

Rapport de Stage Cycle: ING2

Analyse des traces d'une activité de résolution de problèmes

Younes Jallouf

Août 29, 2020



La date de la soutenance: 10 Septembre, 2020

Non confidentiel: , Confidentiel IGN , Confidentiel Industrie

Commanditaire :

Thierry VIEVILLE,

Laboratoire d'Innovation et Numérique pour l'Éducation (UCA, laboratoire LINE)
190 609 313 00019, 8542Z / Enseignement supérieur, INSPE de Nice, 21 avenue Liégeard

Encadrement du stage :

Victor Coindet, ENSG, enseignant référent
Thierry Vieville, LINE, maître de stage

Responsable pédagogique du cycle :

Jean-François Hangouët, IGN/ENSG/DIAS

Stage:

Stage du 15/06/2020 au 01/09/2020

Diffusion Web : Internet , Intranet ENSG

Nombre de pages : 32 dont 09 d'annexes

Système: L^AT_EX

1 Remerciements

Au terme de ce travail, je tiens à exprimer ma profonde gratitude et mes chaleureux remerciements à mon encadrant externe Mr. Thierry VIEVILLE pour sa collaboration, sa disponibilité et ses précieux conseils.

Je tiens également à exprimer mes sincères remerciements et ma forte reconnaissance à Mr. Victor COINDET, mon encadrant interne, pour son soutien, sa bienveillance et sa servabilité.

Je remercie tout les élèves qui ont contribué à l'activité Creacube en l'occurrence Elie ALBAN et Hugo TARDY, pour leurs efforts.

Finalement, je tiens à remercier le corps professoral de l'école nationale des sciences géographiques pour leur engagement à améliorer et faire évoluer la filière, ainsi que toute personne ayant aidé de près ou de loin à réaliser ce projet.

2 Résumé

Ce travail de stage a comme ambition l'étude et l'analyse d'une activité de résolution de problèmes à travers une démarche créative. Cette activité est nommée CreaCube et consiste à pousser les intervenants à déchiffrer le jeu et comment le réussir en les poussant à penser, réagir et finalement mettre en oeuvre la solution proposée. On intervient tout au long de cette procédure en notant les différents événements dans l'ordre et en prenant en considération la notion temporelle. Ces événements notés feront après l'objet d'une étude qui modélise la démarche de résolution ainsi que les éléments qui ont contribué au résultat final. On vise donc à améliorer le rythme de ces tâches en considérant un système qui remplace l'enregistrement manuel direct ou après l'activité par une analyse automatique/semi-automatique. On fait donc recours à l'apprentissage automatique en fournissant au script une vidéo de l'activité et elle retourne un enregistrement des événements ainsi que le temps d'incidence.

En fin de compte on fournit au script un ensemble d'événements stochastiques ce qui signifie une sortie non vraiment utile ou utilisable. Pour cela on a classifié par un code l'ensemble des données réalisées manuellement afin de s'en servir pour entraîner notre modèle pour qu'il soit capable de reconnaître les configurations dans chaque vidéo. On est donc entrain de mettre en place une approche de prédiction avec la probabilité enregistrée comme l'élément qui nous intéresse le plus pour valoriser notre performance.

En effet, l'entraînement récurrent de notre réseau de neurones a montré que cette approche est en fait prometteuse malgré le peu de données dont on dispose, par contre le code en lui même n'est pas assez facile à manipuler pour un usage fréquent. Pour cela, on a pensé à mettre en place un service web accessible sur une machine virtuelle à l'INRIA localement ou par internet et qui retourne un fichier json rempli directement si le nom de la vidéo est déjà reconnu, sinon il fait le traitement et retourne le fichier json avec les probabilités en question. Outre cela, un panneau de fonctionnalités partielles a été mis en oeuvre telles le chargement d'un modèle entraîné pour un nouveau passage ou même l'interception d'un entraînement avec des points de contrôle et de vérification. En plus, une interface d'utilisateur graphique a été mise en place pour une meilleure visualisation et une meilleure rectification des erreurs et des fautes potentielles par le script.

Key words: CreaCube, démarche créative, automatique/semi-automatique, apprentissage automatique, prédiction, probabilité, performance, entraînement, modèle, service web, machine virtuelle, json, interface d'utilisateur.

3 Abstract

The present work offers a study and an analysis of an activity aiming the resolution of a problem through creative solutions. This activity goes by the name of CreaCube and is based on the process of deciphering the aim of the game by inciting the participant to develop and design and then create an external rendering of the proposed solution. Throughout these tasks, it is up to us to intervene via logging the occurring events at first, and then inspecting them afterwards in order to modelize the thinking pattern and the different parts contributing to the final results.

Our principal aim is thus to enhance the performance and the pace of these missions by piloting a scheme where we replace the manual logging and analysis by an automated/semi-automated approach. This idea implies recourse to machine learning solutions where we feed the script videos of the activity and it returns the log of every event recorded along their name and timing. Clearly, we will be feeding the script some stochastic events, which means a random questionable output barely useful, this is where the database of the manually made results comes in handy and will serve as a base to train our model to recognize the patterns in every video. This being said, we are conducting an approach of prediction where the probability of each occurrence is the currency and the token of the performance.

The results of the recurring training of our machine learning model showed promising results of this approach despite the low disposition of data, however the script on itself is hardly accessible for daily use. And here interferes the idea of building a web service hosted on a virtual machine and capable of processing tasks locally and remotely. This service was built to either give the wanted results, which in our case is a file with the events and their probability, or to process a video if its json matching name is not existing on the server. Along this, the work above includes some partial functionalities like loading a trained model to pass through it again or to save training results with checkpoints if the training phase is interrupted for a given reason. We made the processing phase graphically possible by a UI(user interface) to enable better visualization and the ability to rectify mistakes or errors that may occur while predicting.

Mots clés: CreaCube, creative solution, automated/semi-automated, machine learning, prediction, probability, performance, training, model, web service, virtual machine, json, user interface.

Sommaire

1 Remerciements	3
2 Résumé	4
3 Abstract	5
4 Introduction	1
5 Contexte du projet et objectif	2
5.1 Contexte	2
5.2 Objectif	3
5.3 État de l'art	4
5.3.1 Apprentissage automatique et Sciences de l'éducation	4
5.3.2 En neurosciences computationnelles	5
6 Présentation du problème	6
7 Modélisation de la solution	7
7.1 Situation de la solution	7
7.2 Modélisation	8
7.2.1 Extraction des images	8
7.2.2 Apprentissage et prédition	9
7.2.3 Web Service	12
8 Mise en oeuvre	13
8.1 Management du projet	13
8.2 Préparation des données d'entraînement	14
8.3 Entrainement	15
8.4 Prediction et output	17
8.5 Web Service et déploiement	18
8.6 Test du service déployé	20
8.7 <i>Long Short-Term Memory</i>	22
9 Conclusion	23
10 Bibliographie	24
A Annexe : Interface Créacube	27
B Annexe : structure du réseau LSTM	28
C Annexe : brain-computational models	30
D Annexe : Pooling layer	31
E Annexe : Postman	32

4 Introduction

A défaut de piloter des études sur l'apprentissage des machines qui évoluent fascinamment, certains chercheurs optent pour assembler ce domaine aux études appliquées à l'apprentissage humain. Une démarche inédite qui lie les Neurosciences Computationnelles(Neurosciences Computationnelles) et les Science cognitive et qui est potentiellellement en mesure de répondre pratiquement à des questions de haut niveau, telle la relation entre l'Affordance et la résolution des problèmes comme la traite la psychologie cognitive, ou même l'affluence de certains traits personnels dans les tactiques de résolution.

Ce stage s'inscrit dans cette perspective et vise l'application du machine learning dans l'extraction des motifs dans les learning analytics de l'expérience, dite creacube, d'initiation à la pensée informatique par la manipulation d'objets tangibles. Le stage est aussi en prolongement d'un projet de recherche mené par une équipe trinome contenant deux collègues de classe Hugo Tardy et Elie Alban que je tiens à remercier pour leurs efforts. Je remercie également et particulièrement mon tuteur Thierry Viéville pour cette opportunité ainsi que Margarida Romero, Laura Cassone et Christine Dupard et toute l'équipe AEx avec qui j'ai travaillé tout au long de ce stage.



Figure 1: Un apprenant entrain de lire les règles de l'activité CreaCube.

Référence: CreaCube-PlayfulRobotics [1]

¹CreaCube: GALA2018-CreaCube-PlayfulRobotics-R17, Margarida Romero

5 Contexte du projet et objectif

5.1 Contexte

Le stage fait partie d'une étude globale qui vise, dans le cadre d'un panel de tâches d'initiation à la pensée informatique par la manipulation d'objets tangibles et souvent robotique, le développement d'un modèle de l'apprenant et de la tâche qu'il résout aussi.

Une telle modélisation relève un défi pour les sciences cognitives et les sciences de l'éducation qui doivent prendre en compte une multitude de propriétés à savoir le profil de l'apprenant, sa manipulation des objets, ses émotions et d'autres paramètres. Parmi les activités proposées dans ce cadre, on trouve un jeu de cubes que le Laboratoire d'Innovation et Numérique pour l'Éducation (LINE) a suggéré. Cette activité consiste en un jeu de 4 cubes robotiques, chacun dispose d'une fonctionnalité spécifique (Des roues, une batterie ...). L'apprenant en question doit réaliser un assemblage qui roule d'un point de départ vers un point d'arrivée sans son intervention chose qui l'incite à découvrir les combinaisons possibles.

Les différents événements relevés sont présentés dans l'interface Cracube, visible en Annexe A.

La plupart des tâches concernant les données de cette activité requièrent un travail manuel que ça soit au niveau de l'acquisition ou au niveau de la manipulation et de la validation. Le stage donc s'inscrit dans l'idée de semi-automatiser, voir automatiser la mesure des traces d'apprentissage grâce à des algorithmes d'apprentissage automatique suite à l'étude de faisabilité faite durant le projet de recherche à l'ENSG.



Figure 2: Manipulation des cubes par un apprenant

5.2 Objectif

L'objectif du stage est de mettre en place un ensemble d'algorithmes de machine learning déployés autant qu'un Web Service, et permettant l'extraction des figures et combinaisons réalisées par un apprenant avec une interface graphique de validation de ces configurations en temps réel. Il est donc proposé par la suite d'extraire des relations temporelles et causales entre les différents éléments observés par le script.

Ces algorithmes feront objet d'une évaluation afin d'estimer leur précision par rapport à la réalisation manuelle des tâches. Celà permettra de savoir dans quelle mesure un script d'apprentissage automatique peut engendrer ou pas une erreur d'interprétation.

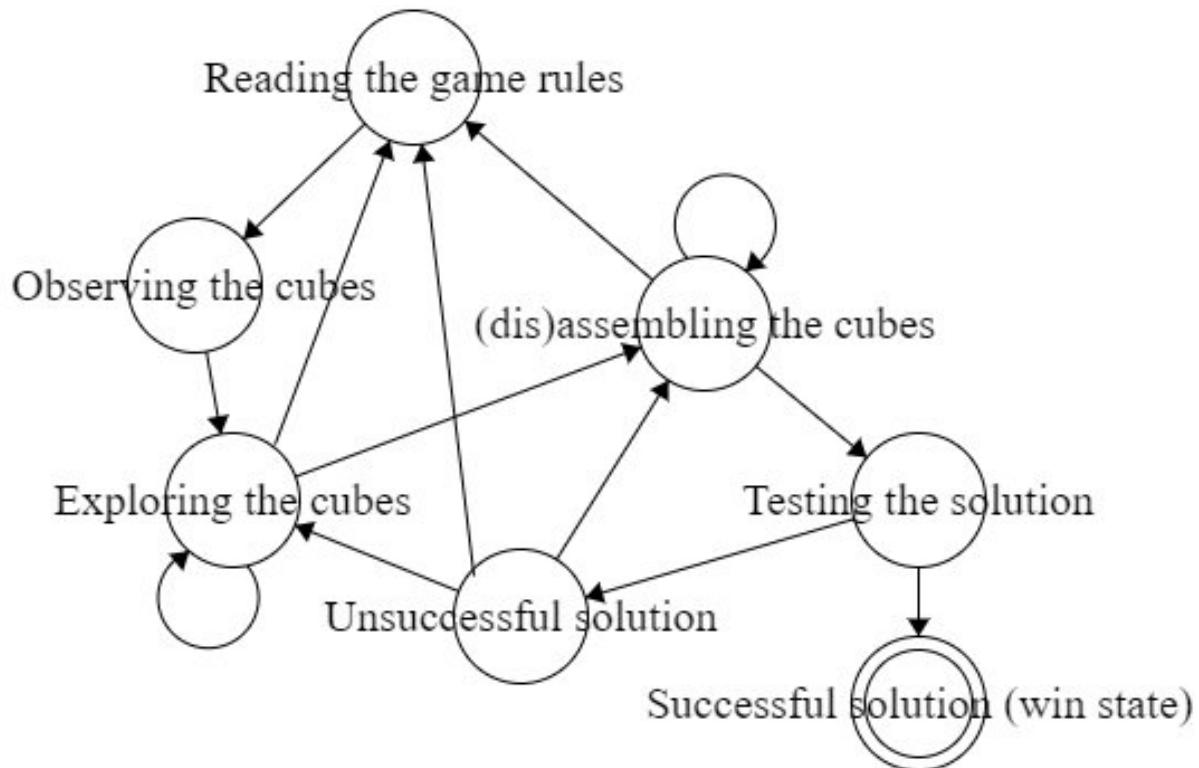


Figure 3: Démarche de résolution de l'activité CreaCube.

Référence: CreaCube-PlayfulRobotics [1]

5.3 État de l'art

5.3.1 Apprentissage automatique et Sciences de l'éducation

D'après l'article *Innovating Pedagogy 2020*². L'intelligence artificielle s'est inscrite dans le domaine de l'éducation sur plusieurs dimensions et des systèmes d'apprentissage sont déjà mis en place. Avec ce déploiement rapide de ces outils, il est devenu indispensable d'introduire les plus jeunes à la pensée informatique.

Le rapport confronte quatre grands aspects lors de l'étude des changements concrets apportés par l'intelligence artificielle(Artificial intelligence) au sein du domaine de l'éducation à savoir, des systèmes d'enseignement intelligents, l'évaluation automatique des examens écrits et des agents conversationnel. Le premier aspect est les perspectives pothumanistes, c'est à dire la mise en place d'un ensemble de barrières entre les notions d'humains et de non-humains à travers un pannel de pratique afin de remettre en question les différentes vision possibles de ces catégories [2]³. Les autres aspects sont, respectivement, l'apprentissage à travers l'Open Data, l'apprentissage multisensoriel et finalement la contribution des algorithmes d'intelligence artificielle dans l'amélioration des ces aspects dans le cadre de l'éducation.

Cognitive computational neuroscience [3]⁴

“What I cannot create, I do not understand” tel était le principe du physicien Richard Feynman et ainsi s'est développé la neuroscience. Des models alors ont été crées afin de comprendre le principe de cognition dans le cerveau et ont été décomposés sur des taches attribuées à des composantes computationnelles. Cette décomposition a permis l'évaluation des performance de chaque composante même si l'ensemble demeure toujours en-deçà de l'intelligence humaine. Ces models ont varié entre des models box-and-arrow, des models psycho-physiques vers des models brain-computational models (brain-computational models), voir Annexe C. Ces derniers ont fait le pont entre la théorie et l'expérience par le biais des calculs mathématiques et ont permis l'explication de certaines fonctions cognitives telle la reconnaissance visuelle des objets. Cette avancée a construit la base des couches de convolution et les couches de Pooling Layer(sous-échantillonage) dans les réseaux de neurones. Avec ces couches, les models cognitifs ont pu être améliorés en terme de performance on les appliquant sur des jeux d'entraînement différents et hétérogènes tout en permettant une avancée révolutionnaire au niveau de la reconnaissance et la classification des images. Et comme insite l'article [4] sur les réseaux neuronaux, ces réseaux ont pour but la reproduction d'une région du cerveau tel le cortex visuel avec des flux permettant le décodage de la position, la forme, la taille et la catégorie des objets figurants dans une image. Cependant, le travail mené durant le stage vise non seulement un résultat généré par la prise en compte des différentes structures cérébrales, mais également le corps et l'environnement extérieur comme facteurs contribuant à l'enchaînement des actions et idées lors de la résolution d'un problème.

²Innovating Pedagogy 2020: Open University Innovation Report 8. Milton Keynes: The Open University.

³Posthumanist Performativity: Toward an Understanding of How Matter Comes to Matter

⁴Nikolaus Kriegeskorte Pamela K. Douglas. "Cognitive computational neuroscience".

<https://drive.google.com/file/d/1UIt5sMHNX3pJM3RBEdN15wqcYPYN2gGj/view?usp=sharing>

5.3.2 En neurosciences computationnelles

Bio-inspired analysis of deep learning on not-so-big data using data-prototypes ⁵

Dans le domaine de l'intelligence artificielle(Artificial intelligence), le *machine learning* fait souvent allusion à l'utilisation des réseaux de convolution qui représentent une série de couches avec des fonctions mathématiques spécifiques. Ainsi chaque neurone d'une couche n'est pas connecté à tous les éléments de la couche suivante (*not fully connected layers*). Ce traitement hiérarchisé se fait dans un seul sens, et plus on avance dans les couches, plus les propriétés décelées sont significatives.

Même si l'approche est "performante" dans de nombreux cas d'application du quotidien (voitures autonomes, reconnaissance faciale), elle demeure fortement liée aux données d'entraînement : pour arriver à un résultat, il est nécessaire d'en avoir de grandes quantités à disposition. En pratique, cette quantité est souvent limitée, surtout lorsqu'il s'agit d'une thématique précise non encore étudiée. Notre objectif de détection des configurations de cubes fait partie de ces exceptions.

Afin de tirer pleinement profit d'un petit jeu de données, des études ont été réalisées sur l'interprétabilité du processus d'apprentissage et de ses résultats et c'est dans cette philosophie que les auteurs de l'article ont travaillé à établir une méthode générique pour interpréter tous les CNN. Cette méthode est cependant fortement influencée par la description des données et surtout les (hyper paramètres) (paramètres à priori de l'entraînement). Les chercheurs ont alors proposé la mise en œuvre d'algorithmes capables d'ajuster automatiquement ces avant le lancement de l'entraînement.

Pourquoi chercher à interpréter l'apprentissage des réseaux de neurones ? Le but est de comprendre les raisons des choix derrière une prédiction du réseau de neurones et pouvoir l'expliquer afin que le manque de données d'entraînement soit compensé par une détection des prédictions "aberrantes" pour améliorer les résultats et c'est exactement le rôle des prototypes présentés dans le schéma du modèle proposé. Il s'agit d'une couche du réseau au niveau supérieur qui fournit aux couches plus profondes une information supplémentaire qui n'est pas juste l'appartenance à une certaine catégorie de classification mais aussi une représentation macroscopique de l'espace des propriétés afin de détecter si une catégorie appartient à une certaine région, si cette dernière contient des exemples adverses et si une catégorie est à considérer dans les étapes qui suivent. Finalement cette information supplémentaire mais significative améliorera le processus de prédiction.

⁵drumond:hal-01954911

6 Présentation du problème

L'idée globale est de pouvoir extraire une information à partir d'un ensemble de séquences d'événements présentés sous forme de vidéo. Pour cela plusieurs pistes se sont dévoilées comme le traitement d'images une par une ou la prise en compte de l'aspect temporelle et de travailler sur les vidéos directement à travers le pannel des outils du domaine Computer Vision.

Un ensemble de tests faits durant le projet de recherche sur ce thème a montré l'inefficacité de l'approche descriptive qui consiste à extraire des informations sur les cubes à l'aide de méthodes classiques d'analyse d'image. Il s'agissait de déterminer automatiquement des paramètres tels que la position des cubes dans l'image, leur taille à l'écran, leur position relative (le cube rouge est-il proche du cube blanc ?) à l'aide d'algorithmes de segmentation. Cependant les résultats (voir figure ??) étaient fortement dépendants des paramètres et des conditions d'acquisition et rendaient la bonne segmentation des cubes difficile en raison des mains, des ombres et de la qualité de la vidéo.

Une deuxième alternative était aussi mise en oeuvre. Il s'agit d'une utilisation directe des réseaux de neurones pré-entraînés pour obtenir des informations élémentaires : position des cubes et leur position relative, position des mains, déplacement vertical, déplacement horizontal. Puis dans un second temps de déduire les événements en fonction de règles conditionnelles sur ces informations élémentaires. Parmi les réseaux pré-entraînés étudiés on trouve :

- *EgoHand* [5], réseau de détection de main en vue à la première personne.
- *Salient Object Subitizing* [6], réseau capable de détecter le nombre d'élément pertinents dans une image.

Donc au lieu de mener une approche basée sur les images et risquer de perdre de l'information en éliminant les liens entre elles, le choix était fixé sur une entrée de type vidéo et une sortie de type fichier json/vidéo.

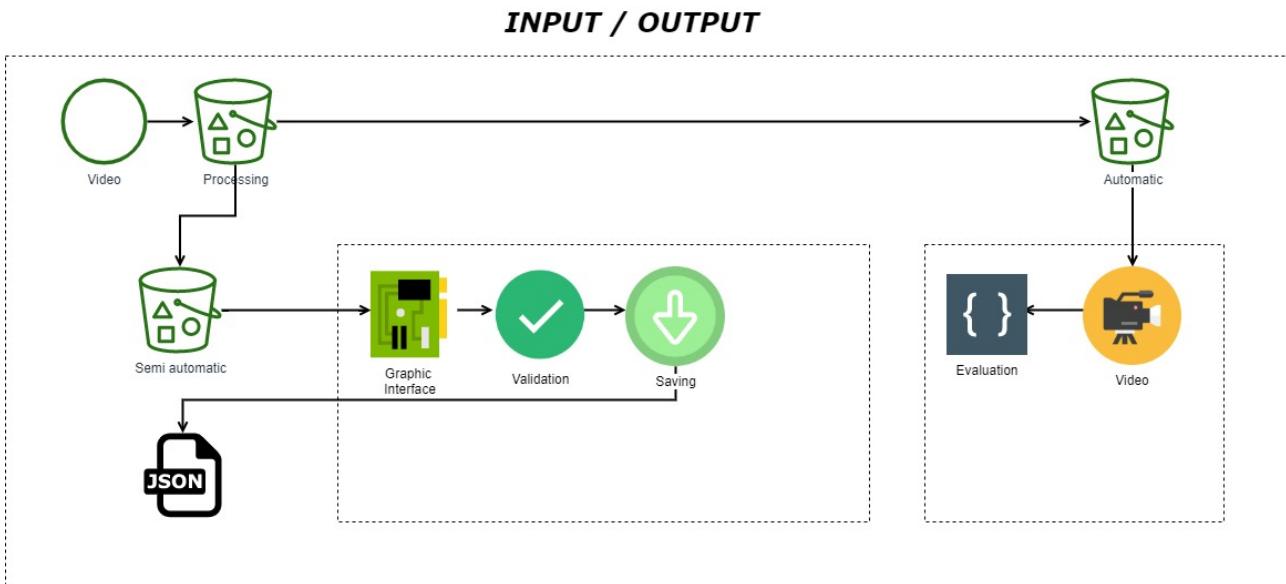


Figure 4: Diagramme entrée/sortie

Du coup, pour atteindre un tel objectif, il fallait opter sur des algorithmes d'apprentissage automatique construit progressivement afin de pouvoir :

- Maintenir l'enchaînement des configurations.
- Exiger une grande précision lors de l'extraction de ces configurations.
- Pouvoir mettre en place une interface de validation en temps réel et un ensemble de données pour le post traitement.

En parallèle, la solution doit respecter d'autres exigences telles:

- Les données d'entraînement de ces algorithmes doivent être préparées par un script qui fait partie de la solution.
- Tout script et outil doivent être déployés et accessible comme un service webService Web.

7 Modélisation de la solution

7.1 Situation de la solution

Au niveau de l'API développée par le LINE et toute les personnes qui contribuent sous le projet AIDE, la solution en question en fait partie comme un ensemble de fonctionnalités accessibles en service web Service Web ainsi qu'en locale. Pour l'accès comme service, il s'agit d'un service web REST qui à travers de, soit des requêtes web direct soit des traitements indirects, peut invoquer un des scripts disponibles sur des données en entrée que nous passons comme paramètre (directement en URL ou dans un formulaire). La totalité de cette disposition est présentée dans l'image ci-dessous:

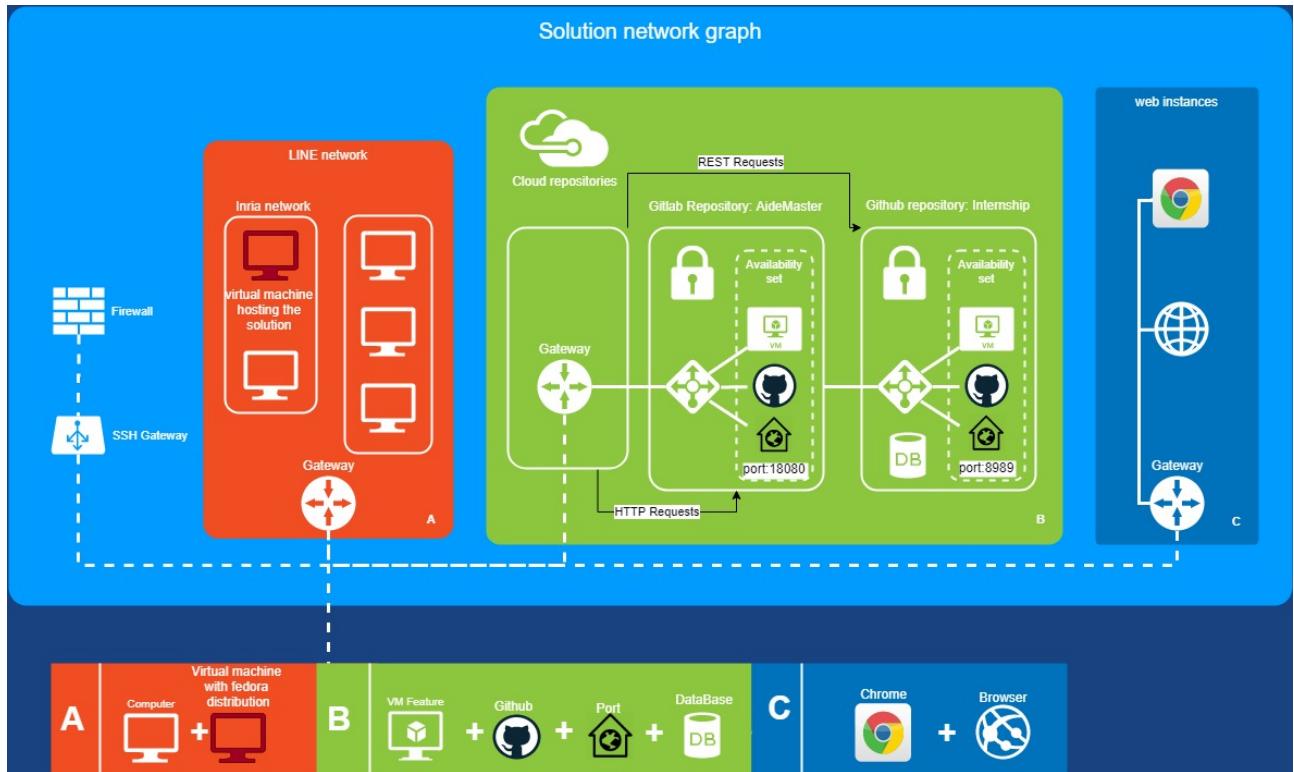


Figure 5: Situation de la solution au niveau du réseau

7.2 Modélisation

7.2.1 Extraction des images

Le principe de cette extraction est créer des images à partir des vidéos existantes et des fichiers json relatives. C'est à dire, on aura des images annotées par la configuration présente au moment donné. L'annotation dont en parle ici est sous forme de dossiers et sous-dossiers qui regroupent ces images comme dans le schéma suivant:

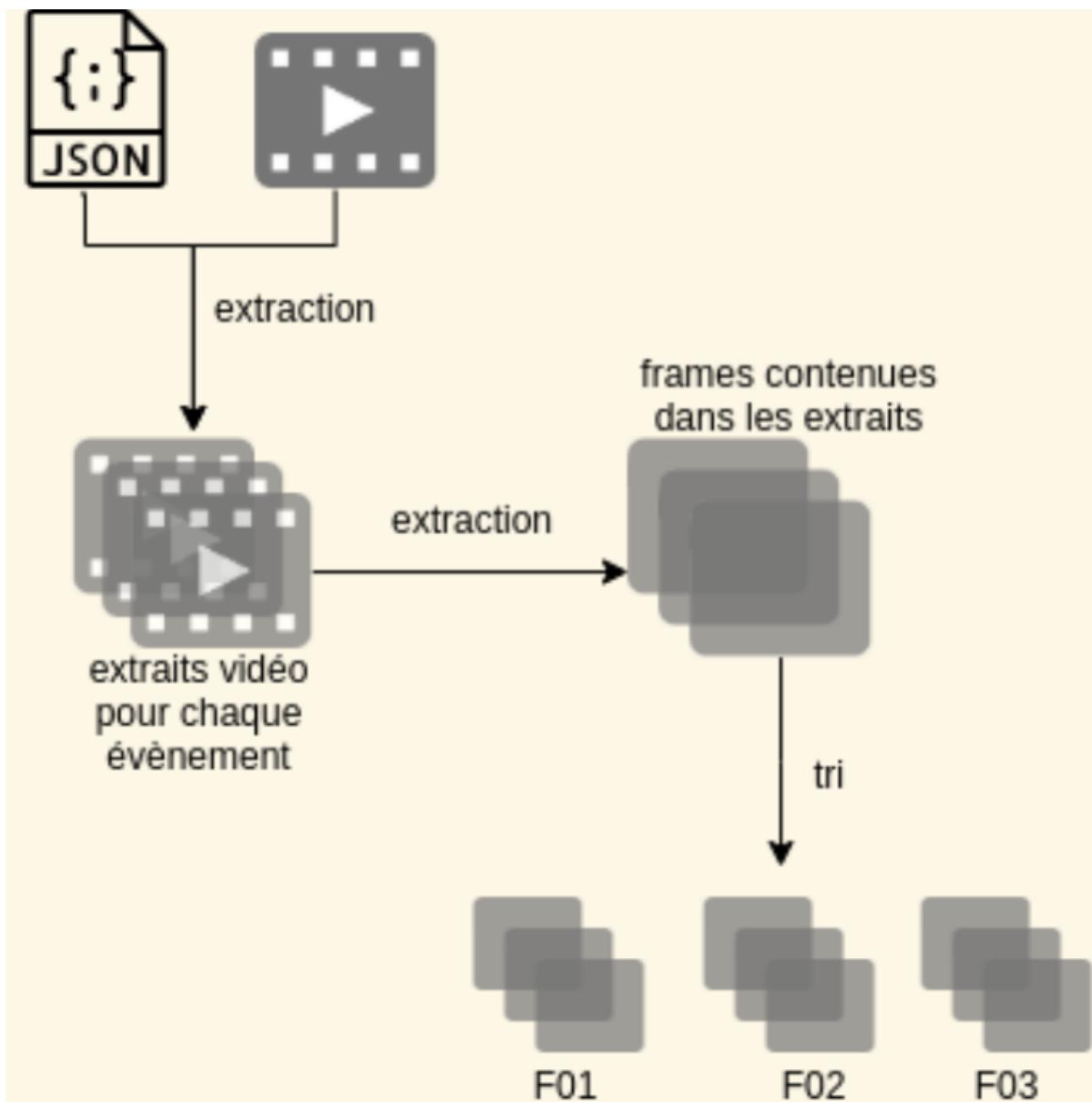


Figure 6: Extraction des frames

Cet ensemble de frames est la base de l'apprentissage du réseau de neurones et ces images feront l'objet de plusieurs itération qui vont améliorer la précision comme l'explique la partie suivante.

7.2.2 Apprentissage et prédition

Afin de résoudre une problématique à l'aide d'un réseau convolutif, plusieurs enjeux contribuent au choix de la conception et la méthode à suivre mais globalement la taille et la quantité du jeu de données sont les plus décisifs. Pour cela, il s'avère tellement lourd de suivre une démarche *from scratch* alors qu'on peut se servir de travaux libres qui sont faits sur des bases de données de grande taille. La question qui advient naturellement est alors : comment récupérer les fruits d'un apprentissage déjà effectué sur un autre réseau pour l'appliquer à ce cas ?

Il existe des démarches qui permettent ce processus comme :

- Le *fine-tuning* :

L'idée est de remplacer le classifieur au sommet du CNN et d'interrompre le model existant tout en retenant les poids et fixant les premières couches (pour palier à l'*over fitting*), après il s'agit d'ajouter des couches qui prennent des poids initialises préalablement et qui sont moins génériques par rapport au début d'un CNN normal.

- Les modèles prêts à l'utilisation :

L'autre astuce, non suffisante lors de changement des classes mais efficace en terme de temps de traitement, consiste à utiliser des modèles entraînés entièrement sur lesquels on a accès à tous les poids. Le format le plus courant qu'on trouve généralement est le format .caffé (contenant les poids du réseau) associé au format .pickle (qui contient les différentes classes).

Vu la multitude de configurations que la solution doit prendre en compte, un entraînement dès le début consommera beaucoup de temps et de ressources sans garantie d'un résultat satisfaisant, le choix est alors fixé sur le principe de fine-tuning. L'illustration suivante regroupe les différentes configurations à prendre en compte dans la classification:

CREACUBE 2. Activity										
AS00									B01	
AS01									B02	
AS02									P01. Imbalance	
AS03									P02. Rotation	
FL01									T01. No test	
FL02									T02. Drop Out / Abandon	
Repositioner cube même forme									T03. Succeed	
START U00									E04. Amazement/Surprise/Distraction	
									E05. Vigilance/Anticipation/Interest	

Figure 7: Interface CreaCube (Illustration du LINE)

La partie existante du réseau est en fait un ResNet50 qui se base sur 50 couches pré-entraînées sur plus d'un million d'images avec plus de 1000 objets détectable. Un ensemble de couches est ajouté à ce réseau tout en interrompant la phase finale de prédiction du ResNet50 (couche softmax).

Le choix du *fine-tuning* est du surtout à la création d'un nouvel ensemble de classes (configurations) et puisque les deux paramètres globaux d'un tel choix sont la taille et la similarité aux données. Pour l'entraînement il y'avait les 4 possibilités suivantes :

- Le nouveau jeu de données est large et ne ressemble pas à celui de l'entraînement:

Dans ce premier cas on est vraiment confiants de la probabilité faible de sur-apprentissage surtout si après un entraînement partiel sur les nouvelles couches, on repasse sur la totalité du réseau.

- Le nouveau jeu de données est large et ressemble à celui de l'entraînement:

Dans ce cas, on peut faire du fine-tuning mais sans entraîner la partie de convolution encore et en ajoutant au-dessus juste un classifieur linéaire.

- Le nouveau jeu de données est petit et ressemble à celui de l'entraînement:

Par contre ici, le manque de données rend un entraînement du réseau de convolution dangereux vu le grand risque de sur-apprentissage et alors il vaut mieux faire une classification linéaire⁶.

- Le nouveau jeu de données est petit et ne ressemble pas à celui de l'entraînement:

C'est dans ce contexte que s'inscrit la solution puisque les images d'origine au niveau du Resnet50 couvrent un éventail large d'activités alors que l'activité Creacube est plus concentrée sur une manipulation de cubes à courte distance. L'approche qui s'avère instinctive dans ce cas est de créer un modèle *from scratch* sauf qu'il est bénéfique de commencer à partir des poids initialisés au niveau du Resnet50 afin d'avoir assez de données et de confiance en entraînant le modèle. Le modèle de convolution ResNet50 suit la philosophie que plus notre réseau est profond (en terme du nombre de couches), plus on extrait des propriétés supplémentaires. Il faut palier à la dégradation de la précision d'entraînement en utilisant une démarche résiduelle (au lieu d'une approximation de la fonction $F(x)$ on donne une approximation de son résidu $H(x) - x$ et donc la fonction principale devient $F(x) + x$)⁷.

⁶https://en.wikipedia.org/wiki/Support-vector_machine

⁷voir la troisième partie de l'article: <https://arxiv.org/pdf/1512.03385.pdf>. La performance de ce Resnet étant fournie dans le répertoire GitHub: <https://github.com/KaimingHe/deep-residual-networks>

Pour les couches supplémentaires appartenant au modèle, elles ont été ajoutées grâce à tensorflow (version 1.15) dans l'ordre croissant:

- **AveragePooling2D:** Réduction du nombre de connexions faites par les couches de convolution en amont au profit des couches suivantes. C'est ce qu'on appelle le Downsampling.
- **Flatten:** Lancer un processus de flattenning sur un tenseur(conteneur de données à N dimensions) est équivalent à retenir une seule dimension des N dimensions qui existent.
- **Dense (Rectified Linear Unit activation):** Dans ce type de couches on parle d'une connexion complète entre tous les neurones des deux couches qui se suivent. Cela requiert une fonction d'activation qui est dans notre cas une ReLU qui tout simplement pour chaque valeur x retient $\max(0, x)$.
- **Dropout:** Cette couche permet une sélection de quelques neurones aléatoirement pour que leur contribution dans la couche suivante soit ignorée temporairement et du coup aucune mise à jour des poids n'est faite. Le but est d'éviter un sur apprentissage (overfitting).
- **Dense (Softmax activation):** Il s'agit d'une couche dense encore mais la fonction d'activation étant une softmax alors si elle retient N valeurs quelconques elle applique une fonction exponentielle afin de les transformer en probabilités (valeurs comprises entre 0 et 1 et dont la somme est égale à 1).

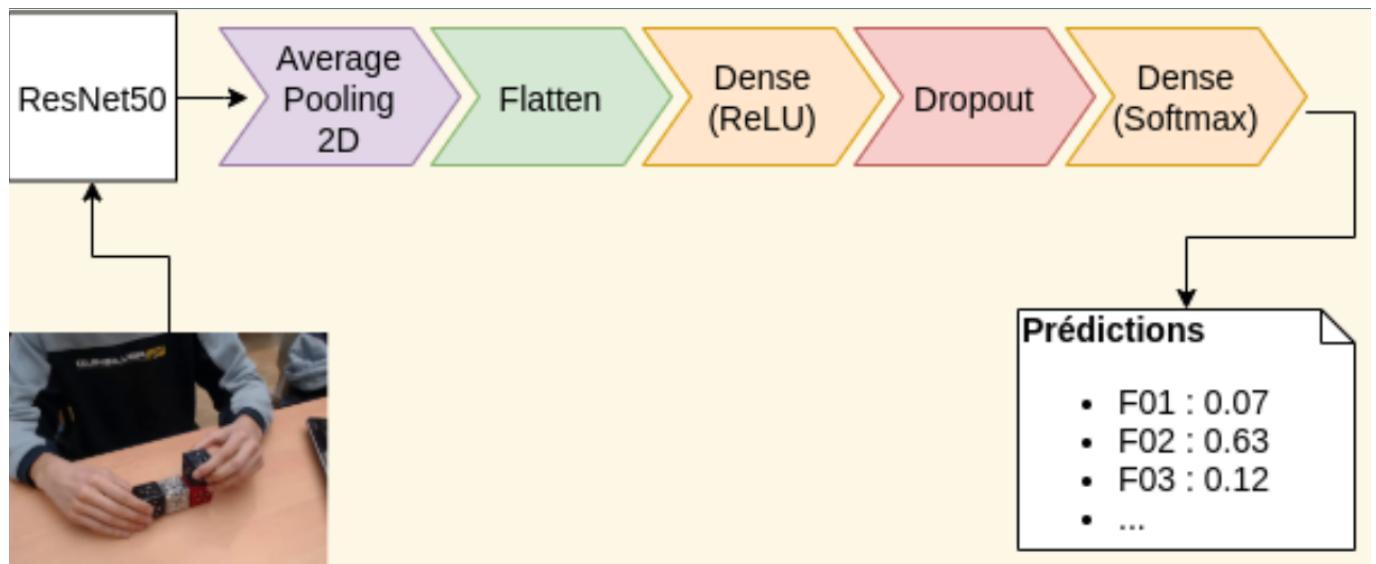


Figure 8: Réseau de neurones utilisé

7.2.3 Web Service

Ayant un chemin local ou sur le cloud vers une vidéo qu'on souhaite traiter en lançant le script dessus, un service web(Service Web) devait se mettre en place afin que cette manipulation d'effectue à travers des requêtes qui s'occupent d'appeler le script avec les paramètres adéquats.

Cependant, quelques vidéos sont confidentielles est non accessibles, du coup un traitement à part était instauré. La seule différence lors de l'appel du service est le chemin de la vidéo et la solution s'occupe du reste.

Le diagramme de classe suivant modélise la partie service web (Service Web):

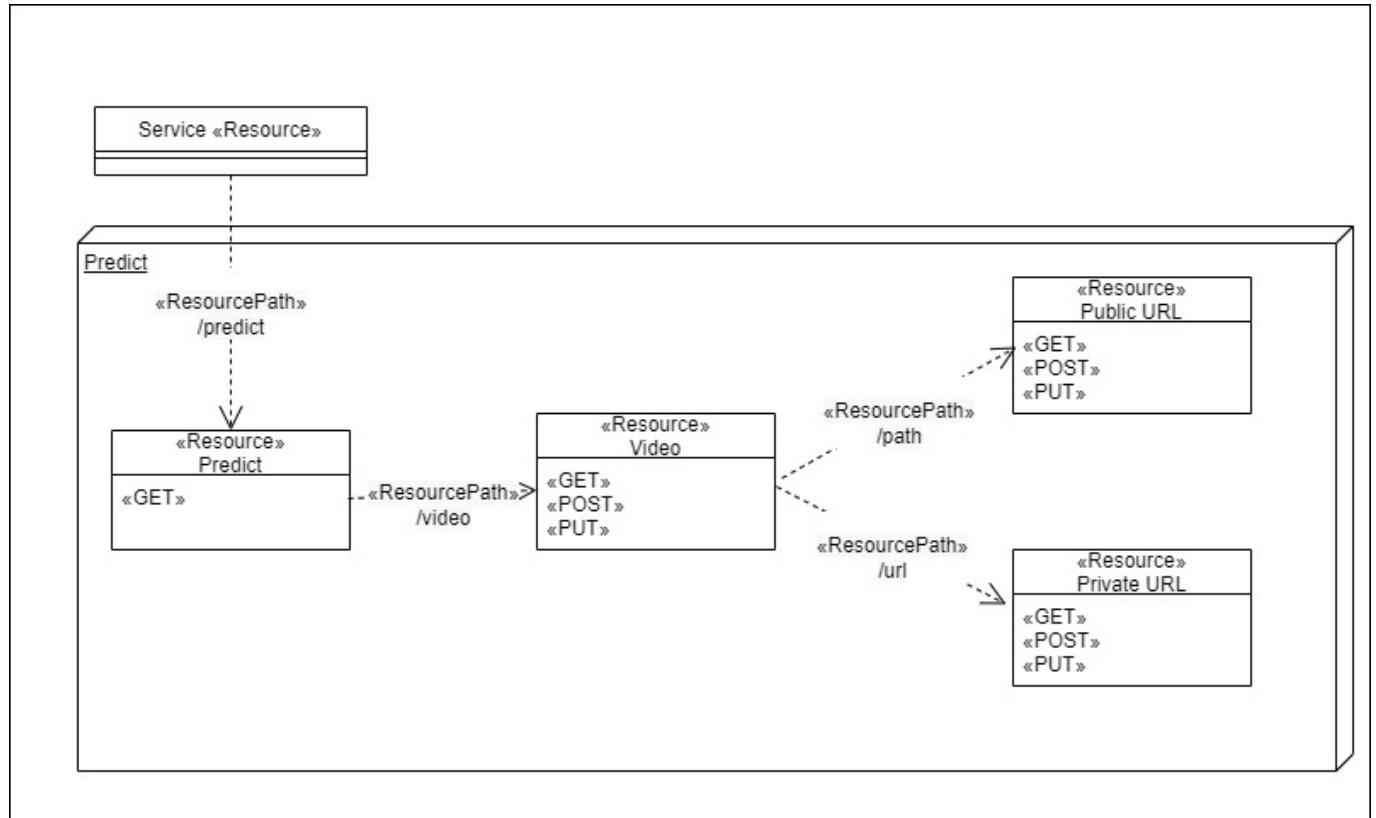


Figure 9: diagramme de classe du web service

8 Mise en oeuvre

8.1 Management du projet

Le stage étant entièrement à distance, l'échange souvent quotidien est maintenu sous forme de mails ou de tâches réalisables sur aide-aex⁸. Pour la gestion des délais et de l'avancement, le diagramme de Gantt permet une visualisation en pourcentage d'avancement ainsi que l'état actuel pour chaque partie et chaque tâche dans cette partie sur un échantillonnage d'une semaine.



⁸<https://team.inria.fr/mnemosyne/fr/aide/> : Une plateforme confidentielle regroupant les contributeurs du projet Aide: Analyse exploratoire

8.2 Préparation des données d'entraînement

L'entraînement du réseau requiert des données multiples et hétérogènes. Le groupement des vidéos des apprenants ainsi que les vidéos acquises lors du projet de recherche a facilité la première tâche de création de données. Toutefois, un traitement additionnel devait être fait afin d'extraire les motifs de ces vidéos en utilisant les fichiers json associés et qui sont sous la forme de:

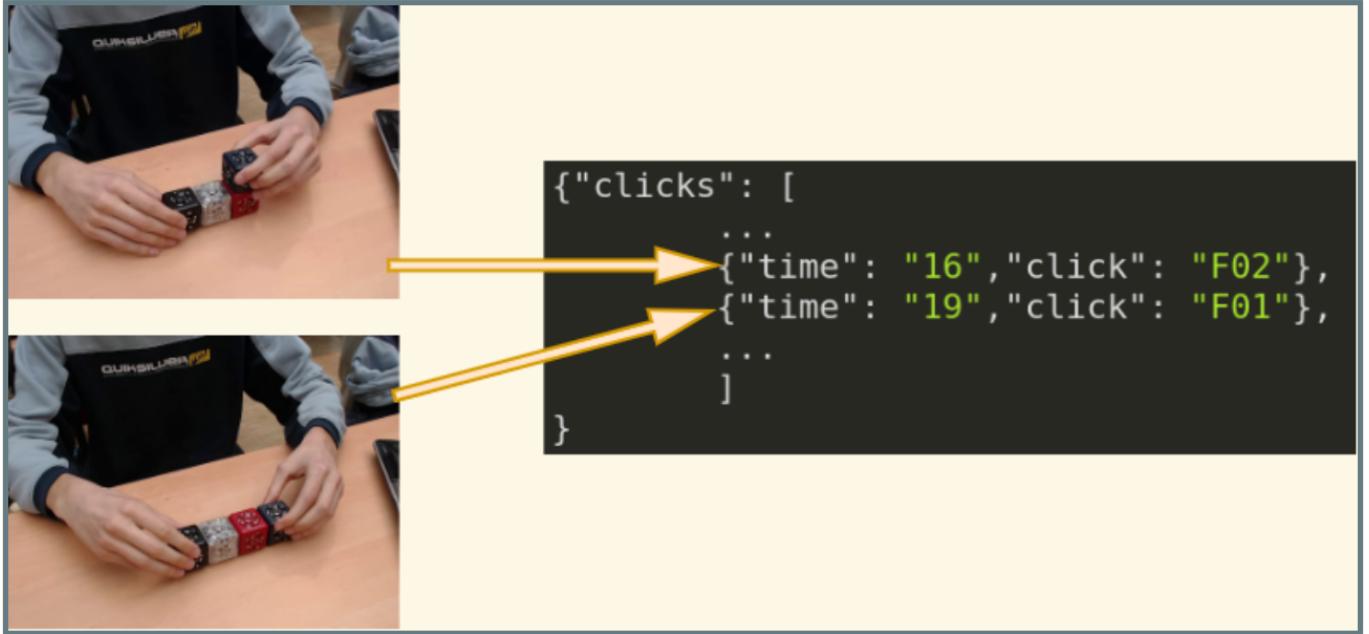


Figure 10: Structure des fichiers JSON

En couplant les vidéos et les json, on lance le script d'extraction qui fournit l'ensemble des données labélisées et prêtes pour lancer l'entraînement. A noter que durant toutes ces procédures, il faut s'assurer que les taux de données d'apprentissage soient plus petits que ceux utilisés pour entraîner le réseau de convolution pré-entraîné vu que ses poids sont considérés bons alors que pour nos couches, ils sont aléatoirement initialisés. On évite ainsi de distordre la partie pré-entraînée.

8.3 Entrainement

Ayant un jeu de données assez large, le premier entraînement était effectué sur un échantillon de 9000 images prises aléatoirement par un script parmi les 180 000 images générées. Cet entraînement de réseau était fait sur un total de 500 époques⁹, et a fourni les résultats suivants:

	precision	recall	f1-score	support
F01	0.84	0.85	0.84	195
F02	0.82	0.77	0.79	166
F03	0.66	0.70	0.68	128
F04	0.60	0.67	0.63	168
F05	0.77	0.66	0.71	109
F06	0.77	0.80	0.79	179
F07	0.86	0.89	0.88	166
F08	0.78	0.70	0.74	123
F09	0.67	0.73	0.70	159
F10	0.89	0.73	0.80	132
F11	0.79	0.69	0.74	107
F12	0.79	0.75	0.77	153
F13	0.74	0.75	0.74	122
F14	0.74	0.83	0.78	127
F15	0.69	0.58	0.63	71
F16	0.84	0.94	0.89	157
F17	0.82	0.91	0.86	123
F18	0.76	0.82	0.79	147
F19	0.76	0.67	0.71	141
accuracy			0.77	2673
macro avg	0.77	0.76	0.76	2673
weighted avg	0.77	0.77	0.77	2673

Figure 11: Résultats d'entraînement sur 9000 images

⁹Une époque égale un passage complet ou une itération sur les données

Les termes d'évaluation de l'entraînement du réseau se base sur la matrice de confusion [7] ¹⁰:

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Figure 12: Matrice de confusion

- **Precision:** Définit la proportion des éléments prédits qui sont corrects.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

Figure 13: Formule de calcule de Precision

- **Recall:** Définit la proportion des éléments corrects qui sont bien prédits.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

Figure 14: Formule de calcule de Recall

- **f1-score:** Définit une balance entre la précision et le recall lors d'une distribution inégale des classes de prédiction.

$$F1 = 2 \times \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Figure 15: Formule de calcule de f1-score

Ces éléments sont primordiales lors de l'évaluation du modèle après chaque entraînement. La précision de 77% est juste pour évaluer la partie entraînement et ne reflète en aucun sens la performance du réseau lors de la prédiction sur un nouveau jeu de données.

¹⁰<https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall>

8.4 Prediction et output

Une fois on juge la précision du modèle suffisante pour lancer un test, on passe à la prédiction avec un nouveau jeu de données non traité par le réseau de neurones. Une fois le script de prédiction lancé, une fenêtre graphique apparaît avec la vidéo au centre et les options en bas comme dans la figure suivante:

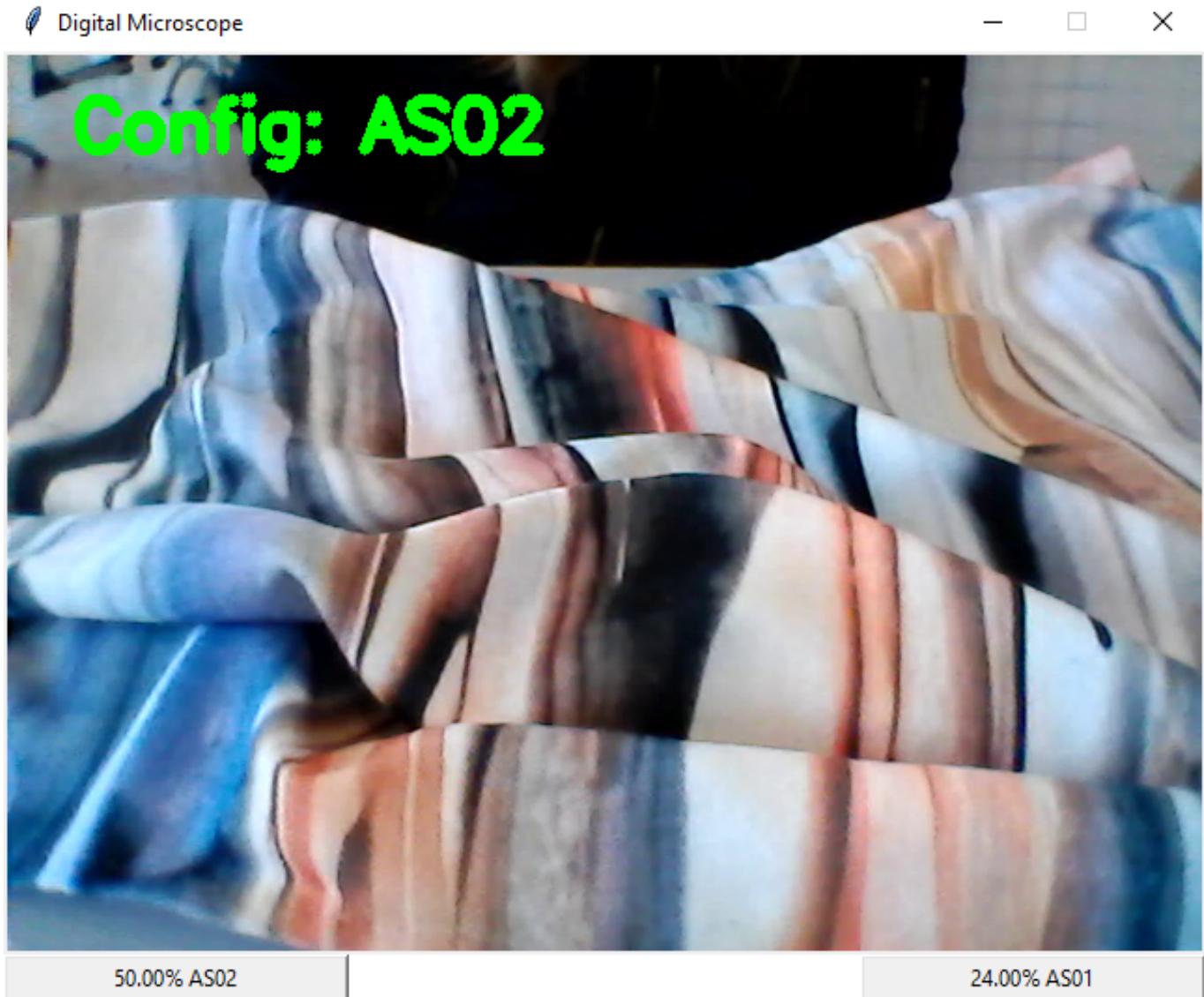


Figure 16: Interface graphique de prédiction

Les deux boutons montrent la précision de prédiction et sont cliquables afin de permettre la correction de résultats. Entre ces derniers il y'a une zone de texte qui permet d'écrire d'autres configurations en cas d'échec de l'algorithme. Un seuil est défini au début afin de permettre la sélection des éléments les plus précis pour l'écriture sur le fichier json.

```
-master > result.json
Web_Service\README.md × outputs\result.json × README.md × test_on_cloud.py × script-to-train ×
1   {"1": {"start": "00:00:0.06", "end": "00:00:0.12", "label": "F18", "prob": "60.00%"},  
2   "43": {"start": "00:00:2.61", "end": "00:00:2.67", "label": "F04", "prob": "18.00%"},  
3   "250": {"start": "00:00:15.15", "end": "00:00:15.21", "label": "F12", "prob": "13.00%"},  
4   "314": {"start": "00:00:19.03", "end": "00:00:19.21", "label": "F03", "prob": "20.00%"},  
5   "324": {"start": "00:00:19.63", "end": "00:00:19.69", "label": "F12", "prob": "22.00%"},  
6   "354": {"start": "00:00:21.45", "end": "00:00:21.51", "label": "F03", "prob": "22.00%"},  
7   "445": {"start": "00:00:26.97", "end": "00:00:27.03", "label": "AS01", "prob": "17.00%"},  
8   "493": {"start": "00:00:29.88", "end": "00:00:29.94", "label": "F03", "prob": "16.00%"},  
9   "507": {"start": "00:00:30.72", "end": "00:00:30.78", "label": "AS01", "prob": "16.00%"},
```

Figure 17: JSON Output

8.5 Web Service et déploiement

Une connexion en SSH est faite vers une machine virtuelle localisée à l'inria qui représente le serveur de production pour les développements faits en local. Cette connexion se fait en OpenSSH qui permet un lien SSH entre Windows(local) et Linux(server). Premièrement il fallait faire un ticket aux personnes en charge pour autoriser l'accès, puis quand celà était fait il a fallut générer une clé (publique et privée) et communiquer la clé publique pour l'autoriser sur le serveur. Ensuite une configuration était établie de ma part en parallèle avec celle du serveur ou il fallait faire un bounce(saut) avec le serveur sop-inria comme jump-host¹¹.

```

yjallouf@ssh-sop:~ 
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\WINDOWS\system32> Start-Service ssh-agent
PS C:\WINDOWS\system32> ssh-add C:\Users\youne\.ssh\id_rsa
Enter passphrase for C:\Users\youne\.ssh\id_rsa:
Identity added: C:\Users\youne\.ssh\id_rsa (C:\Users\youne\.ssh\id_rsa)
PS C:\WINDOWS\system32> ssh yjallouf@aide-line.inria.fr
ssh: connect to host aide-line.inria.fr port 22: Connection timed out
PS C:\WINDOWS\system32> ssh -q yjallouf@ssh-sop.inria.fr nc %h %p
nc: port range not valid
PS C:\WINDOWS\system32> ssh -q yjallouf@ssh-sop.inria.fr %h %p
bash: line 0: fg: no job control
PS C:\WINDOWS\system32> ssh -q yjallouf@ssh-sop.inria.fr %p 80
bash: line 0: fg: no job control
PS C:\WINDOWS\system32> ssh -q yjallouf@ssh-sop.inria.fr
Last login: Wed Jul 15 14:46:43 2020 from 87-88-191-79.abo.bbox.fr
SSH server at INRIA Sophia Antipolis - restricted access

Limited quotas. Instead of putting files on this server, you should
setup a "SSH transparent bounce" as described at:

https://wiki.inria.fr/support/EN:SAM:SSH_connection-EN

Last login: Wed Jul 15 14:46:43 2020 from 87-88-191-79.abo.bbox.fr
SSH server at INRIA Sophia Antipolis - restricted access

Limited quotas. Instead of putting files on this server, you should
setup a "SSH transparent bounce" as described at:

https://wiki.inria.fr/support/EN:SAM:SSH_connection-EN

[yjallouf@ssh-sop ~]$ ssh yjallouf@aide-line.inria.fr
The authenticity of host 'aide-line.inria.fr (138.96.124.28)' can't be established.
RSA key fingerprint is ae:87:52:11:4d:9e:97:ba:5f:7e:6d:bf:c0:ed:3b:68.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'aide-line.inria.fr,138.96.124.28' (RSA) to the list of known hosts.
yjallouf@aide-line.inria.fr's password:
Permission denied, please try again.
yjallouf@aide-line.inria.fr's password:
Connection closed by 138.96.124.28
[yjallouf@ssh-sop ~]$

```

Figure 18: Connexion en SSH

Une fois la connexion établie et la solution déployée, on lance le serveur nodejs comme suivant:

```

(venv2) D:\Internship\Web_Service>npm start

> cubicspotter@1.0.0 start D:\Internship\Web_Service
> node main.js

started on: 8989
listening on port 8989!
Connected to mongo server.
{
  videos: NativeCollection {
    ...
  }
}

```

Figure 19: Lancement du serveur Node

¹¹C'est un système de gestion d'accès qui constitue une couche de sécurité avant d'accéder à la machine cible.

Tout les prérequis étant satisfais pour le bon lancement des scripts, il suffit donc d'avoir le chemin complet local ou sur le cloud d'une vidéo publique ou juste le nom de la vidéo si elle est privée¹².

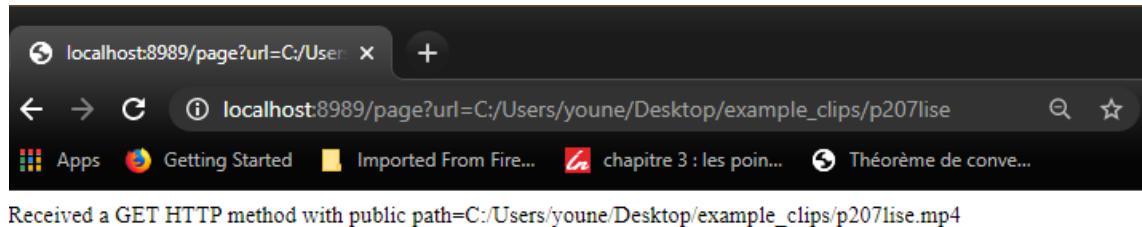


Figure 20: lancement sur une vidéo publique

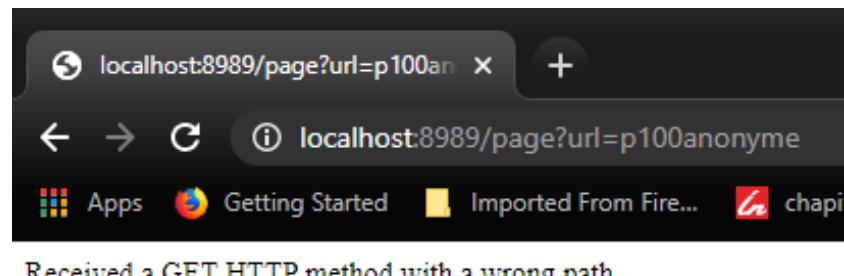


Figure 21: Lancement sur une vidéo privée qui n'existe pas

Le bon fonctionnement de cette partie du stage a été testé sur l'outil Postman voir annexe E, qui permet une vérification minitieuse de l'envoie et la réception des requetes web et des services REST. Le principe de ce test est expliqué sur l'article [8].

L'ensemble des outils développés, les guides de fonctionnement et le guide d'installation de chaque module sont disponibles sur le répertoire GitHub: github.com/izaganami/Internship

¹²les vidéos privées sont stockés dans le répertoire suivant: file:///workspace/private/videos/

8.6 Test du service déployé

Afin d'imiter un service web avec un nom de domaine et un port pour la partie test, on a opté pour le service ngrok¹³. Il s'agit d'ou outil permettant l'exposition des serveurs locaux derrière le pare-feu(firewall) et les Network address translations¹⁴ vers le public par des tunelles sécurisées. La figure suivante explicite le fonctionnement de ngrok:

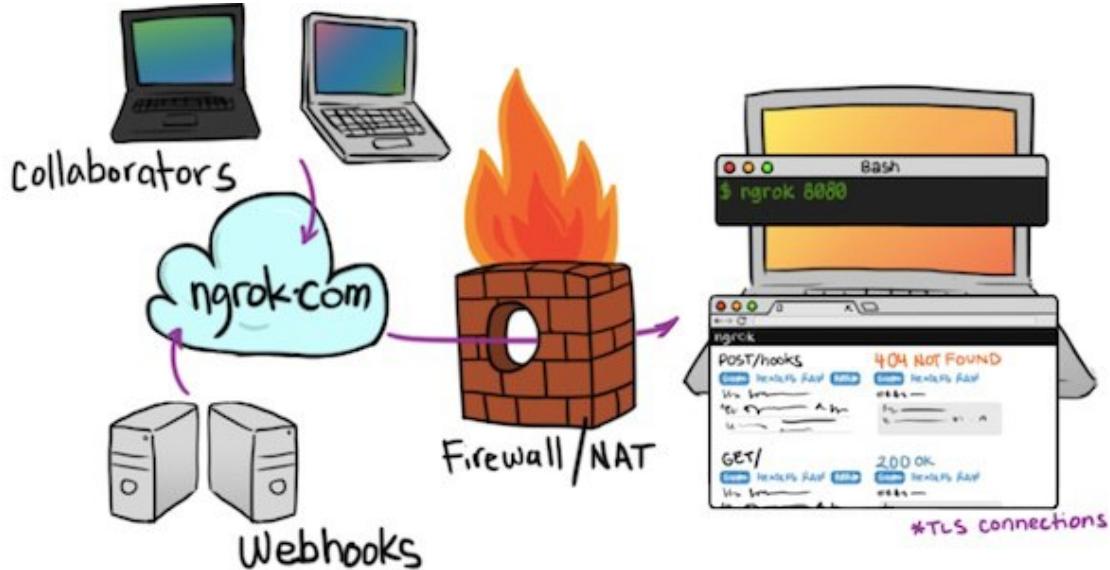


Figure 22: Schéma du ngrok. Référence du 11/08/2020: <https://www.cachem.fr/ngrok-tunnel-applications-locales/>

Une fois l'outil installé par le gestionnaire de packets npm, le lancement de l'outil se fait comme suit en spécifiant le port sur utilisé par le serveur:

```
(venv2) D:\Internship\Web_Service>ngrok http 8989
ngrok by @inconshreveable

Session Status          online
Account                 Younes  (Plan: Free)
Version                2.3.35
Region                 United States (us)
Web Interface          http://127.0.0.1:4040
Forwarding             http://0abddc3a58d5.ngrok.io -> http://localhost:8989
Forwarding             https://0abddc3a58d5.ngrok.io -> http://localhost:8989

Connections            ttl     opn     rt1     rt5     p50     p90
                        0       0      0.00    0.00    0.00    0.00
```

Figure 23: Schéma du ngrok. Référence du 11/08/2020: <https://www.cachem.fr/ngrok-tunnel-applications-locales/>

Comme indiqué ci-dessus, l'accès peut être fait en http comme en https, par contre vu que la version de l'outil utilisé est gratuite, le nom de domaine est automatiquement généré avec un chaîne alpha-numérique.

¹³<https://ngrok.com/product>

¹⁴Network address translation: Network address translation

Après le lancement, les pages 20 et 21 web déployées sur le serveur, prennent en compte les feuilles de styles et les scripts javascript associés et deviennent comme suit:

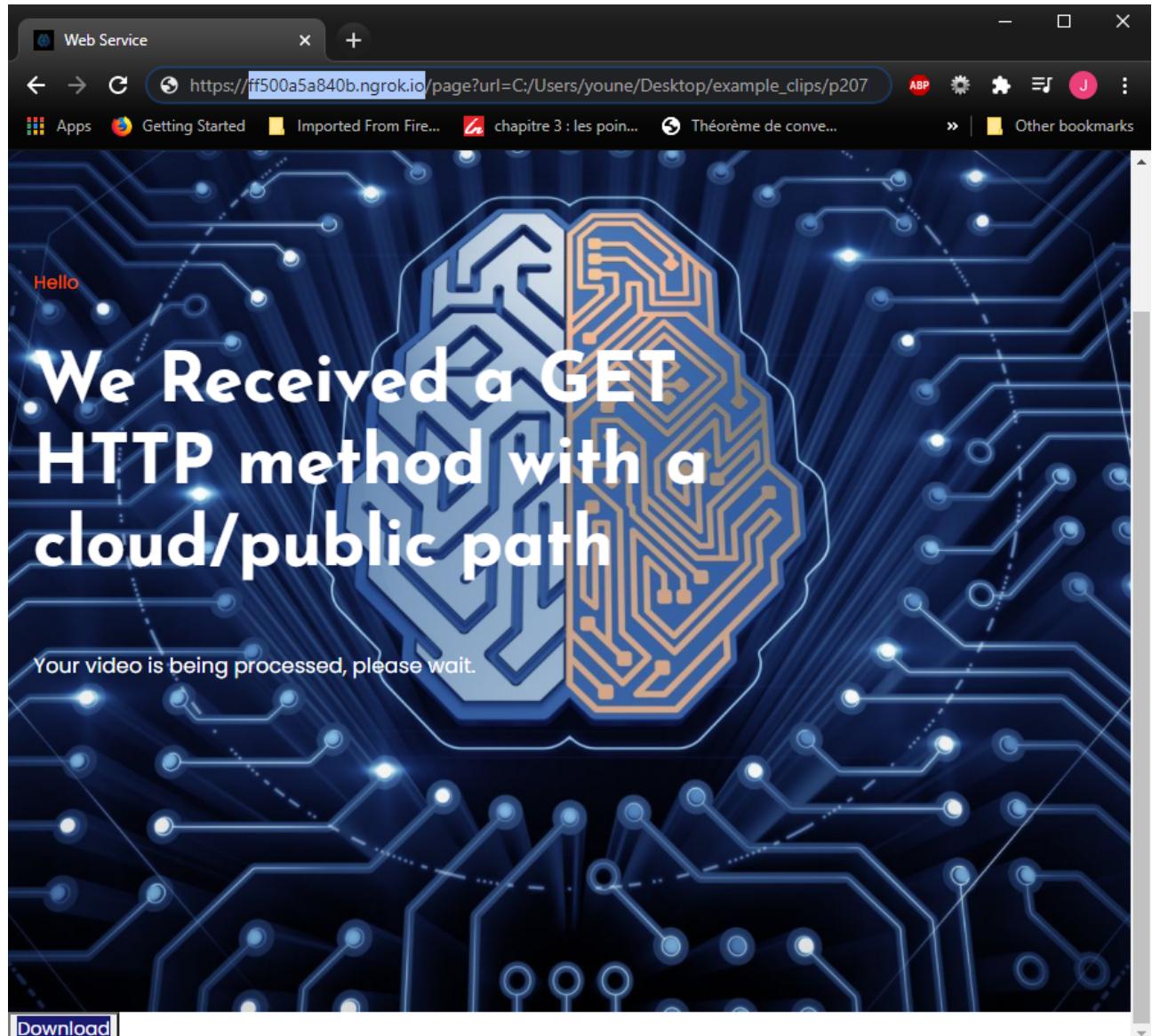


Figure 24: Page web lors du lancement en Cloud

Le bouton en bas nommé "Download" permet le téléchargement du fichier json associé dès que le processus de prédiction est fini. Le lien en bleu dans la zone URL correspond au nom de domaine du serveur généré aléatoirement par ngrok.

8.7 Long Short-Term Memory

Pour prendre en compte la dimension temporelle dans l'analyse des vidéos, un réseau de neurones récursif a été entraîné *Long Short-Term Memory* avec en élément d'entrée les caractéristiques issues du CNN pré-entraîné.

En retirant la dernière couche du CNN (le softmax), on a accès à l'avant-dernière couche, qui regroupe les caractéristiques d'une image ayant traversé le réseau. Ces caractéristiques sont envoyées au réseau LSTM dans un ordre précis pour que ce dernier synthétise l'information et améliore les prédictions.

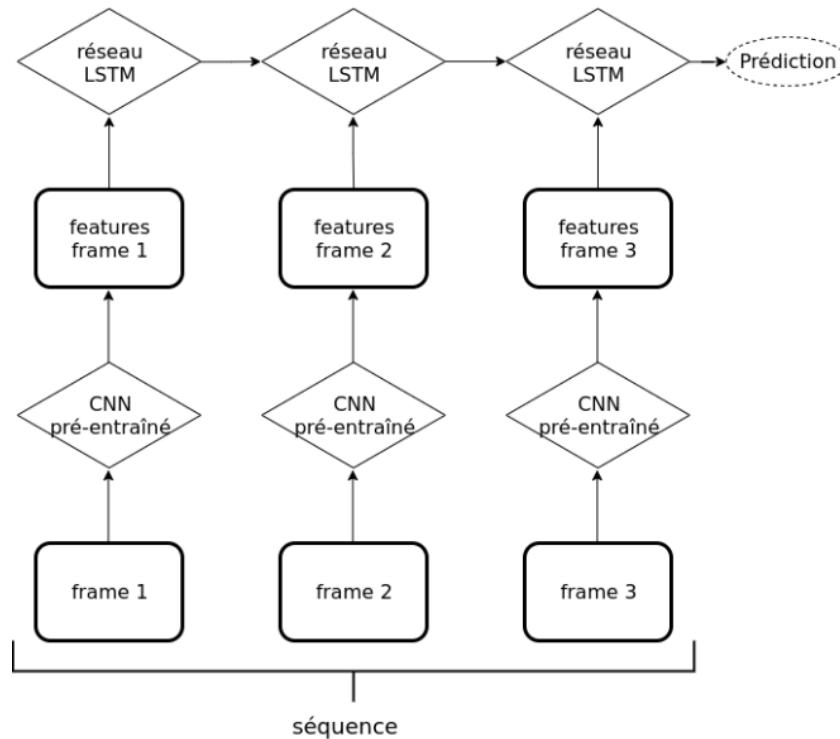


Figure 25: Schéma du réseau LSTM utilisé

9 Conclusion

Après avoir lancer le traitement semi-automatique sur des nouveaux échantillons, le temps requis pour faire le processing est devenu remarquablement minime avec une maîtrise plus grande des entrées et des sorties, on est arrivé donc à la conclusion qu'une démarche basée sur l'apprentissage automatique, mais avec un contrôle en parallèle, est fructueuse et beaucoup plus avantageuse chose qui offre la possibilité d'investir dans des modèles plus compliqués et plus couteux au niveau du temps et du matériel.

Malgré quelques difficultés du travail distant et de la configuration double du projet sur Windows et sur Linux sous la distribution Fedora32, ainsi que certaines compléxités au niveau du code, cette expérience était tellement enrichissante et j'ai quand même pu récolter de précieuses connaissances dans plusieurs domaines.

Ce stage m'a permis d'affronter la vie opérationnelle dans le domaine de développement informatique, d'améliorer mon savoir en SSH, Big Data et l'apprentissage automatique, ainsi que compléter ma formation théorique et acquérir le sens de responsabilité et de gestion des serveurs distants et de leur maintenance pour pouvoir faire mes premiers pas dans le monde professionnel.

J'ai de la chance d'effectuer ce stage au sein de LINE ou j'ai eu l'occasion d'avoir des connaissances sur le domaine d'éducation et sur les diverses tâches assurées par son personnel qui n'a cessé de me soutenir tout au long de cette période.

Le travail effectué constitue une étude prolongeable de l'approche de traitement automatique/semi-automatique puisque les modèles de réseaux de neurones pourront faire l'objet d'un re-entraînement plus intensif et sur un jeu de données encore plus grand. Les couches ajoutées au ResNet50 peuvent être multipliées en fonction de la performance de la machine hôte et du rendement désirables par le script. Outre le côté métier, le service peut être déployé pour le public avec un nom de domaine adéquat, plus de tunelles et un nombre intéressant de requêtes par seconde.

10 Bibliographie

Références bibliographiques

- [1] Margarida Romero, Dayle David, and Benjamin Lille. “CreaCube, a playful activity with modular robotics”. In: *International Conference on Games and Learning Alliance*. Springer. 2018, pp. 397–405.
- [2] Karen Barad. “Posthumanist performativity: Toward an understanding of how matter comes to matter”. In: *Signs: Journal of women in culture and society* 28.3 (2003), pp. 801–831.
- [3] Nikolaus Kriegeskorte and Pamela K Douglas. “Cognitive computational neuroscience”. In: *Nature neuroscience* 21.9 (2018), pp. 1148–1160.
- [4] Dan C. Ciresan, Ueli Meier, and Jürgen Schmidhuber. “Multi-column Deep Neural Networks for Image Classification”. In: *CoRR* abs/1202.2745 (2012). arXiv: 1202.2745. URL: <http://arxiv.org/abs/1202.2745>.
- [5] Sven Bambach et al. “Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.
- [6] Bryan C Asuncion. *Determination of Salient Design Elements through Eye Movements, Aesthetics, and Usability*. Mississippi State University, 2018.
- [7] ADJ van Dijk. “Machine Learning Crash Course: Classification”. In: (2020).
- [8] Anshu Soni and Virender Ranga. “API Features Individualizing of Web Services: REST and SOAP.” In: *International Journal of Innovative Technology and Exploring Engineering* 8 (2019), pp. 664–671.
- [9] Sepp Hochreiter et al. *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*. 2001.

List of Figures

1	Un apprenant entrain de lire les règles de l'activité CreaCube. Référence: CreaCube-PlayfulRobotics [1]	1
2	Manipulation des cubes par un apprenant	2
3	Démarche de résolution de l'activité CreaCube. Référence: CreaCube-PlayfulRobotics [1]	3
4	Diagramme entrée/sortie	6
5	Situation de la solution au niveau du réseau	7
6	Extraction des frames	8
7	Interface CreaCube (Illustration du LINE)	9
8	Réseau de neurones utilisé	11
9	diagramme de classe du web service	12
10	Structure des fichiers JSON	14
11	Résultats d'entraînement sur 9000 images	15
12	Matrice de confusion	16
13	Formule de calcul de Precision	16
14	Formule de calcul de Recall	16
15	Formule de calcul de f1-score	16
16	Interface graphique de prédiction	17
17	JSON Output	17
18	Connexion en SSH	18
19	Lancement du serveur Node	18
20	lancement sur une vidéo publique	19
21	Lancement sur une vidéo privée qui n'existe pas	19
22	Schéma du ngrok. Référence du 11/08/2020: https://www.cachem.fr/ngrok-tunnel-applications-locales/	20
23	Schéma du ngrok. Référence du 11/08/2020: https://www.cachem.fr/ngrok-tunnel-applications-locales/	20
24	Page web lors du lancement en Cloud	21
25	Schéma du réseau LSTM utilisé	22
26	Interface CreaCube (Illustration du LINE)	27
27	Le manque de dépendances à long terme en RNNs. Référence du 25/07/2020 : https://colah.github.io/posts/2015-08-Understanding-LSTMs/	28
28	Modèle répétitif sous un RNN. Référence du 25/07/2020 : https://colah.github.io/posts/2015-08-Understanding-LSTMs/	28
29	Interactions inter-couche dans un LSTM. Référence du 25/07/2020 : https://colah.github.io/posts/2015-08-Understanding-LSTMs/	29
30	les disciplines liant le cerveau humain et la machine pour créer des brain-computational models. Référence du 20/07/2020 : http://www.rochester.edu/UCIS/cognitivescience.html	30
31	Pooling avec f=max sur une fenêtre de 2x2. Référence du 22/07/2020 : "The best explanation of Convolutional Neural Networks on the Internet!", par: Harsh Pokharna	31
32	le pooling dans un CNN de reconnaissance d'objets. Référence du 22/07/2020 : "The best explanation of Convolutional Neural Networks on the Internet!", par: Harsh Pokharna	31
33	Interface Postman	32

Glossary

Affordance l'affordance signifie l'ensemble des interactions spontanées entre un apprenant et son environnement. 1, 26

Artificial intelligence Intelligence artificielle. 4, 5, 26

brain-computational models Brain-computational models signifie les modèles qui considèrent le cerveau humain comme une machine afin de bien comprendre son fonctionnement. 4, 6, 25, 26, 30

hyper paramètres C'est ensemble de paramètres qu'on fixe avant le lancement d'un entraînement d'un réseau de neurones. Ils sont fixes et non modifiables par la récursivité de l'entraînement. 5, 26

LINE Laboratoire d'Innovation et Numérique pour l'Éducation. 2, 26

Network address translation Network address translation est une méthode permettant le passage d'une adresse IP vers une autre en modifiant l'en-tête lors du transit dans le routeur.. 20, 26

Neurosciences Computationnelles Les neurosciences computationnelles (NSC) sont un champ de recherche des neurosciences qui s'applique à découvrir les principes computationnels des fonctions cérébrales et de l'activité neuronale, c'est-à-dire des algorithmes génériques qui permettent de comprendre l'implémentation dans notre système nerveux central de nos fonctions cognitives. Wikipedia, référence du 10/07/2020, [https://fr.wikipedia.org/wiki/Neurosciences_computationnelles].1, 26

Pooling Layer Il s'agit d'une couche dans un réseau de neurones qui contient une fonction progressive qui réduit la taille d'une représentation en réduisant le nombre de paramètres. Voir annexe D. 4, 26

REST Representational state transfer. C'est une architecture software qui définit un ensemble de règles et de contraintes à respecter lors de la mise en place d'un web service. 7, 19, 26

Science cognitive Il s'agit des études inter-disciplinaires du cerveau humain et de l'intelligence. Elle vise la caractérisation de la nature des connaissances humaines en forme et en contenu ainsi que comment ces connaissances sont acquises, traitées et utilisées. Référence du 11/07/2020, <https://bcs.mit.edu/research/cognitive-science>. 1, 26

Service Web Un Web Service est un protocole d'interface informatique de la famille des technologies web permettant la communication et l'échange de données entre applications et systèmes hétérogènes dans des environnements distribués. Référence du 13/07/2020, https://en.wikipedia.org/wiki/Web_service.6, 7, 12, 26

A Annexe : Interface Créacube

CREACUBE 2. Activity									
AS00									
AS01									
AS02									
AS03									
FL01 Turn cube w/o reloc	U00. Play instruction	U02. Questioning instructions	U03. No cubes in hand (no manipulation)	U04. Hands up with 1 cube	U05. Hands up with 2 cubes	U06. Hands up with 3 cubes	U07. Hands up with 4 cubes	T01. No test	P08. Doesn't move (wheels)
FL02 Représenter cube même forme	U01. Stop instructions	B01. Trial/error	B02. Analytical /systemic	B03. Hypothesizing	B04. Ego preservation	B05. Complaining	E01. Ecstasy/Joy/Serenity	T02. Drop Out / Abandon	P09. Doesn't move (on/off)
START U00								E03. Admiration/Trust/Acceptance	P10. Doesn't move (capteur)
								E04. Terror/Fear/Apprehension	P11. Doesn't move (invers)

Figure 26: Interface Creacube (Illustration du LINE)

B Annexe : structure du réseau LSTM

Le début de cette perspective a commencé avec les réseaux de neurones récursives (RNN). En théorie ces réseaux ont été capable de gérer les dépendances à long terme entre les couches et les paramètres, sauf que en pratique ils ont connu un échec dans ce sens. Le scientifique (1991) [German] and Bengio, et al. (1994) [9] était derrière la découverte du manque d'apprentissage des connexions d'information sous les RNNs.

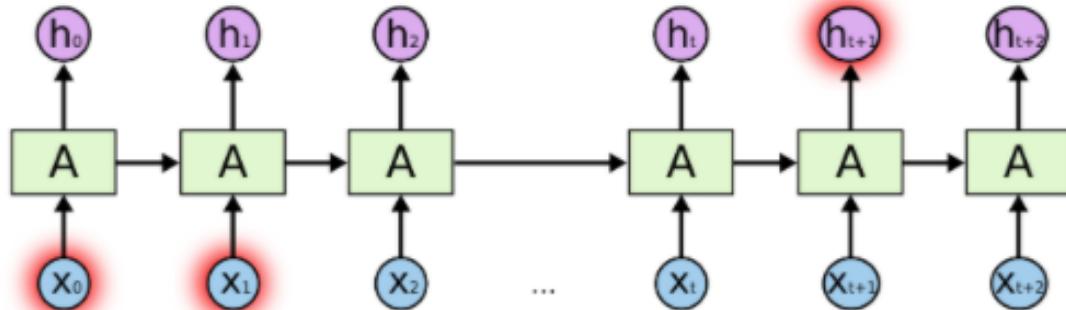
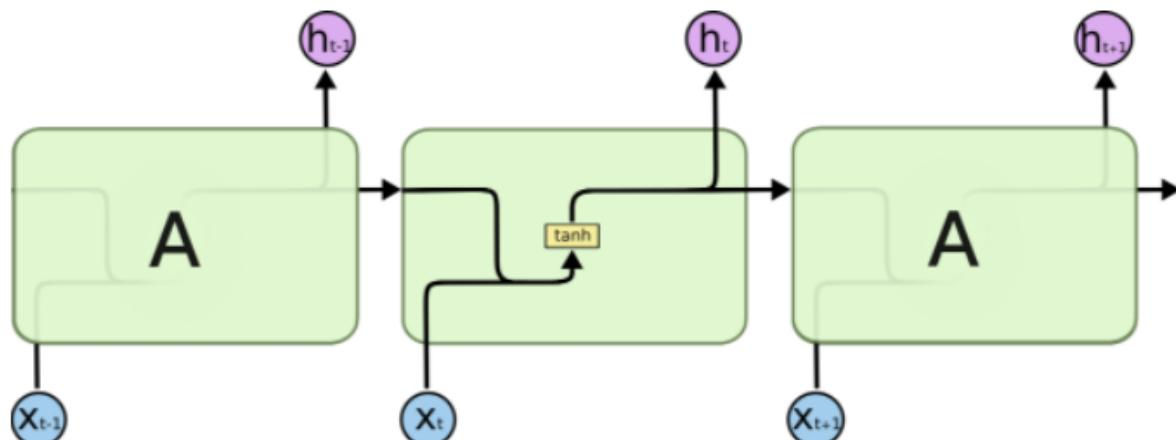


Figure 27: Le manque de dépendances à long terme en RNNs.

Référence du 25/07/2020 : <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

Normalement, les RNNs forment une chaîne de modules répétitives d'un réseau de neurones. Leur structure est simple, par exemple on peut avoir des couches de tangente hyperboliques (tanh) comme suit:

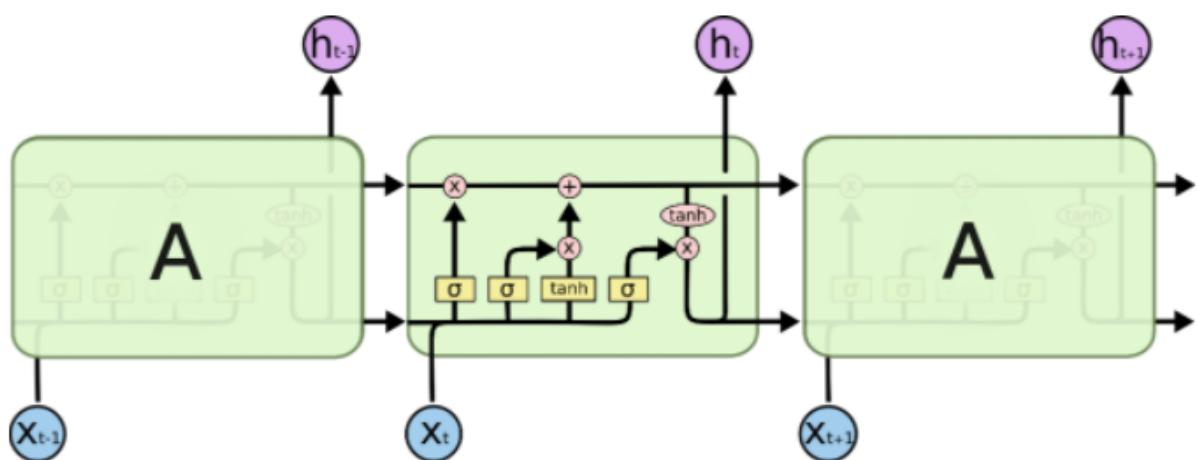


The repeating module in a standard RNN contains a single layer.

Figure 28: Modèle répétitif sous un RNN.

Référence du 25/07/2020 : <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

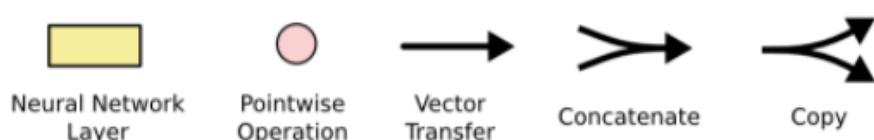
Par contre, sous un LSTM, ces modules répétitifs ont des structures différentes et n'ont pas une seule couche comme dans les RNNs mais plutôt quatre couches qui interagissent entre elles.



The repeating module in an LSTM contains four interacting layers.

Figure 29: Interactions inter-couche dans un LSTM.

Référence du 25/07/2020 : <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>



C Annexe : brain-computational models

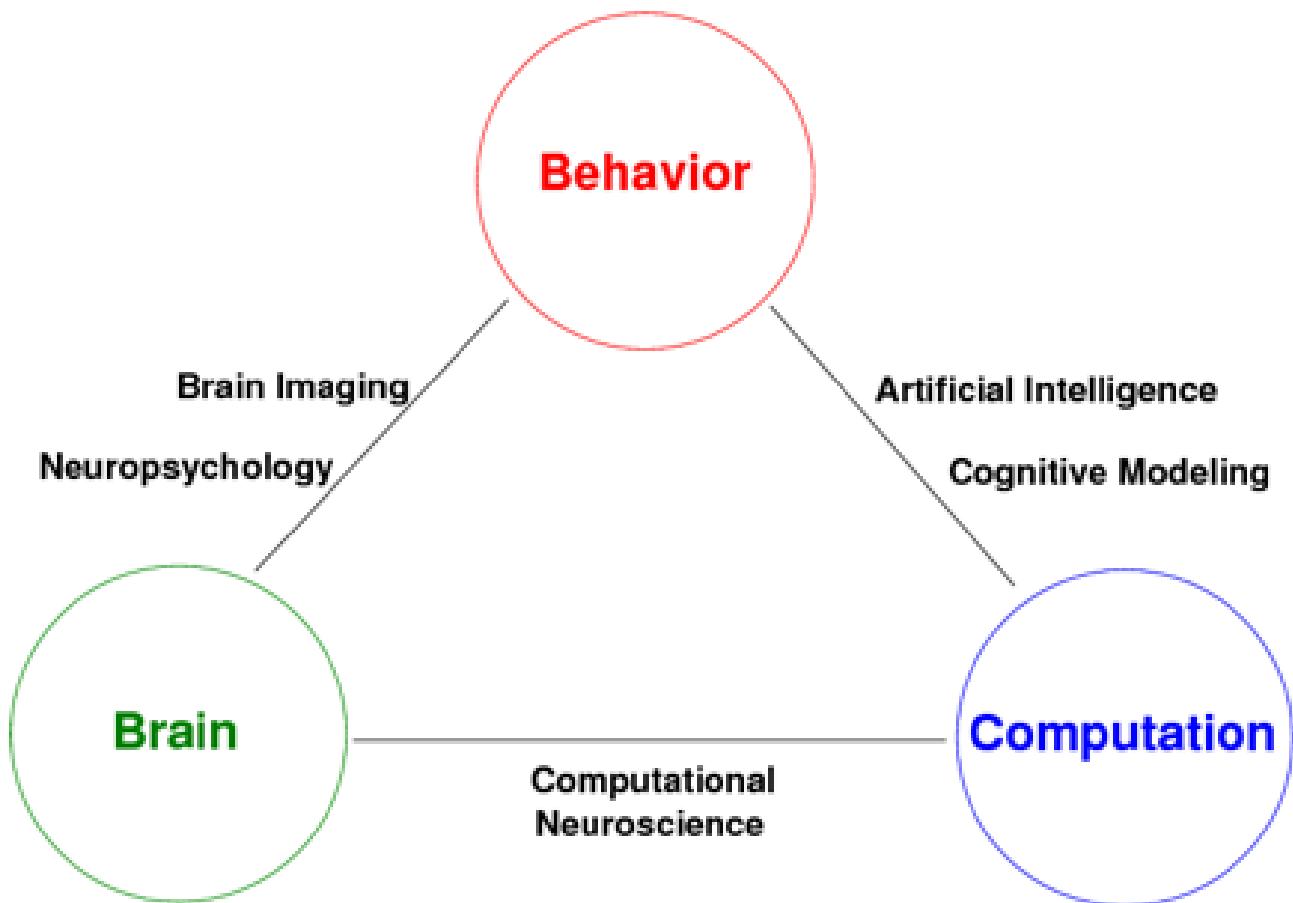


Figure 30: les disciplines liant le cerveau humain et la machine pour créer des brain-computational models.
Référence du 20/07/2020 : <http://www.rochester.edu/UCIS/cognitivescience.html>

D Annexe : Pooling layer

La couche du pooling peut se faire à travers des multiples fonctions, les plus utilisées sont la moyenne(average) et le maximum entre deux couches qui se suivent. Les figures suivantes représentent respectivement la fonction pooling avec le maximum et la situation des couches de pooling dans un réseau de convolution CNN.

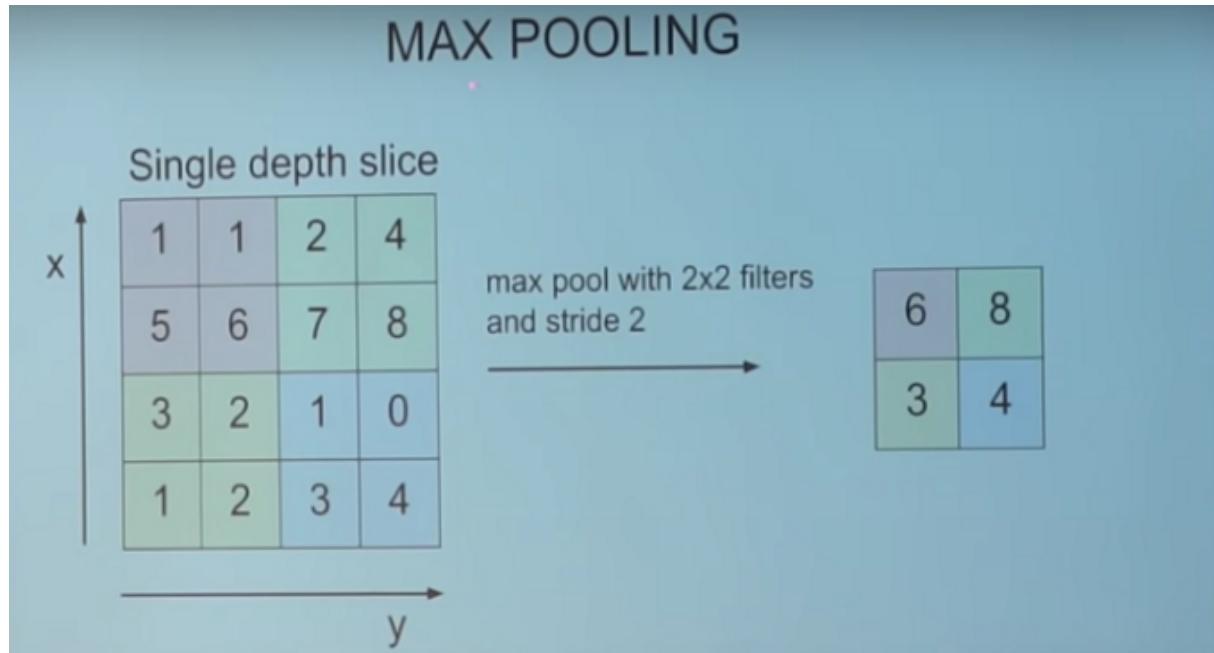


Figure 31: Pooling avec $f=\max$ sur une fenêtre de 2×2 .

Référence du 22/07/2020 : "The best explanation of Convolutional Neural Networks on the Internet!", par: Harsh Pokharna

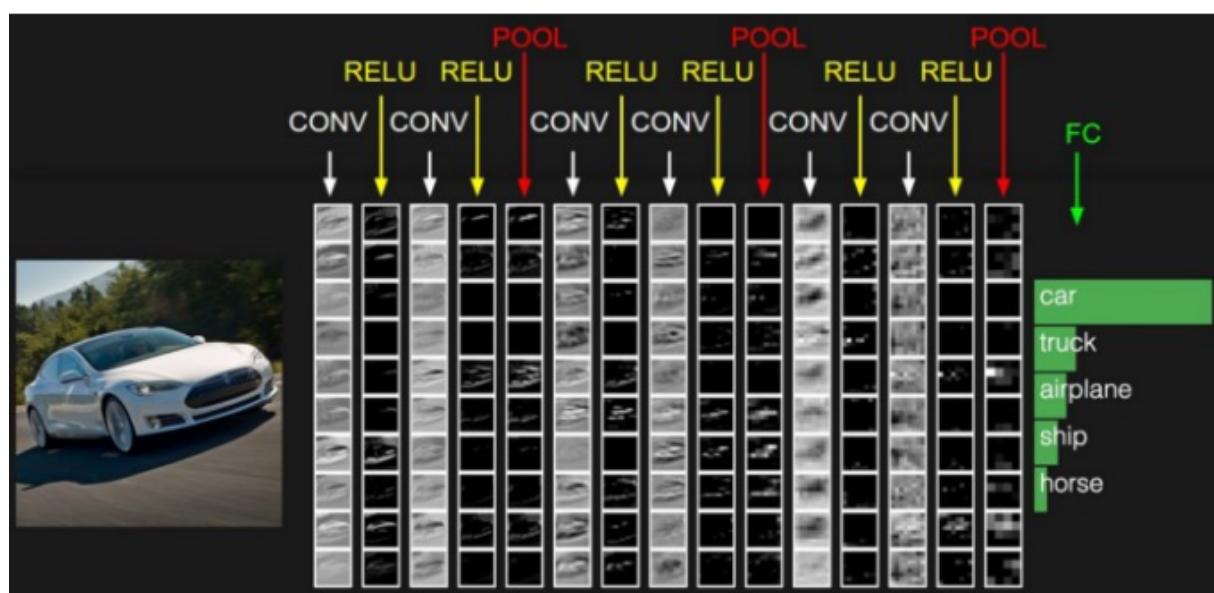


Figure 32: le pooling dans un CNN de reconnaissance d'objets.

Référence du 22/07/2020 : "The best explanation of Convolutional Neural Networks on the Internet!", par: Harsh Pokharna

E Annexe : Postman

L'interface de gestion des web-requests sur Postman ressemble à la figure suivante:

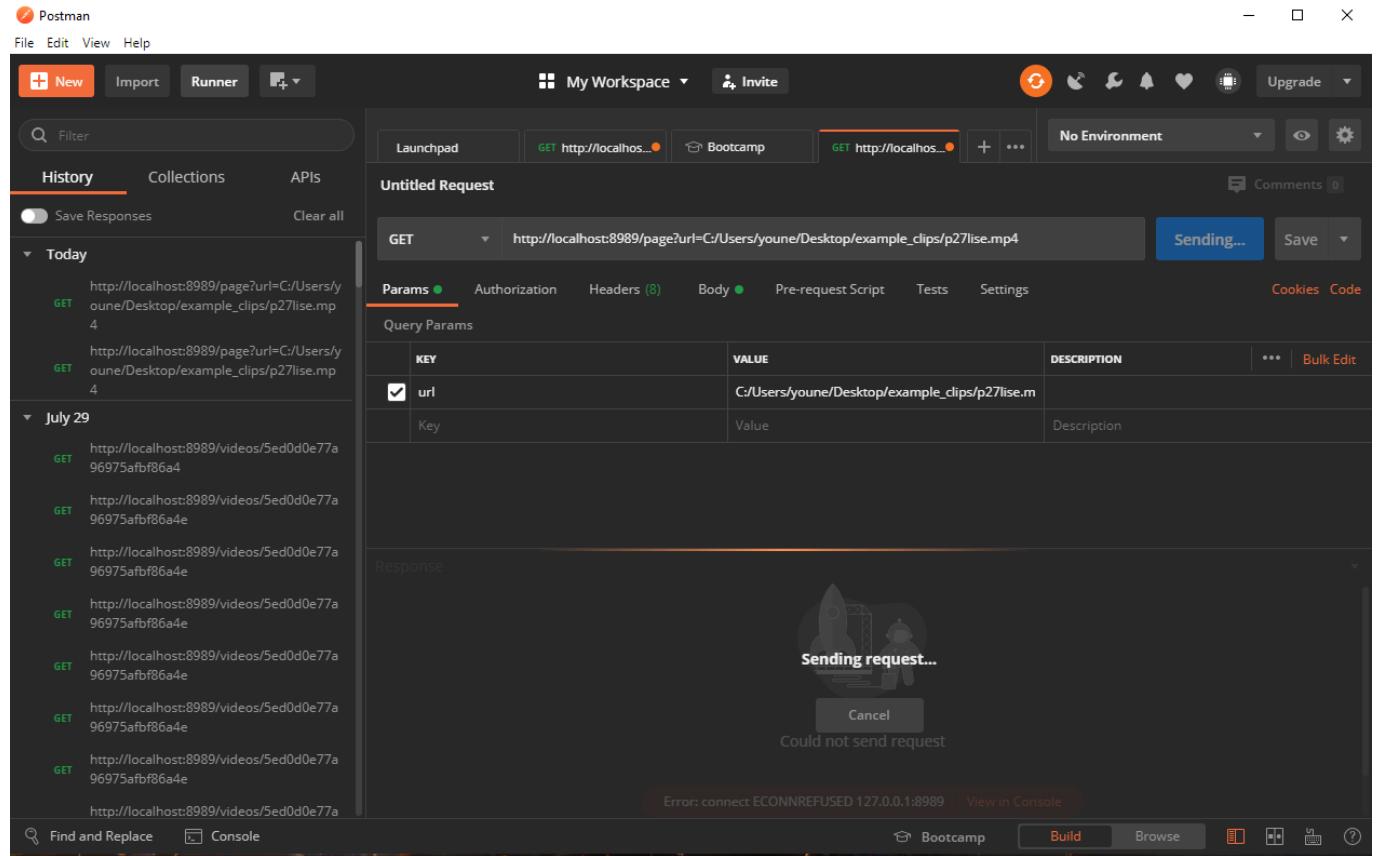


Figure 33: Interface Postman