Projet N°2 - Computer Vision / NLP

Winter 2021

Date de rendu du code : *dernier commit le jeudi 8 avril à 23h*
Restitution orale : *Vendredi 9 avril (25 min + 25 min)*

1. Objectifs du projet

Ce projet vise à appréhender une problématique de Data Science sur des sujets **Computer Vision** ou **NLP** sans avoir à disposition une large base de données pré-labellisée par le métier. C'est un cas habituel au vu du temps nécessaire pour constituer de telles ressources de qualité et le caractère innovant de ces approches. Il s'agira donc de s'appuyer au maximum sur des techniques de transfert learning, d'augmentation des données que vous aurez labellisées vous-même, d'utilisation de données open-source.

⇒ Le sujet est **libre** pour que la créativité puisse s'exprimer au mieux ! La partie 2 vous aidera à cadrer votre sujet.

Des chercheurs pourront déjà avoir touché au sujet choisi ou à certaines de ses composantes. Ce n'est pas discriminant au contraire; être capable de comprendre une démarche, réutiliser du code, réentrainer un modèle, vérifier des résultats, rajouter sa propre contribution (nouvelles images labellisées, changer des éléments du réseau ou certains paramètres d'entraînement, ...) sont dans le spectre de ce projet.

La liste ci-dessous présente l'ensemble des points sur lesquels vous serez évalués pour ce premier projet :

- ❏ Respect des **guidelines** (décrites dans la partie 3)
- ❏ Savoir **restituer** et **vulgariser** l'approche ainsi que la démarche scientifique : Pourquoi ce modèle pour ce sujet ? Pourquoi ces données-ci ? ...
- ❏ Savoir mettre en place un **pipeline** complet de deep learning (data augmentation, monitoring de métriques de validation, évaluation).
- ❏ Fournir un code **propre** et de **qualité**
- ❏ **Mobiliser** l'ensemble des connaissances et des outils appris jusqu'à présent

2. Guidelines

Afin de vous aider à cadrer votre projet, vous trouverez, ci-dessous, un ensemble d'attendus sur votre code et la démarche.

Prérequis : Constituer une équipe de 2 à 3 personnes maximum.

Code

- Créer un repository Gitlab sous ce groupe gitlab.
- Le code devra être modulaire et bien architecturé (template DDD recommandé).
- Le code devra être fonctionnel et facilement rejouable sur un environnement quelconque. Fournir en particulier les **packages pré requis** et un moyen d'obtenir les données utilisées/poids des modèles (soit directement, soit via un **script de téléchargement** visant ces éléments stockés sur un drive).
- Un **script d'entraînement** (préciser la durée d'entraînement) et un autre de **prédiction** devront être fournis. Par exemple sous la forme d'une ligne de commande :

```
python train.py --image_dir data/images_train/ --epochs 10 ...
python predict.py --image_dir data/images_test/ --gpu 0 ...
```

*Note : l'utilisation d'un **fichier de configuration** pour les paramètres du modèle et pour les paramètres d'entraînement/prédiction est vivement conseillé.*

- Le projet devra posséder un README
- Support de présentation clair et concis focalisés sur la démarche scientifique
- Une ***mini-app*** visuelle (cf. Partie 4).

Démarche

- Choisir une problématique parmi les sujets qui vous passionnent ! Par exemple, on peut penser vis à vis de l'actualité à la santé et les challenges sous-jacents au COVID (vérification de la distanciation sociale, priorisation de mails, analyse de textes scientifiques, ou sur une note plus douce les sentiments/degrés d'attention lors d'un call vidéo, le nombre de personnes applaudissant à votre fenêtre, ...) mais le sujet peut être tout autre.
  *Note : si développement d'une mini-app (cf. Partie 3), privilégier des cas d'usage où des **données peuvent être extraites ou générées facilement** pour qu'un utilisateur puisse tester (ex : prise de photo, saisie de texte, etc.)*

- De manière générale, une bonne source d'inspiration sur ce qu'il est possible de faire est trouvable sur ce site : https://www.paperswithcode.com/sota.
  Il est tout à fait possible voir recommandé dans le temps imparti de réutiliser du code si l'on se l'approprie, que l'on démontre sa compréhension du modèle en exposant forces et faiblesses (slides de restitution) et qu'on l'applique à son propre jeu de données constitué. Plus particulièrement pour le NLP vous pouvez

regarder du côté de https://huggingface.co/models pour des poids pré-entraînés.

- Des outils de labellisation sont précisés plus bas pour vous permettre de créer vos données. Il est essentiel dans ce projet d'en utiliser un, même pour peu d'éléments; l'idée est de travailler sur leur utilisation en mode collaboratif.

- Pour entraîner vos modèles vous pouvez passer par le cloud (crédits GCP) si vous n'avez pas de GPU à disposition. Sachez que d'autres plateformes existent comme https://vast.ai/ ou encore Google Colaboratory. Cela n'est pas une course à la performance et il n'est pas nécessaire de s'entraîner des jours durant pour spécialiser un modèle pré-entraîné par ex.

3. Format de restitution

En plus du code packagé et des éléments de restitution de la démarche scientifique restituée à l'oral, il sera bien vu de fournir une mini-application web de restitution des résultats (assez simple).
Certains outils sont relativement faciles à prendre en main sans grosse connaissance préalable dans le front-end :

- https://www.streamlit.io/ (recommandé car plus simple et intuitif)
- https://dash.plotly.com/ (nécessite plus de temps)

L'idée n'est pas de multiplier les features disponibles dans l'application mais juste d'avoir une **interface de visualisation des résultats.** Par exemple une zone d'upload d'une image/vidéo/texte, un bouton pour lancer la prédiction et une visualisation du résultat.

4. Données

En plus des images/vidéos ou textes que vous pouvez générez vous-même voici des liens utiles permettant de trouver des **jeux de données**, n'hésitez pas à vous appuyez sur ces ressources ou d'autres ressources que vous jugerez pertinentes :

- https://www.datasetlist.com/
- https://huggingface.com/datasets
- https://www.visualdata.io/
- https://datasetsearch.research.google.com/

Afin de réaliser de la **data augmentation** ce package est suggéré pour l'image :
⇒ https://github.com/aleju/imgaug

Enfin pour les **outils d'annotation** de vos données en voici une liste non exhaustive :
⇒ https://www.datasetlist.com/tools/

Suivant le sujet certains sont plus appropriés que d'autres; CVAT et docanno permettent de faire nombre d'entre eux et en collaboratif.

Et surtout HAVE FUN !