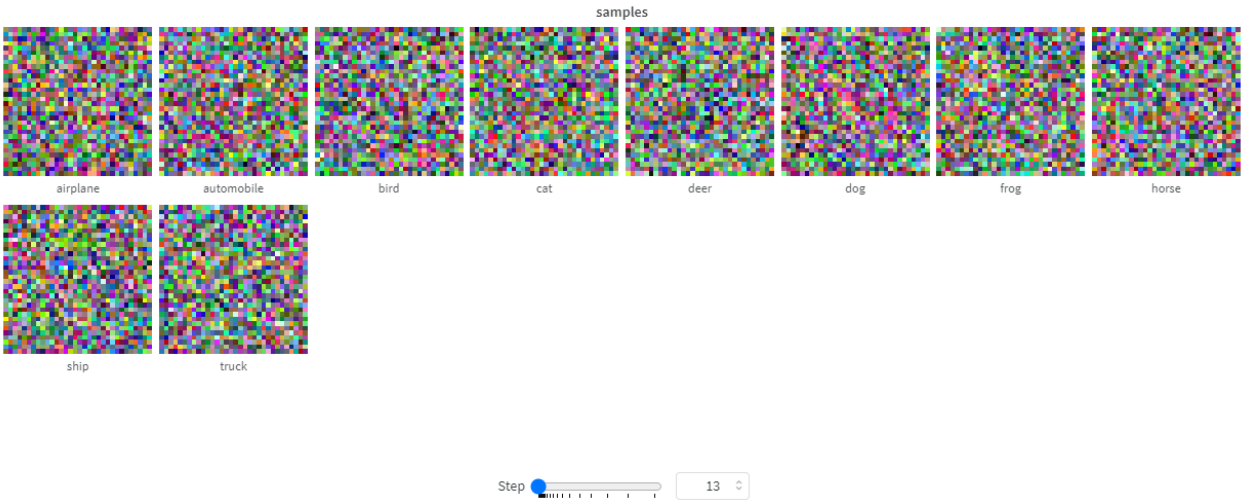
توفر هذه المكتبة محوّل transformer مدرب مسبقاً T5-Large (T5: Text-to-Text Transfer Transformer)، وصف اسمه Imagen يتألف من نماذج احتمالية لانتشار تقليل الضوضاء المتتالية (DDPM: Denoising Diffusion Probabilistic Models) مشروطة بالترميز النصي من النموذج T5 المذكور آنفاً (شبكة انتباه attention network). يوضح الشكل 17 الآلية التي تعمل بها هذه الشبكة.

قمنا بتجميع مكونات النموذج (المرمز T5 ومولد الصور Imagen) وربطهما معاً باستخدام نص برمجي script قمنا بكتابته. ومن ثم تمت كتابة نص آخر يقوم بالتدريب. قمنا بتدريب النماذج التالية من الصفر، حيث لم نستخدم نموذج توليد صور من التوصيفات النصية مدرب مسبقاً وحاولنا التعديل عليه وإنما تمت عملية التدريب من الصفر وبنصوص برمجية قمنا بكتابتها. للقيام بالتدريب كنّا بحاجة وحدة معالجة بيانية GPU (Graphics Processing Unit) لأن التدريب على وحدة المعالجة المركزية CPU كان بطيء جداً، لذلك تم التدريب على google colab[14]. ضمن الموارد المتاحة على ال colab لم نستطع التدريب على صور أكبر من (32x32 بكسل) وحجم دفعة batch size (عدد الصور التي يتم تمريرها للنموذج في كل خطوة أثناء التدريب) أكثر من 64، حيث يؤدي ذلك إلى تجاوز سعة الرام RAM المتاحة والتي تبلغ 15 GB.

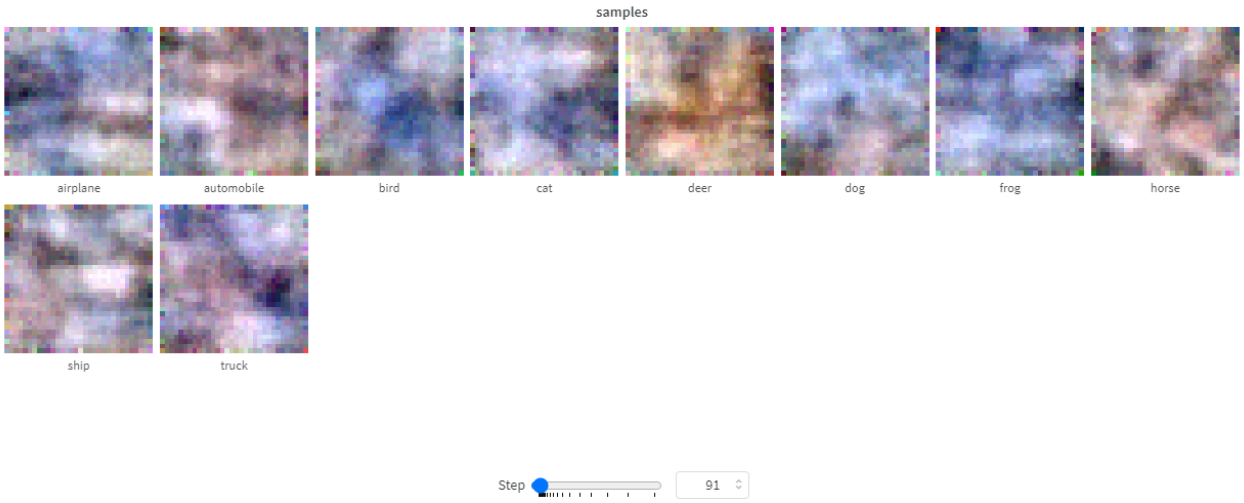
بدايةً قمنا بتدريب النموذج على مجموعة المعطيات CIFAR-10[15]. تحتوي CIFAR-10 على مجموعة من 60000 صورة ملونة (32x32 بكسل) تنتمي إلى 10 فئات مختلفة. هذه الفئات هي السيارات والطائرات والقطط والكلاب والطيور والسفن والشاحنات والضفادع والخيول. أي أن مجموعة المعطيات هذه عبارة عن ثنائيات مؤلفة من الصورة ونص يعبر عن الصف أو الفئة التي تنتمي إليها. من المشاكل التي تعاني منها هذه المجموعة أن النص المرافق لكل صورة لا يحتوي توصيف دقيق للصورة، مما سيعيق عمومية النموذج الناتج كما سنرى لاحقاً. لمتابعة تقدم النموذج أثناء التدريب، يتم توليد الصور الموافقة للنص التالي ‘a photo of a / an label’ لجميع الصفوف أو الفئات labels الموجودة في مجموعة المعطيات، كل عدد معين من الخطوات. عند بداية التدريب، يقوم النموذج بتوليد صور مضجّجة جداً، مع تقدم التدريب وازدياد عدد خطوات التدريب تتحسن الصور وتصبح واضحة الملامح إلى حدٍ ما، بما معناه أن التدريب يتم بشكل صحيح. توضح الأشكال التالية (الشكل 18،الشكل 19، الشكل 20 والشكل 21) بعض العينات المأخوذة في مراحل مختلفة من التدريب كل منها مرفقاً بخطوة التدريب التي أخذت عندها.

[14] https://colab.research.google.com

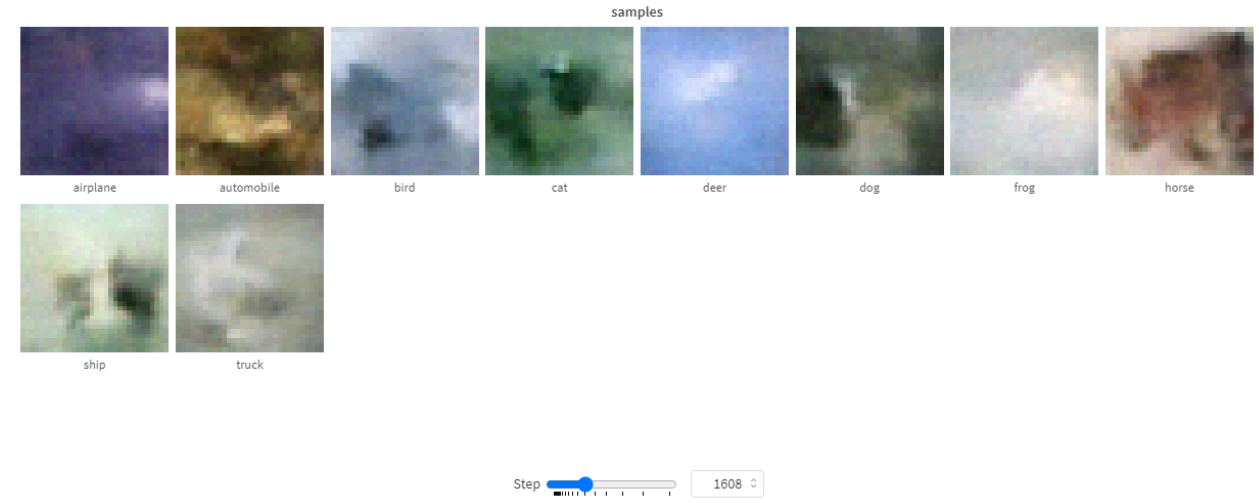
[15] https://www.cs.toronto.edu/~kriz/cifar.html



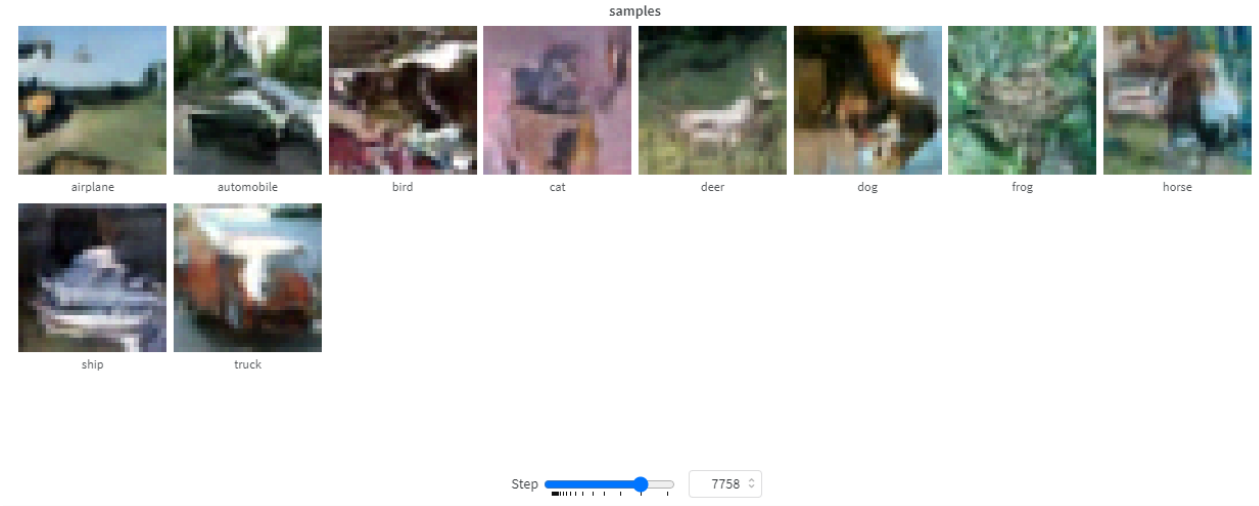
الشكل 18. عينات الصور المولَّدة في بداية تدريب النموذج على CIFAR-10 عند خطوة التدريب 13



الشكل 19. عينات الصور المولَّدة أثناء تدريب النموذج على CIFAR-10 عند خطوة التدريب 91

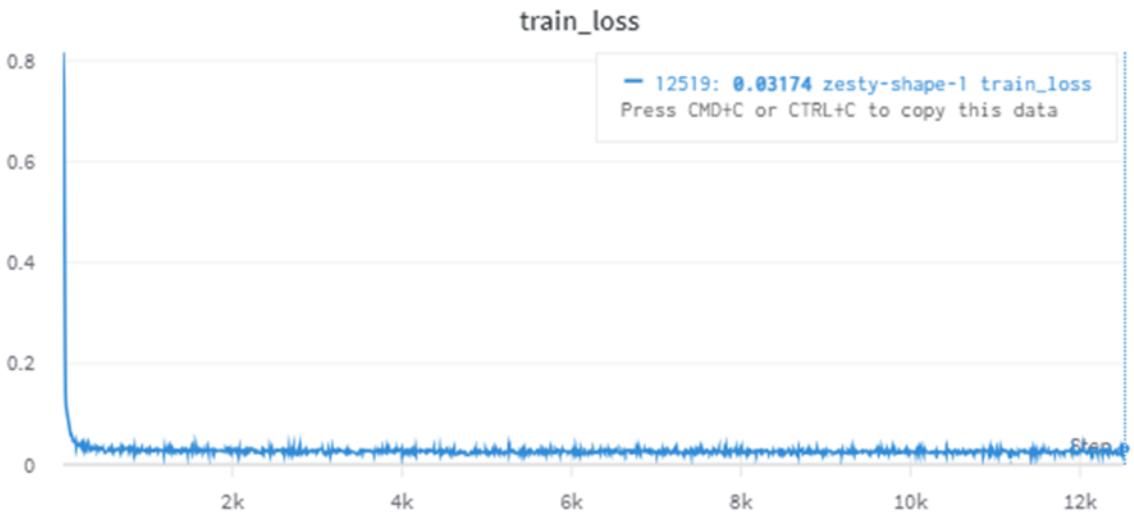


الشكل 20. عينات الصور المولَّدة أثناء تدريب النموذج على CIFAR-10 عند خطوة التدريب 1608

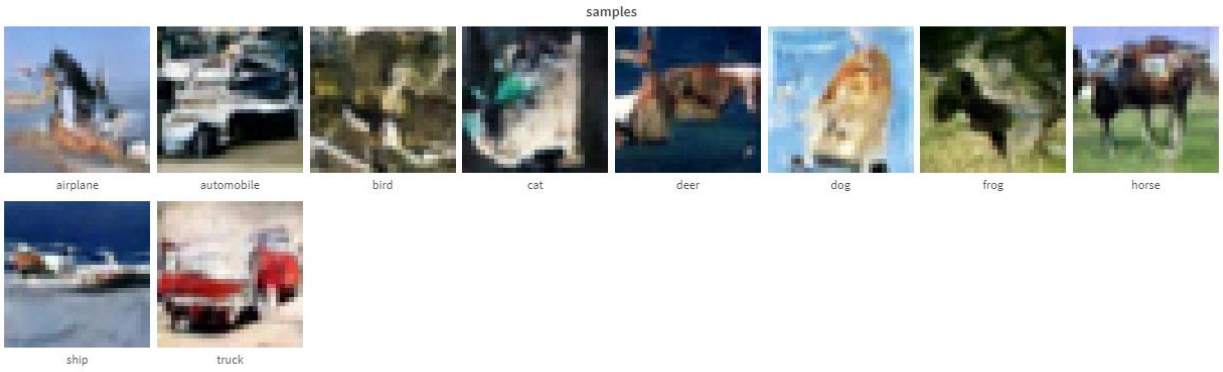


الشكل 21. عينات الصور المولَّدة قبيل انتهاء جلسة التدريب الأولى للنموذج على CIFAR-10 مع خطوة تدريب 7758

تم الوصول إلى حوالي 12000 خطوة تدريب خلال الجلسة الأولى مع الدقة (الخسارة) الموضحة في الشكل 22، وكانت الصور المولدة كما هو وضح في الشكل 23.



الشكل 22. خسارة التدريب على مجموعة البيانات CIFAR-10 في المرحلة الأولى

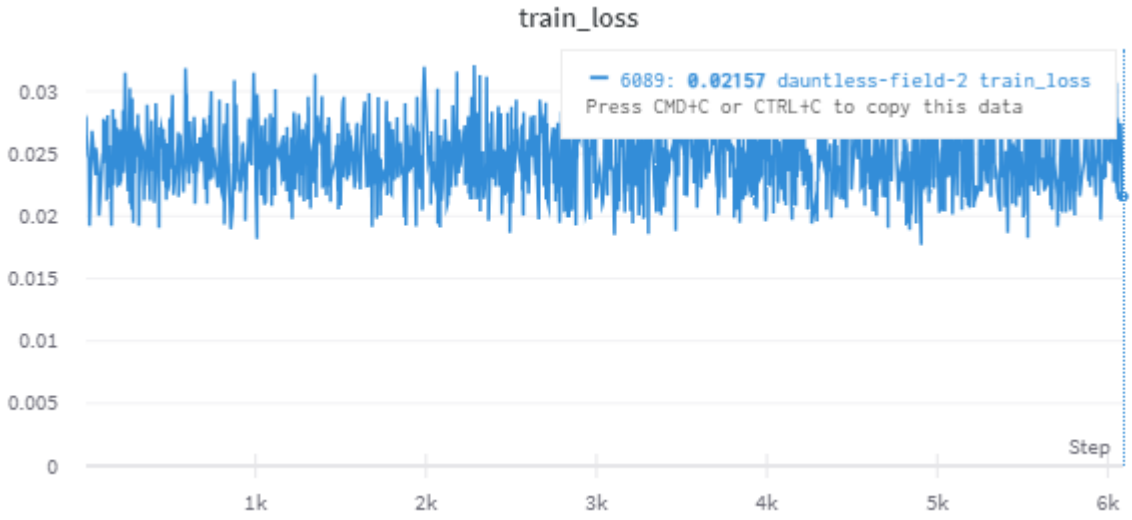


الشكل 23. عينات الصور المولَّدة عند انتهاء جلسة التدريب الأولى للنموذج على CIFAR-10

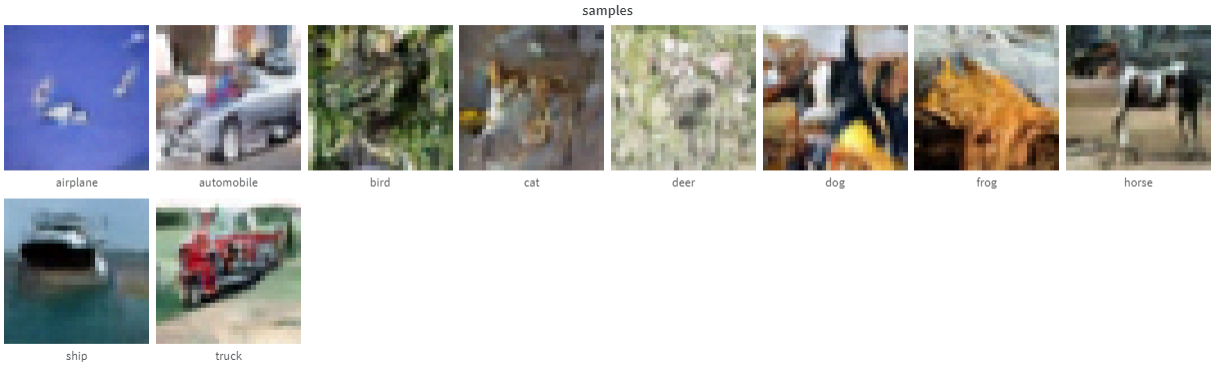
كما نرى فإن أغلب الصور توافق إلى حد مقبول التوصيف الموافق لها مثل horse, automobile, deer, ship, truck بينما هناك صور لا تطابق الوصف المطلوب (الشكل 23). هناك عدة أسباب محتملة لهذه النتيجة نذكر منها:

- زمن التدريب: كما نعلم مثل هذه النماذج تحتاج زمن تدريب كبير جداً وهذا لم يكن متاحاً بسبب زمن جلسات ال colab المحدود.
- نظراً للانقطاعات التي كانت تحدث أثناء التدريب، تم اللجوء إلى حفظ النموذج model دورياً (كل 1000 خطوة تدريب) كنقطة تدقيق مؤقتة checkpoint لنستخدم الأخيرة منها في استئناف التدريب مرة أخرى. بهذه

الطريقة تم استئناف التدريب مرتين على النموذج السابق، في الأولى تم تحقيق 6000 خطوة إضافية لنحصل على النتائج الموضحة في الشكل 24 والشكل 25.

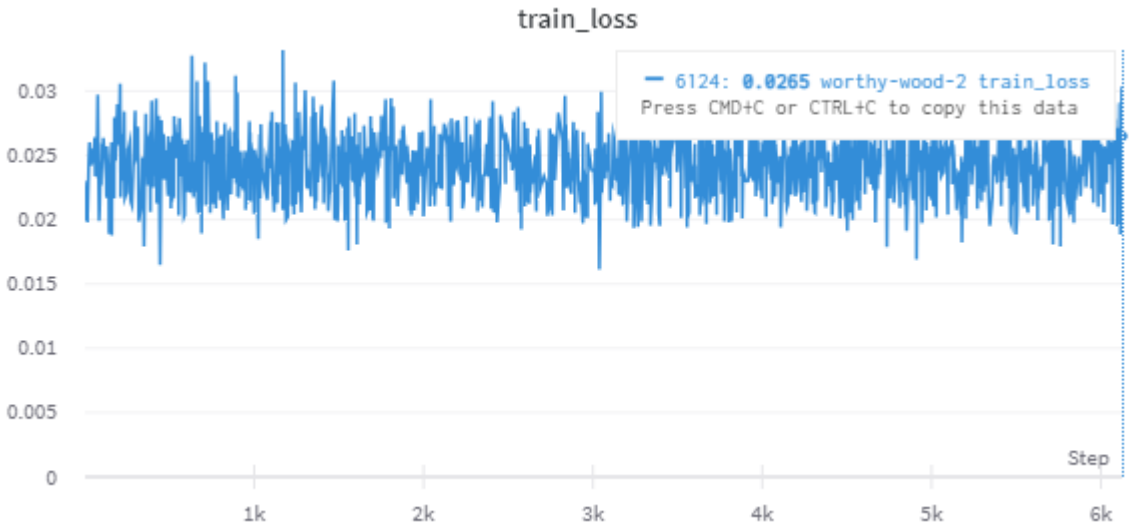


الشكل 24. خسارة التدريب على مجموعة البيانات CIFAR-10 في المرحلة الثانية



الشكل 25. عينات الصور المولَّدة عند انتهاء جلسة التدريب الثانية للنموذج على CIFAR-10

نلاحظ أن تابع الخسارة استقر حول قيمة معينة، وهذا ما يدل على أن النموذج بدأ بالتقارب والتدريب يتم بشكل صحيح (الشكل 24)، كذلك هذا ما نلاحظه على الصور المولدة حيث أصبحت أغلب الصور الموجودة توحي بالتوصيف النصي المستخدم لتوليدها (الشكل 25الشكل 25). في الثانية، تم تحقيق 6000 خطوة أخرى والوصول إلى النتائج الموضحة في الشكل 26 والشكل 27.



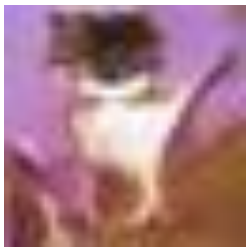
الشكل 26. خسارة التدريب على مجموعة البيانات CIFAR-10 في المرحلة الثالثة

الشكل 27. عينات الصور المولَّدة عند انتهاء جلسة التدريب الثالثة للنموذج على CIFAR-10

يستمر تابع الخسارة بالثبات على نفس القيمة السابقة، لنلاحظ التحسن الواضح في أداء النموذج من خلال الصور التي يتم توليدها، حيث يمكن مشاهدة سيارة في الصورة التي تم توليدها من النص 'a photo of an automobile' (الشكل 27)، كذلك الأمر في كل من صورة الطيارة والطير والكلب والضفدع والحصان والشاحنة. ولكن هذا التحسن يظهر فقط على الصفوف التي تم التدريب عليها فقط، حيث أن هذا النموذج لا يستطيع تمييز الألوان أو الأعداد أو أي سياق نصي مترابط يتضمن ما لم يتم تدريبه عليه. فعلى سبيل المثال، عند طلب توليد صورة موافقة للنص 'red car' تظهر لنا الصورة الموجودة في الشكل 28.

الشكل 28. الصورة المولّدة من التوصيف النصي "سيارة حمراء" باستخدام النموذج المدرب على مجموعة البيانات CIFAR-10

يوحي الشكل العام للصورة رغم عدم الوضوح الموجود بهيئة سيارة، ولكن كما نرى لا يوجد لون أحمر في الصورة. كذلك الصورة الموجودة في الشكل 29 والتي تم توليدها من النص 'blue dog'، حيث من الواضح وجود حيوان في الصورة يوحي بشكل كلب ولكن ليس هناك مراعاة للألوان فلا وجود للون الأزرق في الصورة.



الشكل 29. الصورة المولّدة من التوصيف النصي "كلب أزرق" باستخدام النموذج المدرب على مجموعة البيانات CIFAR-10

يعود السبب في ذلك كما ذكرنا سابقاً، مجموعة المعطيات هذه (مصممة لتصنيف الصور والتعرف عليها وغير مناسبة لتوليدها انطلاقاً من نص) لا تتضمن وصف وافي للصورة وإنما هو عبارة عن الصف الذي تنتمي إليه هذه الصورة فقط.

في محاولة تحسين النموذج السابق، اتجهنا للتدريب على مجموعة معطيات أخرى مستخدمة في هذا المجال. مجموعة المعطيات COCO (Common Objects in Context)[16] هي واحدة من أكثر مجموعات البيانات شهرةً واستخداماً في مجال معالجة الصور وفهمها وتعلم الآلة. تم تطويرها لدعم تحديات فهم الصور والكائنات في سياقها. تحتوي مجموعة COCO على أكثر من 200,000 صورة متنوعة وملونة بأحجام مختلفة، تشمل ما يزيد عن 80 فئة مختلفة من الكائنات والأشياء، مثل الأشخاص والسيارات والحيوانات والأثاث والمأكولات والمزيد. حجم هذه المجموعة كبير جداً حوالي 20 GB وعدد الصور الموجودة في مجموعة التدريب training dataset الخاصة بها يفوق المئة ألف صورة، وبالتالي

[16] https://cocodataset.org

تستغرق المعالجة الأولية لهذه المجموعة زمن كبير جداً لذلك وعلى سبيل الاختبار تم تدريب النموذج على مجموعة التحقق validation dataset المؤلفة من 5000 صورة كل منها مزود بخمس توصيفات مختلفة.

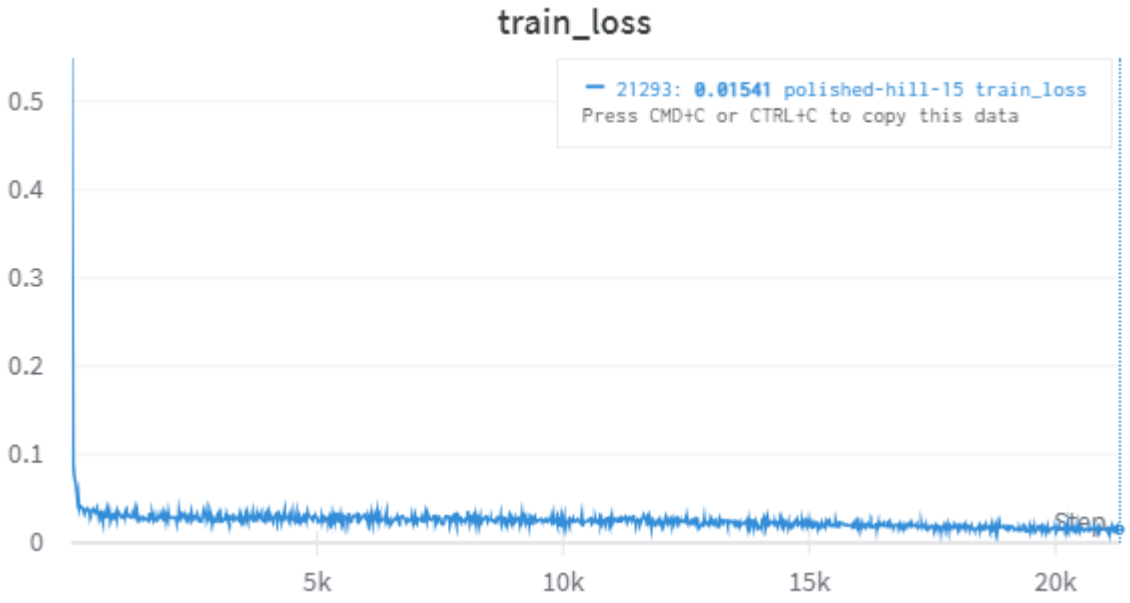
تتضمن المعالجة الأولية لمجموعة المعطيات هذه ثلاث مراحل:

1– تم تكرار كل صورة 5 مرات وإرفاق كل منها بتوصيف من التوصيفات الخمسة المرفقة. وبالتالي أصبح إجمالي الصور 25000 صورة.

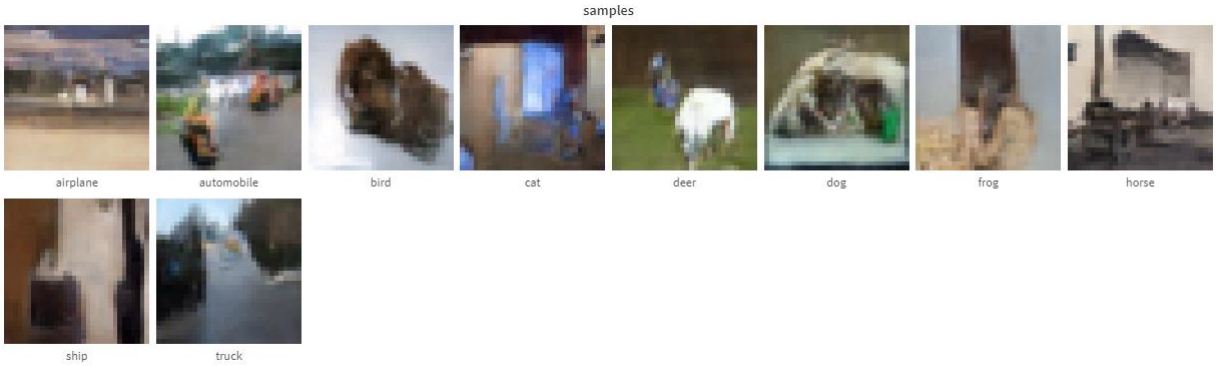
2– معالجة الصور: بسبب عشوائية وعدم انتظام حجم الصور الموجودة، ونظراً لأن النموذج لا يقبل إلّا بحجم صورة ثابت لجميع صور الدخل، فتم اعتماد حجم 32x32 وهو أكبر حجم استطعنا استخدامه ضمن موارد google colab. كذلك تم تحويل الصورة إلى Tensor لنتمكن من إدخالها إلى النموذج.

3– معالجة النص المرافق للصورة: يخضع كل توصيف لعملية تقطيع Tokenization، ليجري إدخاله بعدها إلى محوّل Transformer مدرب مسبقاً T5 للحصول على شعاع التضمين الخاص به Embedding، قبل إدخاله إلى المحوّل يجري جعل جميع التوصيفات متساوية في الطول (تم اختيار هذا الطول ليكون 32 رمز Token) فإذا تجاوز طول التوصيف هذا الطول يجري اقتطاعه عند هذا الطول ما يسمى بعملية القص Truncation، أما في حال كان أقل من هذا الطول يجري إضافة حشو للتوصيف بحيث يصل إلى الطول المطلوب من خلال عملية تسمى الحشو Padding. في عملية الحشو يجري إضافة رموز خاصة تسمى رموز الحشو padding tokens إلى التوصيف حتى يصل الطول المطلوب، ثم يتم إضافة قناع انتباه للشعاع بعد التقطيع attention mask لتجنب تعلم رموز الحشو المضافة أثناء التدريب. يأخذ القناع قيمه في [0، 1] حيث 1 لرموز التوصيف الأساسية،و0 لرموز الحشو المضافة.

استغرقت هذه المعالجة حوالي الساعة (5000 صورة فقط) لذلك كان من الصعب استخدام مجموعة التدريب الكاملة حيث ستحتاج إلى أكثر من 24 ساعة وهذا غير متاح نظراً للجلسات المحدودة المتاحة. عند تدريب النموذج على هذه المجموعة حصلنا على 21000 خطوة تدريب. يوضح الشكل 30 خسارة التدريب في هذه المرحلة، بينما يوضح الشكل 31 الصور التي أصبح النموذج قادراً على توليديها في نهاية هذه المرحلة.



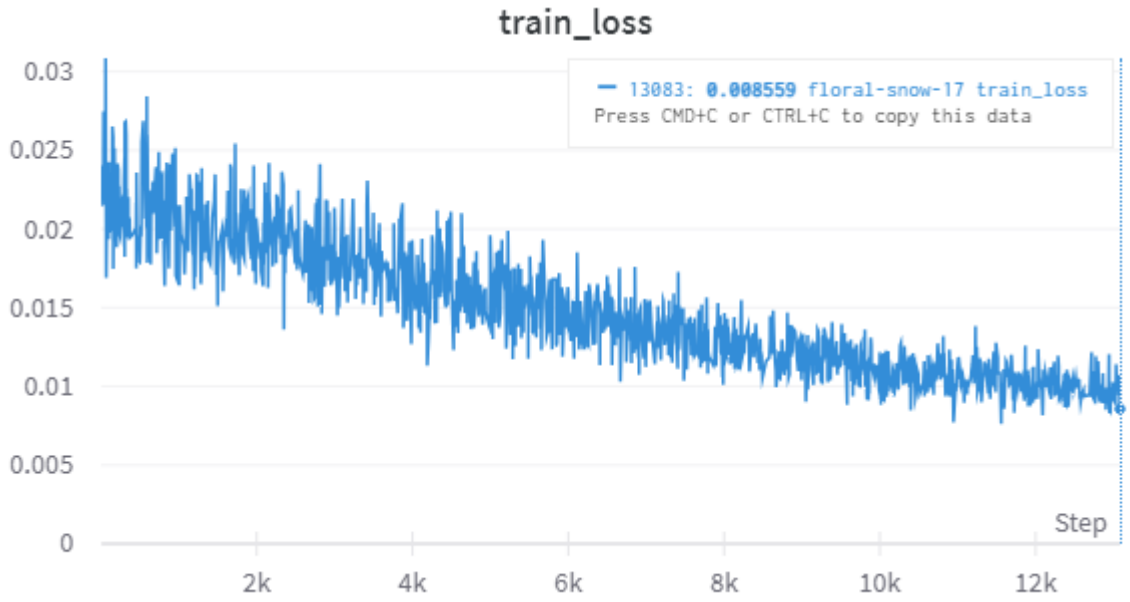
الشكل 30. خسارة التدريب على مجموعة البيانات COCO في المرحلة الأولى



الشكل 31. عينات الصور المولَّدة عند انتهاء جلسة التدريب الأولى للنموذج على COCO

كما نلاحظ هناك نتيجة على بعض الصور مثل bird (الشكل 31)، ولكن في الصورة dog (الشكل 31) نلاحظ وجود زوج من الكلاب علماً أن الوصف النصي الذي تم توليد الصورة منه هو 'a photo of a dog' وبالتالي كنا نتوقع وجود كلب واحد في الصورة. غير أن أغلب الصور لا تصل أساساً إلى الوصف النصي المطلوب. يمكن تفسير النتائج السابقة تبعاً لسببين:

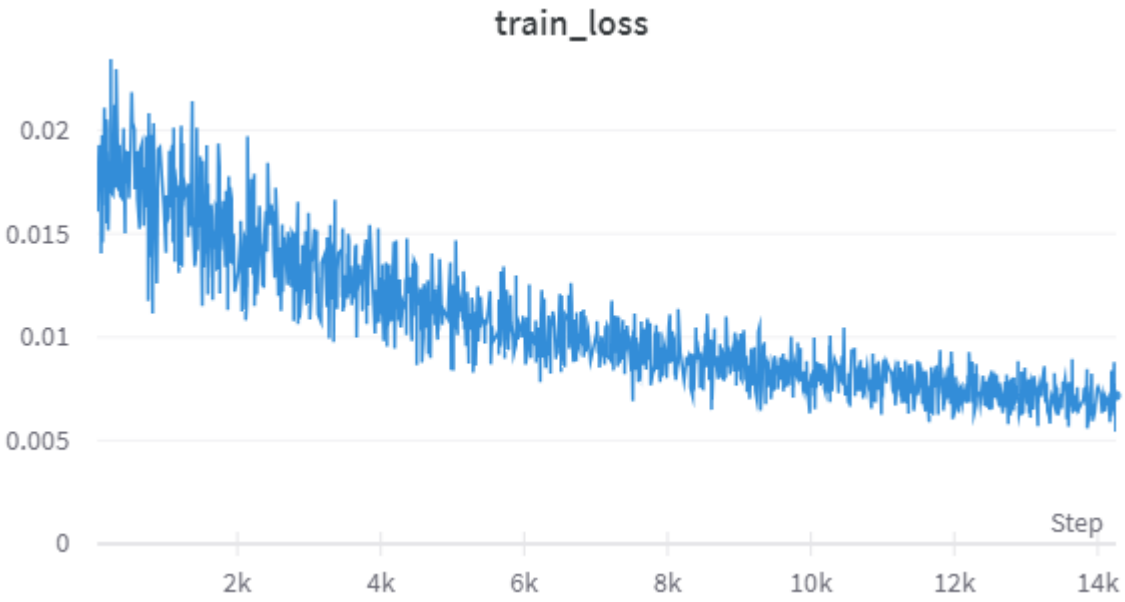
1- نظراً لكون عدد الفئات في مجموعة المعطيات COCO أكبر بكثير من المجموعة السابقة CIFAR-10 كذلك عدد الأغراض التي تندرج تحت كل فئة، فمن المنطقي أن يحتاج النموذج للتدريب عليها زمن تدريب أكثر بكثير من التدريب السابق. لذلك اضطررنا أيضاً إلى استئناف التدريب مرة أخرى على النموذج الذي حصلنا عليه، تم تحقيق 13000 خطوة إضافية وكانت النتائج الموضحة في الشكل 32 والشكل 33.



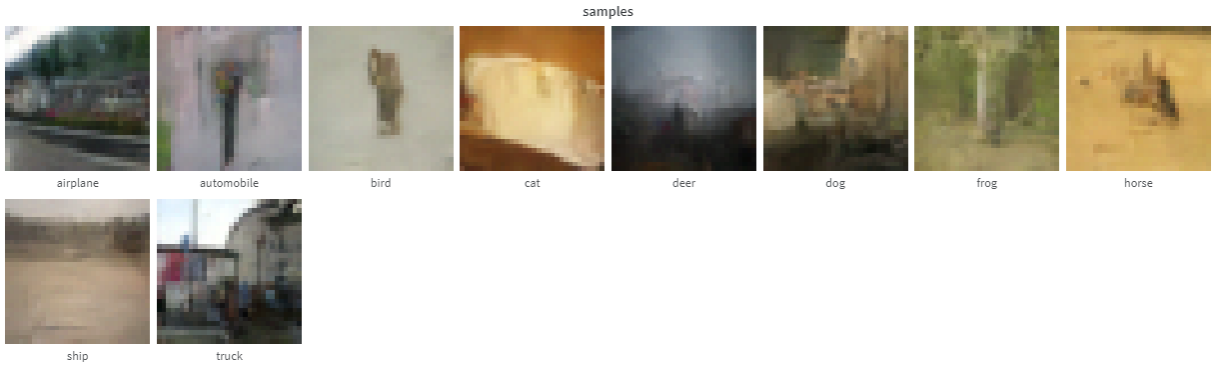
الشكل 32. خسارة التدريب على مجموعة البيانات COCO في المرحلة الثانية

الشكل 33. عينات الصور المولَّدة عند انتهاء جلسة التدريب الثانية للنموذج على COCO

نلاحظ أن خطأ التدريب ينخفض ولكن مع ذلك فإن الصور لا تتحسن. تم استئناف التدريب مرة ثانية أنجز خلالها 14000 خطوة أخرى والنتائج الموضحة في الشكل 34 والشكل 35.



الشكل 34. خسارة التدريب على مجموعة البيانات COCO في المرحلة الثالثة



الشكل 35. عينات الصور المولَّدة عند انتهاء جلسة التدريب الثالثة للنموذج على COCO

نظراً لكون زيادة عدد ساعات التدريب لم يعط النتيجة المرجوة. نعزو النتيجة التي تم الوصول إليها إلى السبب الثاني والذي سنأتي على ذكره في 2.

2- لم نستطع التدريب على مجموعة COCO المخصصة للتدريب بسبب الوقت الضخم جداً الذي تتطلبه معالجتها وأيضاً التدريب عليها. لذلك كان البديل الوحيد المتوفر هو مجموعة التحقق، وهي مجموعة صغيرة جداً مقارنة بعدد الفئات والأغراض الموجودة فيها، وبالتالي لن تعطي هذه المجموعة نتيجة جيدة مهما زدنا زمن التدريب، إضافة إلى أنه توجد مشكلة أخرى وهي مشكلة تصغير الصور أثناء المعالجة الأولية، حيث أدى تصغير الصور من حجمها الأساسي

في مجموعة المعطيات إلى 32x32 إلى تشويهها وضياع خواصها، ولكن كما ذكرنا لم نستطع التدريب على حجم أكبر من هذا الحجم نظراً للموارد الضخمة جداً التي يتطلبها (التدريب على حجم أكبر من هذا غير متاح باستخدام وحدة المعالجة البيانية GPU التي يوفرها المعهد العالي أيضاً).

في محاولة أخرى لتحسين أداء النموذج، تم اللجوء إلى التدريب باستخدام وحدة المعالجة البيانية GPU المتوفرة في المعهد العالي على مجموعة المعطيات Flickr-8k[17]، وهي مجموعة معيارية جديدة لوصف الصور باستخدام النصوص، تتألف من 8000 صورة مقترنة بخمس تسميات توضيحية مختلفة تقدم وصفاً واضحاً للكيانات والأحداث البارزة في الصورة. تم اختيار الصور من ست مجموعات مختلفة على Flickr، ولا تميل إلى احتواء أي أشخاص أو مواقع معروفة، ولكن تم اختيارها يدوياً لتصوير مجموعة متنوعة من المشاهد والمواقف. تم تحميل هذه المجموعة من موقع Kaggle[18]. فيما يلي أهم الخطوات التي تم القيام بها لإنجاز هذا التدريب:

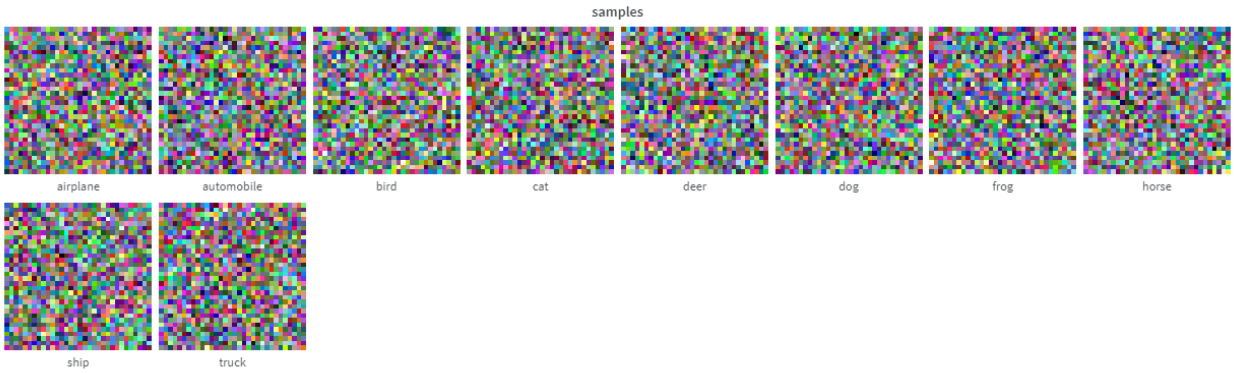
1. تم القيام بتحميل مجموعة البيانات Flickr-8k على مخدم وحدة المعالجة البيانية GPU Server الخاص بالمعهد العالي.
2. تحضير مجموعة البيانات للتدريب: تبين بعد تحميل هذه المجموعة أنها عبارة عن مجلد تحت اسم Images يحوي الصور التي سيتم التدريب عليها، مرفق بملف نصي باسم captions يربط كل صور من الصور في المجلد Images بخمس توصيفات نصية. تم القيام أولاً بقراءة الملف النصي على أنه ملف CSV وذلك باستخدام مكتبة pandas، للحصول على ثنائيات كل منها يمثل اسم صورة معينة (الاسم المحفوظة به في المجلد Images) والتوصيف النصي لها. بعد ذلك، تم إنشاء الصف FlickrDataset الذي يقوم بفتح الصور وربطها مع توصيفها النصي في أزواج يمكن الوصول لها بشكل مباشر بشكل مشابه للوصول لعنصر في مصفوفة، إضافة إلى تطبيق التحويلات على كل منهما (الصورة وتوصيفها) في حال وجودها.
3. التحويلات التي تم تطبيقها على كل من الصور والتوصيفات النصية مشابهة لما تم القيام به على COCO في التدريب السابق. فقد تم تصغير حجم الصورة إلى 32x32 لعدم وجود الموارد الكافية إضافة إلى تحويلها إلى Tensor، كذلك بالنسبة للتوصيفات النصية، فقد تم تقطيع كل منها إلى رموز Tokenization ومن ثم تحويلها إلى شعاع تضمين طوله 32 رمز، يتم قصه إذا ازداد طوله عن هذا الطول وبالعكس، إذا كان أقل من هذا

[17] https://www.kaggle.com/datasets/adityajn105/flickr8k

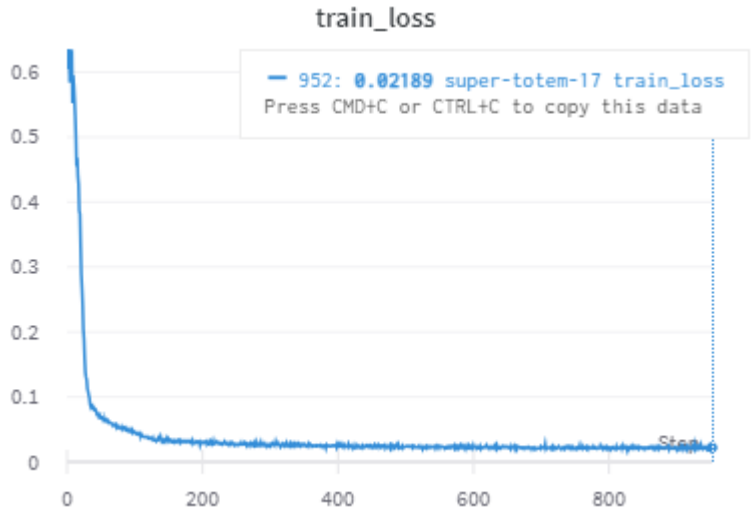
[18] https://www.kaggle.com

الطول يتم إضافة حشو Padding ليصل الطول المطلوب، حيث يجب أن جميع أشعة التضمين متساوية الأبعاد لنتمكن من بدأ التدريب.

4. البدأ بالتدريب: بدأنا تدريب النموذج وبشكل مشابه لعملية تدريب النموذجين السابقين، فقد تمت كتابة نص برمجي script يقوم بتوليد الصور مع كل عدد معين من الخطوات لمراقبة تطور التدريب، وكيفية تأثيره على أداء النموذج. في بداية التدريب كانت الصور عبارة عن ضجيج فقط، كما يوضح الشكل 36. مع تقدم التدريب تنخفض خسارة التدريب، ويبدأ الضجيج بالاختفاء تدريجياً لتظهر مكانه صور ذات ملامح يمكن تمييزها، يمكن رؤية ذلك في الشكل 36 و الشكل 37 والشكل 38.



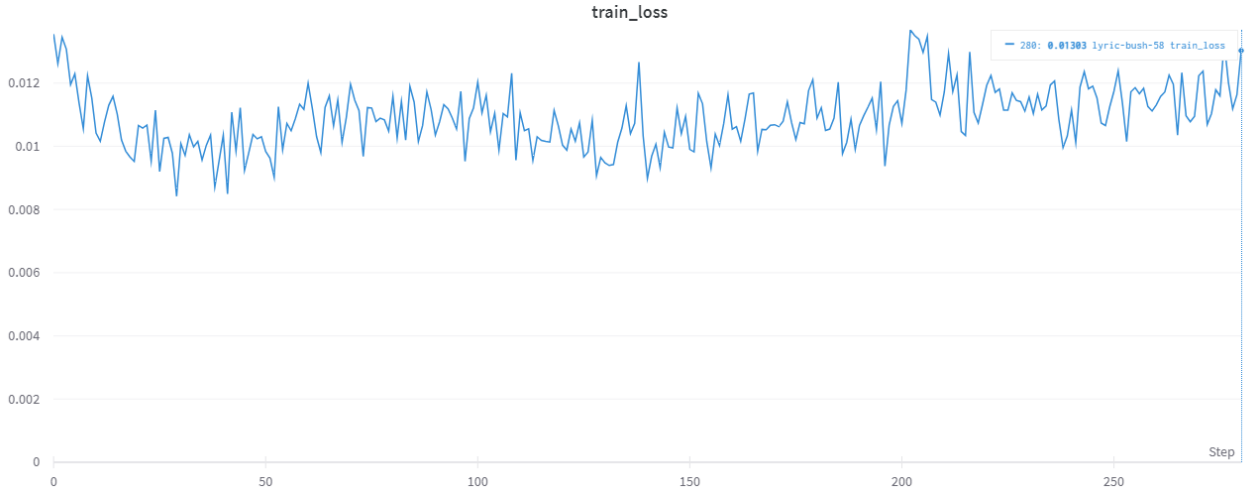
الشكل 36. عينات الصور المولَّدة في بداية تدريب النموذج على Flickr-8k



الشكل 37. خسارة التدريب على مجموعة البيانات Flickr-8k

الشكل 38. تطور الصورة المولدة من التوصيف النصي "صورة لطائرة" مع تقدم التدريب

من الأمور التي تمت ملاحظتها خلال هذا التدريب، أنه كان شديد البطيء حيث لم نستطع تحقيق أكثر من 20000 خطوة تدريب، علماً أن هذا الرقم قد استغرق حوالي الأسبوعين ليتم الوصول إليه. ما زال النموذج يفتقر إلى التدريب كما تشير النتائج(الشكل 38). تم استئناف التدريب مراراً وتكراراً بسبب الانقطاعات التي تخللته، حيث جرى حفظ نقطة تدقيق مؤقتة كل 200 خطوة لنتمكن من متابعة التدريب عليها عند حدوث انقطاع. مع نهاية 20000 خطوة تدريب كانت النتائج الموجودة في الشكل 39 والشكل 40.



الشكل 39. خسارة التدريب خلال عملية استئناف التدريب الأخيرة على مجموعة البيانات Flickr-8k

الشكل 40. الصور المولَّدة مع انتهاء 20000 خطوة تدريب على Flickr-8k

يلخص الجدول 1 النتائج التي تم الحصول عليها خلال مرحلة تدريب النماذج التي أجريناها.

مجموعة البيانات المستخدمة في تدريب النموذج	خسارة التدريب
CIFAR-10	0.027
COCO	0.006
Flickr-8k	0.013

الجدول 1. نتائج مرحلة التدريب

تمثل النتائج السابقة أفضل النماذج التي حصلنا عليها بعد التدريب الذي قمنا به. جرى حفظها كنقاط تدقيق مؤقتة checkpoints ليتم استخدامها في موقع الوب.

2.5. بناء واجهة التخاطب

سوف نستعرض في هذا القسم خطوات بناء واجهة التخاطب مع المستخدم (تطبيق الوب) التي سنوظف فيها النماذج التي دربناها في القسم السابق.

1.2.5. المتطلبات وحالات الاستخدام

سنورد في هذه الفقرة المتطلبات الوظيفية وغير الوظيفية للنظام، وكذلك مخطط حالات الاستخدام وسيناريو العملية.

1.1.2.5. المتطلبات الوظيفية وغير الوظيفية

يجب أن يكون النظام قادراً على تلبية المتطلبات الوظيفية التالية:

- السماح لمستخدم الموقع بإدخال توصيف نصي ليتم بناءً عليه توليد الصورة الموافقة باستخدام نموذج تعلم عميق مدرب على هذه المهمّة.

- عرض الصورة بعد أن يتم توليدها، وذلك بأربعة قيم مختلفة للدقة (32x32, 64x64, 128x128, 256x256).
- السماح للمستخدم بتحميل download الصورة بالدقة الأصلية للنموذج المستخدم في توليدها.

بينما تتلخص المتطلبات غير الوظيفية في:

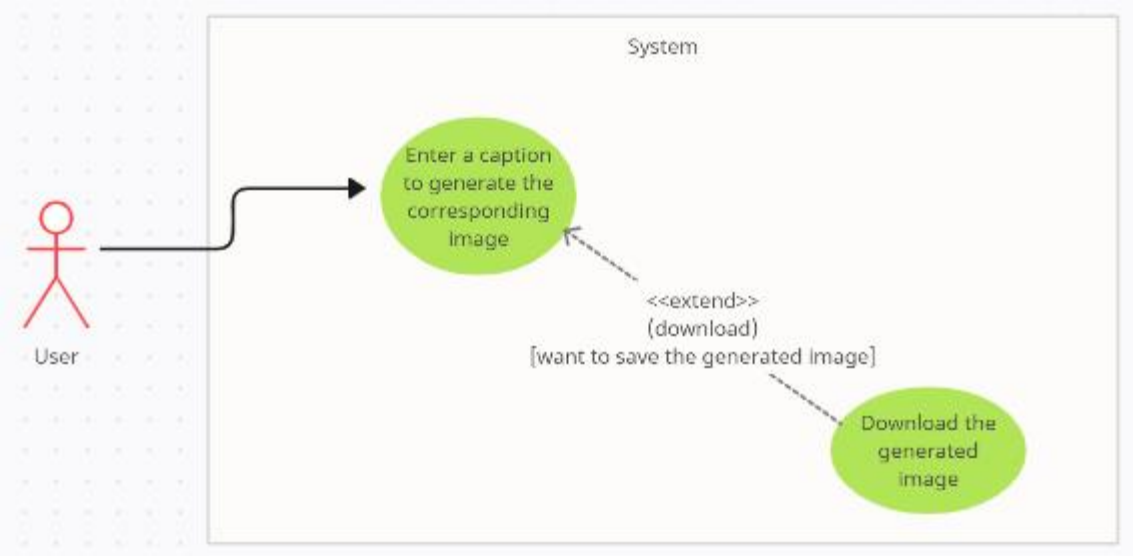
- سهولة الاستخدام.
- اعتماد واجهة تخاطب بيانية.

2.1.2.5. حالات الاستخدام

توجد حالتي استخدام رئيسيتين للموقع:

1. إدخال التوصيف النصي المراد توليد الصورة على أساسه.
2. تحميل الصورة المولَّدة.

يبين الشكل 41 مخطط حالات الاستخدام لموقع الوب.



الشكل 41. مخطط حالات الاستخدام الخاص بالموقع

فيما يلي سرد حالتي استخدام الموقع اللتين تم ذكرهما:

- **حالة الاستخدام إدخال التوصيف النصي المراد توليد الصورة بناءً عليه Enter a caption**
  - اسم الحالة: إدخال التوصيف النصي المراد توليد الصورة على أساسه.

- الملخص: في هذه الحالة يقوم المستخدم بإدخال التوصيف النصي للصورة واختيار نموذج التعلم العميق. ليقوم النظام بعد ذلك بعرض الصورة المولدة للمستخدم مع إمكانية القيام بتحميلها عند الطلب.
- الفاعلون: المستخدم
- الظروف السابقة: لا يوجد.
- الظروف اللاحقة: تم عرض الصورة المولدة للمستخدم مع إمكانية القيام بتحميلها عند طلب ذلك.

- **حالة الاستخدام تحميل الصورة download**

- اسم الحالة: تحميل الصورة.
- الملخص: في هذه الحالة يقوم المستخدم بحفظ الصورة المولدة على جهاز الحاسب الخاص به.
- الفاعلون: المستخدم
- الظروف السابقة: وجود صورة مولدة بناءً على توصيف نصي مُدخَل من المستخدم.
- الظروف اللاحقة: تم تحميل الصورة لدى المستخدم.

يمكن توقع السيناريو الناجح لاستخدام الموقع على الشكل التالي:

1. دخول المستخدم إلى صفحة الموقع الرئيسية.
2. قيام المستخدم بإدخال التوصيف النصي للصورة، واختيار نموذج التعلم العميق المراد استخدامه لتوليدها.
3. قيام النظام بعرض الصورة المولدة من قبل النموذج المختار.
4. طلب المستخدم تحميل الصورة في حال رغبته بذلك.
5. قيام النظام بتحميل الصورة على حاسب المستخدم.

2.2.5. بناء موقع الوب

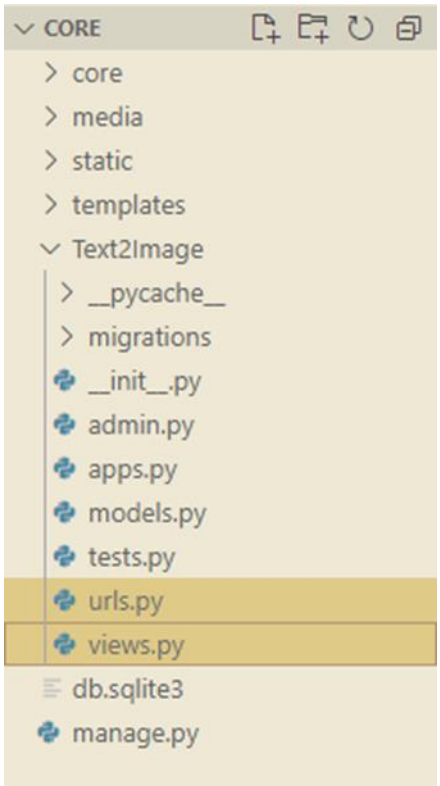
منصة الوب جانغو Django

Django[19] هو إطار عمل framework عالي المستوى يستخدم في تطبيقات الوب، ويشجع فكرة التطوير السريع والتصميم النظيف. يوفر هذا الإطار الكثير من متاعب تطوير الوب بحيث يتم التركيز على كتابة التطبيق فقط. تم استخدام Django من أجل بناء صفحة وب لتوليد الصور بناء على التوصيف النصي.

[19] https://www.djangoproject.com

جرى أولاً بناء مجلد العمل، ومن ثم أنشأنا بداخله تطبيق يدعى Text2Image. قمنا بإضافة إعدادات التطبيق configuration الموجودة في ملف apps.py التابع لمجلد التطبيق إلى قائمة INSTALLED_APPS في ملف settings.py الموجود في المجلد core من أجل تسجيل التطبيق، وكذلك ضمّنا رابط url خاص به بحيث أصبح الوصول لموقع التطبيق تحت الرابط التالي: localhost/Text2Image/index. يحتوي مجلد التطبيق Text2Image على المجلدات والملفات الآتية (الشكل 42):

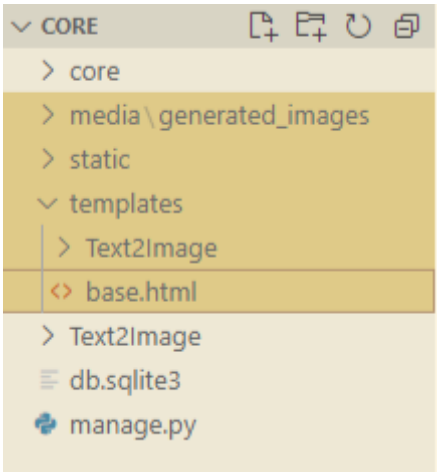
- views: يتضمن هذا المجلد مجموعة توابع كل تابع منها يدعى view. يتم استدعاء كل view لمعالجة طلب معين من المستخدم، وإعادة صفحة ويب له على شكل HTTP Response بحيث تتضمن هذه الصفحة نتائج معالجة هذا الطلب .
- urls: لربط كل رابط معين مع view يتم استدعاؤها عند طلب هذا الرابط.



الشكل 42. الهيكلية العامة المستخدمة في بناء موقع الوب

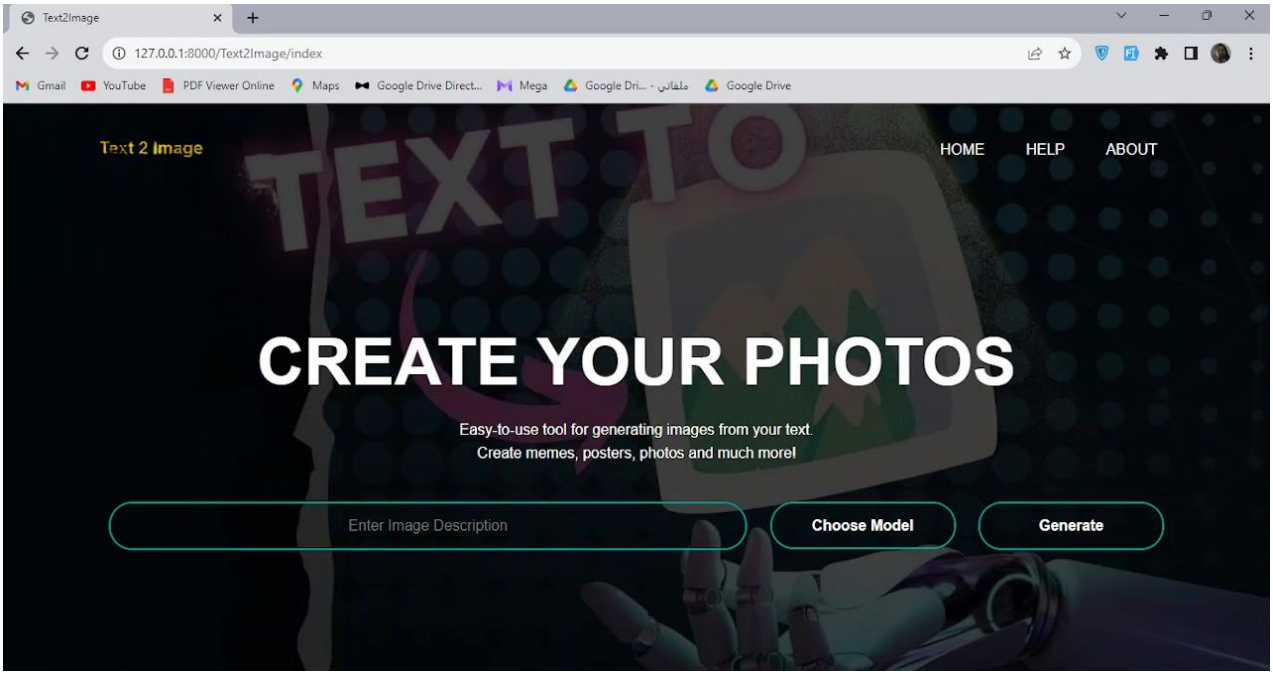
في حين أنه تم إنشاء كل من المجلدات التالية ضمن مجلد العمل الأساسي، وذلك لتحقيق مبادئ إعادة الاستخدام وقابلية الصيانة المطلوبة عند إنشاء أي تطبيق برمجي:

- templates: يتضمن هذا المجلد ملفات ال HTML الخاصة بكل تطبيق ضمن مجلد يحمل اسم التطبيق، إضافة إلى ملفات ال HTML المشتركة بين جميع التطبيقات.
- static: يتضمن عدة مجلدات هي css, js, img, md لحفظ ملفات CSS وملفات JavaScript وكذلك لحفظ الصور ونماذج التعلم العميق التي سيجري استعمالها في الموقع.
- media: يتضمن مجلد اسمه generated_images تخزن فيه الصور التي يتم توليدها ليتمكن المستخدم من تحميلها في حال أراد ذلك.



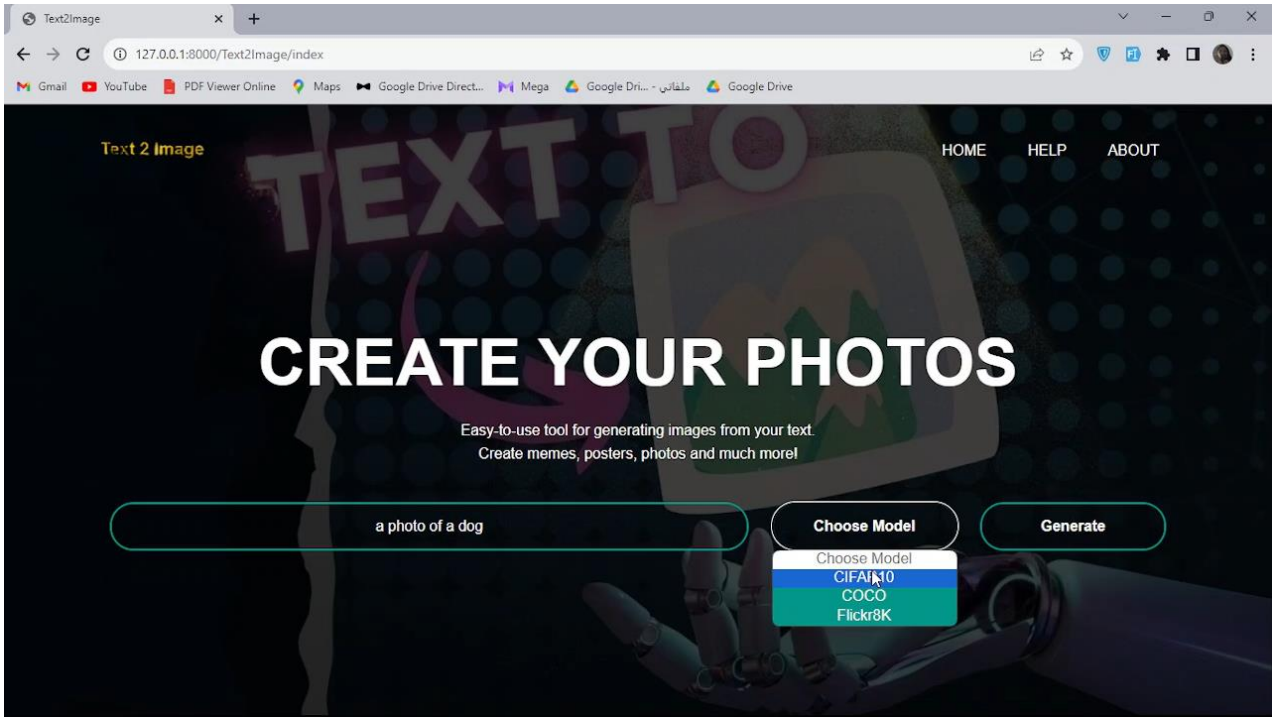
الشكل 43. ملفات media, static, templates ضمن مجلد العمل

يقوم المستخدم بالوصول إلى الصفحة الرئيسية من خلال الرابط الذي ذكرناه. يوضح الشكل 44 الصفحة الرئيسية للموقع.



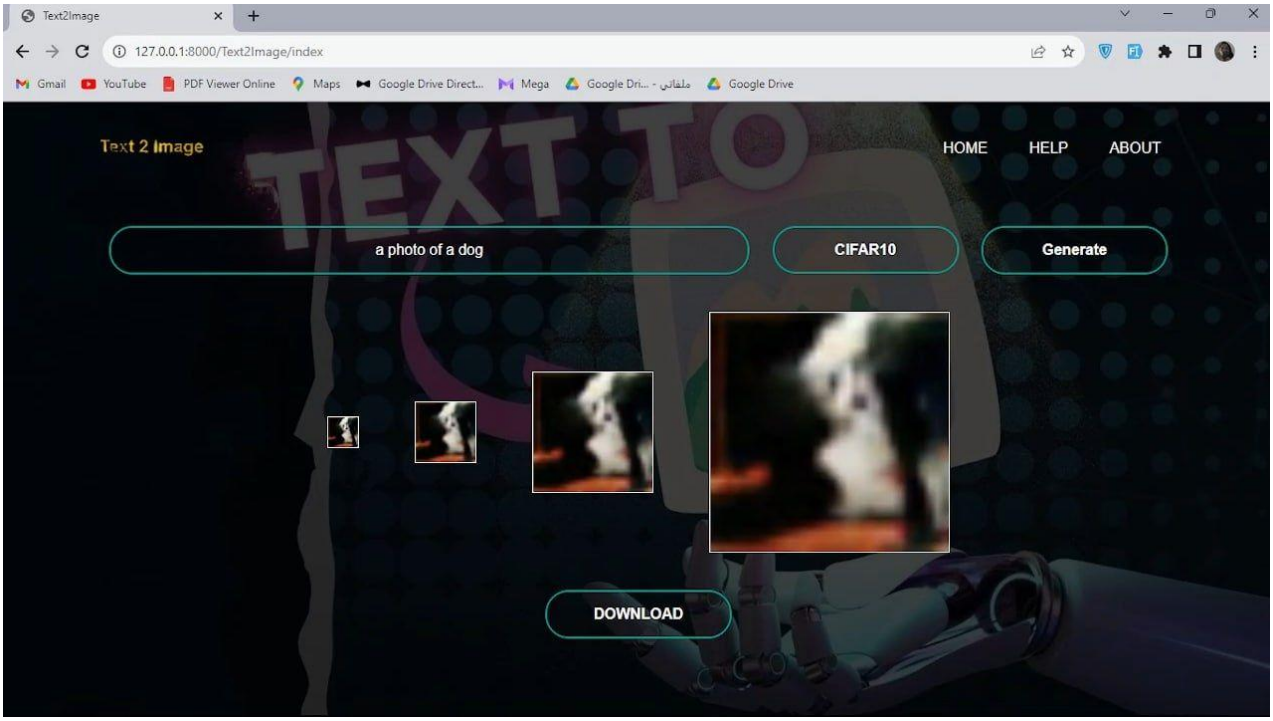
الشكل 44. الواجهة البيانية الرئيسية لموقع الوب

يقوم المستخدم بإدخال التوصيف النصي المراد توليد الصورة على أساسه واختيار النموذج المطلوب استخدامه في التوليد (الشكل 45)، ثم يضغط زر توليد أو إنشاء Generate، عند الضغط على الزر Generate، يتم تمرير النص إلى السيرفر وتطبيق النموذج عليه. يجري أولاً تضمين النص باستخدام محوّل النموذج نفسه، ومن ثم يتم توليد الصورة الموافقة وحفظها في مجلد السيرفر كما أسلفنا ليتمكن المستخدم من تحميلها لاحقاً.

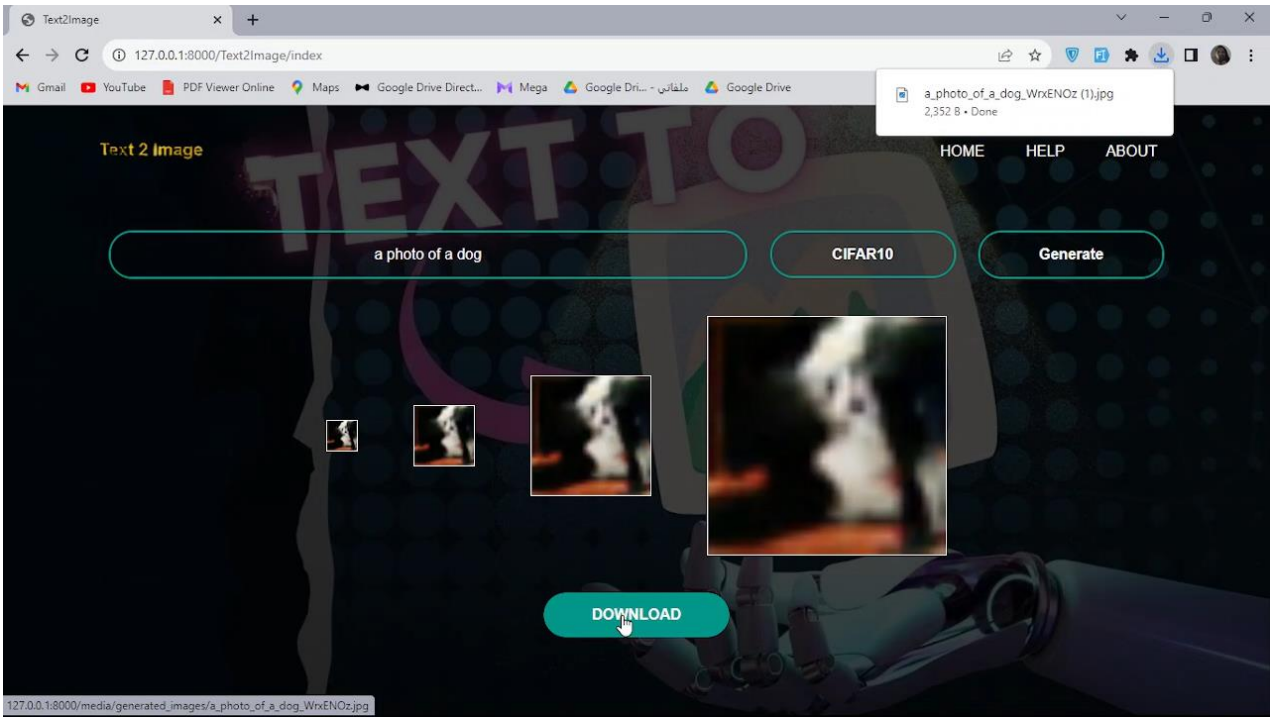


الشكل 45. توضيح كيفية إدخال التوصيف النصي واختيار نموذج التعلم العميق

عندما تنتهي عملية المعالجة تتم إعادة صفحة وب (الشكل 46) للمستخدم تحوي الصورة المولَّدة بأحجام مختلفة مع زر تحميل DOWNLOAD. عندما يضغط المستخدم زر التحميل DOWNLOAD يتم تحميل الصورة لدى المستخدم تلقائياً باستخدام المتصفح (الشكل 47).



الشكل 46. صفحة الوب المستخدمة لعرض الصورة المولَّدة



الشكل 47. تحميل الصورة المولَّدة

الخاتمة

تم في هذا العمل بناء نموذج لتوليد الصور من التوصيف النصي باستخدام تقنيات التعلم العميق، وذلك بعد القيام بدراسة مرجعية موسّعة للتعرف على أهم التقنيات المستخدمة في هذا المجال. جرى تدريب النموذج السابق على مجموعات بيانات مختلفة، حيث تم في كل مرحلة تفسير النتائج التي تم الحصول عليها والاستفادة منها في محاولة تحسين الأداء الذي يقدمه هذا النموذج. واجه هذا العمل العديد من الصعوبات والتحديات فيما يتعلق بالموارد الحسابية الضخمة وزمن التدريب الطويل الذي يتطلبه المشروع، وجرت محاولة الاستفادة من جميع الموارد المتاحة في محاولة تحسين النموذج وتطويره. في محاولة لتجاوز الصعوبات السابقة التي اعترضت سير المشروع تم اللجوء إلى حفظ النموذج خلال التدريب بشكل دوري كنقاط تدقيق مؤقتة checkpoints وذلك لاستئناف التدريب عليها في حال انقطاع التدريب. تم في النهاية الوصول إلى نتائج مقبولة نوعاً ضمن الموارد التقنية والعتادية المتاحة من حيث الربط بين التوصيف والصورة وكذلك من حيث وضوح الصورة. كانت نتيجة التدريب السابق ثلاثة نماذج جرى وضعها في الاستخدام من خلال مكاملتها مع موقع وب يتيح للمستخدم توليد الصورة الموافقة للتوصيف النصي الذي يدخله، كذلك يتيح له تحميلها أيضاً في حال أراد ذلك.

الآفاق المستقبلية

تطور مجال توليد الصور من التوصيف النصي Text-to-Image Generation بشكل ملحوظ في السنوات الأخيرة، ومن المتوقع أن تستمر هذه التطورات في المستقبل. نذكر بعض الافاق المستقبلية لهذا المجال:

- زيادة دقة الصور المولّدة: العمل على زيادة دقة وجودة الصور المولدة. قد يتم ذلك باستخدام تقنيات التعلم العميق مثل الشبكات العصبونية الصنعية العميقة Deep Neural Networks والاستفادة من النماذج الأكثر تقدماً، فعلى سبيل المثال أدى استخدام سلسلة متتالية من شبكات U-Net على خرج شبكة التوليد لزيادة دقة الصورة بشكل ملحوظ.
- توليد صور متعددة الزوايا والأوضاع: قد تنتقل التطورات إلى توليد صور متعددة الزوايا والأوضاع من وصف نصي واحد، مما يسهم في تحسين واقعية الصور المولدة وزيادة تنوعها.
- تطبيقات متعددة: يمكن أن يشمل مجال توليد الصور من التوصيف النصي مجموعة متنوعة من التطبيقات منها ما هو تجاري أيضاً، بما في ذلك الفنون الرقمية والتصميم والتعليم والألعاب والتسويق.
- توليد صور ثلاثية الأبعاد والواقع المعزز: قد تتجه التطورات أيضاً نحو توليد صور ثلاثية الأبعاد وتجارب واقع معزز، مما يوفر تجارب أكثر تفاعلية وواقعية للمستخدمين.

بشكل عام، يمكن توقع مزيد من التطورات في مجال توليد الصور من التوصيف النصي مع تقدم التكنولوجيا والبحث المستمر في مجال تعلم الآلة والذكاء الصنعي.

المراجع

[1] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

[2] Aggarwal, C. C. (2018). Neural networks and deep learning. Springer, 10, 978-3.

[3] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," in IEEE Transactions on Neural Networks and Learning Systems, doi: 10.1109/TNNLS.2021.3084827.

[4] Ian Goodfellow et al. “Generative adversarial nets.” In: Advances in neural information processing systems 27 .(2014)

[5] Wang, Q., Ma, Y., Zhao, K., & Tian, Y. (2022). A comprehensive survey of loss functions in machine learning. Annals of Data Science, 9(2), 187-212.

[6] M. Mirza and S. Osindero. Conditional generative adversarial nets. arXiv:1411.1784, 2014.

[7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. NeurIPS, 2020.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. arXiv:1706.03762, 2017.

[9] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

[10] Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.

[11] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[12] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In ICCV, 2017.

[13] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text-to-image synthesis. In ICML, 2016.

[14] Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., and He, X. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. arXiv:1711.10485, 2017.

[15] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. ArXiv, abs/2205.11487, 2022.

[16] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Bob McGrew Pamela Mishkin, Ilya Sutskever, and Mark Chen. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In arXiv:2112.10741, 2021.

[17] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. In arXiv, 2022.

[18] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In NeurIPS, 2022.

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. JMLR, 2022.

[20] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In CVPR, 2022.

[21] Yan, X., Yang, J., Sohn, K., Lee, H. (2016). Attribute2Image: Conditional Image Generation from Visual Attributes. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. ECCV 2016.

[22] Farhadi, A., Hejrati, M., Achar, P., Rashtchian, C., & Hockenmaier, J. (2010). Every picture tells a story: Generating sentences from images. In European conference on computer vision (pp. 15-29).

[23] Ramzan, S., Iqbal, M. M., & Kalsum, T. (2022). Text-to-Image Generation Using Deep Learning. IEEC 2022, p. 16.