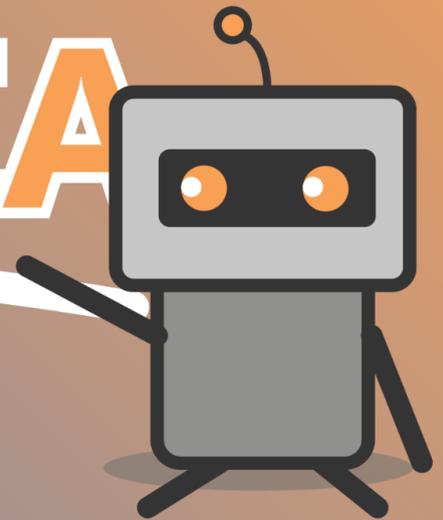




Crea IA



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : CREA'IA

> PRÉSENTATION GÉNÉRALE :

- *Idée et objectifs*
- *Origines et intérêts du projet*
- (...)

Créa'IA est une application permettant de concevoir des modèles d'intelligence artificielle, des plus simples aux plus complexes, regroupant en une unique plateforme toutes les étapes nécessaires à l'élaboration d'un modèle.

L'un des objectifs principal de l'application est de la rendre la plus accessible possible, pour y créer sa "première intelligence artificielle". Pour respecter cet objectif, nous avons ajouté une dimension éducative au projet, en incluant des tutoriels imagés, et en axant nos efforts sur la mise en place d'une interface ergonomique et intuitive, qui guide les utilisateurs vers l'élaboration d'un modèle d'intelligence artificielle complet, entraîné et exécutable. De plus, pour assembler une plateforme à objectif éducatif, nous avons créé l'ensemble des ressources graphiques, des icônes aux images explicatives, pour expliquer de manière imagée les concepts sur lesquels repose l'intelligence artificielle.

Le projet est né lorsque notre professeur de NSI avait suggéré d'entraîner une intelligence artificielle comme projet durant l'année, mais en faisant des recherches pour comprendre le fonctionnement et la mise en place d'une IA, nous avons rapidement réalisé la difficulté de la tâche. En effet, il n'existe que très peu de logiciels libres et gratuits qui permettent de réaliser cela, le plus souvent ces logiciels sont compliqués à prendre en main et demandent des bases très solides en informatique et en mathématiques, ainsi qu'une connaissance renforcée en réseaux neuronaux artificiels et en ingénierie des données.

Nous interagissons de plus en plus avec les intelligences artificielles au quotidien, et c'est pourquoi il est important de comprendre leur fonctionnement et de rendre cette technologie accessible à tous, sans prérequis ou formation.

> ORGANISATION DU TRAVAIL :

- *Présentation de l'équipe (prénom de chaque membre et rôle dans le projet)*
- *Répartition des tâches*
- *Organisation du travail (répartition par petits groupes, fréquence de réunions, travail en dehors de l'établissement scolaire, outils/logiciels utilisés pour la communication et le partage du code, etc.)*

Notre équipe est constituée de deux membres :

- **Hugo**, qui s'est occupé principalement du développement de l'interface, des ressources graphiques et de l'implémentation des algorithmes d'apprentissage et de prédiction de l'intelligence artificielle.
- **Clément**, qui s'est chargé de la mise en place et de l'élaboration du serveur et de la création du modèle de démonstration contenant près de 800 images étiquetées.

Nous avons organisé notre travail principalement à l'aide de Discord, où nous pouvions communiquer en temps réel. De plus, une grande partie du travail a été fait pendant les vacances scolaires, ce qui nous permettait de faire quotidiennement le point sur l'avancement du projet.

Pour le partage du code, nous avons utilisé git ainsi que GitHub pour que chaque membre puisse avoir accès au code et le publier de manière organisée, ce qui permet de faciliter le flux de travail.

LES ÉTAPES DU PROJET :

- *Présenter les différentes étapes du projet (de l'idée jusqu'à la finalisation du projet)*

Premièrement, l'idée est née de la frustration de ne pas pouvoir trouver de logiciel libre et gratuit permettant d'avoir accès aux fondations d'une technologie de plus en plus répandue. Nous avons alors proposé l'idée à notre professeur de NSI, Mr. MEDEN qui nous a guidé sur certains concepts mathématiques qui permettaient le fonctionnement des intelligences artificielles.

Dans un second temps, nous nous sommes documentés sur le fonctionnement en profondeur des réseaux neuronaux et sur les fondations informatiques et mathématiques du fonctionnement d'une intelligence artificielle. Nous avons donc regardé de nombreuses vidéos, tutoriels, lus des articles et documentations pour maîtriser le sujet, et permettre l'élaboration de l'application.

Puis, nous avons décidé des technologies à utiliser : langages, bibliothèques et environnements. En pesant le pour et le contre, nous avons déterminé qu'utiliser Python avec PyTorch, le tout reposant sur un serveur Flask était le choix le plus approprié.

Ensuite, nous avons conçu les premiers prototypes, d'abord graphiques puis fonctionnels de l'application, à chaque avancement, nous nous sommes concertés en essayant d'obtenir des avis extérieurs pour affiner notre travail.

Et enfin, une fois l'application terminée, nous avons réservé une petite période de temps pour faire des essais, déboguer l'application, faire un grand nombre d'entraînements et calculer l'efficacité des modèles pour proposer des préconfigurations d'hyper paramètres les plus adaptés possible, tout cela pour s'assurer de proposer une application complète, fonctionnelle et accessible.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

- *Avancement du projet (ce qui est terminé, en cours de réalisation, reste à faire)*
- *Approches mises en œuvre pour vérifier l'absence de bugs et s'assurer de la facilité d'utilisation du projet*
- *Difficultés rencontrées et solutions apportées*

Dans sa forme actuelle, le projet est entièrement opérationnel pour l'entraînement de modèles d'IA relevant de la classification d'images. Malheureusement, nous aurions souhaité ajouter deux autres types d'intelligences artificielles : la reconnaissance d'objets ainsi que la classification de texte, mais nous avons choisi de renforcer l'application en offrant une expérience complète sur un type unique de modèle.

De plus, la catégorie "Entraînement" peut être hostile aux personnes non initiées, nous avons comme projet d'expliquer en détails et simplement chaque hyperparamètre pour que tous les utilisateurs puissent les ajuster pour créer des modèles plus performants et personnalisés. Pour pallier ce problème, nous avons proposé des pré-configurations qui détaillent simplement l'objectif de chaque prédisposition d'hyperparamètres, mais nous pensons que cela n'est pas suffisant.

Au-delà de ça, pour nous assurer que notre application était simple à prendre en main et intuitive, nous avons demandé des retours à des personnes extérieures au projet, et le plus souvent étrangères à l'informatique. Nous avons donc compilé les suggestions et critiques du projet pour essayer de l'affiner le plus possible : éclaircir les zones d'ombre ou rendre la navigation davantage ergonomique, le tout pour assurer une plateforme accessible à tous et à caractère éducatif.

Pour vérifier l'absence de bugs, nous avons chacun lu et vérifié le code dans son entièreté, de plus, nous avons suivi le processus d'installation sur différentes machines et avons fait tourner le code sur des ordinateurs puissants mais aussi des ordinateurs aux ressources limitées. Dès qu'un bug était constaté, il était rapidement résolu, en s'assurant qu'une erreur similaire n'a pas été reproduite autre part dans le code. Ce protocole permet d'essayer de minimiser au maximum la présence de bugs au sein de la plateforme.

L'une des difficultés principales que nous avons rencontrée lors de la mise en place du projet est l'aspect mathématique des intelligences artificielles, qui s'avéraient souvent très avancées en abordant des notions que nous n'avons jamais rencontrées en cours. Mais notre professeur de NSI a pu nous aider et nous expliquer toute la base mathématique nécessaire à la prise en main de cette technologie.

> OUVERTURE :

- *Idées d'améliorations (nouvelles fonctionnalités)*
- *Stratégie de diffusion pour toucher un large public (faites preuve d'originalité !)*
- *Analyse critique du résultat (si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?)*

Dans son état actuel, l'application ne couvre qu'un type (bien qu'il soit le plus répandu) de modèles d'IA, il serait intéressant d'étendre cela non seulement à la reconnaissance d'objets et à la classification de texte, mais également à des types qui ont émergé plus récemment : complétion de texte (tel que ChatGPT) et création d'images, qui restent encore majoritairement monopolisés par des modèles fermés au public et parfois payant.

De plus, nous avons masqué une partie essentielle lors de la conception d'une intelligence artificielle : la structuration du réseau neuronal. Nous utilisons des réseaux pré-entraînés (resnet18) pour faciliter l'apprentissage, mais l'ajout d'une option permettant de construire son propre réseau, couche par couche, pourrait être un réel atout et augmenterait considérablement les possibilités qu'offre l'application.

Nous pensons que Créa'IA est une application qui à le potentiel d'être utilisée par le grand public pour se préparer à l'arrivée de plus en plus importante des intelligences artificielles dans notre quotidien. Cette application pourrait être utilisée dans des contextes éducatifs, mais également par des développeurs qui souhaiteraient intégrer des IA dans leurs projets, pour accomplir des tâches non réalisables avec des algorithmes classiques.

En somme, si nous devions le refaire, nous aurions mieux organisé notre temps, pour pouvoir proposer toutes les fonctionnalités qui étaient initialement prévues, et de faire davantage appel à des points de vues extérieurs pour vérifier et contrôler la facilité d'utilisation de l'application, qui est un enjeu majeur du projet.

DOCUMENTATION

- *Spécifications fonctionnelles (guide d'utilisation, déroulé des étapes d'exécution, description des fonctionnalités et des paramètres)*
- *Spécifications techniques (architecture, langages et bibliothèques utilisés, matériel, choix techniques, format de stockage des données, etc)*
- *Illustrations, captures d'écran, etc*

I Readme

Installation

Afin de pouvoir utiliser l'application, il est nécessaire d'installer les dépendances situées dans le fichier `requirements.txt` :

```
pip install -r requirements.txt
```

Vous devez aussi avoir préalablement installé SQLite3 sur votre machine, et de vous assurer que votre version de Python est inférieure ou égale à 3.11.0 et supérieure à 3.9 (>3.11 n'est pas pris en charge par PyTorch).

Utilisation

Le code source de l'application se trouve dans le dossier `sources/`. Nous avons également mis à disposition dans le dossier `sources/chiens_chats/` un jeu de données de +800 images de chats et de chiens que vous pouvez utiliser pour tester l'application. Toutes les images sont libres de droits et proviennent de Wikimedia Commons.

Pour lancer l'application, vous devez vous rendre dans le dossier `sources/` et exécuter la commande :

```
flask run
```

Une fois l'application lancée, vous pouvez vous rendre à l'adresse 127.0.0.1:5000 pour y accéder.

Sources externes

Voici une liste de l'ensemble des sources externes que nous avons utilisées pour réaliser ce projet :

- Flask, utilisé pour le serveur web de l'application.
- PyTorch, utilisé pour l'entraînement des modèles d'IA et pour la prédiction.
- resnet18 qui est un modèle de réseau de neurones pré-entraîné pour la classification d'images.
- Toutes les ressources graphiques ont été réalisées dans le cadre de ce projet.
- Les images de chats et de chiens proviennent de Wikimedia Commons et sont libres de droits.

II Fonctionnalités

Vos modèles :

**modèle1**
Classification d'images

**test**
Classification d'images

+ Créer un nouveau modèle

L'application vous permet de gérer et de construire plusieurs modèles simultanément.

Vous pouvez importer et supprimer des images pour constituer le jeu de données utilisé par votre modèle.

806 éléments Importer des éléments

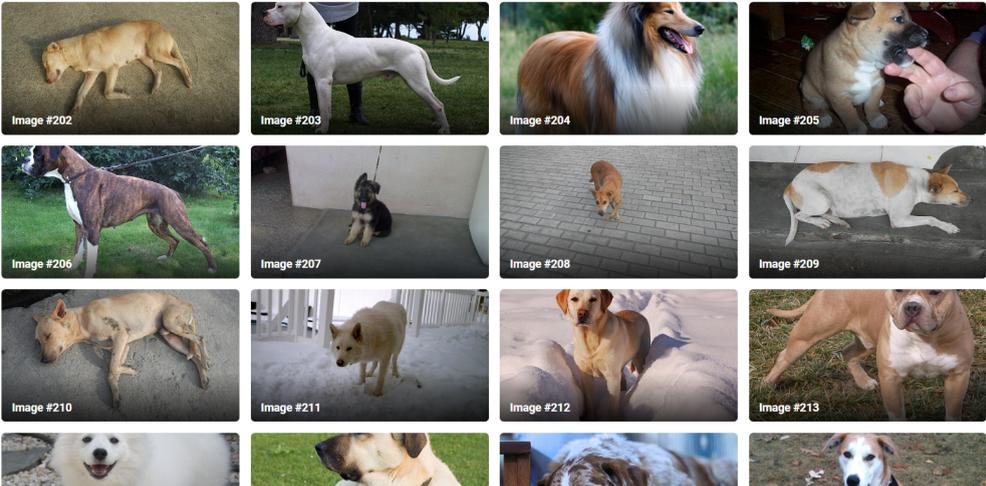


Image #202 Image #203 Image #204 Image #205
Image #206 Image #207 Image #208 Image #209
Image #210 Image #211 Image #212 Image #213



Image #1816

Choisissez une ou plusieurs étiquettes qui correspondent à cette image.

✓

L'étiqueteur vous permet d'étiqueter l'ensemble de votre jeu de données directement au sein de l'application.

Entraînez votre modèle à l'aide de pré-configurations ou des nombreux hyperparamètres avancés.

Vous pouvez aussi estimer la qualité de votre jeu d'entraînement avec une note donnée de F- à S+.

Préconfigurations	Hyperparamètres	Qualité
<p>Standard Une configuration d'hyperparamètres standard pour un entraînement équilibré du modèle.</p> <p>Rapide Une configuration d'hyperparamètres pour un entraînement rapide, avec un compromis sur la précision.</p> <p>Léger Une configuration d'hyperparamètres adaptée à des ordinateurs peu puissants, permettant un entraînement plus rapide avec une précision acceptable.</p> <p>Élaboré Une configuration d'hyperparamètres améliorés pour un modèle plus complexe, avec une meilleure précision mais nécessitant des ressources plus importantes.</p> <p>Profond Une configuration d'hyperparamètres adaptée à l'entraînement de réseaux de neurones profonds, nécessitant des ressources importantes.</p>	<p>Nombre d'epochs <input type="text" value="1"/></p> <p>Taille du batch <input type="text" value="1"/></p> <p>Taux d'apprentissage <input type="text" value="0,0001"/></p> <p>Dimension des images <input type="text" value="224"/></p> <p>Momentum <input type="text" value="0,9"/></p> <p>Dégradation des pondérations <input type="text" value="0,0001"/></p>	<p>Votre jeu d'entraînement contient un total de 806 éléments étiquetés.</p> <p style="text-align: center; font-size: 2em;">S</p> <p>Vous pouvez produire un modèle renforcé et très fiable avec votre jeu de données !</p> <p style="text-align: right;">Démarrer l'entraînement</p>

Epoch 1/10

Temps restant : 00h 03m 35s

282 / 806

Précision

66.67%

Obtenez des statistiques sur l'entraînement de votre modèle en temps réel.

De plus, l'application est compatible avec CUDA pour accélérer grandement le processus d'entraînement, si votre carte graphique est compatible.

Modèle entraîné

Gérez et supprimez les modèles que vous avez entraînés.

Choisissez le modèle entraîné sur lequel vous souhaitez faire des prédictions.

modele1682708184.pth

Entraîné il y a 51 minutes

Choisir

Supprimer

modele1682706580.pth

Entraîné il y a 1 heure

Choisir

Supprimer

modele1682696920.pth

Entraîné il y a 3 heures

Choisir

Supprimer

modele1682687107.pth

Entraîné il y a 6 heures

Choisir

Supprimer

modele1682615377.pth

Entraîné le 27/04/2023 19:09:37

Choisir

Supprimer

modele1682610465.pth

Entraîné le 27/04/2023 17:47:45

Choisir

Supprimer



Résultats

Chat 99%

Chien 1%

Essayez votre modèle en chargeant des images personnalisées sur lesquelles demander une prédiction !

III Technologies utilisées

Flask est le cœur du projet, c'est sur ce micro-framework que repose l'entièreté de l'application. Nous avons choisi Flask car il s'agit d'un outil très complet et puissant, qui convient parfaitement au projet.

Nous avons essayé de rendre la navigation la plus fluide possible, en minimisant les interruptions dues à des rafraichissements de page. Pour cela, nous avons utilisé **l'API Fetch** qui permet de récupérer des informations dynamiquement depuis le serveur, sans passer par une réactualisation de la page. Pour cela, nous avons aussi mis en place une API et de nombreux endpoints dans le routage de l'application.

Pour ce qui est de l'entraînement des intelligences artificielles, nous avons choisi **PyTorch**, un framework open source pour le machine learning très populaire et performant, qui offre une grande flexibilité pour la construction et l'entraînement de modèles et une documentation très complète.

Nous pouvons également mentionner le logiciel **Adobe Illustrator**, avec lequel nous avons pu concevoir l'ensemble des ressources graphiques utilisées dans l'application.