الجمهورية العربية السورية

المعهد العالي للعلوم التطبيقية والتكنولوجيا

قسم هندسة المعلوميات – اختصاص برمجيات وذكاء صنعي

العام الدراسي 2024-2025

إعداد:

مريم خيربك

إشراف:

د. آصف جعفر د. عمر حمدون ما. عماد قرحيلي

ص

مشروع أعد لنيل إجازة الهندسة باختصاص هندسة البرمجيات والذكاء الصنعي بعنوان:

**بيئة ذكية لتحليل الصور الطبية لأمراض نادرة اعتماداً على تقنيات التعلم العميق**

# فهرس المحتويات

[فهرس المحتويات 2](#_Toc204711193)

[جدول المصطلحات 5](#_Toc204711194)

[مقدمة عامة 6](#_Toc204711195)

[1. الإطار العام للمشروع 7](#_Toc204711196)

[1.1 أهداف المشروع 7](#_Toc204711197)

[2.1 متطلبات النّظام 7](#_Toc204711198)

[1.2.1 المتطلبات الوظيفية 8](#_Toc204711199)

[2.2.1 المتطلبات غير الوظيفية 8](#_Toc204711200)

[2. تنجيز النظام 9](#_Toc204711201)

[1.2 المخطط العام للنّظام المراد 9](#_Toc204711202)

[2.2 آلية العمل 10](#_Toc204711203)

[1.2.2 اختيار مجموعة البيانات Mendeley Data 10](#_Toc204711204)

[2.2.2 توليد التوصيفات 11](#_Toc204711205)

[3.2.2 النموذج المستخدم 12](#_Toc204711206)

[3.2 آلية التدريب 12](#_Toc204711207)

[1.3.2 بنية النموذج النهائي 12](#_Toc204711208)

[2.3.2 تجهيز مجموعة التدريب 13](#_Toc204711209)

[3.3.2 تحميل البيانات وتنسيق الدفعات 14](#_Toc204711210)

[4.3.2 حساب مصفوفة logits 15](#_Toc204711211)

[5.3.2 دالة الخسارة 17](#_Toc204711212)

[6.3.2 التدريب 21](#_Toc204711213)

[3. الاختبارات والنتائج 22](#_Toc204711214)

[1.3 اول تجربة 22](#_Toc204711215)

[4. الأدوات المستخدمة 23](#_Toc204711216)

[1.4 PostgreSQL 23](#_Toc204711217)

[2.4 React 23](#_Toc204711218)

[3.4 Python 23](#_Toc204711219)

[4.4 FastApi 23](#_Toc204711220)

[5.4 Json Web Token (JWT) 23](#_Toc204711221)

[6.4 Git 23](#_Toc204711222)

[7.4 GitHub 23](#_Toc204711223)

[5. التنجيز البرمجي لنظام الويب 24](#_Toc204711224)

[1.5 معماريّة المشروع Architecture 24](#_Toc204711225)

[1.1.5 البنية النظيفة 24](#_Toc204711226)

[2.1.5 خواص النظام المبني وفق البنية النظيفة 24](#_Toc204711227)

[3.1.5 طبقات النظام المنجز 25](#_Toc204711228)

[4.1.5 أهم الأنماط التصميمة المستخدمة في بناء النظام 26](#_Toc204711229)

[2.5 واجهات التّطبيق 26](#_Toc204711230)

# جدول المصطلحات

|  |  |
| --- | --- |
| المصطلح | المعنى/الدلالة |
| Zero-shot Learning | التعلم دون أمثلة - التصنيف دون تدريب مسبق على الفئة |
| Vision-Language Models (VLM) | نماذج فهم الصورة واللغة |
| Contrastive Language-Image Pretraining (CLIP) | التدريب التبايني المسبق للنصوص والصور |
| VIT |  |
| Swin Transformer |  |
| Tensors | موتّرات |
| logits | مصفوفة احتمالات ما قبل التفعيل |
|  |  |

# مقدمة عامة

تُعد مسألة عدم توفر بيانات تدريب كافية إحدى أبرز التحديات التي تواجه أنظمة التصنيف المعتمدة على التعلم الموجه (Supervised learning) خاصة في المجالات الحساسة كالقطاع الطبي. إذ تتطلب النماذج التقليدية توافر كمية كبيرة من البيانات المصنفة لكل حالة مرضية، وهو أمر غير ممكن في كثير من الأحيان، سواء بسبب ندرة بعض الأمراض، أو حتى صعوبة توفير صور للبيانات المطلوبة لأسباب تتعلق بالخصوصية أو غيره، أو ظهور أمراض جديدة وغير معروفة كما حدث أثناء جائحة "كوفيد 19"، أو نتيجة انتقال أفراد من بيئات جغرافية مختلفة إلى مناطق جديدة لا تعد ضمنها هذه الأمراض مألوفة، أي أمراض تصيب الأفراد المنتمين لعرق معين بشكل أكبر من غيرهم.

لإيجاد حلول مناسبة لمثل هذه المسألة يكون بناء نظام قادر على التكيّف مع الحالات غير المتوقعة دون الحاجة إلى بيانات تدريبية متخصصة أمر بالغ الأهمية. ومن هنا تبرز الحاجة إلى اعتماد مقاربات بديلة قادرة على التعميم خارج نطاق البيانات المرئية سابقاً. تُمثل تقنية التعلم بدون أمثلة مسبقة (zero-shot Learning) حلاً واعداً في هذا السياق، حيث تتيح تصنيف الحالات بناءً على وصفها النصي فقط، دون الحاجة إلى وجود صور تدريبية سابقة. والتي تزداد فاعليتها عند استخدام نماذج قوية لفهم الصور والنصوص ضمن إطار نموذج متعدد الوسائط (multimodal framework)، مما يفتح أفاقاً واسعة نحو بناء أنظمة طبية أكثر مرونة واستجابة.

**الفصل الأول**

# الإطار العام للمشروع

نهدف في هذا الفصل إلى عرض تعريف عام بالمشروع، وذكر الدوافع والهدف الذي نسعى لتحقيقه، ويتم ذلك من خلال تحديد أهداف المشروع والإشارة إلى المتطلبات الوظيفية وغير الوظيفية الخاصة بهذا المشروع.

## أهداف المشروع

تواجه الأنظمة الطبية تحديات جوهرية عند التعامل مع الحالات النادرة أو غير المسبوقة، حيث يتعذر في كثير من الأحيان توفير بيانات تدريبية كافية تغطي كافة الأمراض المحتملة، خاصة في ظل الأزمات الصحية المفاجئة أو التنقل البشري المستمر عبر المناطق الجغرافية المختلفة. الأمر الذي يستدعي حلولاً أكثر مرونة تعتمد على نماذج قادرة على التعميم خارج حدود البيانات التي تم تدريبها عليها.

بناءً على ذلك، يهدف هذا المشروع إلى تطوير نظام ذكي لتصنيف الأمراض الطبية من الصور الشعاعية والسريرية اعتماداً على التوصيفات النصية للأمراض، دون الحاجة إلى تدريب مسبق على كل مرض بشكل خاص، مستفيدين من قوة النماذج متعددة الوسائط، ويحقق المشروع الأهداف التالية:

1. الهدف الأول: هو حل المسألة من منظور الذكاء الصنعي، حيث نحاول توظيف التطورات الأخيرة في مجال التعلم العميق -خاصة في مجالي معالجة النصوص والصور- وصولاً لمقاربة قادرة على تصنيف الصور الطبية بناءً على وصفها النصي فقط، عبر آلية التعلم بدون أمثلة (zero-shot Learning)، مما يتيح له التعامل مع أمراض غير معروفة سابقاً ضمن بيئة التدريب.
2. الهدف الثاني: دمج النموذج ضمن تطبيق برمجي صحي ذكي، يسمح للمستخدمين (أطباء ومساعديهم) برفع صور طبية والحصول على اسم المرض الأكثر تشابهاً، حيث جرت مراعاة الأسس والمبادئ المتعارف عليها في هندسة البرمجيات وصولاً لتطبيق قابل للتوسع وسهل الصيانة.
3. الوصول لاستجابة سريعة وفعالة دون التضحية بدقة التشخيص، مع إمكانية تفسير النتائج عبر عرض الجمل النصية التي ساهمت في تحقيق أعلى تشابه (Top-K phrases)

## متطلبات النّظام

نبيّن فيما يلي المتطلبات الوظيفية وغير الوظيفية لهذا التطبيق.

### المتطلبات الوظيفية

أن يتيح النّظام -والذي يهدف لتخديم نموذج ذكاء اصطناعي ضمن تطبيق ويب يخدم مركز صحي- للمستخدم ما يلي:

* إمكانية تسجيل الدخول وفقاً لدور المستخدم (إداري، مساعد، طبيب).

حسب الدور البرمجي:

1. الإداري Admin:

* إضافة مستخدمين وتحديد أدوارهم.
* تعديل معلومات المستخدمين.

1. المساعد (السكرتير):

* إضافة مريض جديد إلى قاعدة البيانات.
* تعديل معلومات المريض الشخصية.
* عرض قائمة الأمراض التي أضيفت لكل مريض

1. الطبيب:
   * الوصول إلى المرضى الذين يتابعهم الطبيب فقط.
   * عرض وتشخيص الحالات المرضية للمريض.
   * رفع صورة والوصول إلى تشخيص ذكي باستخدام نموذج CLIP.
   * تحليل الصورة لحصول على Top-5 احتمالات للأمراض.
   * الحصول على الجمل النصية الأعلى تشابهاً التي ساهمت في التصنيف

### المتطلبات غير الوظيفية

* سهولة الاستخدام: عن طريق تصميم واجهات بسيطة وواضحة يمكن التنقّل بينها بسهولة.
* إعادة الاستخدام Reusability: عن طريق تصميم النظام بشكل يتيح إمكانيّة استخدام المكونات البرمجيّة المختلفة في مشاريع أخرى أو تطبيقات مختلفة دون الحاجة لتعديلات كبيرة
* قابليّة التّوسع: عن طريق تنجيز النّظام بشكل مرن يسمح بإضافة مكوّنات أو وظائف جديدة دون التّأثير على المكونات أو الوظائف الموجودة بالفعل.

# تنجيز النظام

في هذا الفصل سيتم توضيح بيئة العمل المستخدمة في تنجيز النظام، وسيتم عرض بعض الأمثلة من النظام

## المخطط العام للنّظام المراد

إن النظام المقترح يهدف إلى تصنيف صور رنين مغناطيسي MRI لأمراض نادرة أو قليلة الانتشار فلم يتم تدريب النموذج عليها بشكل كافٍ إما لعدم توفر البيانات أو لعدم انتشار هذه الأمراض في أماكن جغرافية محددة من خلال الـ Few-Shot learning، كما يهدف لتصنيف أي مرض جديد لم يتوفر له بعض مجموعة بيانات كما في حالة انتشار جائحة كورونا من خلال Zero-Shot learning.

فتم تصميم النظام بحيث يأخذ صورة الرنين المغناطيسي كدخل له، مع وصف نصي منسّق بصيغة JSON لمجموعة الأمراض، يذكر بهذا الوصف اسم الأمراض مع عدد من العبارات التي تعنى بتوصيف كل منها، فيتمكن نموذج الذكاء الاصطناعي من تحليل الحالة وتصنيف المرض التي توضحه الصورة المدخلة.

تتم معالجة الصورة لتصبح بأبعاد تناسب النموذج، وبعدها يتم تطبيق تطبيع Normalization لقيم البكسلات لتتناسب مع خصائص النموذج، تمرر الصورة بعد المعالجة إلى نموذج يقوم باستخراج التمثيل الشعاعي للصورة (Image Embedding) للتعبير عن الصورة بشكل عددي.

وتتم معالجة النص من خلال تحويله إلى tokens عبر tokenizer، التي سيتم تمريرها إلى نموذج لاستخراج التمثيل الشعاعي لكل عبارة (Text Embedding).

يتم بعد الحصول على التمثيلين الشعاعين حساب التشابه بين الصورة وكل عبارة ونختار وفقاً لذلك النص الأكثر تشابهاً فيكون هو المرض المعبر عن الصورة.

Inputs

MRI image

JSON  
Descriptions

Vision Encoder

Preprocessing

Image Embedding

Tokenizer

Text Encoder

Text Embedding

Similarity Calculation

Outputs

## آلية العمل

في الفقرات القادمة سيتم توضيح خطوات العمل المتبعة بهدف تنجيز النظام السابق:

### اختيار مجموعة البيانات Mendeley Data

تمثل مجموعة البيانات الطبية المتوفرة عبر منصة Mendeley Data مصدراً غنياً لتدريب نماذج الذكاء الاصطناعي في مجال التصوير الطبي. تتضمن المجموعة 34,192 صورة طبية موزعة على 40 فئة مرضية، مع تركيز خاص على صور الرنين المغناطيسي (MRI) للدماغ، الجهاز العصبي، والعمود الفقري، تم جمعها من عدة مستشفيات.

تحتوي المجموعة على 26 فئة تتعلق بأمراض الدماغ، بالإضافة إلى فئات تغطي اضطرابات عصبية، أمراض الحبل الشوكي، واضطرابات جهازية. تشمل المجموعة أمراضاً نادرة مثل متلازمة ووكر-واربورغ (808 صورة)، Pachygyria مع نقص تنسج المخيخ (592 صورة)، مرض Moyamoya مع نزيف بطيني (864 صورة)، مرض Hallervorden-Spatz (1064 صورة)، وضمور العضلات من نوع Fukuyama (920 صورة).

كما تتضمن المجموعة أمراضًا شائعة مثل ورم الشبكية مع انتشار داخل الجمجمة (1200 صورة)، متلازمة شوغرن (1056 صورة)، مرض أوسغود-شلاتر الثنائي مع التهاب المفاصل المزمن (665 صورة)، الورم الليفي العصبي من النوع الأول مع الورم الدبقي البصري (944 صورة)، التصلب الحدبي (664 صورة)، وأورام الدماغ (1048 صورة).

تقدم الصور في هذه المجموعة بتنسيقات قابلة للمعالجة، مثل JPEG, PNG، مع تنظيم الملفات ضمن مجلدات حسب الفئة المرضية (Class-wise folder structure). يُسهّل عملية التحميل والمعالجة باستخدام مكتبات تعلم عميق شائعة مثل PyTorch وTensorFlow، حيث يمكن التعامل مع كل مجلد كـ label مباشر للصورة.

يمكن الاستفادة من هذه المجموعة في:

* تصنيف الصور الطبية (Medical Image Classification): من خلال بناء نموذج يصنّف صورة MRI إلى إحدى الفئات الأربعين.
* التعلّم بدون إشراف مسبق (Zero-shot/Few-shot): بفضل وجود فئات نادرة ذات عينات محدودة، تعتبر المجموعة مناسبة جدًا لتقييم النماذج التي تتعامل مع محدودية البيانات.

من خلال هذه المواصفات كانت مجموعة البيانات هذه مناسبة للنظام المراد بناءه.

### توليد التوصيفات

نظراً لطبيعة المشروع، ومن كون النموذج يأخذ ملف نصي بصيغة JSON يوضح توصيف للأمراض المراد التدريب عليها، أو تصنيفها فكانت المرحلة اللاحقة لاختيار مجموعة البيانات هي إنشاء التوصيفات المعبرة عن كل فئة مرضية، بشكل يعكس خصائصها السريرية والتصويرية، حيث أن مجموعة البيانات المستخدمة وفرت 40 مرض مع أسماء هذه الأمراض فقط كتصنيفات labels، هذا النقص في البيانات الوصفية استدعى تصميم مرحلة مستقلة مخصصة لبناء مجموعة توصيفات نصيّة ذات جودة عالية تمثّل كل مرض بطريقة تسمح للنموذج بتعلّم الارتباط بين الصور والنصوص بشكل فعال.

في البداية، تم اختبار عدّة نماذج من نوع (Vision-Language Models) لتوليد أوصاف نصية، من ضمنها نموذج BLIP، والنموذج nlpconnect/VIT-gpt2-image-captioning، بالإضافة إلى نموذج متخصص في المجال الطبي image-captioning-base-chestXray-finetuned. أظهرت النتائج أن هذه النماذج لم تستطع توفير أوصاف دقيقة تعكس تفاصيل الأمراض بشكل واضح، بل كانت مخرجاتها إما عامة أو غير مناسبة للسياق الطبي المطلوب، ما استدعى اللجوء إلى استراتيجية أخرى.

فكان الحل المقترح لبدء لب المشروع هو اللجوء إلى ChatGPT مدعوماً بهندسة موجهات (Prompt engineering) مصممة بعناية، هدفت إلى توليد توصيفات متعددة لكل مرض بأسلوب منهجي، تم استخدام 8 موجهات مختلفة لتغطية ثلاثة أنماط من الوصف:

1. خمسة أوصاف مرئية (image-level) كل منها يتراوح بين 60-100 كلمة.
2. ثلاثة أوصاف مبنية على الموقع التشريحي أو التوزيع المرضي (location-based) بطول يتراوح بين 50-80 كلمة.
3. وصفان للتصنيف السريري أو التشخيصي (summaries classification) بطول 40-60 كلمة بأسلوب تقني سريري.

حيث تم تعزيز هذه التوصيفات عبر تمرير حوالي 20 صورة طبية من مجموعة البيانات المستخدمة، ليتم تعديل التوصيف ليصبح أفضل بناءً على الصور، الأمر الذي ساعد في إنتاج توصيفات أكثر واقعية وشمولاً للسمات المرضية.

### النموذج المستخدم

كما هو واضح من الدراسة المرجعية فإن النموذج المقترح لهذا العمل هو نموذج من نوع VLM وهو CLIP. لكن لكون التعامل في هذا المشروع مع صور ونصوص طبية بشكل خاص، حيث أنه يعد مجال دقيق جداً، فتم استخدام النموذج medCLIP، وهو نموذج مفيد للتعلم التبايني بين الصور والنصوص الطبية، حيث أنه يهدف إلى تحسين جودة التمثيلات المتعددة الوسائط (صورة-نص) في المجال الطبي، مما يجعله مناسباً للتطبيقات الطبية ذات الموارد المنخفضة.

إن نموذج medCLIP، كما أشرنا سابقاً، يحتوي مرمّز للصورة ومرمّز للنصوص، نتكلم في الفقرات التالية عن العمل المنجز أثناء مراحل المشروع، وبعده يتم ذكر التجارب ونتائجها.

## آلية التدريب

### بنية النموذج النهائي

تم تدريب النموذج باستخدام إطار medCLIP الذي يعتمد على مبدأ التعلم التبايني (Contrastive Learning) بين الصور والتوصيفات النصية المقابلة لها. يهدف هذا الأسلوب إلى تعزيز التشابه بين الصور والنصوص التي تنتمي إلى نفس الفئة المرضية، وفي الوقت ذاته تقليل التشابه بين الصور والنصوص التي تنتمي لفئات متغايرة، مما يسمح للنموذج بتعلّم تمثيلات متعددة الوسائط (Image-Text Representations) فعالة.

وللوصول إلى البنية النموذجية المثلى، أُجريت مجموعة من التجارب على الاحتمالات المختلفة الممكنة لكل من مرمّز الصور ومرمّز النصوص (الموضح بالتفصيل في قسم التجارب العملية). بعد تحليل النتائج ومقارنة الأداء، تم الاستقرار على البنية التالية:

* Swin Transformer-base كمرمّز للصور (Image encoder)، حيث تم إلغاء تجميد جميع الطبقات وتدريب النموذج بالكامل خلال فترة التدريب، مما يسمح للنموذج بالتكيف بدقة مع خصائص الصور الطبية في مجموعة البيانات.
* BioClinicalBERT كمرمّز للنصوص (Text encoder). وهو نموذج لغوي كبير (LLM) مُدرَّب مسبقاً على نصوص طبية سريرية. لذا تم تجميد جميع طبقات النموذج الأساسية للحفاظ على التمثيل اللغوي الطبي المدرب مسبقاً، والاكتفاء بتحديث طبقة الإسقاط (Projection Layer) فقط خلال التدريب.

### تجهيز مجموعة التدريب

قبل البدء في عملية التدريب، تم تطبيق مجموعة من الخطوات التحضيرية على كل من الصور والتوصيفات النصية، بهدف توحيد الشكل العام للبيانات وضمان توافقها مع بنية النموذج.

أولاً: معالجة الصور:

* تنسيق الصور: تم التأكد من أن جميع الصور بصيغة قابلة للمعالجة مثل (JPEG, PNG) حيث تم تحميلها من المجلدات المصنفة حسب الفئة المرضية، وتحويلها جميعاً إلى RGB.
* إعادة التحجيم: تم تعديل أبعاد جميع الصور لتكون بقياس ثابت قدره 224×224 بكسل.
* التحويل إلى موتّرات (Tensors): تحويل الصور إلى مصفوفات عددية مناسبة لشبكة الرؤية باستخدام مكتبة PyTorch.
* التطبيع (Normalization): تُطبع الصور باستخدام المتوسط والانحراف المعياري لمجموعة ImageNet لضمان توافق القيم مع النموذج الأولي.

ثانياً: معالجة النصوص:

* تنظيف التوصيفات: تم التحقق من أن جميع التوصيفات النصية خالية من الفراغات الزائد، وتحويل جميع المحارف الموجودة لأحرف صغيرة.
* التجزئة: تم استخدام الـ AutoTokenizer الخاص بـ BioClinicalBERT من مكتبة HuggingFace لتجزئة النصوص الطبية إلى رموز (tokens) بما يتوافق مع بنية النموذج.
* توحيد الطول: تم تحديد طول أقصى (Max Length) لكل توصيف نصي عند 128 رمزاً، مع تطبيق padding للنصوص القصيرة، والتأكد أن العبارات المستخدمة لن تتجاوز هذا الطول.
* تحويل إلى موتّرات (Tensors): بعد التجزئة وتوحيد الطول، تم تحويل التوصيفات إلى تنسيقات قابلة للإدخال إلى النموذج (attention\_mask, input\_ids)[[1]](#footnote-1).

#### حساب درجة تفرّد العبارات النصية (Phrase Uniqueness)

توصيف الأمراض بعبارات طبية، يمكن أن يسبب مشكلة بسبب التشابه في الأعراض بين عدد من الأمراض، على سبيل المثال:

إذا كانت لدينا صورة رنين مغناطيسي تنتمي لفئة "ورم دبقي"، والعبارة النصية المرتبطة بها هي:

**"صورة رنين مغناطيسي تُظهر كتلة ورمية غير منتظمة الحدود** **ذات شذوذ في الإشارة وتباين محيطي، قد تشير إلى ورم دبقي"،** يتم حساب درجة تفرّد هذه العبارة عبر قياس تشابهها مع عبارات أمراض أخرى كعبارة **"صورة رنين مغناطيسي تُظهر منطقة منزوعة الميالين غير منتظمة الحدود قد تشير إلى تصلب لويحي"،** أو عبارة من مرض مشابه مثل **"صورة رنين مغناطيسي تُظهر ورماً سحائياً منتظم الحدود داخل الدماغ".**

إن التشابه بالمعاني الدلالية للأمراض قد يسبب التباساً للنموذج، فلحل هذه المشكلة تم حساب درجة تفرّد العبارات النصية ضمن ملف التوصيفات المنشأ (Phrase Uniqueness) بهدف تمييز الأوصاف الأكثر تحديداً ودلالةً بين الأمراض المختلفة عن الأوصاف العامة أو الشائعة التي قد تتكرر بين أكثر من فئة مرضية. تُحتسب درجة التفرّد لكل عبارة نصية عن طريق تمثيل العبارة بشكل متّجه في فضاء تضمين دلالي (Semantic Embedding Space) باستخدام نموذج Sentence-BERT، ثم يتم قياس تشابه كوساين (Cosine Similarity) بين هذه العبارة وجميع العبارات الأخرى التي تنتمي لأمراض مختلفة ضمن مجموعة البيانات. رياضياً، تُحسب درجة تفرّد العبارة من خلال العلاقة التالية:

حيث هي العبارة النصية المُراد حساب تفردها، و هو تمثيلها الدلالي، و هو تمثيل دلالي لأي عبارة نصية أخرى من مرض مختلف.

إذا كانت العبارة قريبة جداً من وصف مرض آخر، تحصل على قيمة تشابه مرتفعة وبالتالي درجة تفرّد منخفضة (قريبة من الصفر). وبالمقابل، إذا كانت العبارة تميّز المرض بشكل واضح عن باقي الأمراض (أي التشابه منخفض)، فإنها تحصل على درجة تفرّد عالية تقترب من الواحد الصحيح.

هذا النهج يسمح للنموذج بمنح أهمية أكبر للعبارات التي توفّر إشارات دلالية واضحة تميز بين الأمراض المتشابهة في صور الرنين المغناطيسي، مما يُعزّز من دقّة التصنيف وقابلية التعميم في مجال التصنيف الطبي.

### تحميل البيانات وتنسيق الدفعات

تم تصميم محمل بيانات (DataLoader) مخصص لتحميل الصور والتوصيفات النصية بشكل متزامن، مع الحفاظ على التوازن بين الفئات المرضية، وضمان التوافق بين تنسيقات الصور والنصوص داخل كل دفعة تدريبية.

* تنظيم البيانات: تم تنظيم مجموعة الصور ضمن مجلدات مصنّفة حسب اسم الفئة المرضية. بالمقابل، تم تحميل التوصيفات النصية من ملف JSON يحتوي على مفتاح الفئة (اسمها) وقائمة توصيفات مرتبطة بها.
* آلية الربط: لم يتم الاعتماد على اقتران مباشر (Image-Caption Pairs)، بل تم استخدام آلية مطابقة على مستوى الفئة (Category-level pairing)، حيث يتم أخذ صورة من فئة معينة، وتربط عشوائياً مع توصيف (أو مجموعة توصيفات) من نفس الفئة، وذلك بهدف تحسين أداء النماذج متعددة الوسائط في المجالات الدقيقة (fine-grained domains)، حتى لا يتم تحديد العلاقة بين الصورة والنص بشكل دقيق فهذا يعيق التعلم.

وهو أمر حقيقةً يستحق الاستفاضة بشرحه قليلاً بهدف التوضيح لأهميته في البحث هذا، فكان الهدف من هذه النقطة ما يلي:

* الاستفادة من أوصاف غنية على مستوى الفئة، حتى وإن لم تكن مرتبطة بصورة محددة بعينها، لأن الصور ضمن الفئة نفسها تتشارك خصائص بصرية أو مرضية متشابهة.
* تجاوز الحاجة لتوفير وصف دقيق لكل صورة، وهو أمر مكلف أو غير ممكن عملياً في المجالات الطبية، حيث تتطلب التوصيفات الدقيقة خبرة بشرية، والعديد من الأمور الأخرى كتوافر جاهزية كافية لجمع البيانات، والهدف الأهم هو كما ذكرنا سابقاً في حال تواجد مرض لم تكشف كل أوصافه الدقيقة بعد.
* السماح للنموذج بتعلّم السمات المميزة للفئة ككل بدلاً من الاعتماد فقط على علاقة فردية بين (صورة وتوصيف).
* التقليل من تأثير الضجيج أو التوصيفات الجزئية التي قد لا تعكس الصورة بشكل كامل (مثلاً، في حال كانت الأعراض غير ظاهرة بوضوح في الصورة المختارة).
* رفع أداء التصنيف بدون أمثلة (Zero-shot learning): من خلال تدريب النموذج على التوافق بين التمثيلات الصورية والنصية على مستوى المفهوم الطبي لا المثال الفردي.

إذن، هذه الاستراتيجية تحوّل إشراف الصورة-النص من إشراف دقيق (instance-level supervision) إلى إشراف مرن قائم على المفهوم (concept-level supervision)، مما ينعكس إيجاباً على قابلية النموذج للتعميم على أمراض جديدة أو صور غير مرئية مسبقاً.

* توليد الدفعات: خلال كل خطوة تدريب (training step)، يتم تكوين دفعة (mini-batch) تحتوي على:
* B صورة طبية من فئات مختلفة.
* لكل صورة، يتم اختيار توصيف (أو عدة توصيفات) من فئة الصورة لتكوين B × num of phrases إدخال.
* تتم مطابقة كل صورة مع توصيف المقترن بها، ويُحسب التشابه بين الصورة وكافة النصوص في الدفعة.
* تهيئة الدخل: لكل دفعة يتم تمرير input\_ids وattention\_maask، الخاصة بالنصوص بعد ترميزها بـ BioClinicalBERT tokenizer، بينما تمرر الصور بشكل مباشر إلى مرمّز SWIN بعد تطبيق المعالجة المناسبة.
* تطبيق خسارة تباينية مدروسة تأخذ بعين الاعتبار التشابه بين الفئات.

### حساب مصفوفة logits

في هذا النظام، تلعب مصفوفة احتمالات ما قبل التفعيل الـ logits دورًا محوريًا في حساب دالة الخسارة التباينية، وهي توضّح "الدرجات الخام" التي تعكس درجة التشابه بين كل صورة وكل توصيف نصي ضمن الدفعة التدريبية، قبل تحويلها إلى احتمالات. يتم توليد مصفوفة الـ logits عبر سلسلة من الخطوات:

* توليد التضمينات العددية (Embeddings)
  + كل صورة i من الدفعة التي حجمها B تُمثل بتضمين عددي
  + كل توصيف نصي j من نفس الدفعة يًمثل بتضمين عددي

حيث D يمثل عدد أبعاد فضاء التضمين (أي طول كل متجه).

* التقييس (L2-Normalization)

قبل حساب التشابه تخضع التضمينات الصورية والنصية لتقييس من النمط L2 بحيث يُصبح طول كل متجه يساوي الواحد. يتم ذلك وفق العلاقة:

* e هو المتجه الأصلي (embedding لصورة أو نص)
* *هو معيار L2*

مما يجعل الضرب النقطي بين المتجهين مكافئاً لحساب تشابه كوساين، حيث أن تشابه كوساين بين المتجهين a وb يحسب كالتالي:

بالتالي عندما نقوم بتقييس a وb يصبح

بذلك يكون عند تمثيل الصور والنصوص بمصفوفات من متجهات embeddings يجب تقييسها قبل حساب مصفوفة التشابه بينهم، حتى يكون الضرب النقطي فيها فعالاً كمؤشر على التشابه الدلالي.

* حساب مصفوفة التشابه (logits matrix)

يتم حساب مصفوفة الـ logits بحجم B×B، حيث كل عنصر فيها يُمثل درجة التشابه الخام بين صورة وتوصيف نصي وفق الصيغة:

حيث logit\_scale هو معامل قابل للتعلم يُضبط أثناء التدريب بهدف تحسين انتشار قيم التشابه واستقرار الحسابات عند تطبيق softmax.

* *تعديل الأوزان السالبة:*

*لزيادة حساسية الخسارة تجاه التشابهات غير المرغوبة بين الفئات* (Negative pairs)*، تتم إعادة وزن العناصر السلبية بناءً على درجة التشابه الدلالي بين العبارات النصية. يُحسب الوزن السلبي لكل زوج غير مطابق (صورة-نص من فئات مختلفة)، ويضاف لوغاريتم هذا الوزن إلى القيم الخام:*

بحيث يتم تقليل مساهمة الأزواج ذات التشابه العالي (عبارات عامة أو غير فريدة) في حساب الخسارة.

* تحويل القيم إلى توزيعات احتمالية (log-softmax)

تطبق دالة log-softmax على كل صف في مصفوفة الـ logits لحساب احتمالية كل توصيف نصي بالنسبة لصورة معيّنة، والعكس عبر تطبيق log-softmax على الأعمدة. بذلك نحصل على توزيع احتمالي يركّز على مطابقة كل صورة مع التوصيف الصحيح لها (والعكس صحيح للنصوص).

* حساب خسارة التباين النهائية

تُحتسب خسارة التباين عبر دمج الخسارتين: (صورة إلى نص) و(نص إلى صورة) بشكل متناظر، بحيث تعكس التوافق بين التضمينات الصورية والنصية ضمن الدفعة الواحدة.

### دالة الخسارة

تم تعديل تابع الخسارة وفق التجارب المختلفة المطبقة، فيما يلي نذكر كل التوابع واحداً تلو الأخر وأثناء ذكر التجارب يتم الإشارة إلى التابع المستخدم:

#### دالة خسارة التباين على مستوى الفئة Category contrastive loss

تم اعتماد دالة خسارة تباينية على مستوى الفئة لتدريب النموذج، وهي مبنية على المعادلتين (1) و(2) الواردتين في الدراسة المرجعية [Saha et al., 2023]- وهما:

تابع الخسارة بالنسبة للصور:

تابع الخسارة بالنسبة للنصوص:

حيث:

* + N حجم الدفعة (batch size)
  + درجة التشابه بين الصورة والوصف النصي محسوبة بواسطة النموذج
  + معامل درجة الحرارة، قابل للتعلم
  + مجموعة المؤشرات j التي تنتمي لنفس الفئة التي تنتمي إليها الصورة أي و من نفس الفئة
  + مجموعة المؤشرات i التي تنتمي لنفس الفئة التي تنتمي لنفس فئة النص

كل عبارة أو صورة تُقارن مع جميع العناصر الأخرى داخل الدفعة، لكن تُحسب الخسارة فقط بين الأزواج الإيجابية (نفس الفئة).

الخسارة الكلية:

تهدف هذه الخسارة إلى تعزيز التعلم من خلال مطابقة كل صورة مع جميع النصوص التي تنتمي إلى نفس الفئة (وليس فقط النص المطابق)، وكذلك مطابقة كل نص مع جميع الصور التي تنتمي إلى نفس الفئة. بذلك يتعلم النموذج تمثيلاً عاماً للفئة. مما يحسن الأداء في التصنيف بدون أمثلة (Zero-Shot Classification) عبر الإشراف الدلالي الذي يعتمد على الفئة بدلاً من الاعتماد على الأزواج الدقيقة فقط (instance-level).

من الناحية الحسابية، يتم إنشاء مصفوفة منطقية (Boolean Mask) تعرف باسم pos\_mask، حيث تكون قيمة العنصر G[i,j]=1 إذا كانت كل من الصورة i، والنص j ينتميان إلى نفس الفئة المرضية. تستخدم هذه المصفوفة لتحديد الأزواج الإيجابية داخل الدفعة تلقائياً بناءً على تصنيفات labels.

تستخدم logits في هذا السياق وهي مصفوفة التشابه الناتجة عن ضرب تضمينات الصور والنصوص ببعضها في الفضاء الموحد.

يتميّز هذا النوع من الإشراف بأنه يسمح للنموذج بتعلم تمثيلات أكثر تعميماً من خلال الاستفادة من جميع التوصيفات النصية الخاصة بالفئة، بدلاً من الاعتماد فقط على الزوج (صورة-نص) الفردي.

وتُعد هذه الخاصية مفيدة بشكل خاص في سياقات التصنيف الطبي الدقيق، حيث تكون الفروقات بين الفئات طفيفة والتوصيفات النصية متنوعة دلالياً.

#### دالة خسارة التباين المرجّحة بناءً على التشابه النصي Similarity-aware weighted infoNCE:

لاعتبارات طبية تقتضي أن بعض الأمراض يحصل وأن يحدث بين توصيفاتها تقاطعات معينة، وخصوصاً في صور الرنين المغناطيسي الدقيقة جداً، يمكن أن يتم توصيف مرضين مختلفين بعدة عبارات، يتقاطع منها عدد معين، تم اللجوء إلى تجارب أخرى باستعمال توابع خسارة تميز هذا الفرق، الفقرة التالية تتكلم عن تابع خسارة مغاير للسابق يستخدم في بعض التجارب.

تم استخدام دالة خسارة تباينية بديلة تأخذ بعين الاعتبار التشابه الدلالي بين العبارات النصية ضمن الدفعة التدريبية، بهدف تحسين قدرة النموذج على التعامل مع الأوصاف النصية الطبية المتشابهة وتقليل احتمالية الالتباس بين الأمراض المتقاربة لغوياً.

نأخذ أوزان العبارات (الموضح في الفقرة 1.2.3.5)، ونجري التعديلات على الأوزان قبل تطبيق دالة softmax:

1. السلبيات: يعتبر الزوج سلبي إذا كان مكون من عبارتين مختلفتين من مرضين مختلفين، لكن هاتين العبارتين متشابهتين لغوياً بدرجة كبيرة، يتم تقليل عقوبة النموذج على الخلط بينهما من خلال خفض وزن الزوج السلبي وفق المعادلة التالية:

حيث sim هو قيم الشبه المحسوب باستخدام نموذج Sentence-BERT، و معامل ضبط الحساسية، و قيمة صغيرة جداً لمنع الوزن من أن يكون صفري.

1. تعديل قيم logits: يتم إدخال الوزن بطريقة لوغاريتمية log-space على مصفوفة التشابه logits كالآتي:

هذه الخطوة تقلل من تأثير الأزواج السلبية شديدة التشابه، وتمنع النموذج من التعلُّق الزائد بصياغات نصية شبه متطابقة لا تحمل فرقاً دلالياً مهماً.

1. بعد هذا التعديل، تُحسب الخسارة باتجاهين:

الخسارة تحسب بالاتجاهين، كما الحالة السابقة، الأول L-img تعبر عن مدى تمييز الصورة للعبارات الصحيحة من بين كل العبارات، والثاني L-txt تعبر عن مدى تمييز العبارة للصورة الصحيحة من بين كل الصور.

ويؤخذ الخسارة النهائية على أنها متوسط الاتجاهين

تساعد هذه الطريقة المُعدَّلة لدالة الخسارة النموذج على التعلم بشكل أكثر استقراراً وتوازناً، حيث يصبح حساساً بدرجة كافية للاختلافات الدلالية الجوهرية بين الأمراض، بينما يتجنب الانجراف إلى التمييز بين أوصاف متشابهة جداً لا تعكس فرقاً طبياً واضحاً. وبذلك نحصل على تمثيلات تباينية موثوقة وأكثر دقة في سياق التصنيف الطبي بدون أمثلة.

#### دالة خسارة التباين الموزونة بالتفرد والتشابه (Uniqueness-aware weighted infoNCE):

دالة الخسارة المستخدمة في هذا النظام هي تطوير لدالة InfoNCE السابقة، ولكن مع تعديلات خاصة لمراعاة التشابه بين العبارات واختلاف مستوى تفرّد كل عبارة. تعامل العبارات كعناصر لكل صورة، شبيهة بما سبق مع تعديل بسيط.

نجعل كل ثنائي (صورة-عبارة) يمثل عنصراً مستقلاً لنحتسب تشابه كل عبارة مع جميع صور الدفعة يتم ذلك عن طريق ما يلي:

* كل صورة في الدفعة لديها K من العبارات الموجبة
* يُعاد تشكيل بيانات الدفعة بحيث يكون لدينا B×K عنصر.
* نحسب تضمينات الصور img-emb وتضمينات العبارات text-emb ونسويها إلى الشكل B, D)×(K

ثم يكون ما يلي:

1. الأوزان الإيجابية هي لأي عبارة ينتمي مرضها إلى نفس فئة الصورة.
2. يأخذ أرقام العبارات (txt\_ids) ويجلب من المصفوفة الجاهزة ‎S\_full‎ تشابه‑كوساين مسبق من sentence-BERT الحساب بين كل زوج عبارات في الدفعة.
3. إذا كانت عبارتان إيجابيتان أي لنفس المرض ومتشابهتين لغويّاً جداً نقلّل وزنهما؛ فالنموذج أصلاً يعرف هذا التشابه، ولا نريده أن يتعلّق بعبارات مكرَّرة، نقلل الوزن وفق المعادلة:

حيث معامل ضبط الحساسية، و قيمة دنيا صغيرة لتجنب الصفر.

1. السلبيات التي تشابهها اللغوي كبير، أو إحدى عباراتها غير فريدة تعطى خطأ أكبر، أي خطأ مع هذه السلبيات يُعاقب أكثر لأنّها ستسبب التباس.

حيث هما أوزان التفرد للعبارتين، حيث تفرد العبارة الأولى، و تفرد العبارة الثانية.

أما معاملات تحكم

1. قبل حساب الـ Softmax نضيف log(weight) لكل زوج، بحيث تتحكم هذه الأوزان في مقدار تأثير كل زوج في حساب الخسارة النهائية، بطريقة ديناميكية مستقرة.
2. حساب الخسارة

نطبق الدالة السابقة على الـ logits المعدلة، ثم نأخذ المتوسط على الأزواج الإيجابية فقط.

الخسارة تحسب بالاتجاهين، الأول L-img تعبر عن مدى تمييز الصورة للعبارات الصحيحة من بين كل العبارات، والثاني L-txt تعبر عن مدى تمييز العبارة للصورة الصحيحة من بين كل الصور.

نأخذ الخسارة النهائية على أنها متوسط الاتجاهين

فيكون باستخدام هذه الاستراتيجية، لا يتعلم النموذج فقط التوافق بين الصورة والعبارة المرتبطة بها، بل يصبح حساساً للاختلافات الدقيقة بين عبارات الأمراض المختلفة، ويتجنب الانجراف نحو الصيغ النصية الأكثر تكراراً أو عمومية. النتيجة النهائية هي تمثيل تبايني أكثر قوة وموثوقية، قادر على التمييز بين الحالات المرضية حتى عند وجود تداخل لغوي كبير بين أوصافها النصية.

### التدريب

كما ذكرنا سابقاً فإن SWIN هو قابل للتدريب بكامل طبقاته، أما BioClinicalBERT مجمّد بكامل طبقاته، عدا رأس الإسقاط، عند إجراء التدريب، ومع إعادة الإشارة إلى أن عدد الأمراض الكلي في مجموعة البيانات هو 40 مرض، تم تقسيم البيانات كالتالي:

1. تدريب على 24 مرض من هذه الأمراض بحوالي 850 إلى 900 صورة من كل صف.
2. تدريب على 8 أخرى فيها 3 صور فقط تجربة few-shot learning أي تعلم بأمثلة قليلة.
3. وهناك 8 أمراض بقيت مجموعة اختبار بهدف تجريب zero-shot learning تعلم بلا أي مثال.
4. تم تقسيم البيانات في صفوف التدريب إلى 80% للتدريب و20% للتحقق من كل صف.

تم تدريب النموذج باستخدام خوارزمية AdamW، استخدمت قيمة صغيرة لمعدل التعلم الخاص بـ SWIN قدرها 10-5، مع وزن تلاشي 10-4، بينما استخدم معدل أعلى لطبقة رأس الإسقاط النصي 10-3 بدون وزن تلاشي، بهدف تسريع تعلم الرأس مع الحفاظ على استقرار الطبقات الداخلية المجمدة.

كذلك تم استخدام تقنية EMA المتوسط الأسي المتحرك Exponential Moving Avarage لتحديث نسخة إضافية من النموذج تسمى ema-model، تُستخدم لتقييم الأداء على مجموعة التحقق، مما يساعد في تقليل تأثير التذبذب الناتج عن تحديثات الوزن الحادة خلال التدريب، ويزيد من ثبات الأداء بين الدفعات.

**الفصل السادس**

# الاختبارات والنتائج

يعرض هذا الفصل الاختبارات

## اول تجربة

1. التجربة الأولى:

Encoder SWIN تم تدريب جزء منه وجزء من BERT

1. التجربة الثانية:

Encoder SWIN تم تدريبه كاملا ثم نقل الاوزان مرة ثانية بهدف تدريب BERT كاملاً

بعد هذه السلسلة تم الاستقرار على استخدام ما يلي:

كنتيجة للمرحلة السابقة تم التوصل إلى أن أفضل خيار هو استخدام SWIN-base، كمرمز للصورة، مع تدريبه كاملاً، مع BioClinicalBERT مجمد عدا رأس الإسقاط.

1. تم العمل على عبارات بسيطة تقوم هذه العبارات على فقط عرض اسم المرض كما لو أنها {is Disease This}
2. تم العمل على عبارة واحدة تقوم هذه العبارة بتوصيف المرض
3. تم العمل على 10 توصيفات للمرض بحيث يتم اخذ عبارة واحدة لكل مرض من التوصيفات
4. تم العمل على قرن 3 عبارات من التوصيفات بالمرض واخذ انسب عبارة
5. تم العمل على اخذ عبارة واحدة من كل category مذكورة في الفقرة السابقة
6. تم العمل على توزين العبارات وفق تكرارها وقياس Loss جديد
7. تم العمل على اخذ عبارة واحدة من كل category مذكورة في الفقرة السابقة

أفضل نتيجة مرفقة باسم FinalOne

# الأدوات المستخدمة

يعرض هذا الفصل الأدوات المستخدمة لتنجيز المشروع

## PostgreSQL

هو نظام إدارة قواعد بيانات علائقية مفتوح المصدر، معروف بمتانته وقابليته للتوسع والالتزام بمعايير SQL. يدعم أنواع بيانات متقدمة واستعلامات معقدة، مما يجعله مناسباً للتعامل مع مجموعات البيانات الكبيرة والمعقدة. يُستخدم PostgreSQL على نطاق واسع في تطبيقات الويب خصوصاً تلك التي تتطلب تخزيناً واسترجاعاً موثوقاً للبيانات.

## React

هي مكتبة JavaScript مفتوحة المصدر، تُستخدم لبناء واجهات المستخدم، خاصة لتطبيقات الويب. تم تطويرها بواسطة شركة Facebook، وتستخدم على نطاق واسع لإنشاء واجهات مستخدم ديناميكية وتفاعلية. تتبع React البنية القائمة على المكونات، مما يساعد المطورين على بناء مكونات واجهة المستخدم التي يمكن إعادة استخدامها وتكوينها معاً لإنشاء واجهات مستخدم معقدة.

## Python

هي لغة برمجة عالية المستوى ومفتوحة المصدر، تُستخدم على نطاق واسع في تطوير البرمجيات، علم البيانات، الذكاء الاصطناعي، وتطبيقات الويب. تدعم Python نماذج برمجة متعددة مثل البرمجة الكائنية والوظيفية. بفضل مكتباتها الغنية وإطار عملها القوي، أصبحت Python خيارًا شائعًا لتطوير خوادم الويب، مثل FastAPI وDjango، كما تُستخدم بشكل أساسي في تدريب وتشغيل نماذج الذكاء الاصطناعي.

## FastApi

هو إطار عمل حديث ومفتوح المصدر لتطوير واجهات برمجة التطبيقات (APIs) باستخدام لغة Python. تميز بسرعته وكفاءته العالية، حيث يعتمد على المعايير الحديثة مثل OpenAPI وJSON Schema. يسمح FastAPI بكتابة كود نظيف وقابل للتوسعة مع دعم قوي للتوثيق التلقائي للواجهات. مما يجعله مناسبًا لتطبيقات الويب التي تتطلب أداءً عاليًا، مثل أنظمة الذكاء الاصطناعي.

## Json Web Token (JWT)

الرمز (JWT) هو غرض JSON يُستخدم لنقل المعلومات بين طرفين بشكل آمن عبر الويب. يتم استخدام JWT على نطاق واسع في النطبيقات الحديثة كآلية مصادقة عديمة الحالة. يعكس cookies رموز JWT يمكن استخدامها للمصادقة في تطبيقات الويب، وتطبيقات الجوال، وتطبيقات سطح المكتب (Desktop application).

## Git

هو نظام تحكم في الإصدارات (version control system) مفتوح المصدر، يستخدم لتتبع التغييرات في الرماز المصدري (source code) أثناء تطوير البرمجيات. حيث يسمح Git لمطورين متعددين بالتعاون في مشروع واحد من خلال إدارة التغييرات التي تم إجراؤها على الرماز المصدري، مما يجعل من السهل تنسيق العمل وتتبع التغيرات ضمن المشروع.

## GitHub

هي منصة تعمل ضمن الويب توفر أدوات للتحكم في الإصدارات والتعاون بين المطورين لتطوير البرامج، حيث أنها مبنية على نظام التحكم في الإصدارات Git. تسمح هذه المنصة بتخزين الكود البرمجي ومشاركته بين مجموعة من المطورين كما تقدم مجموعة من الأدوات للتعامل مع بيئات التطوير المستندة إلى السحابة (cloud) واستضافة مواقع الويب الثابتة.

**الفصل السابع**

# التنجيز البرمجي لنظام الويب

في هذا الفصل نقدّم شرحاً عن معماريّة النّظام وتنجيزه البرمجي إضافةً الى المكاتب والبيئات المستخدمة

## معماريّة المشروع Architecture

من المهم قبل البدء في بناء أي نظام برمجي تحديد منهجية واضحة للبناء، بهدف الوصول إلى رماز مصدري قابل للاختبار والتعديل عند الحاجة، سيتم في هذا المشروع اعتماد البنية النظيفة، كما سيتم الاستفادة من بعض الأنماط التصميمية التي تهدف لحل مشاكل التصميمية التي يمكن أن تواجهنا. بدايةً سنقوم بتعريف بعض المفاهيم التي سنستخدمها لاحقاً.

### البنية النظيفة

نمط تصميمي معماري، يهدف بشكل رئيسي إلى إنشاء نظام برمجي قابل للصيانة ومستقل عن تفاصيل التنفيذ. يعزز الاستقلالية بين مكونات النظام المختلفة، بالتالي يسمح لهذه المكونات بالتطور بشكل مستقل دون التأثير على النظام بأكمله، ومن خلال تطبيقه يتوفر لدينا رماز مصدري قابل للاختبار بالتالي يسهل إدارة وتوسيع النظام البرمجي المبني على أساس هذه البنية.

### خواص النظام المبني وفق البنية النظيفة

من أهم مبادئ البنية النظيفة أنها تخضع لمبدأ عكس التبعية الذي هو مبدأ أساسي جداً في توجيه تنظيم المكونات والوحدات داخل النظام البرمجي وكذلك كيفية تفاعل هذه المكونات مع بعضها، وهذه القاعدة تنص على أن التبعيات بين طبقات النظام المبني تشير دائماً إلى الداخل أي نحو لب النظام، والذي يعني أن مكونات النظام العليا لا تعتمد على المكونات في المستوى السفلي، بل العكس. مما يضمن أن التغييرات في المكونات السفلى لا تؤثر على المكونات العليا، وذلك يعزز من مرونة النظام، حيث أن التبعية هنا تنطلق من الطبقات الخارجية (والتي هي الأكثر ثباتاً وتجريداً) نحو الطبقات الداخلية.

ولتوضيح أكثر فإن النظام المبني وفق البنية النظيفة يتميز بعدة خواص أساسية، منها:

* الاستقلالية: يمكن تطوير كل مكون من مكونات النظام بشكل منفصل، مما يسهل العمل الجماعي ويقلل من التعقيدات، حيث يتمتع بالاستقلالية عن:
  + أطر العمل
  + تخزين البيانات
  + واجهات المستخدم
* قابلية الصيانة: يسهل تعديل النظام وإضافة ميزات جديدة دون الحاجة إلى إعادة بناء النظام بالكامل.
* قابلية الاختبار: يمكن اختبار كل مكون من مكونات النظام بشكل مستقل، مما يعزز من جودة البرمجيات.

### طبقات النظام المنجز

#### طبقة المجال Domain Layer

هي الطبقة الأهم في المشروع حيث تحتوي على القواعد الأساسية للعمل (business rules). تتمثل هذه الطبقة بطبقة الكيانات (entities) بما يتوافق مع البنية النظيفة. لا تعتمد هذه الطبقة على أي تبعيات (dependency) من طبقات أخرى باستثناء الأدوات المشتركة التي تتعلق فقط بمفاهيم العمل التي تساعد في التنفيذ. كما تحوي على جميع الواجهات interfaces التي سيتم تنجيزها في طبقات أخرى.

#### طبقة التطبيق Application Layer

طبقة تعرف بطبقة حالات الاستخدام (use case layer) في البنية النظيفة. هذه الطبقة لا ترتبط بأي تبعية (dependency) لأي طبقة أخرى سوى طبقة المجال. تكمن وظيفتها في تنفيذ منطق العمل المطلوب وتنظيم التفاعلات بين المجالات الفرعية، في هذه الطبقة تم فصل حالات الاستخدام إلى قسمين بهدف تحقيق Command Query Responsibility Segregation، سنتكلم عنه في فقرة لاحقة.

#### طبقة البنية التحتية Infrastructure Layer

تعد طبقة البنية التحتية جزءاً أساسياً من الإطار الهيكلي للتطبيق، حيث تتعامل مع الطبقات الخارجية التي تدعم البنية التحتية للتطبيق. تعتمد هذه الطبقة على طبقة التطبيق (dependency) لتنفيذ المهام التشغيلية للنظام، مثل تخزين البيانات في قواعد المعطيات، وتنجيز الواجهات المعرفة في طبقة المجال.

#### بنية العرض Presentation Layer

تندرج طبقة العرض ضمن الطبقة الخارجية للبنية التحتية التطبيقية. تعتمد هذه الطبقة على تبعية (dependency) لجميع الطبقات المذكورة سابقاَ، حيث تقوم باستقبال الطلبات من المستخدمين باستخدام بروتوكول HTTP، وتعد المشغل الأساسي للتطبيق، تتولى أيضاً التحقق من صلاحية الطلبات المرسلة إلى طبقة التطبيق (Application)، ثم تعيد الإجابات المستلمة من هذه الطبقة إلى المستخدم.

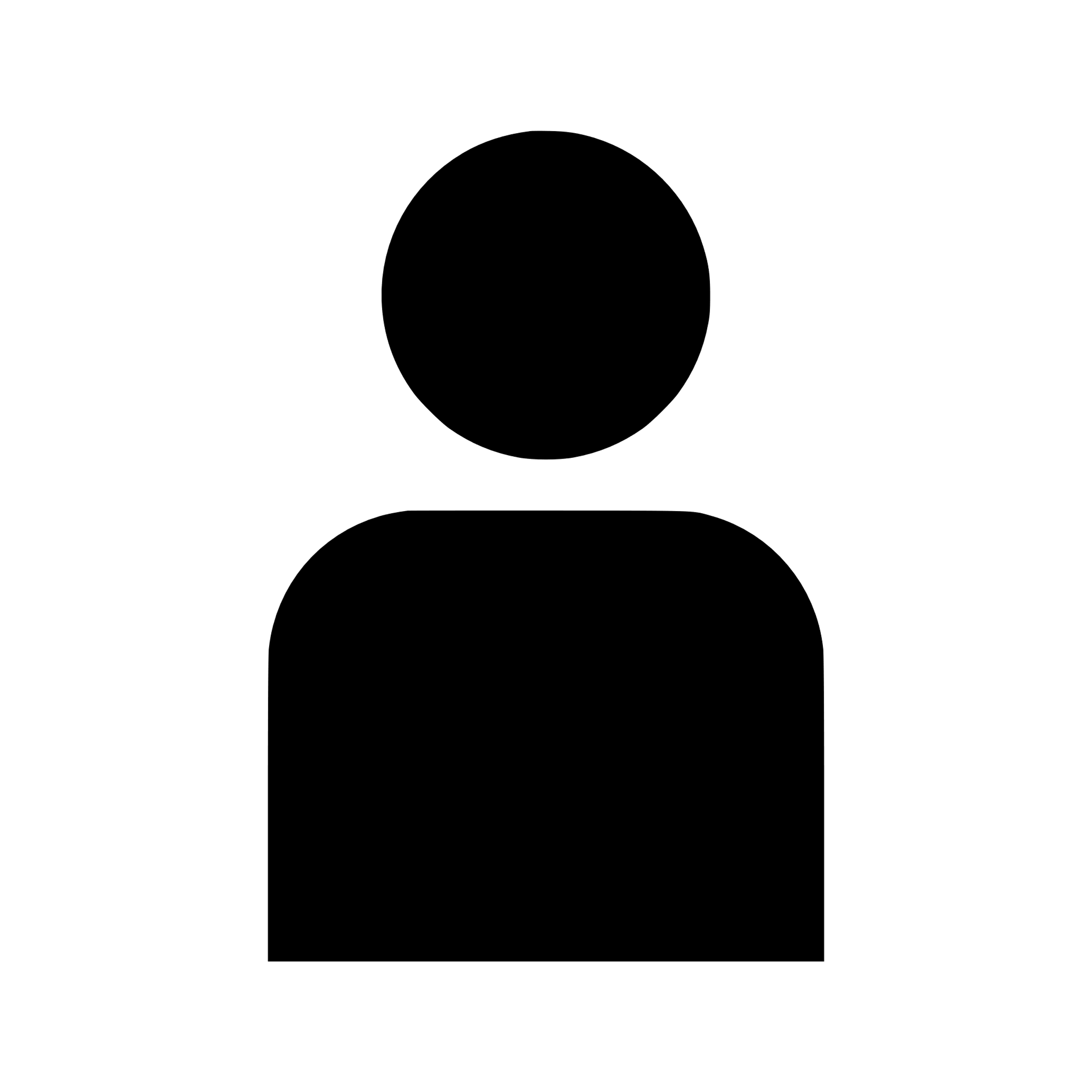
Application

Front-end

Back-end

UIs

Domain



End User

HTTP Request

HTTP Response

AI Models

Presentation

Infrastucture

### أهم الأنماط التصميمة المستخدمة في بناء النظام

#### Repository

ن

#### Unit of work

ن

#### Command query

ن

#### Singelton

تأتي أهمية هذا النمط من كونه يقوم باستدعاء الاوزان مرة وحدة عند بداية التطبيق فيوفر وقت استدعائهم كل مرة عند الطلب.

## واجهات التّطبيق

تم تنفيذ واجهات التطبيق باستخدام react.

1. Attention\_mask قائمة بوليانية من 0 أو 1 تستخدم ليعلم النموذج أي الرموز يجب أن تؤخذ وأيها يتجاهله بسبب ال padding أي الإضافة بهدف توحيد طول العبارات [↑](#footnote-ref-1)