



ÉCOLE
CENTRALE LYON

Edition 2018-19

Projets de l'option Informatique

Mohsen Ardabilian, Daniel Muller

Projets

1. Generative Adversarial Text to Image Synthesis.....	2
2. Image-to-Image Translation with Conditional Adversarial Networks.....	3
3. Image Caption Generation	4
4. Video caption generation.....	5
5. Chat-bot.....	6
6. Text Mining et analyse de sentiments ET / OU Conversation Chatbot*	7
7. Textual Big Data : Application du "Deep Learning" au traitement des langues naturelles	8
8. Etude et réalisation d'un environnement d'apprentissage par renforcement.....	9
9. « Next-nextcloud » : mise à niveau et développement/déploiement d'un dropbox pour le secteur humanitaire	10
10. Interactions homme / robot humanoïde par la voix.....	12
11. Deep Learning pour la prédiction de l'émotion induite par les vidéos	13
12. Analyse de trajectoires GPS et Visualisation.....	14

1. Generative Adversarial Text to Image Synthesis

Tuteur(s) : Liming.Chen@ec-lyon.fr

Contact : Liming.Chen@ec-lyon.fr

Automatic synthesis of realistic images from text would be interesting and useful, but current AI systems are still far from this goal. However, in recent years generic and powerful recurrent neural network architectures have been developed to learn discriminative text feature representations. Meanwhile, deep convolutional generative adversarial networks (GANs) have begun to generate highly compelling images of specific categories, such as faces, album covers, and room interiors. In this project, we will develop a novel deep architecture and GAN formulation to effectively bridge these advances in text and image modeling, translating visual concepts from characters to pixels. We will demonstrate the capability of our model to generate plausible images of birds and flowers from detailed text descriptions.

This white and yellow flower have thin white petals and a round yellow stamen



References

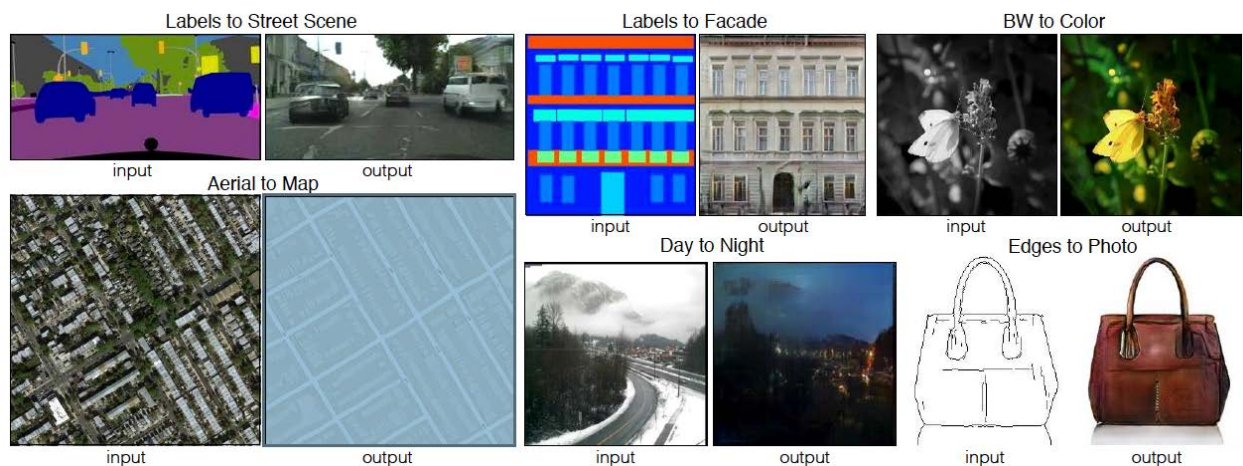
<https://arxiv.org/pdf/1605.05396.pdf>

2. Image-to-Image Translation with Conditional Adversarial Networks

Tuteur(s) : Liming.Chen@ec-lyon.fr

Contact : Liming.Chen@ec-lyon.fr

This project will conditional adversarial networks as a general-purpose solution to image-to-image translation problems. These networks not only learn the mapping from input image to output image, but also learn a loss function to train this mapping. This makes it possible to apply the same generic approach to problems that traditionally would require very different loss formulations. We will demonstrate that this approach is effective at synthesizing photos from label maps, reconstructing objects from edge maps, and colorizing images, among other tasks. As a community, we no longer hand-engineer our mapping functions, and this work suggests we can achieve reasonable results without hand-engineering our loss functions either.



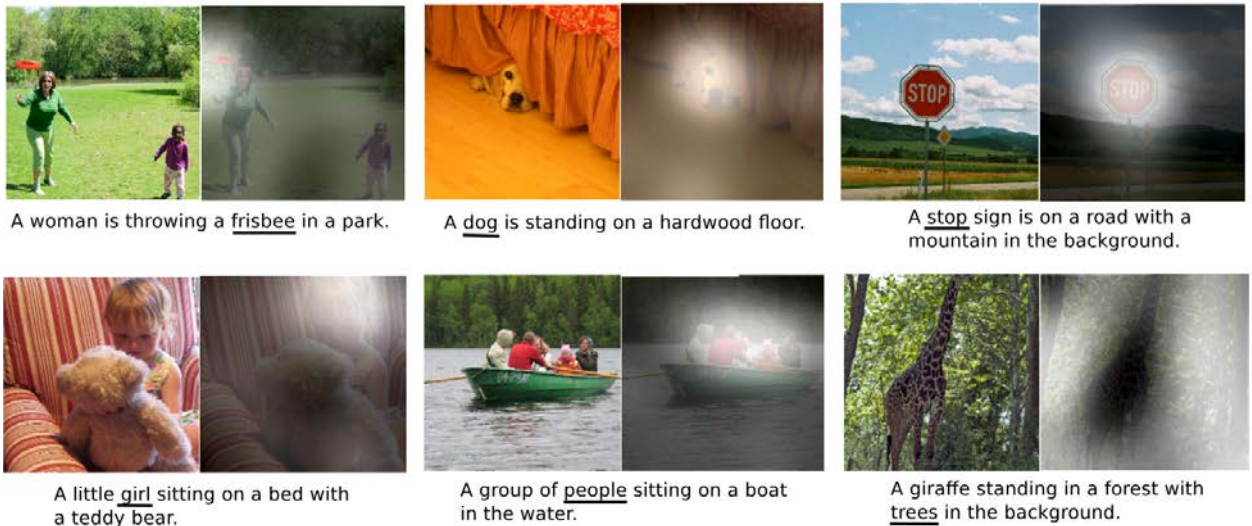
References

<https://arxiv.org/pdf/1611.07004v1.pdf>

3. Image Caption Generation

Tuteur(s) : Liming.Chen@ec-lyon.fr
Contact : Liming.Chen@ec-lyon.fr

Inspired by recent work in machine translation and object detection, this project introduces an attention-based model that automatically learns to describe the content of images. We will train this model in a deterministic manner using standard backpropagation techniques and stochastically by maximizing a variational lower bound. Also, we will show through visualization how the model is able to automatically learn to fix its gaze on salient objects while generating the corresponding words in the output sequence. The use of attention with state-of-the-art performance will be validated on three benchmark datasets: Flickr8k, Flickr30k and MS COCO.



References

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention", ICML, 2015








<https://arxiv.org/pdf/1502.03044.pdf>

4. Video caption generation

Tuteur(s) : Liming.Chen@ec-lyon.fr
 Contact : Liming.Chen@ec-lyon.fr

Real-world videos often have complex dynamics; and methods for generating open-domain video descriptions should be sensitive to temporal structure and allow both input (sequence of frames) and output (sequence of words) of variable length. To approach this problem, we propose a novel end-to-end sequence-to-sequence model to generate captions for videos. For this we exploit recurrent neural networks, specifically LSTMs, which have demonstrated state-of-the-art performance in image caption generation. Our LSTM model is trained on video-sentence pairs and learns to associate a sequence of video frames to a sequence of words in order to generate a description of the event in the video clip. Our model naturally is able to learn the temporal structure of the sequence of frames as well as the sequence model of the generated sentences, i.e. a language model. We evaluate several variants of our model that exploit different visual features on a standard set of YouTube videos and two movie description datasets (M-VAD and MPII-MD).

Topics sequence-to-sequence learning, RNN, LSTM, Attention, Schedule sampling, beam search

						
(1)	(2)	(3)	(4)	(5)	(6a)	(6b)

Temporal Attention (GNet+3D-conv_{att}):

(1) At night , SOMEONE and SOMEONE step into the parking lot.

(2) Now the van drives away.

(3) They drive away.

(4) They drive off.

(5) They drive off.

(6) At the end of the street , SOMEONE sits with his eyes closed.

S2VT (Ours):

(1) Now, the van pulls out a window and a tall brick facade of tall trees . a figure stands at a curb.

(2) Someone drives off the passenger car and drives off.

(3) They drive off the street.

(4) They drive off a suburban road and parks in a dirt neighborhood.

(5) They drive off a suburban road and parks on a street.

(6) Someone sits in the doorway and stares at her with a furrowed brow.

DVS:

(1) Now , at night , our view glides over a highway its lanes glittering from the lights of traffic below.

(2) Someone's suv cruises down a quiet road.

(3) Then turn into a parking lot .

(4) A neon palm tree glows on a sign that reads oasis motel.

(5) Someone parks his suv in front of some rooms.

(6) He climbs out with his briefcase , sweeping his cautious gaze around the area.

References

<https://www.cs.utexas.edu/users/ml/papers/venugopalan.iccv15.pdf>
<https://gist.github.com/vsubhashini/38d087e140854fee4b14>

5. Chat-bot

Tuteur(s) : Liming.Chen@ec-lyon.fr
Contact : Liming.Chen@ec-lyon.fr

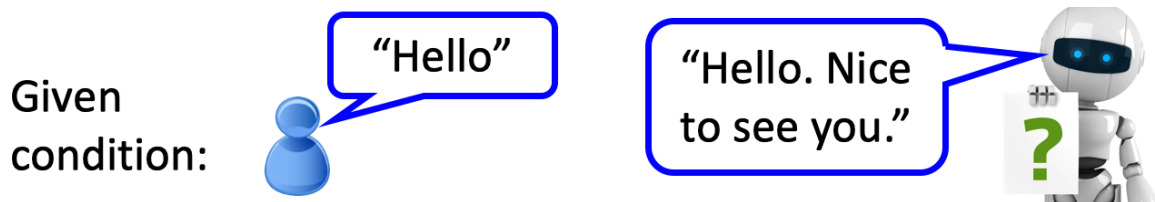
A generative ChatBot based on Sequential Neural Network and Deep Learning which can be trained on any desired dataset for specific purposes. Instead of ordinary ChatBots which are based on hard-coded responses, it can understand context and respond accordingly.

"Victor", as an instance, represents one of the several possibilities that are possible by training a sequential neural network. Going into the specifications, it is constructed of three layers of the sequential neural network each containing 128 neurons. The encoder and decoder have a vocabulary size of 20,000 each. The corpus used for training is a collection of dialogues from 617 movies containing about 220,000 conversations (Cornell Movie Dialogue Corpus).

The real "catch" about Victor is that it can be trained on any data we like, it can learn however we want it to. It gives contextual answers instead of hard-coded responses. It draws information from previous conversational exchanges and is thus, closer to how humans interact. The ability to be trained on any data is not only limited to English, it can be trained in any language we want it to any that is the real beauty of it.

The bot performs extremely well on casual conversations that are prevalent in movies given the fact that we had a relatively short time to train it and thus, several parameters had to be compromised (number of neurons in each layer etc.). Talking about future improvements, we can train it on even a bigger dataset and use more layers or more number of neurons in each layer. Also, we could add the feature of training it simultaneously while using it which would further increase its accuracy with experience.

Topic A generative ChatBot based on Sequential Neural Network and Deep Learning



References

<https://github.com/codedak/chatbot>

6. Text Mining et analyse de sentiments ET / OU Conversation Chatbot*

Tuteur(s) : Alexandre.Saidi@ec-lyon.fr

Contact : Alexandre.Saidi@ec-lyon.fr

On souhaite étudier le flot d'informations textuelles (Twittes, avis de consommateurs, ...) s'exprimant sur un sujet (une émission de télé, produits d'un site, avis sur un restaurant, ...) pour en dégager le sentiment (l'opinion, polarité de l'avis – positive/négative/neutre).

Le but sera ensuite de

- prédire un gagnant possible de ViaVoice ! (une émission / un sujet similaire dont l'issue n'est pas connue)
- calculer une note (nombre d'étoiles) à partir du texte d'un avis (cf. Amazon)
- analyser plus profondément un texte pour dégager une tendance. Par exemple, un client qui compte changer de "crèmerie" et exprime ses sentiments/opinions en se livrant dans un texte : "je me demande si ma crèmière se moque pas de moi en me filant son 06 sur une plaque de beurre rancit ; pourtant, il lui restait bien un "Paris-Brest" en vitrine !".

Pour ce faire, on utilisera les outils modernes employés pour extraire de l'information d'un corpus textuel (e.g. Word2vect mais aussi toute autre technique, sans aller à l'analyse syntaxique).

* Un sujet différent et/ou une extension de ce sujet peut s'intéresser à la compréhension d'un Chatbot. Il s'agira de pouvoir tenir une conversation non triviale.

7. Textual Big Data : Application du "Deep Learning" au traitement des langues naturelles

Tuteur(s) : Alexandre.Saidi@ec-lyon.fr

Contact : Alexandre.Saidi@ec-lyon.fr

On souhaite étudier le Deep Learning (apprentissage profond = Réseaux de neurones adaptés) au problème traitement de (grande) base de données textuelles.

Une des techniques employées (par Google notamment) est la technique Word2Vect. On souhaite étudier et étendre cette technique aux phrases / documents.

Parmi les objectifs on peut citer : résumé automatique, classification, Question / Réponse, ..

Différents types de réseaux de Neurones (LSTM, Récurrents, ...) peuvent être également envisagés / employés. Ce projet s'attachera à faire le point sur ces outils.

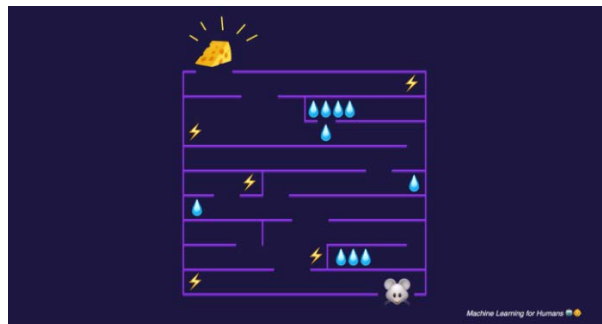
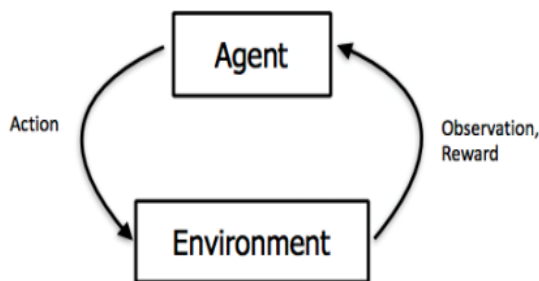
Mots-clés

BIG data, TextMining, Machine Learning, Reconnaissance de Formes.

8. Etude et réalisation d'un environnement d'apprentissage par renforcement

Tuteur(s) : Alexandre.Saidi@ec-lyon.fr
Contact : Alexandre.Saidi@ec-lyon.fr

Dans de multiples domaines, l'apprentissage par renforcement (*reinforcement Learning* : RL) est une technique d'apprentissage non supervisé sur la base du schéma simplifié (à gauche) suivant.



Par exemple, dans le labyrinthe (à droite) exploré par une souris, on cherche à réaliser un apprentissage (*faire apprendre à la souris*) afin d'obtenir le (maximum de) fromage. La souris peut recevoir une *récompense* maximale si elle atteint le fromage, une récompense moindre si elle accède à l'eau et une *punition* aux endroits avec une décharge électrique. Elle doit in fine et per elle-même apprendre à accéder à la nourriture et ne pas se faire électrocuter.

Ce type d'apprentissage a de nombreuses applications, en particulier en robotique où, au lieu de faire exécuter un algorithme prédéfini au robot, on procède à un apprentissage par renforcement qui est davantage dynamique et adaptatif.

Le but de ce projet est de

- Etudier le paradigme d'apprentissage par renforcement (de multiples références bibliographiques disponibles).
 - . Le *Markov Decision Processes* (MDPs) feront partie de cette étude en tant qu'outil classique utilisé pour la prise de décision en apprentissage par renforcement,
 - . En cas de progrès rapide, Il est possible de s'intéresser à l'étude de l'apprentissage par renforcement "profond" ainsi qu'aux différentes autres études récentes sur le sujet (Q-Learning, Policy Learning, Value Learning, Deep RL, ...).
- Réaliser une application visuelle (cf. figure en haut à droite) qui démontre, au fur et à mesure, la progression de l'agent (une souris, un robot, etc.) dans l'apprentissage de son environnement et de sa *mission*. A la fin de cet apprentissage, l'agent saura de manière efficace comment accéder à l'objectif le mieux récompensé (accès au fromage et à l'eau).

Le livrable l'étude bibliographique, le code de la démonstration, un rapport, ...

9. « Next-nextcloud » : mise à niveau et développement/déploiement d'un dropbox pour le secteur humanitaire

Commanditaire : CartONG (<http://cartong.org/>)
Camille Scheffler, c_scheffler@cartong.org
Martin Noblecourt, m_noblecourt@cartong.org



Tuteur(s) : René Chalon, bât E6, rene.chalon@ec-lyon.fr
Contact : René Chalon, bât E6, rene.chalon@ec-lyon.fr

Contexte

CartONG est une organisation non gouvernementale spécialisée dans la cartographie et la gestion de l'information pour les organismes humanitaires. L'association est basée à Chambéry et emploie des personnes un peu partout dans le monde, d'où l'intérêt d'un outil centralisé pour tous et partout.

L'association fonctionne pour l'instant avec l'outil Nextcloud (<https://nextcloud.com/>) qui permet de centraliser, de partager et synchroniser en interne des documents et des agendas. Il s'agirait donc de l'améliorer en ajoutant des fonctionnalités telles que l'édition de document en commun, la possibilité de communication via un chat, et d'autres fonctionnalités que les étudiants auront également défini comme utile dans leurs réflexions préalables.

Objectifs de réalisation

L'objectif est d'optimiser les fonctionnalités de Nextcloud afin de le transformer en véritable intranet en identifiant des outils complémentaires pouvant être connectés à Nextcloud.

Pour ce faire il sera nécessaire de passer par une étape de débogage du Nextcloud, certains bugs ayant déjà été identifiés par CartONG et compilés dans une fiche qui sera fournie aux étudiants.

Une première phase devra identifier des outils open source pouvant être intégrés à Nextcloud en évaluant la complexité d'intégration. Deux outils sont demandés prioritairement :

- Édition collaborative de documents en ligne
- Communication via un chat

Dans une deuxième phase, les outils devront être installés sur un serveur de développement afin de tester et d'évaluer la faisabilité et l'utilisabilité (PoC=Proof of Concept). Des développements logiciels complémentaires peuvent être envisagés pour permettre l'intégration mais il ne s'agit pas ici de développer des outils de toute pièce.

Une importance sera donnée à la sécurité (niveau de permissions/accès) ainsi qu'à la documentation technique pour permettre l'installation et l'administration en interne par l'ONG avec des ressources limitées.

La mise en place d'un système de backup serait également très appréciée et utile, mais reste optionnelle en fonction de l'avancement des étudiants sur le projet.

Livrables

- Rapport final présentant les solutions envisagées, les choix effectués (et argumentés), la mise en œuvre du PoC en détaillant les étapes d'installation et les problèmes rencontrés.
- Un serveur de test opérationnel
- Une documentation technique détaillant l'installation complète et le paramétrage du serveur
- Une documentation technique sur l'administration du serveur
- Le code source des développements effectués et la documentation technique associée.

Moyens mis à disposition pour la réalisation du travail

- Une personne de CartONG sera là tout au long du projet pour répondre aux questions et assurer le suivi
- Un expert technique sera également présent pour soutenir les élèves sur le plan technique.
- L'ONG mettra à disposition un serveur de test.

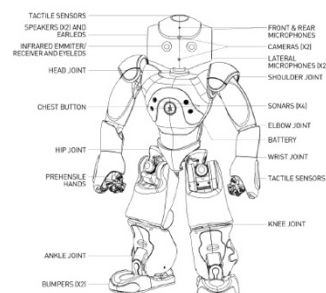
10. Interactions homme / robot humanoïde par la voix

Commanditaire : LIRIS, UMR 5205 CNRS, <https://liris.cnrs.fr/>

Tuteur(s) : Emmanuel Dellandrea, Emmanuel.Dellandrea@ec-lyon.fr

Maxime Petit, Maxime.Petit@ec-lyon.fr

Contact : Emmanuel Dellandrea, Emmanuel.Dellandrea@ec-lyon.fr



L'objectif de ce projet est de doter le robot humanoïde Nao¹ de la capacité à interagir avec un humain en utilisant la voix. Des projets d'étudiants de l'année dernière ont permis de développer un système permettant à Nao de reproduire le mouvement d'un humain capté par une caméra Kinect, ainsi que de reconnaître son expression faciale. L'objectif ici est donc de consolider cette interaction en apportant la modalité voix.

Une étude des différentes solutions technologiques permettant la reconnaissance vocale avec les capteurs actuellement disponibles (micros Nao, Kinect, ordinateur) devra être réalisée dans un premier temps, puis une fois la solution la plus adaptée retenue, le développement du système d'interaction devra être réalisé.

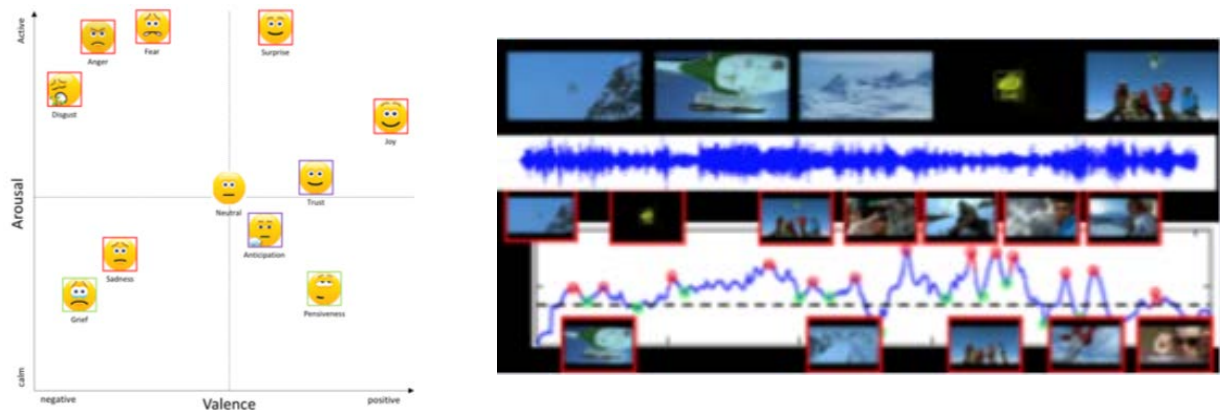
¹ <https://www.ald.softbankrobotics.com/fr/robots/nao>

11. Deep Learning pour la prédiction de l'émotion induite par les vidéos

Commanditaire : LIRIS, UMR 5205 CNRS, <https://liris.cnrs.fr/>

Tuteur(s) : Emmanuel Dellandrea, Emmanuel.Dellandrea@ec-lyon.fr

Contact : Emmanuel Dellandrea, Emmanuel.Dellandrea@ec-lyon.fr



La reconnaissance de l'émotion induite par les vidéos consiste en l'identification de l'émotion ressentie par une personne regardant une vidéo. Cela a de nombreuses applications possibles telles que la recommandation de vidéos basée sur l'émotion, l'indexation de vidéos, le filtrage de vidéos...

Les techniques d'apprentissage profond (ou Deep Learning) ont permis de réaliser des progrès spectaculaires pour la reconnaissance de nombreux concepts visuels tels que personne, visage, chat, voiture, ... L'étape suivante est de s'intéresser à des notions plus subjectives et de niveau sémantique plus élevé, telles que les émotions.

Ainsi, l'objectif de ce projet est d'étudier et d'adapter les principes du Deep Learning et des réseaux récurrents à la reconnaissance de l'émotion induite par les vidéos.

12. Analyse de trajectoires GPS et Visualisation

Tuteur(s) : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr

Contact : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr

Des milliers d'étudiants et personnels se déplacent tous les jours vers le campus de l'ECL. Nous souhaitons, pendant ce projet, comprendre ces trajectoires, en réalisant leur cartographie (comme illustrées Figure 1) et en essayant de proposer de nouvelles analyses. En particulier nous nous focaliserons sur les données collectées par Google que tout le monde peut exporter depuis son téléphone via les archives Google Takeout <https://takeout.google.com/settings/takeout>

Lors de ce projet vous êtes invités à générer vos propres trajectoires, soit en utilisant votre smartphone, soit en utilisant des capteurs GPS que nous mettons à votre disposition, et à utiliser les outils que vous développerez pour mieux les comprendre et en proposer de nouvelles analyses et formes de communication visuelle. Vous vous baserez sur des infrastructures d'analyse existantes de ces trajectoires (code Python) illustrées Figure 1.

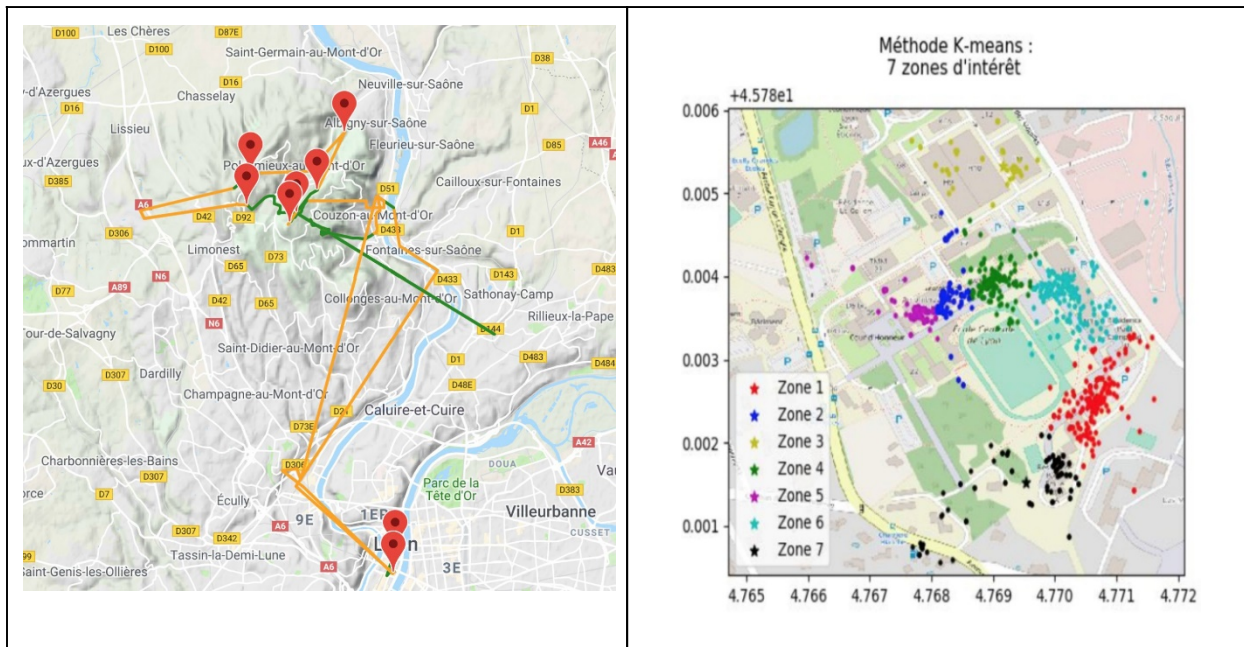


Figure 1. A gauche, exemple de visualisation de trajectoires depuis Lyon vers le campus issues de <https://github.com/AmigoCap/GPSFlow> et le clustering de positions sur le campus issues de <https://github.com/AmigoCap/CarToMob/>. Ces données ont été extraites de téléphones GPS classiques d'étudiants de l'ECL depuis l'export d'archive Google Takeout.

Etapes du projet

- Récupérer des données de déplacement GPS disponibles dans Google Takeout <https://takeout.google.com/settings/takeout> et proposer des affichages dans Open Street Map (<http://wiki.openstreetmap.org/wiki/Planet.gpx>) de la position, vitesse, etc.
- Analyser les données récoltées au moyen de visualisations brutes (Figure 1, gauche) cette fois-ci en Python (en stockant les données sous forme de DataFrame Pandas)
- Proposer des méthodes de filtrage de données [Yu, 2015] permettant de grouper les routes similaires et faire émerger des flots en fonction des horaires ou jours de la semaine vers le campus de l'ECL
- Identifier la modalité de déplacement sur ces routes (train ? vélo ? voiture ? à pied ?) de manière automatique ; comparer vos résultats avec Google Takeout qui propose également une prédiction.
- Proposer de nouvelles formes de représentations de ces trajectoires (ex: utilisation de matrices origines destinations [Dykes, 2010], conversion trajectoires en calories ou coûts de consommation d'essence, ..)

Livrables

Pour chacune de ces étapes, nous attendons un détail des analyses de données et représentation graphique sous forme de Notebook Python. Le langage Python sera en effet le langage privilégié pour ce projet de part son usage relativement facile, et les bibliothèques existantes pour le chargement, nettoyage, analyse et représentation des données. Il sera possible d'utiliser d'autres outils soit clés en main orientés GIS (OSM, ArcGIS), soit créer vos propres outils (JavaScript, D3.js).

Références

[Yu, 2015] Zheng, Yu. "Trajectory data mining: an overview." ACM Transactions on Intelligent Systems and Technology (TIST) 6.3 (2015): 29.

[Dykes, 2010] Flow Trees for Exploring Spatial Trajectories. Jo Wood, Jason Dykes, Aidan Slingsby and Robert Radburn

<https://pdfs.semanticscholar.org/316e/b1db18209cbc47f030cfda7ef81f0b4ea2ad.pdf>

13. GANs (Deep Learning) pour la génération de Visualisation de Données

Tuteur(s) : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr, <http://romain.vuillemot.net/> Théo Jaunet, <https://theo-jaunet.github.io/aboutme/>

Contact(s) : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr

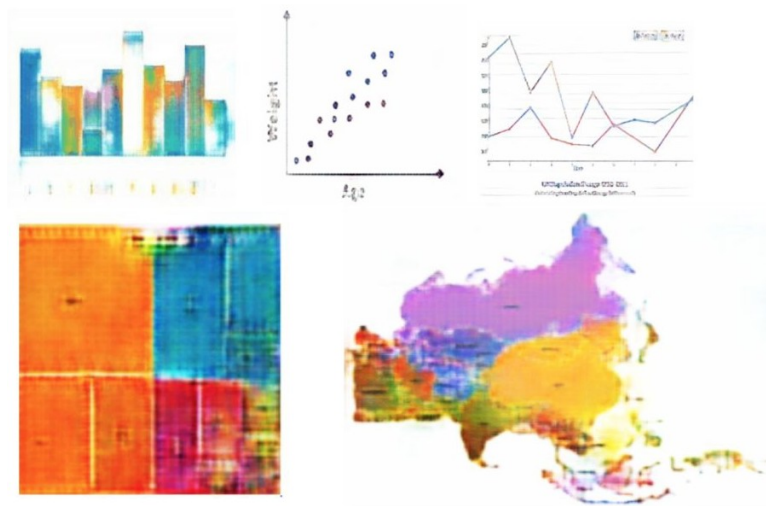
Un generative adversarial network (GAN) est un catégorie de modèle de deep learning, qui a été en mesure de produire des images photoréalistes, souvent impossibles à distinguer des images réelles.

Mais également des travaux plus créatif tel que le transfert de style comme :

<https://arxiv.org/pdf/1703.10593.pdf>.

Ce projet se concentre sur les possibilités que les GANs peuvent apporter au domaine de la visualisation de données (data visualization) et générer des images comme sur la Figure 1. Générer automatiquement de telles visualisations permettrait de rapidement prototyper des dashboards ou des interfaces complexes d'analyse de données.

Figure 1 : Exemples de visualisations générées à partir de GANs: bar chart, scatterplots, line chart, treemap et cartes géographiques.



Ce projet se décompose en plusieurs étapes :

1. **Création d'un jeu de données** de visualisation comprenant 3-4 type différents (e.g., bar chart, line chart, scatter plot, pie chart, etc ...) en étendant un premier jeu réalisé dans <https://github.com/AmigoCap/VizTransfert>
2. **Se familiariser avec le framework pytorch et les GANs** notamment modèle pix2pix <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>
3. **Entraîner un modèle pix2pix** pour générer des viz avec des conditions spécifiques
4. **Enfin, créer une interface web** pour interagir avec le modèle entraîné comme : <http://qandissect.res.ibm.com/qanpaint.html?project=churchoutdoor&layer=layer4> ou <https://affinelayer.com/pixsrv/>

Compétences techniques

- Très bonnes connaissances de Python
- Éventuellement Pytorch, JavaScript, et Fflask
- Goût pour le design et l'exploration de données

14. Visualisation de Données en Réalité Virtuelle

Tuteur(s) : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr, <http://romain.vuillemot.net/> Théo Jaunet, <https://theo-jaunet.github.io/aboutme/>

Contact(s) : Romain Vuillemot, Romain.Vuillemot@ec-lyon.fr

Les casques de réalité virtuelle offrent un nouvel espace de navigation et d'interaction avec des données. La plateforme Amigo du LIRIS dispose de tels casques comme le Samsung Odyssey (Figure 1 gauche) et d'un ordinateur portable Alienware haute performance.

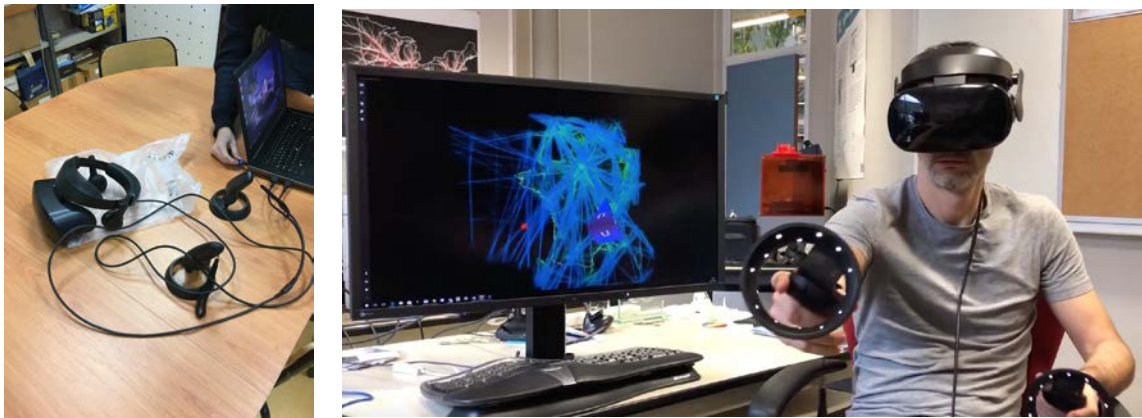


Figure 1. Casque Samsung Odyssey et vidéo de démonstration de son utilisation pour l'exploration de données 3D d'avions : <https://www.youtube.com/watch?v=8j9r9JKlqww>

L'objectif de ce projet est de développer de nouvelles représentations 3D de données pour ces environnements immersifs. En particulier nous souhaitons mettre au point de nouvelles représentations de données complexes telles que les données geo-spatiales (ex : représentation de distances de parcours dans une ville) mais aussi abstraites (ex : séries temporelles, graphes de réseaux sociaux, ..).

Il est nécessaire de former un groupe qui possède de très bonnes connaissances de programmation, idéalement en C# car il s'agira de programmer le framework Unity 3D. Le groupe doit aussi avoir un fort intérêt pour le design et l'analyse de données.

Le matériel sera mis à disposition bâtiment D5 et il sera possible d'accéder à la salle pendant les horaires de projets de l'option, mais également en dehors.

Objectifs

1. Définir un cahier des charges sur les visualisations à réaliser
2. Prendre en main l'environnement de développement Unity3D
3. Créer des premières images et les afficher en 3D
4. Générer de nouvelles visualisations 3D
 - a. A partir de cartes géographiques
 - b. A partir de visualisations abstraites multi-dimensionnelles
5. Proposer de nouvelles interactions avec ces visualisations
6. Documenter et réaliser un démonstrateur
7. Mettre le code en open source sur GitHub

Références

FiberClay: Sculpting Three Dimensional Trajectories to Reveal Structural Insights, Christophe Hurter, Nathalie Henry Riche, Steven Drucker, Maxime Cordeil, Richard Alligier, Romain Vuillemot. PDF: <http://romain.vuillemot.net/publis/infovis18-fiberclay.pdf>