



Département d'informatique
IFT 780 – Réseaux neuronaux
Plan d'activité pédagogique
Hiver 2024

Enseignant

Pierre-Marc Jodoin

Courriel : Pierre-Marc.Jodoin@USherbrooke.ca

Local : D4-1016-1

Téléphone : +1 819 821-8000 x62025

Disponibilités : Du lundi au vendredi de 9 h à 17 h

Responsable(s) : Pierre-Marc Jodoin, Shengrui Wang

Site web du cours : <https://info.usherbrooke.ca/pmjodoin/cours/ift780/>

Horaire

Exposé magistral :	Lundi	14h30 à 15h20	salle À venir
	Mardi	10h30 à 12h20	salle À venir

Description officielle de l'activité pédagogique¹

Cibles de formation : Connaître et comprendre plusieurs types de réseaux de neurones. Savoir les implanter, les entraîner et analyser leur performance. Savoir lire, comprendre, synthétiser et présenter des travaux d'articles scientifiques sur les réseaux de neurones. Pouvoir reproduire les résultats d'un article scientifique ou concevoir un nouveau réseau de neurones puis évaluer sa performance.

Contenu : Apprentissage supervisé par réseaux de neurones : classification et régression avec réseaux à propagation avant et prédiction de cibles. Réseaux de neurones classiques : perceptron multicouches et régression logistique. Réseaux à convolution et architectures profondes (deep learning) modernes : VGG, InceptionNet, ResNet, UNet, etc. Applications à l'imagerie : reconnaissance, segmentation, localisation, transfert de style, etc. Réseaux de neurones récurrents et applications à l'analyse de texte. Modèles génératifs adversaires et réseaux de neurones non supervisés : auto-encodeurs et auto-encodeurs variationnels. Bonnes pratiques : transfert d'entraînement, augmentation de données, normalisation, méthodes d'entraînement modernes, visualisation. Concepts avancés : modèles d'attention, autoML, compression, convolution dilatées.

Crédits 3

Organisation 3 heures d'exposé magistral par semaine
6 heures de travail personnel par semaine

Particularités Ce cours fait suite à IFT603-712 -Techniques d'apprentissage. Si vous n'avez pas suivi ce cours (ou équivalent) il est fortement suggéré d'en étudier le contenu.

¹<https://www.usherbrooke.ca/admission/fiches-cours/ift780>

1 Présentation

Cette section présente les cibles de formation spécifiques et le contenu détaillé de l'activité pédagogique. Cette section, non modifiable sans l'approbation du comité de programme du Département d'informatique, constitue la version officielle.

1.1 Mise en contexte

Ce cours fait écho à l'essor des réseaux de neurones profonds. De nos jours, l'apprentissage automatique joue un rôle central dans de nombreux domaines d'applications, tels que la vision par ordinateur, le traitement automatique du langage, la reconnaissance vocale, les systèmes tutoriels intelligents, la modélisation de l'utilisateur, la robotique, la bio-informatique, les finances, le marketing, les jeux vidéo, la télédétection, etc. Ce cours porte sur la conception, l'analyse (pratique et théorique), l'implémentation et l'application de réseaux de neurones capables d'apprendre leur propre représentation grâce à des données étiquetées.

1.2 Cibles de formation spécifiques

Ce cours vise à introduire les concepts fondamentaux des réseaux de neurones et à présenter une vaste gamme d'approches différentes utilisées dans des systèmes d'apprentissage automatique. À la fin de cette activité pédagogique, l'étudiante ou l'étudiant sera capable de :

1. Comprendre les notions fondamentales des réseaux de neurones ;
2. Comprendre les fondements mathématiques en lien avec leur fonctionnement ;
3. Maîtriser des modèles dédiés à l'analyse d'images ;
4. Comprendre le fonctionnement des modèles dédiés à l'analyse de texte ;
5. Maîtriser les concepts sous-jacents aux modèles génératifs ;
6. Comprendre les limites des réseaux de neurones ;
7. Comprendre comment analyser et visualiser la représentation d'un réseau de neurones ;
8. Connaître les architectures neuronales parmi les plus utilisées ;
9. Apprendre à entraîner un réseau de neurones dans un contexte d'application réelle.

1.3 Contenu détaillé

Thème	Contenu	Nbr. d'heures	Objectifs	Travaux	Lectures
1	Concepts de base (rappel du cours IFT603-712 et un peu plus) : <ul style="list-style-type: none"> • Perceptron et Perceptron multi-couches • Régression logistique, fonctions d'activation • Fonctions de perte, "Softmax" • Propagation avant, rétro-propagation et dérivée en chaîne • "Dropout" • Descentes de gradient stochastique et ses variantes • Vectorisation 	4	1, 2, 9	✓	6.1, 6.2, 6.3 et 6.5 de [10]
2	Réseaux à convolution : <ul style="list-style-type: none"> • Convolution et partage de poids • Cartes de caractéristiques • Notions diverses (tenseur, foudée, dilatation, étirement, etc.) • Calcul de la taille d'un réseau • Aggrégation • Réseaux entièrement convolutifs • Structure d'un réseau à convolution • Rétro-propagation et multiplication matricielle parcimonieuse • Convolution 1D, 2D, 3D et applications à l'analyse de signaux 1D, 2D, 3D 	4	1, 2, 3, 9	✓	9.0 à 9.5 de [23]
3	Réseaux à convolution avancés et architectures convolutives modernes : <ul style="list-style-type: none"> • Normalisation par lot et par couche • Augmentation de données et transfert d'apprentissage • Ensemble de modèles • Différents types de convolutions (séparée, par groupe, mélangée, à trous, etc.) • Variations sur le thème du "dropout" • Architectures convolutive modernes 1D, 2D, 3D • Bonnes pratiques et ajustement des hyperparamètres 	5	1, 2, 3, 8	✓	
4	Segmentation et localisation : <ul style="list-style-type: none"> • Segmentation sémantique vs par instance • Segmentation et localisation par fenêtre • Architectures diverses (Encodeur-Décodeur, U-Net, etc) • Convolution transposée • Métriques d'évaluation 	5	2, 3, 8	✓	

5	<p>Réseaux récurrents :</p> <ul style="list-style-type: none"> • RNN, LSTM, GRU • Jetonisation (Word2Vec) • Modèles d'attention • Transformeurs • Applications à l'analyse et la génération de texte • Sous-titrage 	6	2, 3, 4, 8, 9		10.1, 10.2, 10.3 et 10.10 de [23]
6	<p>Modèles génératifs :</p> <ul style="list-style-type: none"> • Autoencodeurs • Autoencodeurs variationnels et inférence variationnelle • Modèles adversaires génératifs • Apprentissage des représentations • Modèles génératifs appliqués à la génération de texte et de signaux 1D, 2D, 3D 	4			14.1 à 14.5, 20.10.3, 20.10.4 de [23]
7	<p>Visualisation :</p> <ul style="list-style-type: none"> • Saillance par occultation • Saillance via rétropropagation et rétro-propagation guidée • Ascension du gradient • Projection 2D et 3D non linéaires • Applications à la synthèse de texture et transfert de style 	3	7		
8	<p>Considérations avancées :</p> <ul style="list-style-type: none"> • Apprentissage par renforcement • Réseaux neuronaux par graphe • Autres fonctions de coût (Dice, contrastive, focale, Tversky etc.) • Gradients automatiques • MLops • Grappes de calcul (CPU-GPU-TPU) • AutoML • Bonnes pratiques en IA • Réseau de diffusion • Éthique et réseaux de neurones 	5	9		11.4 de [23]

9	<p>Concepts connexes au choix :</p> <ul style="list-style-type: none"> • Auto-entraînement • Apprentissage actif • Adaptation de domaines • Divers modèles de GAN • Attaques adversaires • Réseaux de neurones robustes aux attaques adversaires • Apprentissage faible • Apprentissage décentralisé et distribué • Reconstruction 3D par apprentissage profond • Segmentation panoptique • Prédiction d'incertitude • Compression de réseaux • Recherche d'architectures neuronales • Apprentissage semi-supervisé • Traduction de texte • Méthodes d'optimisation avancées (recuit simulé, Newton, gradient conjugué, BFGS, etc.) • Apprentissage à partir de peu ou pas d'exemples ("FewShot learning" – "ZeroShot learning") • Super-résolution • Débruitage • Système questions et réponses • Réseaux de neurones éthiques • Réseaux de neurones à mémoire • Recherche d'architectures neuronales (NAS) 	3	2, 3, 5, 6, 7, 8, 9		
---	---	---	---------------------	--	--

1. Le cours doit comprendre au moins trois travaux pratiques couvrant tous les sujets marqués «✓» dans le tableau.
2. Les lectures indiquées ne sont là qu'à titre indicatif. L'enseignant est libre de choisir un autre document de référence.

2 Organisation

Cette section propre à l'approche pédagogique de chaque enseignante ou enseignant présente la méthode pédagogique, le calendrier, le barème et la procédure d'évaluation ainsi que l'échéancier des travaux. Cette section doit être cohérente avec le contenu de la section précédente.

2.1 Méthode pédagogique

Les périodes de cours hebdomadaires serviront aux exposés théoriques et aux travaux dirigés. Dès la première semaine de cours, des équipes seront créées. Chaque équipe devra choisir, avant la date limite mentionnée ci-bas, un sujet de présentation orale. L'objectif de la présentation orale est d'explorer à fond un des sujets vus en classe. **Il est d'ailleurs fortement suggéré de vous inspirer de votre projet de recherche si toutefois vous en avez un.** Chaque travail pratique (TP) permettra d'explorer différents sujets vus en classes. Les lectures complémentaires sont mentionnées dans les notes de cours. Des instructions particulières seront données pour chacun des travaux. En ce qui concerne l'examen final, il est **récapitulatif** et seules des notes manuscrites sont admises.

La méthode pédagogique pour ce cours est similaire à celle des autres cours aux cycles supérieurs. Ainsi, à chaque semaine, sera dispensée une séance magistrale de 3 heures. Par contre, à quelques reprises, nous passerons 1 à 3 heures au laboratoire afin de se familiariser avec le code informatique.

Il est à noter que ce cours est à l'intention de personnes étudiantes ayant des **bases solides en techniques d'apprentissage** comme le cours IFT603-712 ainsi qu'en **programmation python**. Si vous n'avez jamais suivi un tel cours, il est fortement suggéré de vous y inscrire, ou à tout le moins, d'en réviser le contenu (<https://jodoin.github.io/cours/ift603/>). Une connaissance de **bash** est préférable.

2.2 Calendrier

Semaine	Date	Thème	Travaux pratiques
1	2024-01-08	1	
2	2024-01-15	1 et 2	
3	2024-01-22	2	
4	2024-01-29	2 et 3	
5	2024-02-05	3	Remise Travail pratique 1
6	2024-02-12	4	
7	2024-02-19	4 et 5	
8	2024-02-26	Examen périodique	
9	2024-03-04	Relâche	Remise Travail pratique 2
10	2024-03-11	5	
11	2024-03-18	5 et 6	Remise Travail pratique 3
12	2024-03-25	6	
13	2024-04-01	8 et 9	
14	2024-04-08	9	
15	2024-04-15	7 et 9	Remise Travail pratique 4
16	2024-04-22	Examen final	

Pas de cours la première semaine en raison du calendrier.

2.3 Évaluation

Travaux pratiques (4)	45 %
Présentation orale	10 %
Examen intra	20 %
Examen final	25 %

Détails à l'intention de tous les étudiantes et étudiants :

- Les devoirs doivent être effectués par **équipe de trois ou quatre**. Les équipes sont sujet à changement au courant de la session.

- Tout code informatique doit être rédigé dans le langage de **programmation Python**. Le code soumis doit être compatible avec la version de Python, c'est-à-dire exécutable sous, installée dans les laboratoires, sous Linux - Ubuntu ;
- Il est fortement recommandé de configurer votre **ordinateur personnel** et de vous en servir pour effectuer les travaux pratiques et le projet. Si vous avez **Windows**, il est suggéré d'installer VMware (gratuits pour les étudiantes et étudiants) et une **VM Ubuntu** (disponible dans le répertoire public). Dû aux technologies utilisées dans le cours, le débogage de l'environnement est difficile sur Windows.
- La qualité du français et de la présentation peut être considérée lors de l'évaluation des travaux ;
- Toute soumission en retard implique une **perte de 10 points par jour** (incluant samedi et dimanche), sauf celles motivées pour des raisons valables et conformes au règlement des études (par exemple, maladie avec attestation d'un médecin). Après 5 jours de retard, **la note sera de 0**.
- Les examens se font seuls en classe. L'intra est d'une durée de 1 h 50 et l'examen final d'une durée de 3 h 00. L'examen final est récapitulatif. Seules les feuilles manuscrites non photocopées sont acceptées pour toute documentation lors des examens.

IMPORTANT : afin d'éviter toute discrimination, les équipes des travaux pratiques seront formées par l'enseignant.

IMPORTANT : en plus de votre code, veuillez soumettre un fichier "gitlab.txt" dans lequel vous donnez le lien vers votre dépôt gitlab. Il est obligatoire d'utiliser gitLab (gitHub, Bitbucket, etc. sont interdits.)

IMPORTANT : en plus de vos documents de travail, vous devez remplir, signer et joindre à votre travail le **formulaire d'intégrité** disponible sur le site web du cours.

IMPORTANT : en plus de vos documents de travail, vous devez remplir et joindre à votre travail l'**agenda d'équipe** disponible sur le site web du cours.

Présentation orale

Chaque équipe aura une présentation orale à faire vers la fin du semestre. Votre présentation doit porter sur un sujet avancé. Pour vous inspirer, vous pouvez sélectionner un sujet parmi ceux mentionnés au thème 9.

IMPORTANT : deux équipes ne peuvent prendre **le même sujet**. Votre présentation doit également suivre les règles de présentation énoncées en classe. Dans tous les cas, le projet doit porter sur un domaine de l'apprentissage profond. Votre présentation doit démontrer que vous avez effectué une revue de littérature et approfondi un article en particulier.

Qualité du code et commentaires

On vous demande du code rédigé 100 % en anglais ou en français, mais pas de français ! Votre code doit aussi être bien documenté, respecter le **standard pep8** (votre IDE devrait fournir des outils pour formater votre code et vous avertir lors d'écarts du standard.) et respecter un standard uniforme pour la nomenclature des variables, des noms de fonctions et des noms de classes. Évitez également les variables « hard codées » empêchant l'utilisation de votre programme sur un autre ordinateur que le vôtre. **Si le code est jugé de qualité insuffisante, des points pourront être retirés.**

Gestion de projet

Comme tout projet qui se respecte, vous devez utiliser un gestionnaire de version de code pour vos travaux pratiques. On vous demande d'utiliser « **git** » via la plateforme « Gitlab » du département d'informatique. On s'attend également à ce que vous fassiez une bonne utilisation de git. Par exemple : évitez de « pousser » du code dans le master sans « merge », évitez les « méga » commits, etc. Une mauvaise utilisation de git peut entraîner **une importante perte de points**.

2.3.1 Qualité de la langue et de la présentation

Conformément à l'article 17 du règlement facultaire d'évaluation des apprentissages² l'enseignante ou l'enseignant peut retourner à l'étudiante ou à l'étudiant tout travail non conforme aux exigences quant à la qualité de la langue et aux normes de présentation.

2.3.2 Plagiat

Le plagiat consiste à utiliser des résultats obtenus par d'autres personnes afin de les faire passer pour sien et dans le dessein de tromper l'enseignante ou l'enseignant. Vous trouverez en annexe un document d'information relatif à l'intégrité intellectuelle qui fait état de l'article 9.4.1 du Règlement des études³. Lors de la correction de tout travail individuel ou de groupe une attention spéciale sera portée au plagiat. Si une preuve de plagiat est attestée, elle sera traitée en conformité, entre autres, avec l'article 9.4.1 du Règlement des études de l'Université de Sherbrooke. L'étudiante ou l'étudiant peut s'exposer à de graves sanctions qui peuvent être soit l'attribution de la note E ou de la note zéro (0) pour un travail, un examen ou une activité évaluée, soit de reprendre un travail, un examen ou une activité pédagogique. Tout travail suspecté de plagiat sera transmis au Secrétaire de la Faculté des

²https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Etudiants_actuels/Informations_academiques_et_reglements/2017-10-27_Reglement_facultaire_-_evaluation_des_apprentissages.pdf

³<https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

sciences. Ceci n'indique pas que vous n'avez pas le droit de coopérer entre deux équipes, tant que la rédaction finale des documents et la création du programme restent le fait de votre équipe. En cas de doute de plagiat, l'enseignante ou l'enseignant peut demander à l'équipe d'expliquer les notions ou le fonctionnement du code qu'elle ou qu'il considère comme étant plagié. En cas d'incertitude, ne pas hésiter à demander conseil et assistance à l'enseignante ou l'enseignant afin d'éviter toute situation délicate par la suite.

2.4 Échéancier des travaux

Les remises se font sur le système turnin (<https://turnin.dinf.usherbrooke.ca/>). Toute soumission en retard implique une **perte de 10 points par jour** (incluant samedi et dimanche), sauf celles motivées pour des raisons valables et conformes au règlement des études (par exemple, maladie avec attestation d'un médecin). Après 5 jours de retard, **la note sera de 0**.

Votre présentation en format PDF, PowerPoint ou Google slides doit avoir été transmise par turnin avant la date limite prévue à l'horaire.

Dans tous les cas, n'oubliez pas de soumettre le **formulaire d'intégrité** et l'**agenda d'équipe**.

2.5 Utilisation d'appareils électroniques et du courriel

Selon le règlement complémentaire des études, section 4.2.3⁴, l'utilisation d'ordinateurs, de cellulaires ou de tablettes pendant une prestation est interdite à condition que leur usage soit explicitement permise dans le plan de cours.

Dans ce cours, l'usage de téléphones cellulaires, de tablettes ou d'ordinateurs est autorisées. Cette permission peut être retirée en tout temps si leur usage entraîne des abus.

Tel qu'indiqué dans le règlement universitaire des études, section 4.2.3⁵, toute utilisation d'appareils de captation de la voix ou de l'image exige la permission de la personne enseignante.

Note : Je réponds aux questions posées par courriel à l'extérieur des périodes de cours.

Si vous avez des questions en dehors des heures de cours, veuillez m'envoyer un courriel pour déterminer un moment pour se rencontrer. Aussi, bien que les appareils électroniques soient acceptés en classe, les étudiantes et étudiants utilisant un ordinateur portable doivent s'installer à l'arrière afin de ne pas distraire le reste de la classe.

3 Matériel nécessaire pour l'activité pédagogique

Tout est disponible ici : <https://info.usherbrooke.ca/pmjodoin/cours/ift780/>

4 Références

- [1] BADRINARAYANAN, VIJAY AND KENDALL, ALEX AND CIPOLLA, ROBERTO : Segnet : A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.
- [2] BAHDAU, DZMITRY AND CHO, KYUNGHYUN AND BENGIO, YOSHUA : Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv :1409.0473*, 2014.
- [3] BROCK, ANDREW AND DONAHUE, JEFF AND SIMONYAN, KAREN : Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv :1809.11096*, 2018.
- [4] BROWN, TOM B AND MANN, BENJAMIN AND RYDER, NICK AND SUBBIAH, MELANIE AND KAPLAN, JARED AND DHARIWAL, PRAFULLA AND NEELAKANTAN, ARVIND AND SHYAM, PRANAV AND SASTRY, GIRISH AND ASKELL, AMANDA AND OTHERS : Language models are few-shot learners. *arXiv preprint arXiv :2005.14165*, 2020.
- [5] ÇIÇEK, ÖZGÜN AND ABDULKADIR, AHMED AND LIENKAMP, SOEREN S AND BROX, THOMAS AND RONNEBERGER, OLAF : 3D U-Net : learning dense volumetric segmentation from sparse annotation. *In International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.
- [6] CHARU C. AGGARWAL : *Neural Networks and Deep Learning*. Springer, 2019.

⁴https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Etudiants_actuels/Informations_academiques_et_reglements/Sciences_Reglement_complementaire.pdf

⁵<https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

- [7] CHEN, LIANG-CHIEH AND PAPANDREOU, GEORGE AND KOKKINOS, IASONAS AND MURPHY, KEVIN AND YUILLE, ALAN L : Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.
- [8] CHENG, JIANPENG AND DONG, LI AND LAPATA, MIRELLA : Long short-term memory-networks for machine reading. *arXiv preprint arXiv :1601.06733*, 2016.
- [9] CHO, KYUNGHYUN AND VAN MERRIËNBOER, BART AND GULCEHRE, CAGLAR AND BAHDANAU, DZMITRY AND BOUGARES, FETHI AND SCHWENK, HOLGER AND BENGIO, YOSHUA : Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*, 2014.
- [10] CHRISTOPHER M. BISHOP : *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006.
- [11] CIREGAN, DAN AND MEIER, UELI AND SCHMIDHUBER, JÜRGEN : Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3642–3649. IEEE, 2012.
- [12] DEVLIN, JACOB AND CHANG, MING-WEI AND LEE, KENTON AND TOUTANOVA, KRISTINA : Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*, 2018.
- [13] DOERSCH, CARL : Tutorial on variational autoencoders. *arXiv preprint arXiv :1606.05908*, 2016.
- [14] EDWARD RAFF : *Inside Deep Learning : Math, Algorithms, Models*. Manning, 2022.
- [15] GIRSHICK, ROSS : Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
- [16] GIRSHICK, ROSS AND DONAHUE, JEFF AND DARRELL, TREVOR AND MALIK, JITENDRA : Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [17] GOODFELLOW, IAN AND POUGET-ABADIE, JEAN AND MIRZA, MEHDI AND XU, BING AND WARDE-FARLEY, DAVID AND OZAIR, SHERJIL AND COURVILLE, AARON AND BENGIO, YOSHUA : Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [18] HE, KAIMING AND GKIOXARI, GEORGIA AND DOLLÁR, PIOTR AND GIRSHICK, ROSS : Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [19] HE, KAIMING AND ZHANG, XIANGYU AND REN, SHAOQING AND SUN, JIAN : Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [20] HOCHREITER, SEPP AND SCHMIDHUBER, JÜRGEN : Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [21] HOWARD, ANDREW G AND ZHU, MENGLONG AND CHEN, BO AND KALENICHENKO, DMITRY AND WANG, WEIJUN AND WEYAND, TOBIAS AND ANDREETTO, MARCO AND ADAM, HARTWIG : Mobilenets : Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv :1704.04861*, 2017.
- [22] HUANG, GAO AND LIU, ZHUANG AND VAN DER MAATEN, LAURENS AND WEINBERGER, KILIAN Q : Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [23] IAN GOODFELLOW, YOSHUA BENGIO ET AARON COURVILLE : *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [24] IOFFE, SERGEY AND SZEGEDY, CHRISTIAN : Batch normalization : Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.
- [25] ISENSEE, FABIAN AND JAEGER, PAUL F AND KOHL, SIMON AA AND PETERSEN, JENS AND MAIER-HEIN, KLAUS H : nnU-Net : a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.
- [26] KARRAS, TERO AND LAINE, SAMULI AND AILA, TIMO : A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.
- [27] KINGMA, DIEDERIK P AND WELLING, MAX : Auto-encoding variational bayes. *arXiv preprint arXiv :1312.6114*, 2013.
- [28] KRIZHEVSKY, ALEX AND SUTSKEVER, ILYA AND HINTON, GEOFFREY E : Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.
- [29] LECUN, YANN AND BOTTOU, LÉON AND BENGIO, YOSHUA AND HAFFNER, PATRICK : Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

- [30] LIU, WEI AND ANGUELOV, DRAGOMIR AND ERHAN, DUMITRU AND SZEGEDY, CHRISTIAN AND REED, SCOTT AND FU, CHENG-YANG AND BERG, ALEXANDER C : Ssd : Single shot multibox detector. *In European conference on computer vision*, pages 21–37. Springer, 2016.
- [31] LONG, JONATHAN AND SHELHAMER, EVAN AND DARRELL, TREVOR : Fully convolutional networks for semantic segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [32] LUONG, MINH-THANG AND PHAM, HIEU AND MANNING, CHRISTOPHER D : Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv :1508.04025*, 2015.
- [33] MIKOLOV, TOMAS AND CHEN, KAI AND CORRADO, GREG AND DEAN, JEFFREY : Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, 2013.
- [34] MILLETARI, FAUSTO AND NAVAB, NASSIR AND AHMADI, SEYED-AHMAD : V-net : Fully convolutional neural networks for volumetric medical image segmentation. *In 2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.
- [35] MIRZA, MEHDI AND OSINDERO, SIMON : Conditional generative adversarial nets. *arXiv preprint arXiv :1411.1784*, 2014.
- [36] PASZKE, ADAM AND CHAURASIA, ABHISHEK AND KIM, SANGPIL AND CULURCIELLO, EUGENIO : Enet : A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv :1606.02147*, 2016.
- [37] PLAUT, ELAD : From principal subspaces to principal components with linear autoencoders. *arXiv preprint arXiv :1804.10253*, 2018.
- [38] RADFORD, ALEC AND METZ, LUKE AND CHINTALA, SOUMITH : Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015.
- [39] RADFORD, ALEC AND NARASIMHAN, KARTHIK AND SALIMANS, TIM AND SUTSKEVER, ILYA : Improving language understanding by generative pre-training. *OpenAI Blog*, 2018.
- [40] RADFORD, ALEC AND WU, JEFFREY AND CHILD, REWON AND LUAN, DAVID AND AMODEI, DARIO AND SUTSKEVER, ILYA AND OTHERS : Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [41] REDMON, JOSEPH AND DIVVALA, SANTOSH AND GIRSHICK, ROSS AND FARHADI, ALI : You only look once : Unified, real-time object detection. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [42] REDMON, JOSEPH AND FARHADI, ALI : YOLO9000 : better, faster, stronger. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.
- [43] REN, SHAOQING AND HE, KAIMING AND GIRSHICK, ROSS AND SUN, JIAN : Faster r-cnn : Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015.
- [44] RONNEBERGER, OLAF AND FISCHER, PHILIPP AND BROX, THOMAS : U-net : Convolutional networks for biomedical image segmentation. *In International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [45] SIMONYAN, KAREN AND ZISSERMAN, ANDREW : Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.
- [46] SOHN, KIHYUK AND LEE, HONGLAK AND YAN, XINCHEN : Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28:3483–3491, 2015.
- [47] SRIVASTAVA, NITISH AND HINTON, GEOFFREY AND KRIZHEVSKY, ALEX AND SUTSKEVER, ILYA AND SALAKHUTDINOV, RUSLAN : Dropout : a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
- [48] SZEGEDY, CHRISTIAN AND LIU, WEI AND JIA, YANGQING AND SERMANET, PIERRE AND REED, SCOTT AND ANGUELOV, DRAGOMIR AND ERHAN, DUMITRU AND VANHOUCHE, VINCENT AND RABINOVICH, ANDREW : Going deeper with convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [49] VASWANI, ASHISH AND SHAZEER, NOAM AND PARMAR, NIKI AND USZKOREIT, JAKOB AND JONES, LLION AND GOMEZ, AIDAN N AND KAISER, ŁUKASZ AND POLOSUKHIN, ILLIA : Attention is all you need. *In Advances in neural information processing systems*, pages 5998–6008, 2017.
- [50] WANG, YI AND LUO, ZHIMING AND JODOIN, PIERRE-MARC : Interactive deep learning method for segmenting moving objects. *Pattern Recognition Letters*, 96:66–75, 2017.
- [51] XU, KELVIN AND BA, JIMMY AND KIROS, RYAN AND CHO, KYUNGHYUN AND COURVILLE, AARON AND SALAKHUTDINOV, RUSLAN AND ZEMEL, RICH AND BENGIO, YOSHUA : Show, attend and tell : Neural image caption generation with visual attention. *In International conference on machine learning*, pages 2048–2057. PMLR, 2015.
- [52] ZEILER, MATTHEW D AND FERGUS, ROB : Visualizing and understanding convolutional networks. *In European conference on computer vision*, pages 818–833. Springer, 2014.

L'intégrité intellectuelle passe, notamment, par la reconnaissance des sources utilisées. À l'Université de Sherbrooke, on y veille!

Extrait du Règlement des études (Règlement 2575-009)

9.4.1 DÉLITS RELATIFS AUX ÉTUDES

Un délit relatif aux études désigne tout acte trompeur ou toute tentative de commettre un tel acte, quant au rendement scolaire ou une exigence relative à une activité pédagogique, à un programme ou à un parcours libre.

Sont notamment considérés comme un délit relatif aux études les faits suivants :

- a) commettre un plagiat, soit faire passer ou tenter de faire passer pour sien, dans une production évaluée, le travail d'une autre personne ou des passages ou des idées tirés de l'œuvre d'autrui (ce qui inclut notamment le fait de ne pas indiquer la source d'une production, d'un passage ou d'une idée tirée de l'œuvre d'autrui);
 - b) commettre un autoplagiat, soit soumettre, sans autorisation préalable, une même production, en tout ou en partie, à plus d'une activité pédagogique ou dans une même activité pédagogique (notamment en cas de reprise);
 - c) usurper l'identité d'une autre personne ou procéder à une substitution de personne lors d'une production évaluée ou de toute autre prestation obligatoire;
 - d) fournir ou obtenir toute aide non autorisée, qu'elle soit collective ou individuelle, pour une production faisant l'objet d'une évaluation;
 - e) obtenir par vol ou toute autre manœuvre frauduleuse, posséder ou utiliser du matériel de toute forme (incluant le numérique) non autorisé avant ou pendant une production faisant l'objet d'une évaluation;
 - f) copier, contrefaire ou falsifier un document pour l'évaluation d'une activité pédagogique;
- [...]

Par plagiat, on entend notamment :

- Copier intégralement une phrase ou un passage d'un livre, d'un article de journal ou de revue, d'une page Web ou de tout autre document en omettant d'en mentionner la source ou de le mettre entre guillemets;
- reproduire des présentations, des dessins, des photographies, des graphiques, des données... sans en préciser la provenance et, dans certains cas, sans en avoir obtenu la permission de reproduire;
- utiliser, en tout ou en partie, du matériel sonore, graphique ou visuel, des pages Internet, du code de programme informatique ou des éléments de logiciel, des données ou résultats d'expérimentation ou toute autre information en provenance d'autrui en le faisant passer pour sien ou sans en citer les sources;
- résumer ou paraphraser l'idée d'un auteur sans en indiquer la source;
- traduire en partie ou en totalité un texte en omettant d'en mentionner la source ou de le mettre entre guillemets ;
- utiliser le travail d'un autre et le présenter comme sien (et ce, même si cette personne a donné son accord);
- acheter un travail sur le Web ou ailleurs et le faire passer pour sien;
- utiliser sans autorisation le même travail pour deux activités différentes (autoplagiat).

Autrement dit : mentionnez vos sources
