

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

OFFRE DE FORMATION MASTER

ACADEMIQUE

| Etablissement | Faculté / Institut | Département |
|---------------|--------------------|--------------|
| UMB – M'SILA | MI | INFORMATIQUE |

Domaine : Mathématiques Informatique

Filière : Informatique

Spécialité : Intelligence Artificielle

رئيس اللجنة البيداغوجية الوطنية
لميدان الرياضيات والإعلام الآلي
أ. د. شيبان عبد الحدين

Année universitaire : 2025- 2026

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

HARMONISATION

OFFRE DE FORMATION

MASTER ACADEMIQUE

| Etablissement | Faculté / Institut | Département |
|---------------|--------------------|--------------|
| UMB – M'SILA | MI | INFORMATIQUE |

Domaine : Mathématiques et informatique

Filière : Informatique

Spécialité : **Intelligence Artificielle**

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

مواصفة عرض تكوين

ماستر أكاديمي

| القسم | الكلية/ المعهد | المؤسسة |
|---------------|--------------------------|-----------------------------|
| الإعلام الألي | الرياضيات والإعلام الألي | جامعة محمد بوضياف - المسيلة |

الميدان: رياضيات واعلام الي

الشعبة: إعلام ألي

التخصص: ذكاء اصطناعي

SOMMAIRE

| | |
|--|-------|
| I - Fiche d'identité du Master | ----- |
| 1 - Localisation de la formation | ----- |
| 2 - Partenaires de la formation | ----- |
| 3 - Contexte et objectifs de la formation | ----- |
| A - Conditions d'accès | ----- |
| B - Objectifs de la formation | ----- |
| C - Profils et compétences visées | ----- |
| D - Potentialités régionales et nationales d'employabilité | ----- |
| E - Passerelles vers les autres spécialités | ----- |
| F - Indicateurs de suivi de la formation | ----- |
| G - Capacités d'encadrement | ----- |
| 4 - Moyens humains disponibles | ----- |
| A - Enseignants intervenant dans la spécialité | ----- |
| B - Encadrement Externe | ----- |
| 5 - Moyens matériels spécifiques disponibles | ----- |
| A - Laboratoires Pédagogiques et Equipements | ----- |
| B- Terrains de stage et formations en entreprise | ----- |
| C - Laboratoires de recherche de soutien au master | ----- |
| D - Projets de recherche de soutien au master | ----- |
| E - Espaces de travaux personnels et TIC | ----- |
| II - Fiche d'organisation semestrielle des enseignements | ----- |
| 1- Semestre 1 | ----- |
| 2- Semestre 2 | ----- |
| 3- Semestre 3 | ----- |
| 4- Semestre 4 | ----- |
| 5- Récapitulatif global de la formation | ----- |
| III - Programme détaillé par matière | ----- |
| IV – Accords / conventions | ----- |

I – Fiche d'identité du Master
(Tous les champs doivent être obligatoirement remplis)

1 - Localisation de la formation :

Faculté : Mathématiques et Informatique

Département : Informatique

2- Partenaires de la formation *:

- autres établissements universitaires :

- entreprises et autres partenaires socioéconomiques :

- Partenaires internationaux :

* = Présenter les conventions

3 – Contexte et objectifs de la formation

A–Conditions d'accès (*indiquer les spécialités de licence qui peuvent donner accès au Master*)

1. Licence LMD informatique SI (systèmes informatique)
2. Licence LMD informatique ISIL (ingénierie des systèmes d'information et logiciels)
3. Licence LMD informatique MIAGE (Méthodes Informatiques Appliquées à la Gestion des Entreprises)
4. Licence LMD informatique IAD (Informatique Appliquée au Droit)

B - Objectifs de la formation (*compétences visées, connaissances pédagogiques acquises à l'issue de la formation- maximum 20 lignes*)

Le Master en « Intelligence Artificielle » donne une connaissance approfondie des algorithmes, méthodes et techniques du domaine de l'intelligence artificielle. Cette formation est ouverte à tout diplômé d'une licence ayant acquis des modules fondamentaux (structures de données, théorie des langages, compilation, les systèmes d'exploitation, logique, et les réseaux de communication, etc.).

La spécialité Intelligence Artificielle permet de former :

- Des étudiants capables de comprendre et de développer des systèmes intelligents dans une grande variété de paramètres. Ils seront en mesure d'analyser, de concevoir et d'implémenter des solutions informatiques à des problèmes de plus en plus complexes en utilisant des méthodes avancées de représentation et de traitement de l'information.
- A l'issue de cette formation, l'étudiant sera capable d'intégrer des technologies d'apprentissage et d'extraire de l'information à partir d'une base de données de grande taille (Big data).

Les étudiants auront les capacités requises pour modéliser, analyser, concevoir et réaliser des systèmes intelligents qui s'adaptent à un environnement changeant; réaliser des systèmes où l'intelligence est distribuée entre de nombreux agents intelligents qui interagissent; raisonner sur le temps et l'espace; développer des systèmes automatiques complexes, autonomes et/ou embarqués où sont impliqués la perception (les capteurs), l'analyse, le raisonnement et la décision.

C – Profils et compétences métiers visés *(en matière d'insertion professionnelle - maximum 20 lignes) :*

Le parcours de la formation se caractérise par deux ans de formation M1 et M2, dans la première année M1, le cursus a pour objectif de donner une formation approfondie en mathématiques discrètes préparant directement aux métiers de la recherche fondamentales et appliquée au sens le plus large. Une spécialisation poussée en M2 est ensuite proposée, elle couvre essentiellement deux axes importants de mathématiques liés à l'informatique, dont la formation contient essentiellement :

Codage, analyse combinatoire, cryptographie, semi groupes et automates finis, informatique théorique, ... etc.

L'objectif de cette spécialité est de fournir aux étudiants un cursus visible et diversifié pour une formation à la recherche dans le domaine des mathématiques discrètes et leurs applications motivées par les problèmes liés aux codages, combinatoire et cryptographie.

D- Potentialités régionales et nationales d'employabilité

Le Master en « Intelligence Artificielle » donne une connaissance approfondie des algorithmes, méthodes et techniques du domaine de l'intelligence artificielle. Cette formation est ouverte à tout diplômé d'une licence ayant acquis des modules fondamentaux (structures de données, théorie des langages, compilation, les systèmes d'exploitation, logique, et les réseaux de communication, etc.).

La spécialité Intelligence Artificielle permet de former :

Des étudiants capables de comprendre et de développer des systèmes intelligents dans une grande variété de paramètres. Ils seront en mesure d'analyser, de concevoir et d'implémenter des solutions informatiques à des problèmes de plus en plus complexes en utilisant des méthodes avancées de représentation et de traitement de l'information.

A l'issue de cette formation, l'étudiant sera capable d'intégrer des technologies d'apprentissage et d'extraire de l'information à partir d'une base de données de grande taille (Big data).

Les étudiants auront les capacités requises pour modéliser, analyser, concevoir et réaliser des systèmes intelligents qui s'adaptent à un environnement changeant; réaliser des systèmes où l'intelligence est distribuée entre de nombreux agents intelligents qui interagissent; raisonner sur le temps et l'espace; développer des systèmes automatiques complexes, autonomes et/ou embarqués où sont impliqués la perception (les capteurs), l'analyse, le raisonnement et la décision.

E – Passerelles vers d'autres spécialités

Une passerelle est autorisée vers les autres spécialités. Toutefois, la passerelle peut être autorisée, après étude par l'équipe de formation, à partir d'un autre parcours jugé proche.

F – Indicateurs de suivi de la formation

L'étudiant aura l'occasion de suivre le module « Initiation à la méthodologie de recherche » au semestre 3. Au semestre 4, durant la préparation de son projet de fin de cycle, l'étudiant est orienté et suivi par un (des) encadreur(s) de l'équipe pédagogique. Si de plus, le thème du projet nécessite un stage en entreprise, un encadrement au sein de celle-ci et également assuré. Le travail est sanctionné à son terme par la rédaction d'un mémoire et une soutenance publique devant un jury spécialisé.

G – Capacité d'encadrement (donner le nombre d'étudiants qu'il est possible de prendre en charge)

De 20 à 30 étudiants.

4 – Moyens humains disponibles

A : Enseignants de l'établissement intervenant dans la spécialité :

4 – Moyens humains disponibles

A : Enseignants de l'établissement intervenant dans la spécialité :

| Nom, prénom | Diplôme graduation + Spécialité | Diplôme Post graduation + Spécialité | Grade | Type d'intervention * | Emargement |
|----------------------|----------------------------------|--------------------------------------|-------|-----------------------|------------|
| Hemmak Allaoua | Ingenieur d'état en Informatique | Doctorat en informatique | Prof. | Cours-TD-TP-Encad. | |
| Sayed Lamri | Ingenieur d'état en informatique | Doctorat en informatique | Prof. | Cours-TD-TP-Encad. | |
| Debbi Hicham | Master en informatique | Doctorat en informatique | Prof. | Cours-TD-TP-Encad. | |
| Kadri Said | Ingenieur d'état en informatique | Doctorat en informatique | Prof. | Cours-TD-TP-Encad. | |
| Meheenni Tahar | Ingenieur d'état en informatique | Doctorat en informatique | MCA | Cours-TD-TP-Encad. | |
| Brahimi Belkacem | Ingenieur d'état en informatique | Doctorat en informatique | MCA | Cours-TD-TP-Encad. | |
| Barkat Abdelabsset | Ingenieur d'état en informatique | Doctorat en informatique | MCA | Cours-TD-TP-Encad. | |
| Mokhtari Rabah | Ingenieur d'état en informatique | Doctorat en informatique | MCA | Cours-TD-TP-Encad. | |
| Bouguerra Abdelbaki | Ingenieur d'état en informatique | Doctorat en informatique | MCI | Cours-TD-TP-Encad. | |
| Chalabi Nour Elhouda | Master en informatique | Doctorat en informatique | MCD | Cours-TD-TP-Encad. | |
| Bougheragueddik | Ingenieur d'état en informatique | Doctorat en informatique | MCI | Cours-TD-TP-Encad. | |
| Amraoui Noureddine | Master en informatique | Doctorat en informatique | MCI | Cours-TD-TP-Encad. | |
| Mellouh Amel | Ingenieur d'état en informatique | Doctorat en informatique | MCS | Cours-TD-TP-Encad. | |
| Hammani Said | Ingenieur d'état en informatique | Doctorat en informatique | WAA | Cours-TD-TP-Encad. | |

* = Cours, TD, TP, Encadrement de stage, Encadrement de mémoire, autre (à préciser)

Etablissement : UMB – M'sila

Intitulé du master : Intelligence Artificielle

Année universitaire : 2025-2026

B : Encadrement Externe :

Etablissement de rattachement :

Etablissement : UMB – M'sila
Année universitaire : 2025-2026

intitulé du master : Intelligence Artificielle

| Nom, prénom | Diplôme graduation + Spécialité | Diplôme Post graduation + Spécialité | Grade | Type d'intervention * | Emargement |
|-------------|------------------------------------|---|-------|--------------------------|------------|
| | | | | | |
| | | | | | |
| | | | | | |

Etablissement de rattachement :

| Nom, prénom | Diplôme graduation + Spécialité | Diplôme Post graduation + Spécialité | Grade | Type d'intervention * | Emargement |
|-------------|------------------------------------|---|-------|--------------------------|------------|
| | | | | | |
| | | | | | |
| | | | | | |

Etablissement de rattachement :

| Nom, prénom | Diplôme graduation + Spécialité | Diplôme Post graduation + Spécialité | Grade | Type d'intervention * | Emargement |
|-------------|------------------------------------|---|-------|--------------------------|------------|
| | | | | | |
| | | | | | |
| | | | | | |

* = Cours, TD, TP, Encadrement de stage, Encadrement de mémoire, autre (à préciser)

5 – Moyens matériels spécifiques disponibles

A-Laboratoires Pédagogiques et Equipements : Fiche des équipements pédagogiques existants pour les TP de la formation envisagée (1 fiche par laboratoire)

Intitulé du laboratoire : Laboratoire d'Informatique et ses Applications de M'sila (LIAM)

| N° | Intitulé de l'équipement | Nombre | observations |
|----|----------------------------------|---------|-------------------------|
| 01 | Micro-ordinateur (pour Windows) | 360 | 20 salles |
| 02 | Micro-ordinateur (avec Internet) | 72 | 04 salles |
| 03 | Micro-ordinateur (pour Unix) | 1 X 40 | Une salle |
| 04 | Micro-ordinateur (pour Windows) | 2 X 30 | 02 Salles avec Internet |
| 05 | Salle multimédia | 16 x 04 | 04 salles |


B- Terrains de stage et formation en entreprise:

| Lieu du stage | Nombre d'étudiants | Durée du stage |
|---------------|--------------------|----------------|
| Condor | 20 | 10 jours |
| Lafarge | 20 | 10 jours |
| Kaizen | 20 | 10 jours |
| Ooredoo | 20 | 10 jours |

C- Laboratoire(s) de recherche de soutien au master :

C- Laboratoire(s) de recherche de soutien au master :

Laboratoire d'Informatique et ses Applications de M'sila (LIAM)

| | |
|--|--|
| Chef du laboratoire : Hemmak Allaoua | |
| N° Agrément du laboratoire : N° 05 du 10/02/2020 | |
| Date : 05/03/2026 |  |
| Avis du chef de laboratoire : Favorable | |

D- Projet(s) de recherche de soutien au master :

| Intitulé du projet de recherche | Code du projet | Date du début du projet | Date de fin du projet |
|---------------------------------|----------------|-------------------------|-----------------------|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

E- Espaces de travaux personnels et TIC :

- Salle centrale des TIC
- Salle de TP de département

Etablissement : Université Mohamed Boudiaf de M'sila Intitulé du master : Intelligence Artificielle
Année universitaire : 2025-2026

II – Fiche d'organisation semestrielle des enseignements

(Prière de présenter les fiches des 4 semestres)

| Unité d'Enseignement | VHS | V.H hebdomadaire | | | | Coeff | Crédits | Mode d'évaluation | |
|--|--------|------------------|-----------|-------------|---------------|----------|-----------|-------------------|--------|
| | 14 sem | C | TD | TP | Travail pers. | | | Continu | Examen |
| UE Fondamentales | | | | | | | | | |
| UEF11 | | 3h | 3h | 3h | 6h | 6 | 10 | | |
| Algorithmique Avancée et Complexité | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| Bases de Données Avancées | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| UEF12 | | 4h30 | | 3h | 3h | 6 | 8 | | |
| Machine Learning | 63h | 3h | | 1h30 | 1h30 | 3 | 4 | 40% | 60% |
| Représentation des connaissances et Raisonnement | 42h | 1h30 | | 1h30 | 1h30 | 3 | 4 | 40% | 60% |
| UE Méthodologique | | | | | | | | | |
| UEM1 | | 3h | 3h | | 3h | 4 | 9 | | |
| Analyse de Données | 42h | 1h30 | 1h30 | | 1h30 | 2 | 5 | 40% | 60% |
| Méthodes d'optimisation | 42h | 1h30 | 1h30 | | 1h30 | 2 | 4 | 40% | 60% |
| UE Transversale | | | | | | | | | |
| | | | | | | | | | |
| UET1 | | 1h30 | | 1h30 | 1h30 | 1 | 2 | | |
| Réseaux avancés | 42h | 1h30 | | 1h30 | 1h30 | 1 | 2 | 40% | 60% |

| UE Découverte | | | | | | | | | |
|-------------------------|-------------|--------------|-----------|-------------|--------------|-----------|-----------|--|------|
| UED1 | | 1h30 | | | | 1 | 1 | | |
| Un module au choix | 21h | 1h30 | | | | 1 | 1 | | 100% |
| Total Semestre 1 | 378h | 13h30 | 6h | 7h30 | 13h30 | 18 | 30 | | |

2- Semestre 2 :

| Unité d'Enseignement | VHS | V.H hebdomadaire | | | | Coeff | Crédits | Mode d'évaluation | |
|-----------------------------|---------------|-------------------------|-------------|-------------|----------------------|--------------|----------------|--------------------------|---------------|
| | 14 sem | C | TD | TP | Travail pers. | | | Continu | Examen |
| UE Fondamentales | | | | | | | | | |
| UEF21 | | 3h | 1h30 | 3h | 4h30 | 6 | 9 | | |
| Deep Learning | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| Systèmes Multi-Agents | 42h | 1h30 | | 1h30 | 1h30 | 3 | 4 | 40% | 60% |
| UEF22 | | 3h | 3h | 1h30 | 4h30 | 6 | 9 | | |
| Vision par ordinateur | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| Gestion de l'incertain | 42h | 1h30 | 1h30 | | 1h30 | 3 | 4 | 40% | 60% |
| UE Méthodologique | | | | | | | | | |
| UEM2 | | 3h | 1h30 | 3h | 4h30 | 4 | 9 | | |
| Modélisation et simulation | 63h | 1h30 | 1h30 | 1h30 | 3h | 2 | 5 | 40% | 60% |
| Virtualisation et Cloud | 42h | 1h30 | | 1h30 | 1h30 | 2 | 4 | 40% | 60% |

| | | | | | | | | | |
|----------------------------------|-------------|-------------|-----------|-------------|-------------|-----------|-----------|-----|------|
| UE Transversale | | | | | | | | | |
| UET2 | | 1h30 | | 1h30 | 1h30 | 1 | 2 | | |
| Gestion de projets informatiques | 42h | 1h30 | | 1h30 | 1h30 | 1 | 2 | 40% | 60% |
| UE Découverte | | | | | | | | | |
| UETD2 | | 1h30 | | | | 1 | 1 | | |
| Un module au choix | 21h | 1h30 | | | | 1 | 1 | | 100% |
| Total Semestre 2 | 378h | 12h | 6h | 9h | 15h | 18 | 30 | | |

3- Semestre 3 :

| Unité d'Enseignement | VHS | V.H hebdomadaire | | | | Coeff | Crédits | Mode d'évaluation | |
|--|--------|------------------|-------------|-----------|---------------|----------|-----------|-------------------|--------|
| | 14 sem | C | TD | TP | Travail pers. | | | Continu | Examen |
| UE Fondamentales | | | | | | | | | |
| UEF31 | | 3h | 3h | 3h | 6h | 6 | 10 | | |
| Intelligence artificielle générative | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| Méta-heuristiques et algorithmes évolutionnaires | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| UEF32 | | 3h | 1h30 | 3h | 4h30 | 6 | 8 | | |
| Apprentissage par Renforcement | 63h | 1h30 | 1h30 | 1h30 | 3h | 3 | 5 | 40% | 60% |
| Reconnaissance des formes | 42h | 1h30 | | 1h30 | 1h30 | 3 | 3 | 40% | 60% |

| | | | | | | | | | |
|-------------------------------|-------------|-------------|-------------|-------------|-------------|-----------|-----------|-----|------|
| UE Méthodologique | | | | | | | | | |
| UEM3 | | 3h | 3h | 1h30 | 4h30 | 4 | 9 | | |
| Natural Language Processing | 63h | 1h30 | 1h30 | 1h30 | 3h | 2 | 5 | 40% | 60% |
| Programmation par Contraintes | 42h | 1h30 | 1h30 | | 1h30 | 2 | 4 | 40% | 60% |
| UE Transversale | | | | | | | | | |
| UET3 | | 1h30 | | | | 1 | 2 | | |
| Méthodologies de recherche | 21h | 1h30 | | | | 1 | 2 | | 100% |
| UE Découverte | | | | | | | | | |
| UED3 | | 1h30 | | | | 1 | 1 | | |
| Un module au choix | 21h | 1h30 | | | | 1 | 1 | | 100% |
| Total Semestre 3 | 378h | 12h | 7h30 | 7h30 | 15h | 18 | 30 | | |

4- Semestre 4 :

| | VHS | Coeff | Crédits |
|--------------------------------|---------------|-----------|-----------|
| PFE avec Mémoire | 750h00 | 18 | 30 |
| Stage dans l'entreprise | | | |
| Ateliers | | | |
| Travail Personnel | | | |
| Autres | | | |
| Total Semestre 4 | 750h00 | 18 | 30 |

5- Récapitulatif global de la formation : (indiquer le VH global séparé en cours, TD, pour les 04 semestres d'enseignement, pour les différents types d'UE)

| VH | UE | UEF | UEM | UED | UET | S4 | Total |
|------------------------------------|----|--------------|-------------|----------|-------------|------|--------------|
| Cours | | 273h | 147h | 63h | 63h | | 483h |
| TD | | 168h | 105h | | | | 273h |
| TP | | 231h | 105h | | 21h | | 357h |
| Mémoire | | - | - | - | - | 750h | 750h |
| Stage dans l'entreprise | | - | - | - | - | | |
| Ateliers | | - | - | - | - | | |
| Travail Personnel | | 399 | 168 | 0 | 42 | | |
| Autres | | - | - | - | - | | |
| | | | | | | | |
| Total | | 1071h | 525h | 63h | 126h | 750h | 1863h |
| Crédits | | 54 | 27 | 3 | 6 | 30 | 120 |
| % en crédits pour chaque UE | | | | | | | 100,00 |

Corbeille des modules de découverte au choix (UED)

- Informatique verte
- Gouvernance et transformation digitale
- Philosophie des sciences et de la technologie
- Droit du numérique et protection des données (RGPD)
- Psychologie cognitive
- Technologies émergentes (Blockchain, IOT, ...)
- Découverte des spécialités de master (une première partie qui permet de découvrir sommairement les autres spécialités du master et une seconde partie qui permet de découvrir en détails les applications potentielles de la spécialité en cours).

III - Programme détaillé par matière (1 fiche détaillée par matière)

Programme détaillé des enseignements du semestre 1 (S1)
Master académique

Spécialité : Intelligence Artificielle (Filière : Informatique)

Intitulé de la matière : Algorithmique avancée et complexité **Semestre :** 1 **Type :** UEF11
VHS : 63h **VHH :** 4h30 **Cours :** 1h30 **TD :** 1h30 **TP :** 1h30
VHS travail personnel : 42h **Coefficient :** 3 **Crédit :** 5

Objectifs de l'enseignement

Cette matière couvre des concepts fondamentaux et des techniques avancées en conception et analyse d'algorithmes en approfondissant les structures de données, les algorithmes de recherche et de tri, ainsi que des approches comme la programmation dynamique et les algorithmes gloutons. La matière fournit aussi une compréhension approfondie de la complexité algorithmique, en mettant l'accent sur les techniques d'analyse, l'importance de la complexité pour l'optimisation, et les méthodes pour prouver la correction des algorithmes.

Connaissances préalables recommandées

Algorithmique, structures de données avancées et programmation.

Contenu de la matière Cours : 21h

1-Introduction à la complexité algorithmique et notions préliminaires

- Définition d'algorithme et d'algorithmique.
- Notion de problème algorithmique et sa résolution.
- Pourquoi analyser la complexité ? Comparaison d'algorithmes pour un même problème.
- Notions de base et types de complexité : temporelle et spatiale.

2. Analyse de la complexité temporelle

- Notation grand O (O): Définition, interprétation, et utilité pour exprimer la complexité asymptotique.
- Calcul de la complexité:
- Identification des opérations élémentaires (comparaisons, affectations, etc.).
- Analyse des boucles et des fonctions récursives.
- Cas le pire, le meilleur, et moyen.
- Exemples de complexités courantes: $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$, etc.

3. Structures de données et complexité

- Tableaux, listes chaînées, arbres, graphes, Tas : opérations courantes et complexité associée.
- Impact de la structure de données sur la complexité d'un algorithme.

4. Techniques d'analyse de complexité

- Diviser pour régner: Stratégie de base, analyse de la complexité (exemple, tri par fusion).
- Programmation dynamique: Principe et optimisation (exemples : problème du sac à dos, plus court chemin).
- Algorithmes gloutons: Principes et analyse de la complexité. (exemples voyageur de commerce, algorithme de Dijkstra).

5. Complexité spatiale

- Mesure de l'utilisation de la mémoire par un algorithme.
- Lien avec la complexité temporelle et compromis possibles.

6. Complexité et Preuve de Correction

- Notions de P, NP, NP-complet: Définitions et exemples de problèmes NP-complets.
- Techniques de Preuve de Correction: Preuve par induction, preuves par contradiction.

7. Optimisation d'algorithmes:

- Utilisation des concepts de complexité pour identifier les points faibles d'un algorithme.
- Techniques d'optimisation (par exemple, simplification de boucles, réduction de la récursivité).

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) •

Examen semestriel en présentiel (60%).

- **Évaluation continue (CC) (40%)** : contrôle continu, travail personnel..

Références bibliographiques

-Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein (2009). Introduction to Algorithms. MIT Press, 3rd edition.

-Ivan Lavallée (2008). Complexité et algorithmique avancée : une introduction. Editions Hermann.

-Jon Kleinberg, Éva Tardos (2005). Algorithm Design. Pearson Publisher, 1st edition.

-Nicolas Hermann et Pierre Lescanne (2005). Est-ce que $P = NP$? Les Dossiers de La Recherche, 20:64–68, août-octobre..

-Robert Sedgewick, Kevin Wayne (2011). Algorithms, Addison-Wesley, 4th edition.

-Förg Flum, Martin Grohe. Parameterized Complexity Theory. Springer Verlag, 2006.

-Udi Manber 1989). Introduction to Algorithms: A Creative Approach. Addison-Wesley.

-J.J.McConnell (2001). Analysis of algorithms: an active learning approach. Jones and Barlett Publishers.

-Cormen, Leiserson, Rivest Stein (2010)., Algorithmique, cours exercices et problèmes, 3ème édition, Dunod.

-Sylvain Perifel (2014). Complexité algorithmique. Ellipses.

-Sanjeev Arora and Boaz Barak (2006). Computational Complexity: A Modern Approach.

Spécialité : Intelligence Artificielle (Filière : Informatique)

Intitulé de la matière : Bases de données avancées **Semestre :** 1 **Type :** UEF11 **VHS :** 63h
VHH : 4h30 **Cours :** 1h30 **TD :** 1h30 **TP :** 1h30
VHS travail personnel : 42h **Coefficient :** 3 **Crédit :** 5

Objectifs de l'enseignement

Cette matière, en l'accent sur les concepts et technologies essentiels pour la conception, la gestion et l'exploitation de systèmes de bases de données modernes, en abordant à la fois les aspects théoriques et pratiques, permet à l'étudiant d'actualiser et d'approfondir ses connaissances des bases de données.

Connaissances préalables recommandées

Concepts de bases sur les bases de données, Langage SQL, Algèbre relationnelle.

Contenu de la matière Cours : 21h

1-Introduction aux bases de données

- Rappels sur les concepts fondamentaux: modèle relationnel, SGBD, langages de requêtes.
- Types de bases de données: relationnelles, NoSQL, objets, etc.
- Architecture d'un SGBD: client-serveur, architectures distribuées.
- Concepts de transaction, concurrence, et récupération de données.

2-Programmation SQL avancée

- SQL avancé: jointures, sous-requêtes, fonctions d'agrégation, vues, procédures stockées.
- Les Triggers
- Les fonctions et procédures stockées
- Traitement et gestion des erreurs
- Langages de manipulation de données pour NoSQL: JSONiq, Cypher.

3-Le modèle Objet-Relationnel

- Présentation du modèle Objet
- Présentation du modèle Relationnel-Objet
- Concepts du modèle RO (types complexes, héritage...)
- Interrogation des BDD Relationnelles-Objet (SQL3)

4-Bases de données NoSQL

- Introduction aux bases de données NoSQL: types de bases de données (clés-valeurs, documents, graphes, colonnes).
- Modèles de données NoSQL: avantages et inconvénients.
- Études de cas: MongoDB, Cassandra, Neo4j.

5-Bases de données distribuées

- Architecture des systèmes distribués.
- Techniques de réplication et de partitionnement.
- Consistance des données dans un environnement distribué: ACID vs BASE.

6-Bases de données dans le cloud

- Modèles de service: IaaS, PaaS, SaaS.
- Cloud computing et bases de données.
- Considérations de sécurité dans le cloud.

7-Performance et sécurité des bases de données

- Optimisation des requêtes SQL.
- Indexation et stratégies d'accès.
- Sécurité: contrôle d'accès, chiffrement des données, audit.
- Gestion de la sécurité dans les bases de données distribuées.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) •

Examen semestriel en présentiel (60%).

- **Évaluation continue (CC) (40%)** : contrôles continus, travail personnel.

Références bibliographiques

-R. Elmasri & S. Navathe (2016). Fundamentals of Database Systems, 7th Edition, Pearson.

- **R. Elmesri, B. Navate (2016). Fundamentals of Database Systems. 7th edition, Pearson Editions**

-T. Connolly & C.Begg (2014). Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition, Pearson.

-Christian Soutou (2013). SQL pour Oracle. Editions Eyrolles.

-Silberschatz, H. Korth & S. Sudarshan (2019). Database System Concepts, 7th Edition, McGraw-Hill.

-Mohamed Fadhel SAAD (2016). PL/SQL sous Oracle 12c. Guide du développeur.

-R.G.G. Cattell (1994). Object Data Management. Addison-Wesley.

-P. Selmer (2012). NOSQL stores and Data analytics tools. Advances in Data Management.

Spécialité : Intelligence Artificielle (Filière : Informatique)

Intitulé de la matière : Machine Learning **Semestre : 1 Type : UEF2**
VHS : 63 **VHH : 4h30** **Cours : 1h30** **TD : 1h30** **TP : 1h30** **VHS travail**
personnel : 42h **Coefficient : 3** **Crédit : 4**

Objectifs de l'enseignement

- Ce module est un approfondissement des bases du Machine Learning : types d'apprentissage, prétraitement des données, extraction de motifs, classification supervisée, validation croisée et optimisation de modèles. Apprendre à connaître les données.
- Connaître et savoir appliquer les fondamentaux théoriques et pratiques du machine learning.
- Connaître et savoir appliquer les techniques d'extraction de connaissance à partir de données quelconques.

Connaissances préalables recommandées

Mathématiques : Algèbre linéaire, probabilités, statistiques, etc.

Contenu de la matière Cours : 21h

1. Introduction et Fondamentaux

- Aperçu du Machine Learning
- Types de ML et cycle de vie d'un projet
- Applications et différences avec le Data Mining

2. Prétraitement des données

- Gestion des valeurs manquantes et outliers
- Normalisation, discrétisation, encodage
- Problème des jeux de données déséquilibrés (Imbalanced Datasets)

3. Régression

- Régression linéaire (simple et multiple)
- Regularisation (Ridge, Lasso)

4. Classification

- Perceptron simple (introduction aux réseaux)
- Arbres de décision et forêts
- k-plus proches voisins
- Naïve Bayes
- SVM (séparateurs linéaires)

5. Clustering

- Mesures de similarité
- k-means, CAH
- DBSCAN

6. Évaluation et validation des modèles

- Métriques pour la régression (MAE, RMSE, R^2)
- Métriques pour le clustering (silhouette, Davies-Bouldin)
- Techniques avancées de validation (stratified CV, nested CV)

- Cross-validation (K-fold, leave-one-out)
- Overfitting vs underfitting (courbes d'apprentissage)
- Optimisation d'hyperparamètres
- Interprétation des résultats (accuracy, precision, recall, F1)

7. Ingénierie des caractéristiques et détection d'anomalies

- Création et sélection de features
- Réduction de dimension (PCA)
- Méthodes de détection (Isolation Forest, LOF)

8. Motifs fréquents et règles d'association

- Concepts de base
- Méthodes pour la recherche des motifs fréquents
- Passage aux règles d'association
- Motifs fréquents séquentiels

9. Études de cas et AutoML

- Projet 1 : segmentation clientèle avancée
- Projet 2 : détection de fraude
- Introduction à **AutoML** (TPOT, Auto-sklearn)

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôle continu, travail personnel.

Références bibliographiques

1. Murphy, K.M., "Machine Learning", MIT Press, 2012.
2. Mohri, M., Rostamizadeh, A., and Talwalkar, A., "Foundations of Machine Learning", MIT Press, 2012.
3. Goodfellow, I., Bengio, Y., and Courville, A., "Deep Learning", MIT Press, 2016.
4. Borwein, J. M., and Lewis, A. S., "Convex Analysis and Nonlinear Optimization: Theory and Examples", Springer, 2006
5. Practical MLOps, Noah Gift & Alfredo Deza, O'REILLY Media, 2021 6.
6. Introducing MLOps, Mark Treveil & the Dataiku Team, O'REILLY Media, 2020

Intitulé de la matière: Représentation des connaissances et Raisonnement

VHS : 42h

VHH : 3h

Cours : 1h30

TD :

TP : 1h30

VHS travail personnel : 42h

Coefficient : 3Crédit : 4

Objectifs de l'enseignement :

L'économie de la connaissance se traduit très concrètement aujourd'hui par le fait que de nombreuses grandes entreprises travaillant sur l'information (NY Times, Bloomberg, Facebook, Microsoft, Google, etc.) ont investi massivement dans la création de "graphes de connaissances". C'est aussi par ce moyen que Google a pu enrichir les réponses aux requêtes des utilisateurs sur son moteur de recherche et sur son assistant vocal. Ces graphes de connaissances représentent des connaissances symboliques et structurées qui sont exprimées à l'aide de langages formels. Diverses formes de logiques permettent de raisonner sur ces connaissances. L'objectif de ce cours est d'introduire les problématiques et les techniques qui s'appliquent à ces types de connaissances en passant en revue les bases logiques, les Structures classiques, le Raisonnement, les Graphes modernes et Applications avancées.

Connaissances préalables recommandées : Modélisation Logique et Systèmes Formels

Contenu de la matière Cours : 21h

1 Introduction et Fondements

1.1 Introduction à la Représentation des Connaissances (RC)

- Pourquoi représenter les connaissances ?
- RC vs Apprentissage statistique.
- Applications : systèmes experts, web sémantique, agents intelligents.

1.2 Concepts de base en logique

- Logique propositionnelle pour la représentation des connaissances
- Logique du premier ordre (FOL)
- Raisonnement : déduction, induction, abduction.

2 Structures classiques de représentation 2.1 Structures de données en RC

- Réseaux sémantiques.
- Frames et scripts.
- Graphes conceptuels.

2.2 Ontologies et Web sémantique

- Ontologies : définitions, rôles.
- Langages : RDF, RDFS, OWL.
- Raisonneurs automatiques (Reasoners).
- Cas d'usage : médecine (SNOMED CT), industrie.

3 Raisonnement automatisé

3.1 Méthodes de raisonnement

- Chaînage avant et arrière.
- Résolution et unification.
- Systèmes experts (ex. : MYCIN, DENDRAL).

3.2 Raisonnement incertain et incomplet en RC

- Logiques floues (fuzzy logic).
- Raisonnement probabiliste : réseaux bayésiens.
- Raisonnement par analogie et par cas (Case-Based Reasoning).

4 Représentation des connaissances dans l'IA moderne 4.1 Graphes de connaissances (Knowledge Graphs)

- Représentation : entités, relations, triplets.
- Langages de requête : SPARQL.

- Applications (Google Knowledge Graph, Wikidata).

4.2 Apprentissage et Représentation (Neuro-symbolic AI) •

Limites des systèmes purement symboliques.

- Intégration Machine Learning et Représentation des connaissances.
- Embeddings de graphes de connaissances (TransE, RotatE, ComplEx).
- Raisonnement avec LLM et Graphes de connaissances.

5 Applications et tendances actuelles

5.1 Traitement automatique des langues (NLP) et RC.

5.2 Chatbots intelligents basés sur KRR.

5.3 Recommandation, diagnostic médical, robotique.

6 Étude de cas

6.1 Construction d'une **mini-ontologie** (ex. domaine médical ou universitaire).

6.2 Utilisation d'un **raisonneur OWL** (ex. **Protégé, Pellet, HermiT**).

6.3 Intégration avec un **graphe de connaissances** (Wikidata, DBpedia).

6.4 Extension : lien avec un modèle d'IA (ex. LLM + Knowledge Graph).

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) •

Examen semestriel en présentiel (60%).

- **Évaluation continue (CC) (40%)** : contrôles continus, travail personnel.

Références bibliographiques

1. Russell, S. & Norvig, P.
Artificial Intelligence: A Modern Approach (4th Edition, Pearson, 2020).
2. Poole, D., Mackworth, A., & Goebel, R.
Computational Intelligence: A Logical Approach (Oxford University Press, 1998).
3. Poole, D., Mackworth, A., & Goebel, R. Brachman, R. J., & Levesque, H. J. *Knowledge Representation and Reasoning* (Morgan Kaufmann, 2004).
4. John F. Sowa
Knowledge Representation: Logical, Philosophical, and Computational Foundations (Brooks/Cole, 2000).
5. Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., & Patel-Schneider, P. F. (Eds.) *The Description Logic Handbook: Theory, Implementation and Applications* (2nd Edition, Cambridge University Press, 2010).
6. Antoniou, G., & Van Harmelen, F. *A Semantic Web Primer* (3rd Edition, MIT Press, 2012).
7. Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., de Melo, G., Gutierrez, C., ... & Zimmermann, A. *Knowledge Graphs* (Synthesis Lectures on Data, Semantics, and Knowledge, Morgan & Claypool, 2021).
8. Hamilton, W. *Graph Representation Learning* (Morgan & Claypool, 2020).
9. Besold, T. R., d'Avila Garcez, A. S., Bader, S., Bowman, H., Domingos, P., Hitzler, P., ... & Silver, D. *Neural-Symbolic Learning and Reasoning: A Survey and Interpretation*
10. Hitzler, P. *Foundations of Knowledge Graphs and Neuro-Symbolic AI* (Lecture notes, Kansas State University, 2022).
11. MIT OpenCourseWare – 6.825: Techniques in Artificial Intelligence (Knowledge Representation and Reasoning) : <https://ocw.mit.edu/courses/6-825-techniques-in-artificial-intelligence-knowledgerepresentation-and-reasoning-fall-2002/>
12. Stanford CS520: Knowledge Graphs (Andrew Ng, Christopher Ré, etc.)

**Programme détaillé des enseignements du semestre (S1) Master
académique**

Intitulé de la matière: Analyse de données

VHS : 42h

VHH : 3h

Cours : 1h30 TD : 1h30

VHS travail personnel : 21h00

Coefficient : 2

Crédit : 5

Objectifs de l'enseignement

L'analyse de données a pour objectif de donner une vision synthétique et interprétable de l'organisation de données ; ainsi, cette matière est une introduction à l'analyse des données en présentant les différents étapes du cycle d'analyse de données afin que l'étudiant prend connaissance de l'importance de l'analyse des données en termes d'interprétation et de prise de décision. Le contenu de cette matière permet à l'étudiant :

- De comprendre les concepts fondamentaux de l'analyse de données.
- De développer des compétences en exploration, nettoyage et transformation de données.
- De comprendre et appliquer différentes techniques d'analyse statistique et prédictive.
- De pouvoir utiliser des outils d'analyse et de visualisation de données.
- D'interpréter les résultats de l'analyse et de les communiquer efficacement. Explorer, décrire et interpréter des données dans leur aspect multidimensionnel.

Connaissances préalables recommandées

Connaissances de base en statistique descriptive et des notions de calcul matriciel

Contenu de la matière Cours : 21h

1– Introduction à l'analyse de données

1. Définition et objectifs de l'analyse exploratoire
2. Typologie des données : qualitatives, quantitatives, ordinales
3. Étapes d'un projet d'analyse de données
4. Préparation des données : nettoyage, codage, traitement des valeurs manquantes

2 – Analyse des relations entre variables

1. Analyse de corrélation (Pearson, Spearman)
2. Test de dépendance entre variables qualitatives (χ^2)
3. Analyse de variance (ANOVA)
4. Régression linéaire simple et multiple : interprétation, qualité d'ajustement

3 – Analyse factorielle

1. **ACP** – Analyse en Composantes Principales (pour données quantitatives)
2. **AFC** – Analyse Factorielle des Correspondances (tableaux de contingence)
3. **ACM** – Analyse des Correspondances Multiples (variables qualitatives multiples)
4. **AFM** – Analyse Factorielle Mixte (variables qualitatives et quantitatives mélangées)

4 – Classification des données

1. Classification ascendante hiérarchique (CAH)

2. Méthodes de partition : k-means, k-medoids
3. Choix du nombre de classes (critères de Calinski-Harabasz, silhouette, etc.)
4. Interprétation des groupes et visualisation des classes

5 – Visualisation des données

5. Graphiques univariés : histogrammes, boxplots, diagrammes circulaires
6. Graphiques bivariés : nuages de points, matrices de corrélation
7. Visualisation multidimensionnelle : cartes factorielles, projection des individus
8. Visualisation interactive avec Python (Matplotlib, Seaborn, Plotly) ou R (ggplot2, FactoMineR)

6 – Séries Chronologiques

1. Modélisation déterministes
2. Analyse de la tendance
3. Les moyennes mobiles
4. Décomposition d'une série chronologique
5. Prévision par lissage exponentiel

7– Études de cas et applications

1. Application sur jeux de données réels (marketing, santé, éducation, etc.)
2. Analyse exploratoire complète guidée
3. Sélection des méthodes selon la nature des données
4. Interprétation critique des résultats statistiques
5. Mini-projet en binôme ou par groupe

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) -
Examen semestriel en présentiel (60%).

- **Évaluation continue (CC) (40%)** : Contrôle continu, travail personnel.

Références bibliographiques

- [1] Saporta G. Probabilités, analyse des données et statistique. 3^{ème} édition, Technip, 2011.
[2] Tenenhaus M. : Statistique. Méthodes pour décrire, expliquer et prévoir, Dunod, 2010.
[3] Lebart L., Morineau A., Piron M. : Statistique exploratoire multidimensionnelle, Dunod, 4^{ème} édition, 2006.
[4] Nakache J.-P., Confais J. : Approche pragmatique de la classification. Technip 2005 [5] Pierre Rigollet, Analysez efficacement vos données, livre Edition Eni, 2013. [6] Nathan Yau, Data visualisation, livre Edition Eyrolles, 2013

Intitulé de la matière : Méthodes d'Optimisation

VHS : 42h VHH : 3h Cours : 1h30 TD : 1h30
VHS travail personnel : 21h00 Coefficient : 2 Crédit : 4

Objectifs de l'enseignement

- Comprendre les fondements théoriques de l'optimisation mathématique.
- Maîtriser les principales méthodes d'optimisation déterministes et stochastiques.
- Savoir formuler un problème d'optimisation dans le contexte de l'IA.
- Appliquer les algorithmes d'optimisation dans les domaines du machine learning, de la vision par ordinateur et de la recherche opérationnelle.
- Découvrir les approches récentes : optimisation convexe, algorithmes de gradient stochastique, métaheuristiques bio-inspirées, optimisation sur graphes et optimisation distribuée.

Connaissances préalables recommandées

Mathématiques générales, Probabilités et statistiques de base, Bases de programmation scientifique (Python, Matlab ou R), Connaissances de base en machine learning souhaitées (mais pas obligatoires).

Contenu de la matière Cours : 21h

1 Fondamentaux de l'optimisation

1.1 Introduction à l'optimisation

- Définition et classification des problèmes d'optimisation.
- Optimisation convexe vs non convexe.
- Normes et distances (euclidienne, p-norme, norme matricielle).

1.2 Optimisation sans contraintes

- Recherche linéaire (line search).
- Méthodes de descente : Gradient Descent, Newton, Quasi-Newton (BFGS).
- Taux de convergence.

1.3 Optimisation sous contraintes

- Multiplicateurs de Lagrange.
- Conditions de Karush-Kuhn-Tucker (KKT).
- Méthodes de pénalisation et de barrières.

2 Optimisation convexe et programmation mathématique

2.1 Optimisation convexe

- Propriétés et théorèmes fondamentaux.
- Dualité et interprétation en IA.

2.2 Programmation mathématique

- Programmation linéaire (Simplex, Méthodes intérieures).
- Programmation quadratique.
- Applications en planification et en apprentissage automatique.

3 Optimisation stochastique et heuristique

3.1 Optimisation stochastique

- Gradient stochastique (SGD).
- Améliorations modernes : Momentum, AdaGrad, RMSProp, Adam.
- Applications à l'entraînement des réseaux profonds.

3.2 Métaheuristiques et optimisation bio-inspirée

- Recuit simulé (Simulated Annealing).
- Algorithmes évolutionnaires (Genetic Algorithms, Differential Evolution).
- Essaim de particules (PSO), colonies de fourmis.
- Applications : optimisation combinatoire, hyperparameter tuning.

4 Optimisation pour l'IA

4.1 Optimisation pour le 'Machine Learning'

- Régularisation et optimisation généralisée.
- Problèmes d'explosion/vanishing gradient et solutions.
- Optimisation sur réseaux convolutifs et récurrents.

4.2 Optimisation distribuée et à grande échelle • Décomposition de problème.

- Algorithmes distribués (ADMM).
- Optimisation sur clusters et GPU.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** interrogations écrites, travail personnel, tests.

Références bibliographiques

- Boyd, S., & Vandenberghe, L. *Convex Optimization*, Cambridge University Press, 2004.
- Nocedal, J., & Wright, S. J. *Numerical Optimization* (2nd Edition), Springer, 2006.
- Goodfellow, I., Bengio, Y., & Courville, A. *Deep Learning*, MIT Press, 2016 (chapitres sur optimisation).
- Kalyanmoy Deb. *Optimization for Engineering Design: Algorithms and Examples*, PHI, 2012.
- Articles récents sur l'optimisation bayésienne, NAS et optimisation distribuée (NeurIPS, ICML, ICLR).

Intitulé de la matière: Réseaux avancés **Semestre :** 1 **Type :** UET1
VHS : 42h **VHH :** 3h **Cours :** 1h30 **TD :** **TP:** 1h30
VHS travail personnel : 21h **Coefficient :** 1 **Crédit :** 2

Objectifs de l'enseignement :

Cette matière vise à consolider et approfondir les connaissances des étudiants sur le fonctionnement des réseaux informatiques, en abordant des notions avancées telles que le routage dynamique, les protocoles de transmission, les protocoles applicatifs ainsi que la configuration IPv6. L'objectif est de préparer les étudiants à analyser, concevoir et gérer des infrastructures réseau plus complexes dans des contextes professionnels ou de recherche.

Connaissances préalables recommandées : Réseaux informatiques (licence)

Contenu de la matière

Cours : 21h 1-

Rappels sur les réseaux

- Modèles OSI et TCP/IP
- Adressage MAC/IP (Protocole ARP)
- Adressage dans les sous-réseaux (FLSM, VLSM, CIDR ..)

2- Protocoles de routage

- Routage statique
- Routage dynamique (RIP, OSPF, BGP)
 - Vecteurs de distance
 - Etat des liens

3- Réseaux locaux et réseaux étendus

- Réseaux locaux Virtuels (Vlan), Protocoles VTP, DTP
- Routage inter-Vlan
- Redondances dans les LANs (STP, EtherChannel, Protocole HSRP)
- Protocole PPP
- VPN (Protocole GRE)

4- Protocoles de transmission

- Les protocoles TCP/UDP
- Gestion des états de la connexion TCP
- Contrôle de flux
- Contrôle de congestion
- Contrôle d'erreur

5- Protocoles applicatifs

- Protocoles web (HTTP et HTTPS)
- Protocoles de messagerie électronique (SMTP, POP et IMAP)
- Services de partage de fichiers (FTP et SMB)
- DHCP
- DNS
- Telnet et SSH

6- Protocole IPv6

- Adressage
- Transition IPv4/IPv6

- Services IPv6... 7- Gestion des réseaux ● LLDP, CDP, NTP
- SNMP
- Qualité de Service (QoS)
- Automatisation de gestion de la configuration (Ansible...)

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) -

Examen semestriel en présentiel (60%).

- **Évaluation continue (CC) (40%) :** Contrôle continu, travail personnel.

Références

- Tanenbaum, A., Feamster, N., & Witherall, D. (2021). Computer networks (6th ed.). Pearson Publisher.
- White, R., & Banks, E. (2017). Computer networking problems and solutions: An innovative approach to building resilient, modern networks. Addison-Wesley Professional.
- Englander, I., & Wong, W. (2021). The architecture of computer hardware, systems software, and networking: An information technology approach. John Wiley & Sons.
- Bonaventure, O. (2021). Computer networking: Principles, protocols and practice.
- Dordal, P. L. (2022). An introduction to computer networks. Department of Computer Science, Loyola University Chicago.

Intitulé de la matière : Une matière au choix **Semestre : 1 Type : UED1**
VHS : 21h **VHH :** 1h30 **Cours :** 1h30 **TD :** **TP :**
VHS travail personnel : **Coefficient :** 1 **Crédit :** 1

Objectifs de l'enseignement : Cette matière permet aux étudiants d'acquérir une vision plus large de leur domaine d'études et favorise une meilleure compréhension des enjeux sociaux, économiques et éthiques liés à l'informatique, ainsi qu'une capacité à communiquer efficacement et à s'adapter à des situations diverses qui peuvent se présenter. En d'autres termes, cette matière permet :

- Une ouverture à d'autres disciplines en relation avec le quotidien, afin de comprendre l'importance de la collaboration dans un environnement pluridisciplinaire.
- L'élargissement des horizons, ce qui favorise la compréhension de la pensée humaine dans le temps et l'espace.
- Développement de compétences transversales : communication, vision holistique, analyse et synthèse, capacité d'adaptation, etc.

Connaissances préalables recommandées

Contenu de la matière Cours : 21h

1. Introduction à la discipline concernée (matière choisie).
2. Présentation des concepts fondamentaux.
3. Applications croisées avec l'informatique
4. Études de cas.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) - **Examen semestriel en présentiel (100%).**

Références bibliographiques : L'enseignant propose des références selon la matière choisie.

Intitulé de la matière: Deep Learning **Semestre:** 2 **Type :** UEF1
VHS : 63h **VHH :** 4h30 **Cours :** 1h30 **TD :** 1h30 **TP :** 1h30 **VHS travail**
personnel : 42h **Coefficient :** 3 **Crédit :** 5

Objectifs de l'enseignement:

Ce module permet d'acquérir des connaissances sur l'apprentissage automatique qui est un pilier très important en intelligence artificielle et comme cas particulier les réseaux de neurones.

Connaissances préalables recommandées

Contenu de la matière Cours : 21h

1 – Introduction au Deep Learning

1. Définition, origine et liens avec le machine learning
2. Réseaux de neurones : principe général
3. Importance des données massives et du calcul parallèle (GPU, TPUs)
4. Cas d'usage (vision, NLP, audio, séries temporelles)

2 – Réseaux de neurones artificiels (ANN)

1. Neurone formel, fonctions d'activation (ReLU, sigmoid, tanh)
2. Propagation avant et rétropropagation
3. Fonction de coût et descente de gradient (SGD, Adam)
4. Overfitting et techniques de régularisation : dropout, early stopping, L2

3– Outils et frameworks pratiques

1. Introduction à TensorFlow et Keras
2. Manipulation de données avec NumPy et Pandas
3. Création, entraînement et évaluation d'un modèle ANN simple
4. Sauvegarde, chargement et visualisation des modèles

4 – Réseaux convolutifs (CNN)

1. Architecture d'un CNN : couches conv, pooling, fully connected
2. Réduction de dimension, détection de motifs locaux
3. Application à la classification d'images (MNIST, CIFAR-10)
4. Techniques : data augmentation, batch normalization

5 – Réseaux récurrents (RNN) et séquentiels

1. RNN simples et limitations (vanishing gradient)
2. LSTM et GRU
3. Application aux séries temporelles et au traitement du langage naturel (NLP)
4. Séquences à séquences (seq2seq) et modèles attentionnels

6 – Modèles avancés et tendances récentes

1. Auto-encoders : réduction de dimension, débruitage
2. GAN (Generative Adversarial Networks) : principe, applications
3. Introduction aux Transformers : architecture et intérêt (vision, NLP)
4. Aperçu de modèles pré-entraînés (BERT, GPT, ResNet)

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôle continu, travail personnel.

Références bibliographiques

- 1- « Réseaux de neurones, Méthodologies et applications », Gérard Dreyfus, Manuel Samuelides, Jean-Marc Martinez, Mirta B. Gordon, Fouad Badran, Sylvie Thiria, Laurent Hérault, 2e édition Eyrolles (29 avril 2004)
- 2- Golovko, V. A. (2017). Deep learning: an overview and main paradigms. Optical memory and neural networks, 26, 1-17.
- 3- L'apprentissage machine - David Rousseau, <https://www.youtube.com/watch?v=utaKQ-obVR8>

Intitulé de la matière: Systèmes Multi-agents **Semestre:** 2 **Type :** UEF
VHS : 42h **VHH :** 3h **Cours :** 1h30 **TD :** 1h30 **TP :**
VHS travail personnel : 42h00 **Coefficient :** 3 **Crédit :** 4

Objectifs de l'enseignement :

- Introduire les concepts fondamentaux des systèmes multi-agents et leur distinction avec les systèmes centralisés.
- Comprendre les notions d'agent, autonomie, coopération, communication et organisation.
- Étudier les architectures d'agents (réactifs, cognitifs, hybrides) et les modèles d'interaction.
- Maîtriser les concepts fondamentaux et récents des systèmes multi-agents, incluant les **Agentic AI** et les basés sur **LLMs**, ainsi que les systèmes **context-aware** et de **coordination décentralisée**.
- Comprendre les mécanismes de **communication, coopération, coordination, contextawareness, et sécurité** des agents.
- Appliquer les SMA dans des domaines avancés : e-commerce, logistique, robotique, simulation sociale, agents de LLM collaboratifs.
- Initier les étudiants aux enjeux modernes tels que scalabilité, robustesse, protocoles inter-agents (Agent-to-Agent), standards émergents (MCP, A2A).

Connaissances préalables recommandées :

IA de base : agents simples, planification, recherche, Programmation (idéalement Java, Python), Notions de réseaux distribués et architecture système, (Optionnel) Fondements en **renforcement learning**, **IA explicable**, **théorie des jeux**.

Contenu de la matière Cours : 21h

1 Les concepts fondamentaux des systèmes multi-agents

- Introduction aux SMA
- Architectures d'agents
- Communication et interaction

2 Évolution et paradigmes récents

- **Agentic AI&Agentic Web** – agents autonomes multi-modules, réseaux d'agents connectés
- **LLM-based MAS** – agents collaborant via langage naturel avec mémoire, planification, rôles assignés
- Protocoles inter-agents émergents : Agent-to-Agent (A2A), Model Context Protocol (MCP), FIPA-ACL

3 Coordination, adaptabilité, et sens du contexte

- **Coordination hybride** : hiérarchique + distribuée, agents humains-agents, MAS pour LLMs
- **Context-aware multi-agent systems (CA-MAS)** – agents réagissant au contexte dynamique dans divers domaines (IoT, gestion de crise...)

4 Apprentissage et communication avancés

- **Multi-agent reinforcement learning avec communication** – émergence des comportements collectifs coopératifs/compétitifs

- Architectures collaboratives dans les SMA (émergence sociale, structures sans programmation explicite)

5 Applications actuelles

- **Robotique autonome** : coordination multi-robots (SMARTS).
- **Smart grids, transport, simulation sociale** : agents adaptatifs dans réseaux d'énergie, trafic, etc.
- **Consensus dynamics** & prise de décision collective

6 Enjeux contemporains

- **Sécurité & robustesse** : vulnérabilités, attaques adversariales, nouveaux modèles de game theory.
- **Interopérabilité & gouvernance** : protocole standards (MCP, A2A), gouvernance de réseaux d'agents

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%)**.
- **Évaluation continue (CC) (40%)** : interrogations écrites, travail personnel.

Références bibliographiques

- Shoham, Y., & Leyton-Brown, K. (2008). **Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations**. Cambridge University Press.
- Russell, S. & Norvig, P. (2021). **Artificial Intelligence: A Modern Approach** (4th ed.), chapitres sur agents et environnements multi-agents.
- Sutton, R. & Barto, A. (2018). **Reinforcement Learning: An Introduction** (2nd ed.), sections sur **multi-agent RL**.
- Panait, L. & Luke, S. (2005). **Cooperative Multi-Agent Learning: The State of the Art**. Autonomous Agents and Multi-Agent Systems.
- Context-Aware MAS: Techniques, Challenges and Future Directions.
- Multi-Agent Coordination across Applications
- Communication-Centric Survey of LLM-Based MAS
- LLMs Working in Harmony: Technologies for LLM-Based MAS
- Multi-Agent Deep RL with Communication.

Intitulé de la matière : Vision par ordinateur **Semestre : 2 Type : UEF1**
VHS : 63h **VHH : 3h** **Cours : 1h30 TD : 1h30** **TP : 1h30**
VHS travail personnel : 42h **Coefficient : 3** **Crédit : 5**

Objectifs de l'enseignement

L'objectif du cours est de donner une vue d'ensemble de la vision artificielle. Formation des images : géométrie, coordonnées homogènes, transformation de coordonnées; photométrie, luminance d'un rayon lumineux, illuminance d'une image; numérisation, projection de la scène tridimensionnelle sur le plan image. Les objectifs du cours sont :

- Introduire les concepts fondamentaux de la perception visuelle artificielle.
- Comprendre les méthodes classiques et modernes de traitement et d'analyse d'images/vidéos.
- Maîtriser les techniques **deep learning avancées** (CNN, ResNet, EfficientNet, Vision Transformers).
- Découvrir les nouvelles tendances : **apprentissage auto-supervisé**, **modèles de diffusion** pour la génération d'images, et **modèles multimodaux** (vision + langage).
- Développer des compétences pratiques avec des outils et frameworks (OpenCV, PyTorch, TensorFlow, Detectron2).

Connaissances préalables recommandées

Algèbre linéaire (vecteurs, matrices, transformations linéaires), Probabilités et statistiques (distribution, inférence, maximum de vraisemblance), Calcul différentiel et optimisation (gradient, descente de gradient), Programmation Python (NumPy, Matplotlib, OpenCV), Bases en **apprentissage automatique** et **deep learning**, **Bases en traitement du signal** (filtrage, convolution, transformée de Fourier).

Contenu de la matière Cours

: 21h

1 Représentation et manipulation d'images •

Formation des images (caméras, capteurs).

- Géométrie, coordonnées homogènes.
- Transformation de coordonnées.
- Photométrie.
- luminance d'un rayon lumineux
- Espaces de couleur (RGB, HSV, YCbCr).
- Opérations de base : filtrage, seuillage, segmentation simple.

2 Détection et extraction de caractéristiques

- Points d'intérêt (Harris, SIFT, SURF, ORB).
- Descripteurs et correspondances.
- Cas pratique : suivi d'objets.

3 Reconnaissance d'objets et vision géométrique

- Modèles géométriques (épipolarité, homographie).
- Calibration de caméra, reconstruction 3D.
- Détection et suivi d'objets (HOG, Viola-Jones).
- Bases des réseaux de neurones convolutionnels.
- Architectures classiques (LeNet, AlexNet, VGG).
- Problèmes : surapprentissage, régularisation, transfert de connaissances.

4 CNN pour la vision

5 CNN avancés

- ResNet, DenseNet, EfficientNet.
- Réseaux pour la segmentation : U-Net, Mask R-CNN.
- Détection d'objets : YOLO, Faster R-CNN, DETR.

6 Vision Transformers (ViT)

- Principe des Transformers appliqués aux images.
- Comparaison CNN vs ViT.
- Applications récentes (classification, segmentation, détection).

7 Vision 3D et vidéo

- Estimation de mouvement et flux optique.
- Vision stéréoscopique.
- Analyse d'actions dans la vidéo.

8 Vision multimodale et auto-supervisée

- Apprentissage auto-supervisé (SimCLR, MAE, MoCo).
- Modèles multimodaux (CLIP, BLIP-2).
- Applications : recherche d'images par texte, VQA (Visual Question Answering).

9 Applications avancées

- Vision médicale (radiologie, imagerie biomédicale).
- Conduite autonome et détection d'environnement.
- Réalité augmentée, biométrie, robotique.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** contrôles continus, travail personnel.

Références bibliographiques

1. Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed.). Springer.
2. Forsyth, D. A. (2023). *Computer Vision: A Modern Approach* (3rd ed.). Pearson.
3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Ressources pratiques

- OpenCV : <https://docs.opencv.org>
- PyTorch Vision : <https://pytorch.org/vision/stable>

Detectron2 : <https://github.com/facebookresearch/detectron2>

Intitulé de la matière : Gestion de l'incertain **Semestre :** 2 **Type :** UEF2
VHS : 42h **VHH :** 3h **Cours :** 1h30 **TD :** 1h30 **TP :**
VHS travail personnel : 42h **Coefficient :** 3 **Crédit :** 4

Objectifs de l'enseignement

Cette matière permet à l'étudiant:

- Comprendre les sources d'incertitude dans les systèmes d'information et en IA (données bruitées, incomplètes, contradictoires).
- Étudier les modèles mathématiques et algorithmiques pour représenter et raisonner avec l'incertitude.
- Maîtriser les approches probabilistes, possibilistes et basées sur la logique floue.
- Introduire les méthodes avancées : inférence bayésienne, réseaux bayésiens, théorie de Dempster-Shafer, modèles probabilistes graphiques.
- Découvrir les applications pratiques : diagnostic médical, systèmes de recommandation, robotique autonome, décision en présence de risques.

Connaissances préalables recommandées

Probabilités et statistiques, algèbre linéaire, calcul différentiel, Notions de logique propositionnelle et du premier ordre, programmation Python (bibliothèques : NumPy, PyMC, scikit-fuzzy), bases en apprentissage automatique (régression, classification, réseaux de neurones).

Contenu de la matière Cours : 21h

1 Introduction à l'incertitude en IA

- Types d'incertitudes : bruit, manque d'information, ambiguïté.
- Exemples d'applications (santé, finance, robotique, traitement du langage).

2 Théorie des probabilités et inférence bayésienne

- Rappels sur la probabilité conditionnelle et le théorème de Bayes.
- Inférence bayésienne et distributions a posteriori.
- Introduction aux chaînes de Markov et modèles cachés de Markov (HMM).

3 Réseaux bayésiens et modèles graphiques probabilistes

- Définition et structure d'un réseau bayésien.
- Inférence exacte et approximative (échantillonnage Monte Carlo, variational inference).
- Applications pratiques (diagnostic médical, vision par ordinateur).

4 Théorie des possibilités et logique floue

- Différences entre approche probabiliste et possibiliste.
- Variables linguistiques et ensembles flous.
- Systèmes experts flous (Fuzzy Inference Systems).

5 Théorie de Dempster-Shafer et raisonnement avec croyances

- Cadre des fonctions de croyance.
- Règle de combinaison de Dempster.
- Cas d'usage : fusion de données incertaines en robotique.

6 Apprentissage sous incertitude

- Probabilistic Machine Learning (Bayesian Neural Networks).
- Dropout comme approximation bayésienne.
- Calibration des modèles de deep learning.

7 Méthodes avancées et actuelles

- Apprentissage par renforcement sous incertitude (POMDP : Partially Observable Markov Decision Process).
- Incertitude épistémique vs incertitude aléatoire.
- Approches récentes : ensembles d'hypothèses, quantile regression, ensembles bayésiens profonds.

8 Applications pratiques

- Diagnostic médical assisté par IA.
- Systèmes de décision sous incertitude en finance et assurance.
- Navigation robotique avec données bruitées.
- IA explicable et gestion de l'incertitude.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** contrôles continus, miniprojet.

Références bibliographiques

1. Barber, D. (2012). *Bayesian Reasoning and Machine Learning*. Cambridge University Press.
2. Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.
3. Klir, G. J., & Yuan, B. (1995). *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall.
4. Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.
5. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Ressources pratiques

- PyMC3 : <https://www.pymc.io>
- scikit-fuzzy : <https://pythonhosted.org/scikit-fuzzy/>
- TensorFlow Probability : <https://www.tensorflow.org/probability>

Intitulé de la matière : Modélisation et simulation **Semestre:** 2 **Type :** UEM
VHS : 63h **VHH :** 4h30 **Cours :** 1h30 **TD :** 1h30 **TP :** 1h30 **VHS travail**
personnel : 42h **Coefficient :** 2 **Crédit :** 5

Objectifs de l'enseignement

- Comprendre les concepts fondamentaux de la modélisation et de la simulation de systèmes complexes (physiques, biologiques, sociaux, informatiques).
- Être capable de construire un modèle formel, de le simuler et d'analyser ses résultats.
- Découvrir les outils et environnements logiciels actuels pour la simulation (Matlab/Simulink, AnyLogic, NetLogo, Python/SimPy).
- Développer des compétences pour appliquer la simulation dans des domaines variés : optimisation de processus, réseaux, logistique, santé, écologie, IA.

Connaissances préalables recommandées Equations différentielles ordinaires, probabilités et statistiques, algèbre linéaire ; programmation (Python, Java ou C++), algorithmique.

Contenu de la matière Cours : 21h

1 Modélisation des systèmes

- Typologie des systèmes : discrets, continus, hybrides, déterministes, stochastiques, adaptatifs.
- Niveaux d'abstraction : micro, méso et macro-modélisation.
- Outils de modélisation formelle :
 - Machines à états finis et automates temporisés.
 - Réseaux de Petri et extensions (réseaux stochastiques, colorés, temporels).
 - Théorie des files d'attente.
 - Modélisation par équations différentielles et modèles stochastiques (Markov, processus de Poisson).
- Introduction à la modélisation multi-agents (SMA) et aux systèmes complexes.

2 Techniques d'évaluation des performances

- Présentation générale des techniques d'évaluation : analytiques, numériques, expérimentales et basées sur la simulation.
- Méthodes mathématiques :
 - Files d'attente (M/M/1, M/M/c, etc.).
 - Modèles markoviens et chaînes de Markov discrètes/continues.
 - Méthodes matricielles et équations de flux.
- Limites des approches analytiques et motivation de la simulation.
- Introduction à l'analyse de sensibilité et à l'incertitude dans l'évaluation des performances.

3 La simulation

- Types de simulation :
 - Simulation continue (équations différentielles, dynamique des systèmes).
 - Simulation discrète à événements.
 - Simulation hybride (combinaison continu/discret/SMA).
 - Simulation Monte Carlo et applications.
- Simulation de systèmes dynamiques complexes (écologie, santé, économie, IA).
- Simulation stochastique et rôle du hasard.

- Génération de nombres pseudo-aléatoires : méthodes, qualité et reproductibilité.
- Tests statistiques de générateurs de nombres aléatoires.
- Échantillonnage et réduction de variance (importance sampling, stratification).
- Analyse, vérification et validation des modèles de simulation.
- Bonnes pratiques pour assurer la crédibilité des résultats.

4 Outils et environnements de simulation

- Langages classiques et historiques de simulation : GPSS, SIMULA, Arena, etc.
- Outils modernes :
 - Matlab/Simulink pour systèmes continus et hybrides.
 - AnyLogic pour simulation multi-paradigmes (DES, agents, dynamique des systèmes).
 - NetLogo et Mesa (Python) pour simulation multi-agents.
 - SimPy (Python) pour la simulation à événements discrets.
- Environnements de co-simulation (FMI, mosaik, OpenModelica).
- Visualisation et graphisme en simulation (dashboards, VR/AR pour simulation interactive).
- Intégration avec l'intelligence artificielle (ex. OpenAI Gym, Unity ML-Agents pour simulation d'apprentissage par renforcement).

5 Étude d'un langage / framework de simulation

- Étude approfondie d'un langage ou framework de simulation au choix (selon orientation du cours) :
 - **SimPy (Python)** : approche pratique de la simulation discrète.
 - **NetLogo** : pour les simulations multi-agents accessibles.
 - **AnyLogic** : outil industriel complet multi-paradigmes.
- Étude de cas : modélisation, mise en œuvre, simulation et analyse.
- Projet pratique : conception d'un modèle (ex. logistique, propagation d'épidémie, réseau informatique, smart city).

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- Examen semestriel en présentiel (60%).
- Évaluation continue (CC) (40%) : interrogations écrites, travail personnel, tests.

Références bibliographiques

- S. S. Lavenberg, Computer systems performance evaluation, Academic press 1983
- Mitrani, Modeling of computer and communication systems, Cambridge university. press 1987
- M. Pidd, Computer simulation and management science, J. Wiley and Sons Ed. 1984
- K. S. Trivedi, Probability and statistics with reliability, queing and computer science • applications, Prentice Hall 1982

Intitulé de la matière : Virtualisation et Cloud **Semestre :** 2 **Type :** UEM
VHS : 42h **VHH :** 4h30 **Cours :** 1h30 **TD :** TP : 1h30
VHS travail personnel : 42h **Coefficient :** 2 **Crédit :** 4

Objectifs de l'enseignement

L'un des concepts les plus stratégiques sur lequel le cloud CloudComputing se base est le concept de virtualisation. Les services fondamentaux du cloud tels que le PaaS et le IaaS sont les services les plus touchés par le concept de virtualisation. Ces services font face à des défis importants tels que le redimensionnement des machines virtuelles suite à des besoins dynamiques de l'utilisateur, l'augmentation des ressources des machines virtuelles en termes de CPU, mémoire ou autre ressource. L'objectif de ce cours est de montrer d'une part l'évolution des technologies et des défis auxquels ont fait face les systèmes informatiques pré-cloud, notamment le clustering et palper ensuite d'une part les différences fondamentales entre les cluster et les cloud et d'autre part les défis et les problèmes techniques auxquels font face les infrastructures (les data center) du cloud.

Connaissances préalables recommandées

Principes des bases de données, les réseaux informatiques et les systèmes d'exploitation.

Contenu de la matière Cours : 21h

1 Introduction, Définition, origines et intérêt du Cloud Computing

- Définition du Cloud Computing
- Origines et évolution
- Caractéristiques et avantages du Cloud
- Importance dans le paysage technologique actuel
- Software as a Service (SaaS)
- Platform as a Service (PaaS) • Infrastructure as a Service (IaaS)

Modèles du déploiement du Cloud.

- Cloud public
- Cloud privé
- Cloud hybride
- Cloud communautaire

4 Les Datacenters et Infrastructures Cloud

- Rôle des datacenters dans le Cloud Computing
- Architecture et fonctionnement des infrastructures cloud • Tendances et innovations dans la conception des datacenters

5 Les techniques de virtualisation.

- Virtualisation des serveurs
- Virtualisation du stockage
- Virtualisation du réseau
- Stockage et traitement des données
- Bases de données dans le cloud
- Data Lake et Data Warehouse

7 Intelligence Artificielle dans le Cloud

- Analyse de données en ligne
- Machine Learning et Deep Learning dans le cloud

8 Sécurité dans le Cloud

- Défis de sécurité dans le cloud computing
- Mécanismes de sécurité et bonnes pratiques

- Solutions de cryptage, gestion des accès et surveillance des menaces **9 Service Level Agreement (SLA).**
- Définition et importance des SLA
- Contenu typique des SLA dans le Cloud Computing • Gestion et suivi des SLA pour assurer la qualité de service.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** contrôles continus, travail personnel.

Références bibliographiques

- Thomas Erl, Ricardo Puttini, Zaigham Mahmood *Cloud Computing: Concepts, Technology & Architecture (2nd Edition)* Pearson, 2023.
- Rajkumar Buyya, Satish Narayana Srirama, Sukhpal Singh Gill *Fog and Edge Computing: Principles and Paradigms* Wiley, 2020.
- Saurabh Mishra, Rajkumar Buyya *Cloud Computing for Machine Learning and Cognitive Applications* Springer, 2021.
- James Bond *The Cloud Computing Book: The Future of Computing Explained* Independently Published, 2022.

Intitulé de la matière : Gestion de projets informatiques **Semestre :** 2 **Type :** UET
VHS : 42h **VHH :** 3h **Cours :** 1h30 **TD :** - **TP :** 1h30 **VHS travail**
personnel : 42h **Coefficient :** 1 **Crédit :** 2

Objectifs de l'enseignement: cette matière permet à l'étudiant :

- De comprendre les principes et les méthodes de gestion de projets informatiques.
- D'exécuter les différentes phases d'un projet informatique, de la conception à la clôture.
- D'acquérir les compétences en planification, organisation, suivi et contrôle de projets.
- D'identifier et gérer les risques liés aux projets informatiques.
- D'utiliser des outils et des techniques de gestion de projets informatiques.
- De travailler en équipe dans un contexte de gestion de projet.

Connaissances préalables recommandées : génie logiciel, systèmes d'information.

Contenu de la matière Cours : 21h

1. Introduction à la gestion de projets informatiques
 - Définition et enjeux de la gestion de projets informatiques.
 - Processus de production d'un projet informatique : processus de réalisation, processus de gestion, processus qualité.
 - Les différents types de projets informatiques (développement logiciel, infrastructure, etc.).
 - Les acteurs impliqués dans un projet informatique.
 - Les relations entre gestion de projet et systèmes d'information.
2. Méthodologies de gestion de projets
 - Méthodologies traditionnelles (cycle en V). • Méthodologies agiles (Scrum, Kanban).
3. Planification et estimation
 - Définition du périmètre et des objectifs du projet.
 - Découpage du projet en tâches (WBS).
 - Établissement du planning (diagramme de Gantt, PERT).
 - Estimation des ressources (temps, budget, personnel, coûts, COCOMO).
 - Utilisation d'outils de planification (Microsoft Project, Asana, etc.).
4. Gestion des risques
 - Identification et analyse des risques (matrice SWOT, diagramme d'Ishikawa)
 - Évaluation de la probabilité et de l'impact des risques
5. Suivi et contrôle
 - Suivi de l'avancement du projet
 - Gestion des changements
 - Communication et reporting
 - Tableaux de bord et de métriques clés de performance (KPI)
6. Gestion des ressources
 - Identification des ressources nécessaires
 - Allocation des ressources aux différentes tâches
 - Gestion des conflits et des priorités
7. Gestion de la qualité
 - Définition des critères de qualité.
 - Mise en place de procédures de contrôle qualité

- Réalisation de tests et de validation
8. Clôture du projet
- Documentation de la clôture du projet.
 - Transfert des connaissances et des compétences.
 - Évaluation de la satisfaction des parties prenantes.
 - Bilan du projet et identification des leçons apprises.
9. Outils de gestion de projets informatiques
- Logiciels de gestion de projets
 - Systèmes de gestion de la connaissance
 - Plateformes de communication et de collaboration

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** contrôles continus, travail personnel..

Références bibliographiques

- Project Management Institute (PMI) 52021). A Guide to the Project Management Body of Knowledge (PMBOK Guide). <https://www.pmi.org/>
- Andrew Stellman, Jennifer Greene (2005). Applied Software Project Management . Series: Theory In Practice.
- Kerzner, Harold (2017). Project Management: A Systems Approach to Planning, Scheduling, and Controlling. Wiley. <https://www.wiley.com/>
- S. Berkun (2008). Making Things Happen: Mastering Project Management (Theory I Practice). O'Reilly.
- R. S. Pressman, B. R. Maxim (2014). Software Engineering: a Practitioner's Approach. McGraw Hill.
- McConnell, Steve (2006). Software Estimation: Demystifying the Black Art. Microsoft Press. <https://www.microsoftpressstore.com/>
- C. Aubry (2022). Scrum un outil convivial pour une agilité radicale. Editions Dunod.
- Stephen H.Kan (2010). Metrics and Models in Software Quality Engineering (2nd Edition), Addison-Wesley Professional.
- Linda Westfall (2009). The Certified Software Quality Engineer Handbook. Quality Press.
- Murali Chemuturi (2010). Mastering Software Quality Assurance: Best Practices, Tools and Techniques for Software Developers. J. Ross Publishing.

Intitulé de la matière : Une matière au choix **Semestre :** 2 **Type :** UED2 **VHS :**
21h **VHH :** 1h30 **Cours :** 1h30 **TD :** **TP :**
VHS travail personnel : **Coefficient :** 1 **Crédit :** 1

Objectifs de l'enseignement : Cette matière permet aux étudiants d'acquérir une vision plus large de leur domaine d'études et favorise une meilleure compréhension des enjeux sociaux, économiques et éthiques liés à l'informatique, ainsi qu'une capacité à communiquer efficacement et à s'adapter à des situations diverses qui peuvent se présenter. En d'autres termes, cette matière permet :

- Une ouverture à d'autres disciplines en relation avec le quotidien, afin de comprendre l'importance de la collaboration dans un environnement pluridisciplinaire.
- L'élargissement des horizons, ce qui favorise la compréhension de la pensée humaine dans le temps et l'espace.
- Développement de compétences transversales : communication, vision holistique, analyse et synthèse, capacité d'adaptation, etc.

Connaissances préalables recommandées

Contenu de la matière Cours : 21h

1. Introduction à la discipline concernée (matière choisie).
2. Présentation des concepts fondamentaux.
3. Applications croisées avec l'informatique
4. Études de cas.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) -
Examen semestriel en présentiel (100%).

Références bibliographiques : L'enseignant propose des références selon la matière choisie.

Intitulé de la matière : Intelligence artificielle générative Semestre: 3 Type : UEF1
VHS : 63h VHH : 3h Cours : 1h30 TD : 1h30 TP : 1h30
VHS travail personnel : 42h Coefficient : 3 Crédit : 5

Objectifs de l'enseignement

- Comprendre les principes théoriques et pratiques de l'IA générative.
- Étudier les principales familles de modèles génératifs (GANs, VAEs, Diffusion Models, LLMs).
- Savoir appliquer ces modèles à différents domaines (vision, langage, audio, multimodalité).
- Analyser les enjeux liés à l'éthique, la sécurité, la propriété intellectuelle et l'usage responsable.
- Développer une expertise pratique avec des frameworks modernes (PyTorch, TensorFlow, HuggingFace).

Connaissances préalables recommandées

Probabilités, statistiques, optimisation, algèbre linéaire, Machine Learning, Deep Learning, Python, bibliothèques ML (NumPy, PyTorch, TensorFlow).

Contenu de la matière Cours : 21h

1 Introduction à l'IA générative

- Définition, historique, et applications (vision, texte, audio, multimodal).
- Différence entre IA discriminative et générative.

2 Fondements probabilistes et mathématiques

- Variables aléatoires, distributions et estimation.
- Apprentissage probabiliste (maximum likelihood, divergence de Kullback-Leibler).
- Approximation variationnelle et Monte Carlo.

3 Modèles génératifs classiques

- Mixtures de Gaussiennes.
- Restricted Boltzmann Machines (RBM).
- Auto-encodeurs classiques et variationnels (VAE).

4 Réseaux Antagonistes Génératifs (GANs)

- Principe et architecture (Generator vs Discriminator).
- Variantes : DCGAN, WGAN, Conditional GAN, CycleGAN, StyleGAN.
- Applications : génération d'images, transfert de style, super-résolution.

5 Modèles de Diffusion

- Principe de la diffusion progressive et débruitage.
- Denoising Diffusion Probabilistic Models (DDPM).
- Stable Diffusion, DALL·E, MidJourney.
- Applications : images, vidéo, audio.

6 Grands Modèles de Langage (LLMs) • Architectures basées sur les Transformers.

- GPT, BERT, LLaMA, Mistral.
- Fine-tuning, prompt engineering, agents génératifs.

7 Multimodalité et Génération Avancée

- CLIP, Flamingo, Kosmos-1.
- Texte ↔ image ↔ audio ↔ vidéo.
- Applications : agents créatifs, synthèse multimodale.

8 Applications industrielles et cas d'usage

- Génération de contenu créatif (design, art, musique).
- Génération de données synthétiques pour l'entraînement.
- Médecine (imagerie synthétique, découverte de molécules).
- Jeux vidéo et réalité virtuelle.

9 Ateliers pratiques

- Implémentation d'un GAN simple en PyTorch.
- Fine-tuning d'un modèle génératif pré-entraîné (HuggingFace).
- Déploiement d'un modèle de génération dans un cas réel.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôles continus, travail personnel.

Références bibliographiques

- **Ian Goodfellow, Yoshua Bengio, Aaron Courville** *Deep Learning* – MIT Press, 2016 (classique, fondements).
- **David Foster**
Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play (2nd Edition) – O'Reilly, 2022.
- **Jakub Langr, Vladimir Bok**
GANs in Action: Deep learning with Generative Adversarial Networks (2nd Edition) – Manning, 2021.
- **Jakub M. Tomczak, Max Welling** *Deep Generative Modeling* – Springer, 2022.
- **Andrew Glassner**
Deep Learning: From Basics to Generative Models – Springer, 2021.
- **Philipp Singer et al.**
Diffusion Models: A Comprehensive Survey of Methods and Applications – ACM Computing Surveys, 2023.

Intitulé de la matière : Méta heuristiques et algorithmes évolutionnaires **Semestre :** 3 **Type :** UEF

VHS : 63h

VHH : 3h

Cours : 1h30

TD : 1h30

TP : 1h30

VHS travail personnel : 42h

Coefficient : 3

Crédit : 5

Objectifs de l'enseignement

- Introduire les **méthodes métaheuristiques** comme outils puissants pour résoudre des problèmes d'optimisation complexes (NP-difficiles, non linéaires, multimodaux).
- Étudier les **algorithmes évolutionnaires** (inspirés de la biologie) et les comparer aux méthodes d'optimisation classiques (descente de gradient, programmation convexe).
- Comprendre comment intégrer les métaheuristiques avec le **Machine Learning et le Deep Learning** (optimisation d'hyperparamètres, architecture search, feature selection).
- Développer la capacité à concevoir, implémenter et expérimenter des approches hybrides (métaheuristiques + apprentissage).
- Appliquer ces techniques à des domaines variés : vision par ordinateur, traitement du langage, robotique, logistique, bio-informatique.

Connaissances préalables recommandées

Probabilités et statistiques, optimisation, algèbre linéaire, **Machine Learning**, Python, bibliothèques ML (NumPy, PyTorch/TensorFlow), **processus stochastiques** et chaînes de Markov.

Contenu de la matière Cours

: 21h

1 Introduction aux métaheuristiques

- Définition et classification des algorithmes d'optimisation.
- Différences entre heuristiques, métaheuristiques et méthodes exactes.
- Applications en IA et en optimisation combinatoire.

2 Algorithmes évolutionnaires (EA)

- Algorithmes génétiques (GA) : codage, sélection, croisement, mutation.
- Stratégies d'évolution (ES).
- Programmation génétique (GP).
- Multi-objective Evolutionary Algorithms (NSGA-II, SPEA2).

3 Autres métaheuristiques inspirées de la nature

- Recuit simulé (Simulated Annealing).
- Optimisation par essaim de particules (PSO).
- Ant Colony Optimization (ACO).
- Artificial Bee Colony, Bat Algorithm, Firefly Algorithm.

4 Métaheuristiques et IA moderne

- Optimisation d'hyperparamètres via GA/PSO (ML & DL).

- Neural Architecture Search (NAS) avec EA.
- Feature selection par métaheuristiques.
- Hybridation ML/DL + métaheuristiques pour tâches complexes (ex. tuning de CNN, RNN).

5 Applications avancées

- Problèmes combinatoires : TSP, bin packing, job scheduling.
- Vision par ordinateur : réglage de CNN, data augmentation générée.
- NLP : optimisation de Transformers et embeddings.
- Robotique : contrôle adaptatif, navigation.
- Bio-informatique : alignement de séquences, prédiction de structures.

6 Évaluation et comparaison

- Mesures de performance : convergence, diversité, temps de calcul.
- Comparaison EA vs optimisation classique.
- Bonnes pratiques pour le design expérimental.

7 Outils et implémentation pratique

- Bibliothèques Python : DEAP, PyGAD, Nevergrad, Optuna.
- Frameworks pour NAS : AutoKeras, ENAS, DARTS.
- Études de cas pratiques (TP/Projet).

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôles continus, travail personnel.

Références bibliographiques

1. Eiben, A. E., & Smith, J. E. (2015). *Introduction to Evolutionary Computing*. Springer.
2. Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. Wiley.
3. Back, T., Fogel, D., & Michalewicz, Z. (2018). *Handbook of Evolutionary Computation*. CRC Press.
4. Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley.
5. Bäck, T., & Schwefel, H. P. (2013). *Evolutionary Computation: An Overview*. Elsevier.

Articles et ressources récentes

- Deb, K. et al. (2002). *A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II*. IEEE Transactions on Evolutionary Computation.
- Real, E. et al. (2019). *Regularized Evolution for Image Classifier Architecture Search*. AAAI.
- Elsken, T., Metzen, J.-H., & Hutter, F. (2019). *Neural Architecture Search: A Survey*. JMLR.
- Hansen, N. (2016). *The CMA Evolution Strategy: A Tutorial*. arXiv.
- OpenAI/DeepMind travaux récents sur l'optimisation évolutionnaire pour RL et DL

Intitulé de la matière : Apprentissage par renforcement **Semestre :** 3 **Type :** UEF
VHS : 63h **VHH :** 4h30 **Cours :** 1h30 **TD :** 1h30 **TP :** 1h30 **VHS travail**
personnel : 42h **Coefficient :** 3 **Crédit :** 5

Objectifs de l'enseignement

- Comprendre les fondements théoriques de l'apprentissage par renforcement.
- Maîtriser les concepts clés : agents, environnements, états, actions, récompenses, politique, fonction de valeur.
- Étudier les principales familles d'algorithmes (tabulaires, fonction d'approximation, deep RL).
- Explorer les extensions récentes : Deep Q-Networks, Policy Gradients, Actor-Critic, modèles multi-agents et hiérarchiques.
- Développer des compétences pratiques pour implémenter et évaluer des agents RL dans des environnements simulés (OpenAI Gym, PyBullet, Atari, MuJoCo).
- Analyser les défis actuels : échantillonnage efficace, exploration/exploitation, généralisabilité, robustesse et sécurité.

Connaissances préalables recommandées

Probabilités et statistiques, optimisation, algèbre linéaire, **Machine Learning**, Python, bibliothèques ML (NumPy, PyTorch/TensorFlow), **processus stochastiques** et chaînes de Markov.

Contenu de la matière Cours : 21h

1 Introduction à l'apprentissage par renforcement

- Définitions et applications (jeux, robotique, systèmes autonomes, optimisation industrielle).
- Différences avec apprentissage supervisé et non supervisé.
- Concepts clés : agent, environnement, récompense, politique, horizon.

2 Fondements théoriques

- Processus de Décision de Markov (MDP).
- Fonctions de valeur et fonction Q.
- Principe d'optimalité de Bellman.

3 Algorithmes classiques (tabulaires)

- Dynamic Programming (Value Iteration, Policy Iteration).
- Monte Carlo Methods.
- Temporal-Difference Learning : SARSA, Q-Learning.

4 Approximation de fonctions

- Représentation des politiques avec réseaux de neurones.
- Q-Learning avec approximation linéaire.
- Défis : instabilité, biais et variance.

5 Deep Reinforcement Learning

- Deep Q-Networks (DQN) : replay buffer, target networks.
- Extensions : Double DQN, Dueling DQN, Prioritized Replay.
- Applications sur Atari, contrôle continu.

6 Méthodes basées sur les politiques • Policy Gradient : REINFORCE.

- Actor-Critic.
- Algorithmes modernes : A2C, A3C, PPO (Proximal Policy Optimization), DDPG, SAC (Soft Actor-Critic).

7 Avancées et extensions

- Multi-agent Reinforcement Learning (MARL).
- Hierarchical RL.
- Exploration avancée (Intrinsic Motivation, Curiosity-Driven RL).
- Offline RL et Batch RL.
- Safe RL et Robust RL.

8 Applications récentes

- Jeux (AlphaGo, AlphaZero, MuZero).
- Robotique et contrôle.
- Systèmes de recommandation.
- Gestion des ressources dans le cloud et réseaux.

9 Enjeux et perspectives

- Défis : échantillonnage inefficace, généralisation, sécurité, explicabilité.
- Perspectives : intégration avec modèles génératifs et LLMs (ex. ChatGPT + RLHF).

10 Ateliers pratiques

- Implémentation de Q-Learning sur un problème simple (Gridworld).
- Implémentation d'un DQN avec PyTorch.
- Expérimentation avec PPO dans OpenAI Gym / MuJoCo.

Références bibliographiques

- **Richard S. Sutton & Andrew G. Barto**
Reinforcement Learning: An Introduction (2nd Edition, MIT Press, 2018).
- **Maxim Lapan**
Deep Reinforcement Learning Hands-On (3rd Edition, Packt, 2023).
- **Csaba Szepesvári**
Algorithms for Reinforcement Learning – Morgan & Claypool, 2020.
- **Lucian Busoniu, Robert Babuška, Bart De Schutter, Damien Ernst**
Reinforcement Learning and Dynamic Programming Using Function Approximators – CRC Press, 2017.

Intitulé de la matière : Reconnaissance des formes **Semestre :** 3 **Type :** UEF2
VHS : 42h **VHH :** 3h **Cours :** 1h30 **TD :** **TP :** 1h30
VHS travail personnel : 42h **Coefficient :** 3 **Crédit :** 3

Objectifs de l'enseignement

- Approfondir les fondements théoriques et pratiques de la reconnaissance des formes en lien avec le machine learning et le deep learning.
- Étudier les modèles probabilistes, géométriques et neuronaux pour la classification, le clustering et la détection de motifs.
- Relier les techniques classiques (statistiques, structurelles) aux approches modernes basées sur le deep learning (CNN, Transformers, autoencodeurs).
- Développer une capacité critique dans le choix des descripteurs, des architectures et des métriques d'évaluation selon le contexte applicatif.
- Appliquer les méthodes à des cas concrets (vision par ordinateur, biométrie, NLP, santé, cybersécurité).

Connaissances préalables recommandées

Algèbre linéaire, probabilités, statistiques, optimisation, programmation en **Python**, maîtrise de **NumPy**, **Scikit-learn**, **PyTorch/TensorFlow**, Machine Learning, Deep Learning

Contenu de la matière Cours : 21h

1 Introduction et rappels

- Définition, types et enjeux de la reconnaissance des formes.
- Rappel : pipeline typique (acquisition → prétraitement → features → classification → évaluation).
- Limites des approches classiques et transition vers l'IA moderne.

2 Représentation et extraction de caractéristiques pour la reconnaissance

- Features statistiques et structurelles (histogrammes, moments invariants, graphes).
- Réduction de dimension : PCA, LDA, t-SNE, UMAP.
- Features apprises automatiquement : autoencodeurs, embeddings.

3 Reconnaissance par les méthodes probabilistes et géométriques

- Classifieur bayésien, Maximum Likelihood, Maximum A Posteriori.
- Distance-based methods : k-NN, clustering (K-means, DBSCAN, Spectral Clustering).
- Mixtures Models (GMM, EM).
- Applications : vision, biométrie, reconnaissance faciale, parole.

4 Reconnaissance par réseaux neuronaux profonds

- CNN avancés pour la reconnaissance visuelle (ResNet, DenseNet, EfficientNet).

Transformers appliqués à la reconnaissance (ViT, Swin Transformer).

- Hybridation CNN + Transformers pour des tâches complexes.
- Applications : vision, biométrie, reconnaissance faciale, NLP.

5 Méthodes génératives et reconnaissance

- Autoencodeurs variationnels (VAE).
- GANs pour la génération et la reconnaissance (DCGAN, StyleGAN).
- Applications : génération d'images réalistes, data augmentation.

6 Évaluation et robustesse

- Métriques : précision, rappel, F1-score, AUC, Balanced Accuracy.
- Cross-validation et régularisation avancée.
- Explicabilité et interprétabilité en reconnaissance des formes (LIME, SHAP, Grad-CAM).
- Robustesse face aux données bruitées, déséquilibrées, adversariales.

7 Applications avancées

- Vision médicale (radiologie, pathologie numérique).
- Cybersécurité (détection d'intrusion, reconnaissance de malware).
- Reconnaissance vocale et biométrie multimodale.
- Recherche d'information multimédia.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôle continu, mini projet.

Références bibliographiques

1. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
2. Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern Classification*. Wiley.
3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
4. Aggarwal, C. C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer.
5. Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed.). Springer.

Articles et ressources récentes

- He, K., et al. (2016). *Deep Residual Learning for Image Recognition (ResNet)*. CVPR.
- Dosovitskiy, A., et al. (2021). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (ViT)*. ICLR.
- Liu, Z., et al. (2021). *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. ICCV.
- OpenAI (2023). *GPT-4 Technical Report* (pour la partie NLP multimodale).

Intitulé de la matière : Natural Language Processing Semestre : 3 Type : UEM
VHS : 63h VHH : 3h Cours : 1h30 TD : 1h30 TP : 1h30
VHS travail personnel : 42h Coefficient : 2 Crédit : 5

Objectifs de l'enseignement

- Comprendre les concepts fondamentaux du traitement du langage naturel et ses applications.
 - Acquérir les bases théoriques et pratiques des modèles statistiques et neuronaux pour le NLP.
- Maîtriser les techniques modernes de représentation du texte (word embeddings, contextual embeddings).
- Apprendre à concevoir, entraîner et évaluer des modèles pour des tâches variées : classification de texte, analyse de sentiments, reconnaissance d'entités nommées, traduction automatique, résumé automatique, question-answering.
- Découvrir les avancées récentes en NLP, notamment les modèles de grande taille (Transformers, LLMs).
- Préparer les étudiants à appliquer le NLP dans des domaines réels : santé, finance, recherche d'information, dialogue homme-machine.

Connaissances préalables recommandées

Probabilités, algèbre linéaire, statistiques de base, programmation en **Python**, structures de données, notions de complexité, Machine Learning, Notions de Deep Learning.

Contenu de la matière Cours : 21h

1 Introduction au NLP

- Défis du traitement automatique du langage (ambiguïtés, polysémie, contexte).
- Applications principales (chatbots, traduction, analyse de sentiments, recherche d'information, résumé).

2 Représentation du texte

- Modèles classiques : sac de mots, TF-IDF, n-grams.
- Word embeddings : Word2Vec, GloVe, FastText.
- Représentations contextuelles : ELMo, BERT, GPT, RoBERTa.

3 Méthodes de base en NLP

- Modèles probabilistes : modèles de langue n-gram, HMM.
- Parsing syntaxique (CKY, dépendances).
- Reconnaissance d'entités nommées (NER).

4 Deep Learning pour le NLP

- Réseaux récurrents (RNN, LSTM, GRU).
- Attention et mécanismes de mémoire.
- Séquence à séquence (Seq2Seq) et applications (traduction, résumé).

5 Les Transformers et modèles pré-entraînés

- Architecture Transformer (encoder-decoder, self-attention).
BERT et modèles dérivés (RoBERTa, DistilBERT, BioBERT).
- GPT et modèles génératifs (ChatGPT, LLaMA).
- Fine-tuning vs Prompt engineering vs In-context learning.

6 Applications avancées du NLP

- Traduction automatique neuronale (NMT).
- Analyse de sentiments et classification de texte.
- Question Answering et dialogue (QA, Chatbots).
- Résumé automatique (extractif vs abstractive).
- Multilinguisme et cross-lingual NLP.

7 Éthique et défis actuels en NLP

- Biais et équité dans les modèles de langage.
- Confidentialité, sécurité et hallucinations des LLMs.
- Vers des modèles explicables et responsables.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôles continus, travail personnel.

Références bibliographiques

1. Jurafsky, D. & Martin, J. H. (2023). *Speech and Language Processing* (3rd Edition, Draft). Prentice Hall.
2. Goldberg, Y. (2017). *Neural Network Methods for Natural Language Processing*. Morgan & Claypool.
3. Eisenstein, J. (2019). *Introduction to Natural Language Processing*. MIT Press.
4. Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). *Recent Trends in Deep Learning Based Natural Language Processing*. IEEE Computational Intelligence Magazine.

Ressources pratiques

1. Hugging Face Transformers (<https://huggingface.co>)
2. Stanford CS224N: *Natural Language Processing with Deep Learning* (cours actualisé chaque année).

Intitulé de la matière : Programmation par Contraintes **Semestre : 3** **Type : UEM**
VHS : 42h **VHH : 3h** **Cours : 1h30 TD : 1h30** **TP :**
VHS travail personnel : 21h **Coefficient : 2** **Crédit : 4**

Objectifs de l'enseignement

Ce cours vise à introduire les étudiants au paradigme de la **programmation par contraintes** (Constraint Programming), largement utilisé pour modéliser et résoudre des problèmes combinatoires complexes. Les objectifs sont :

- Comprendre la notion de problème de satisfaction de contraintes (CSP).
- Apprendre à modéliser un problème réel en variables, domaines et contraintes.
- Étudier les principales techniques de résolution (backtracking, filtrage, heuristiques).
- Appliquer la programmation par contraintes à des cas concrets : planification, emploi du temps, coloration de graphes, optimisation.

Connaissances préalables recommandées

Probabilités, algèbre linéaire, statistiques de base, programmation en **Python**, structures de données, notions de complexité, Machine Learning, Notions de Deep Learning.

Contenu de la matière Cours : 21h

1 Introduction à la Programmation par Contraintes

- Paradigmes de résolution : programmation déclarative vs impérative.
- Définitions : CSP (Constraint Satisfaction Problem), variables, domaines, contraintes.
- Applications réelles : planification, emploi du temps, configuration, allocation de ressources.

2 Modélisation des CSP

- Contraintes unaires, binaires, globales (AllDifferent, Cumulative, Circuit).
- Graphes de contraintes et représentation.
- Modélisation dans des solveurs (MiniZinc, OR-Tools).

3 Algorithmes de Résolution

- Recherche exhaustive et backtracking.
- Forward checking et propagation de contraintes.
- Consistance locale : arc-consistance (AC-3), chemin-consistance, k-consistance.
- Stratégies de recherche : MRV (Minimum Remaining Values), LCV (Least Constraining Value).

4 Heuristiques et stratégies de recherche

- Heuristique MRV (Minimum Remaining Values).
- Heuristique LCV (Least Constraining Value).
- Ordonnancement dynamique des variables.

5 Programmation par Contraintes et Optimisation

- Problèmes d'optimisation par contraintes (COP).
- Contraintes souples et pondérées.
- Stratégies d'optimisation globale : branch & bound, recherche locale, metaheuristiques (Tabu Search, recuit simulé).

6 CP et IA Hybride

Intégration **CP + ML/DL** :

- Prévision par ML puis optimisation par CP.
- Génération de contraintes à partir de modèles neuronaux.

- CP vs SAT/SMT : complémentarités.
- Applications hybrides :
 - Planification robotique (ML pour perception, CP pour planification).
 - Logistique intelligente (prédictions ML + optimisation CP).
 - Énergie intelligente (forecasting ML + allocation CP).

7 Applications pratiques avancées

- Emploi du temps universitaire.
- Problème des n-reines et TSP (voyageur de commerce).
- Planification de soins hospitaliers.
- Optimisation des ressources cloud et réseaux (cloud computing, edge computing).

8 Outils et frameworks

- Solveurs : OR-Tools (Google), Choco, Gecode, IBM CP Optimizer.
- MiniZinc : langage standard de modélisation.
- PyCSP3 : programmation par contraintes en Python.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre)

- **Examen semestriel en présentiel (60%).**
- **Évaluation continue (CC) (40%) :** Contrôles continus, mini projet.

Références bibliographiques

1. **Rossi, F., van Beek, P., & Walsh, T. (Eds.). (2006).** *Handbook of Constraint Programming*. Elsevier.
2. **Dechter, R. (2003).** *Constraint Processing*. Morgan Kaufmann.
3. **Apt, K. R. (2003).** *Principles of Constraint Programming*. Cambridge University Press.
4. **Marriott, K., Nethercote, N., Rafeh, R., Stuckey, P. J., et al. (2014).** *MiniZinc Handbook*. Springer.

Ressources en ligne

- Barták, R. (2021). *Constraint Programming: In Pursuit of the Holy Grail*. AI Communications.
- Lodi, A., & Zarpellon, G. (2017). *On learning and branching: a survey*. TOP.
- Kotthoff, L., Gent, I. P., & Miguel, I. (2016). *An evaluation of machine learning in algorithm selection for constraint solving*. AIJ.
- Google OR-Tools documentation (dernières versions).
- MiniZinc Challenge (compétition internationale, ressources actualisées).

Intitulé de la matière : Méthodologie de recherche **Semestre :** 3 **Type :** UET3
VHS : 21h **VHH :** 1h30 **Cours :** 1h30 **TD :** **TP :**
VHS travail personnel : **Coefficient :** 1 **Crédit :** 2

Objectifs de l'enseignement

Cette matière vise à enseigner aux étudiants les compétences et les connaissances nécessaires pour mener des recherches scientifiques efficaces. Il couvre les étapes clés de la recherche, de la formulation du sujet à la rédaction du rapport final, en passant par la revue de littérature, la conception de la recherche, la collecte et l'analyse des données, ainsi que la présentation orale des résultats.

Connaissances préalables recommandées : rédaction scientifique, langue utilisée.

Contenu de la matière Cours : 21h

1. Introduction à la recherche scientifique

- Définition et objectifs de la recherche scientifique.
- Types de recherche (fondamentale, appliquée, etc.).
- Rôle de la méthodologie dans la recherche.
- L'importance de la maîtrise de la langue utilisée.
- Éthique de la recherche en informatique : plagiat, auto plagiat, IA générative (ChatGpt, Deepseek, etc.)

2. Formulation du sujet de recherche

- Identification et sélection d'un sujet de recherche pertinent.
- Revue de littérature : o Recherche de sources d'information pertinentes (articles scientifiques, livres, etc.). o Analyse critique de la littérature existante (état de l'art).
 - o Identification des lacunes dans l'état de l'art.
- Définition de la problématique o Formulation d'une question de recherche claire et précise.
 - o Définition d'hypothèses de recherche.
- Objectifs de la recherche o Définition des objectifs spécifiques de la recherche cible.
 - o Identification des résultats attendus.

3. Conception de la recherche

- Choix des méthodes de recherche appropriées (expérimentale, qualitative, quantitative, etc.).
- Planification de la collecte des données (échantillonnage, outils de collecte, etc.).
- Élaboration d'un plan de recherche.

4. Collecte des données

- Mise en œuvre du plan de recherche.
- Utilisation des outils et techniques de collecte de données spécifiques à l'informatique.
- Gestion et organisation des données collectées.

5. Analyse des données

- Utilisation de logiciels d'analyse de données.
- Traitement et analyse des données collectées.
- Utilisation de méthodes statistiques ou d'autres techniques d'analyse appropriées. - Interprétation des résultats et discussion des implications.

6. Rédaction du rapport de recherche

- Modes et structures de publication : article, brevet, thèse, livre, poster, communication orale...
- Utilisation de logiciels de rédaction : LaTeX, overleaf, etc.
- Structure et organisation d'un rapport de recherche scientifique.
- Rédaction claire et concise des différentes parties du rapport.
- Présentation des résultats et discussion des conclusions.

- Rédaction et outils de gestion des références bibliographiques (APA, IEEE, Zotero, EndNote, etc.) et des annexes.

7. Présentation orale des résultats

- Préparation et réalisation d'une présentation orale des résultats de recherche.
- Maîtrise de la communication scientifique.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) -
Examen semestriel en présentiel (100%).

Références bibliographiques

-Beaud, Michel (2020). L'art de la thèse. La Découverte.

-Stefan Kottwitz (2021). LaTeX Beginner's Guide: Create visually appealing texts, articles, and books for business and science using LaTeX. 2nd Edition, Packt Publishing.

-Jean-Marie Dubois (2005). La rédaction scientifique : mémoires et thèses : formes régulières et par articles. Estem.

-Michèle Lenoble-Pinson (1996). La rédaction scientifique : conception, rédaction, présentation. Signalétique, De Boeck Université.

-Christine Gérard, Jean Germain (1985). Recherche bibliographique et documentaire : généralités. Faculté de philosophie et Lettres.

Intitulé de la matière : Une matière au choix **Semestre : 3 Type : UED3**
VHS : 21h VHH : 1h30 Cours : 1h30 TD : TP :
VHS travail personnel : Coefficient : 1 Crédit : 1

Objectifs de l'enseignement : Cette matière permet aux étudiants d'acquérir une vision plus large de leur domaine d'études et favorise une meilleure compréhension des enjeux sociaux, économiques et éthiques liés à l'informatique, ainsi qu'une capacité à communiquer efficacement et à s'adapter à des situations diverses qui peuvent se présenter. En d'autres termes, cette matière permet :

- Une ouverture à d'autres disciplines en relation avec le quotidien, afin de comprendre l'importance de la collaboration dans un environnement pluridisciplinaire.
- L'élargissement des horizons, ce qui favorise la compréhension de la pensée humaine dans le temps et l'espace.
- Développement de compétences transversales : communication, vision holistique, analyse et synthèse, capacité d'adaptation, etc.

Connaissances préalables recommandées

Contenu de la matière Cours : 21h

1. Introduction à la discipline concernée (matière choisie).
2. Présentation des concepts fondamentaux.
3. Applications croisées avec l'informatique
4. Études de cas.

Mode d'évaluation (doit être porté à la connaissance des étudiants en début de chaque semestre) -
Examen semestriel en présentiel (100%).

Références bibliographiques : L'enseignant propose des références selon la matière choisie.

V- Accords ou conventions

NON

(Si oui, transmettre les accords et/ou les conventions dans le dossier papier de la formation)

VI - Avis et Visas des organes administratifs et consultatifs

Intitulé du Master : Algèbre et Mathématiques discrètes

| Comité Scientifique de département |
|---|
| Avis et visa du Comité Scientifique : |
| Date : |
| Conseil Scientifique de la Faculté |
| Avis et visa du Conseil Scientifique : |
| Date : |
| Doyen de la faculté |
| Avis et visa du Doyen ou du Directeur : |
| Date : |
| Conseil Scientifique de l'Université |
| Avis et visa du Conseil Scientifique : |
| Date : |

VIII - Visa de la Conférence Régionale

(Uniquement à renseigner dans la version finale de l'offre de formation)