2023/2024

Rapport Projet Data Science

RETOURS TEXTUELS DES INGÉNIEURS DE POLYTECH MONTPELLIER

PRÉSENTÉ PAR

Alexandre Deloire - Rémi Jorge

Jiayi He - Charlène Morchipont

Contexte

L'École d'ingénieur Polytech Montpellier effectue des sondages auprès des anciens élèves afin de recueillir des informations vis à vis de leurs études et insertion professionnelle. Une étude à 6 mois, à 18 mois et à 30 mois des jeunes diplômés est réalisée par l'école.

A part des questions à choix unique ou multiple, des retours textuels sont aussi exprimés par les diplômés de Polytech Montpellier lors de ces enquêtes. Ces données ont été jusqu'à présent peu étudiées.

Nous avons choisi d'extraire les réponses textuelles, de les organiser et de les analyser.

Sujet

Nous allons étudier les retours textuels des anciens élèves de l'École d'ingénieur Polytech Montpellier concernant leur formation et leur insertion professionnelle, collectés lors d'enquêtes menées à 6 mois, 18 mois et 30 mois après l'obtention de leur diplôme.

Problématique

Beaucoup des réponses textuelles abordent les retours sur la formation, en particulier les cours. Les anciens étudiants expriment leurs opinions sur les cours qu'ils jugent utiles, inutiles, absents, et ceux qui auraient pu être bénéfiques ou nécessitent un renforcement afin d'avoir une meilleure insertion professionnelle. Nous proposons d'étudier ces réponses afin de déterminer, pour chaque filière, quels étaient les cours utiles, inutiles, etc.

Objectifs

- Extraire les données des enquêtes
- Nettoyer et organiser les données textuelles collectées auprès des diplômés pour éliminer les erreurs et incohérences.
- Faire une analyse globale des données pour dégager les méthodes appropriés d'analyse
- Implémenter les méthodes d'analyse
- Faire une analyse fine des données textuelles et y dégager des résultats
- Proposer une interface interactive aux décideurs

Introduction

Dans les couloirs de Polytech, on entend souvent des élèves se plaindre de certains cours, tandis que d'autres auraient souhaité un renforcement de certaines matières. En effet, les cours constituent une préoccupation majeure des élèves, car la raison principale de fréquenter Polytech est évidemment l'enseignement dispensé. L'analyse de ces données est donc un atout indispensable pour mieux comprendre l'opinion des élèves, permettant ainsi de prendre des décisions concernant les cours spécifiques à chaque filière et, par conséquent, d'améliorer la formation à Polytech.

Dans un premier temps, il est impératif d'avoir une vue d'ensemble des données dont nous disposons, car nous ne connaissons rien sur ces informations. Voici comment nous avons procédé pour cette étape :

Présentation des données

Nous avons téléchargé les données. Voici quelques caractéristiques à leur sujet, certaines posent d'entrée des problèmes :

- Les données sont stockées dans des fichiers distincts.
- Les données sont au format Excel, mais dans différents formats (datant de différentes années).
- L'encodage des fichiers n'est pas le même, certains sont en latin-1, d'autres en utf-8.

Nous avons développé un programme Python pour extraire, pour chaque fichier, une liste de ces questions ainsi que les caractéristiques de chaque question, notamment :

- La question
- Le type de réponse
- Des informations générales sur les réponses, par exemple si c'est une catégorie ou binaire : l'énumération des valeurs possibles, si c'est un nombre, le min et le max etc.
- Le nombre de réponses
- Le nombre de réponses manquantes
- Le nombre de valeurs uniques

Grâce aux informations extraites, nous avons pu discuter des meilleures questions à utiliser et des méthodes que nous allions employer pour cette étude textuelle. Le programme d'extraction des questions est fourni en annexe.

Ces questions ont retenus notre attention :

- Quels enseignements vous semblent les plus utiles pour l'exercice de votre métier et votre insertion professionnelle ?
- Parmi les enseignements fournis par l'école, quels sont ceux qui mériteraient d'être approfondis ou renforcés ?
- Quels enseignements, absents de votre formation, vous auraient été utiles ?
- Quels enseignements, présents dans votre formation, vous paraissent inutiles ?

Avant d'appliquer les méthodes d'analyse présentées dans ce rapport, il était nécessaire de nettoyer les données qui étaient presque inutilisables. Dans la suite de cette partie on va discuter les problèmes rencontrés et les solutions mises en avant.

Extract, Transform, Load (ETL)

Les formats Excel de différentes années posent problème, car l'encodage des fichiers n'est pas le même. Pire que cela, le fichier de 2020 comprend des erreurs d'encodage qui rendent les phrases avec des accents presque irrécupérables. Voici un exemple de phrase que l'on peut trouver :

“ la comptabilitÈ matiËre, la comptabilitÈ en gÈnÈral, du droit juridique ”

On remarque un problème d'encodage, avec les accents aigus devenant majuscules et graves.

Un autre problème est que les fichiers ont traversé plusieurs systèmes d'exploitation, entre Windows, macOS et Linux, et on retrouve des erreurs de lecture de caractères de fin d'échappement "\r" et "\n", ce qui se traduit, en plus de l'erreur d'encodage, par l’apparition de "_x000D_".

De plus, entre chaque fichier, la formulation des questions change légèrement, avec parfois l'ajout d'un mot, la suppression, voire l'apparition de nouvelles questions et la disparition d'autres. Il faut donc homogénéiser les questions pour que lors des analyses, on puisse retrouver les mêmes questions entre les différents fichiers.

Ensuite, il est nécessaire de convertir tous les fichiers dans un format homogène et récent.

Pour ce faire, nous avons créé un script Python, nommé "clean_up_excel.py", qui se charge de ce nettoyage et de cette conversion. Une fois le nettoyage effectué, il est essentiel de disposer d'un moyen simple d'extraire les données et les colonnes que nous souhaitons traiter dans un format CSV, facilitant ainsi l'analyse et les entraînements. À cet effet, nous avons créé un script Python nommé "xls_to_csv.py", paramétrable avec les noms des colonnes, qui extrait toutes les données nécessaires dans un nouveau fichier.

Une fois cela fait, nous pouvons maintenant commencer à exploiter les données pour effectuer des analyses.

Aperçu des données

Il est important, avant toute analyse approfondie, d'obtenir quelques informations de base sur nos données. Nous avons créé des graphiques simples pour comprendre précisément en quoi consistent les réponses. Les décideurs se posent exactement les mêmes questions.

Par exemple :

- Combien y a-t-il de réponses par année, par filière, par sexe ?
- Combien d'entre eux ont répondu aux questions textuelles ? etc...

En analysant les graphiques, nous remarquons que chaque filière est bien representée ainsi que chaque sexe chaque année, ce qui nous indique que nous pouvons tirer de vraies conclusions de nos analyses appronfondies ci-dessous.

Nous avons aussi fait quelques graphiques pour des questions non textuelles pour pouvoir les mettre en lien plus tard avec les résultats des questions textuelles.

Les graphiques sont fournis avec notre outil de prise de décision interactif et en annexe.

Nuages de Mots

L'analyse de données textuelles est un travail complexe, il ne s'agit pas de chiffres mais plutôt une accumulation de mots qui ensemble créent un sens. L'objectif est donc de pouvoir représenter de la manière la plus simple, la plus intuitive, les données pour que l'utilisateur de notre application puisse avoir un retour fiable de son questionnaire pour notre cas.

Ainsi, nous avons décidé que des nuages de mots seraient une excellente façon de représenter l'opinion des élèves sur les enseignements de leur formation. Nos nuages de mots concernent deux questions distinctes, très pertinentes pour l'analyse :

- Quels enseignements vous semblent les plus utiles pour l'exercice de votre métier et votre insertion professionnelle ?
- Parmis les enseignements fournis quels sont ceux qui mériteraient d'être approfondis ?

Les enseignements délivrés à Polytech sont divers et variés. Certains enseignements sont communs à plusieurs filières mais diffèrent légèrement sur le contenu. Plusieurs obstacles ont dû être surmontés pour arriver à un résultat satisfaisant.

La construction des nuages a été réalisée en plusieurs étapes :

- Privilégier la cohérence des mots en s'informant au préalable sur les enseignements des différentes filières. Avant toute chose, il était impératif de s'informer sur les matières enseignées au sein des autres filières. Repérer les enseignements composés de plusieurs mots ("mécanique des fluides", "chimie organique"...). Trouver les abréviations possibles afin de faire correspondre les mêmes mots (rdd -> résistance des matériaux).
- Dans un deuxième temps il a fallu réaliser des listes de mots, regrouper les enseignements d'une même UE ou d'un même thème. Cela permet de comptabiliser les mots qui signifient la même chose ensemble et de rendre leur poids plus fiable et également cela permet une meilleure visibilité.
- Une fois ce travail fait, il s'agit de la phase de peaufinage. L'objectif était de passer de regroupements conséquents, à des regroupements plus petits et qui permettent de réellement avoir les sous-idées utiles plutôt qu'une nouvelle visualisation des plaquettes des enseignements.
- La dernière étape a été de récupérer, pour chaque filière, le top 5 des enseignements les plus cités.

Les outils utilisés :

- Expressions régulières pour identifier les matières composées de plusieurs mots
- Librairie NLTK, stopword, afin d'ignorer les mots de liaison dans un texte comme : "et", ce sont des mots qui sont souvent répétés donc les garder aurait posé des soucis

Présentation des résultats

Le nuage de mots présentant les enseignements utiles pour l'insertion professionnelle est très intéressant. On remarque réellement des matières qui ressortent, comme par exemple en IG, le web et le software qui sont effectivement des enseignements clés de la formation. Aucune incohérence des réponses n'est visible et on a une bonne compréhension pour chaque filière des enseignements les plus importants.

Pour le nuage de mots présentant les enseignements qui mériteraient d'être approfondis, on remarque que les matières principales s'y retrouvent également et sont souvent à approfondir.

Pour avoir des résultats plus précis et plus détaillés il est intéressant de regarder les résultats de la méthode k-means que nous avons utilisé également.

Analyse de la satisfaction

Une fois les nuages de mots réalisés, nous avons souhaité connaître les tendances générales que reflétaient les commentaires écrits. La question que nous avons souhaité étudier est :

" Vos remarques et commentaires relatifs à votre insertion professionnelle. "

En effet, il est intéressant de savoir si les retours sont plutôt positifs ou plutôt négatifs car l'insertion professionnelle est liée aux enseignements. Ceux-ci nous orientent plus ou moins dans des domaines et les connaissances acquises ou non influencent grandement la réussite dans le monde professionnel. Grâce à ces données, on va pouvoir repérer facilement s'il y a un problème dans une filière, peut-être que les enseignements ne sont plus à la hauteur des attentes du monde du travail, etc...

Obstacles rencontrés

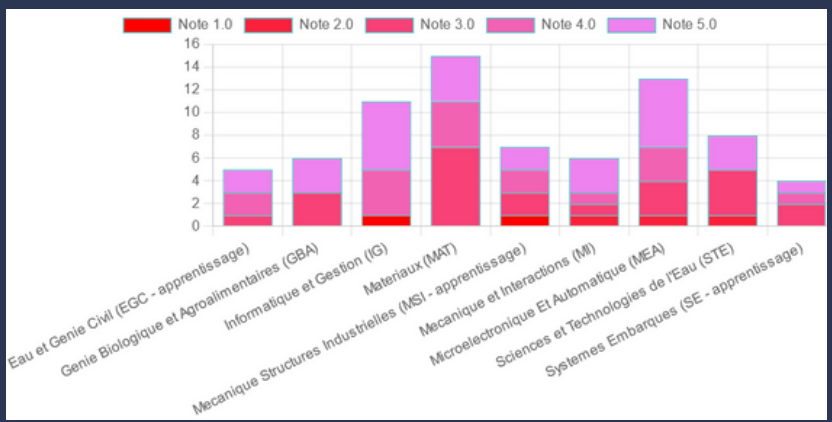
Pour cette analyse, nous n'avons pas de base de connaissance comme nous avions pour les nuages de mots car les commentaires sont très aléatoires. Il a donc fallu trouver un moyen de traiter cette grande masse de données de manière efficace et trouver le sens juste aux commentaires.

La solution trouvée pour y parvenir est l'utilisation d'un réseau de neurones pré-entrainés pour répondre à l'analyse de sentiment. Il en existe plusieurs mais ils ne sont pas tous fiables et trouver un outil précis et cohérent n'a pas été simple.

Nous étions partis dans un premier temps sur l'utilisation de Textblob et NLTK. Malheureusement les résultats de nos tests de fiabilité n'étaient pas concluants. Un commentaire très élogieux était classé neutre, de même pour un commentaire très négatif. Il a donc fallu se renseigner et tester d'autres méthodes possibles avant de trouver le bon outil.

En finalité nous avons utilisé le module transformers avec pipeline qui nous a permis d'avoir un classement des commentaires allant de 1 étoile (négatif) à 5 étoiles (positif). Les résultats étaient d'une très bonne fiabilité et nous avons pu réalisé un graphique qui représente la satisfaction d'un point de vue insertion professionnelle.

Résultat



Graphique représentant la satisfaction des étudiants vis-à-vis de l'insertion professionnelle selon les commentaires écrits

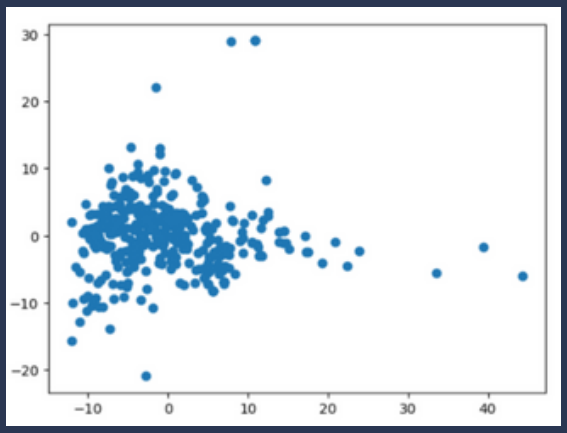
Les résultats obtenus sont très satisfaisants car on a le moyen de réellement quantifier la satisfaction. Grâce à cela, on peut se focaliser sur l'analyse d'une filière en particulier qui aurait reçu plus de retours négatifs et diagnostiquer le problème à l'aide des nuages de mots décrits précédemment. Pour 2022, on remarque par exemple que 50% des réponses de la filière MAT sont de 3 étoiles, il serait donc intéressant d'analyser précisément cette filière afin de voir ce qui fait que cette note a été attribuée.

Classification

Comment aborder l'analyse d'une grande quantité de données textuelles ? L'objectif est de regrouper les commentaires pour identifier les similitudes et obtenir une vision globale des avis exprimés. Nous sommes donc confrontés à un problème de classification.

Conversion du texte en vecteur

Après la sélection et le nettoyage des données, la première étape consiste à transformer le texte en une forme de données compréhensibles par un ordinateur. Pour ce faire, nous convertissons le texte en un vecteur, également appelé embedding. Nous utilisons le réseau de neurones entraîné par spacy pour le français, en optant pour sa version large. Après avoir comparé plusieurs réseaux neuronaux, la version large de spacy s'est avérée performante pour cette étape, ce qui a motivé notre choix. Le résultat est un vecteur en 300 dimensions.

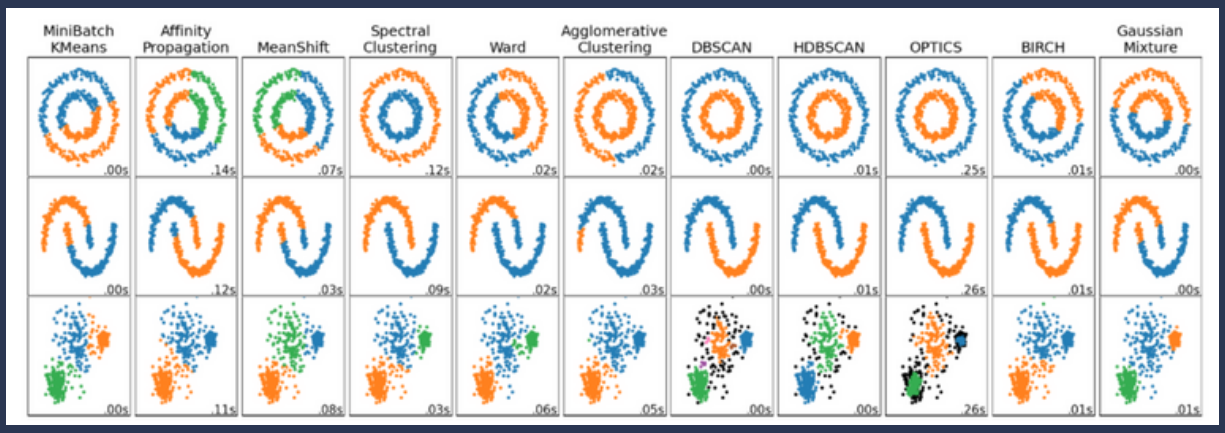


Afin d'obtenir une représentation visuelle, nous utilisons une Analyse en Composantes Principales (ACP) pour réduire la dimension à 2. Dans l'exemple ci-dessous, portant sur la question : "Quels enseignements vous semblent le plus utiles dans l'exercice de votre métier et votre insertion professionnelle ?", nous avons sélectionné l'échantillon STE.

Visualisation des vecteurs projetés en 2 dimensions

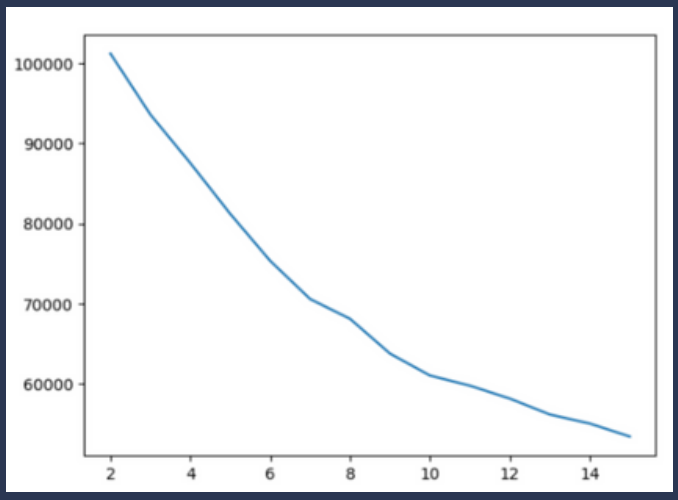
Choix de la méthode et paramètre

La prochaine étape consiste à choisir le modèle de classification. Plusieurs algorithmes existent pour la classification non supervisée. Bien que nous connaissions déjà l'algorithme des K-moyennes (ou KMeans), nous avons souhaité comparer ses performances à d'autres méthodes. Après divers essais avec des données variées, les résultats indiquent que les K-moyennes offrent parmi les meilleures performances de classification, justifiant notre choix.



Différentes méthodes de classification non-supervisée et leur performance en fonction du type de donnée

La méthode des K-moyennes nécessite de définir préalablement le nombre de clusters à former. Il s'agit d'un paramètre crucial, car un choix inapproprié peut entraîner des résultats médiocres. Nous avons exploré plusieurs méthodes pour calculer le nombre optimal de clusters. La première méthode consiste à examiner l'inertie totale moyenne en fonction du nombre de clusters. Puisque les K-moyennes visent à minimiser l'inertie, cela s'avère un bon indicateur. Il suffit ensuite de rechercher un "coude" pour déterminer le nombre optimal.

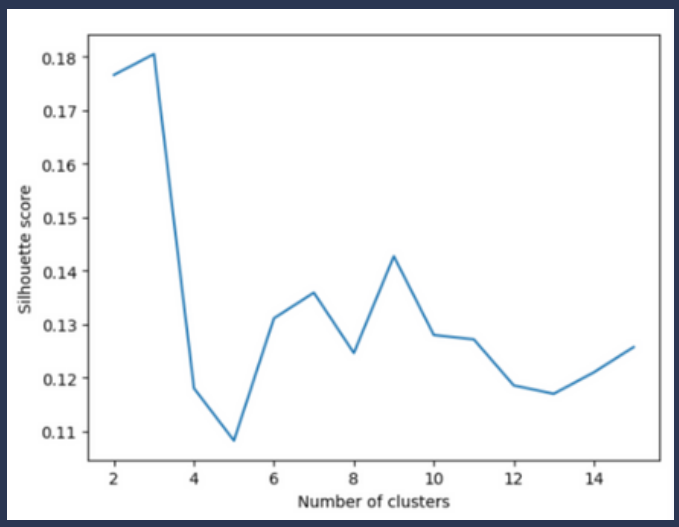


$$\sum_{i=0}^{n} \min_{\mu_j \in C}(||x_i - \mu_j||^2)$$

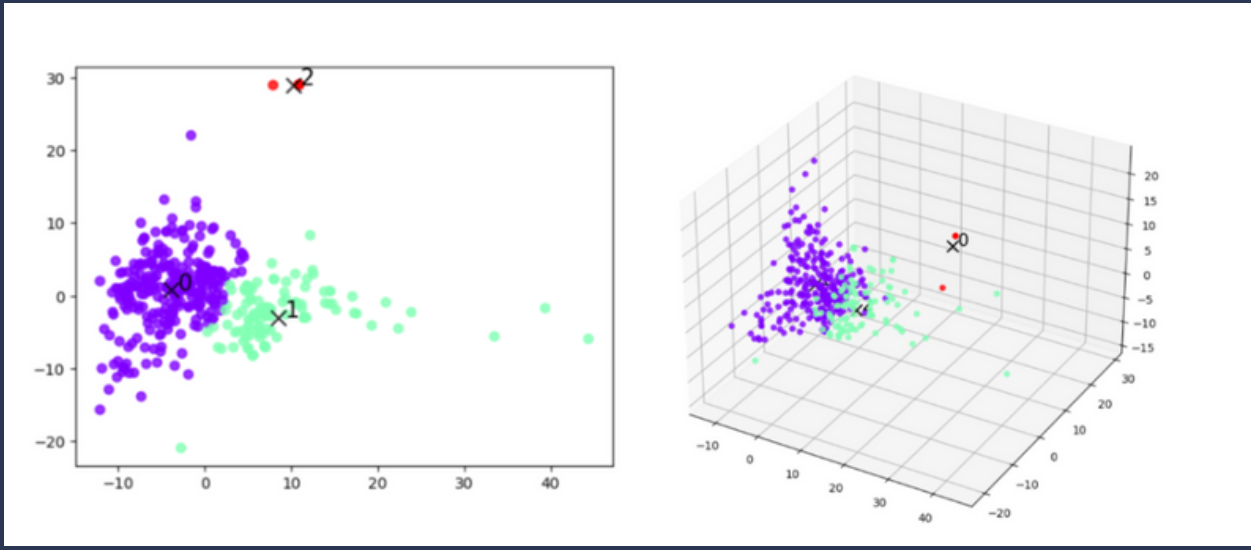
Visualisation de l'inertie en fonction du nombre de cluster et formule de l'inertie utilisé dans K-moyenne.

La deuxième méthode utilise la fonction silhouette_score() de la bibliothèque scikit-learn. Elle évalue l'efficacité d'une classification entre -1 et 1 (-1 étant le moins bon), en mesurant la proximité des clusters les uns par rapport aux autres.

Performance de l'algorithme K-moyenne en fonction du nombre de cluster



Après cette analyse, nous déterminerons un nombre optimal de clusters en fonction de nos objectifs. Dans notre exemple, nous choisissons 3 clusters. Cependant, les K-moyennes sont un algorithme non déterministe. En effet l'algorithme choisi derrière est l'algorithme de Lloyd-Max et la première étape consiste à choisir aléatoirement le centre des clusters. Cette simple étape ne permet pas de reproduire l'expérience à moins de définir un état fixé avec une seed. Pour optimiser la reproductibilité et améliorer le résultat, nous utilisons une seed pour définir un état initial fixe. Ainsi nous obtenons le résultat suivant :



Visualisation des clusters en 2 dimensions et 3 dimensions

Lors de l'observation des commentaires par clusters, nous obtenons :

- Cluster 1 (taille 265 commentaires) : “hydraulique fluviale”, “Hydrologie Hydraulique”, “Génie des procédés Hydraulique urbaine”, “génie des procédés”, ...
- Cluster 2 (taille 118 commentaires) : “Insertion Pro”, “Excel”, “Communication”, “Ecologie”, ...
- Cluster 3 (taille 5 commentaires) : “Tous”, “Tous”, “stage”, “Tous”, “Tous”.

Cette analyse montre que les commentaires semblent bien regroupés selon leurs idées. Le cluster 3 est rapide a analysé du fait de sa petite taille. Cependant ce n’est pas le cas du cluster 1 et 2 comprenant plus des centaines de commentaires. Il serait encore trop long de vérifier cela manuellement.

Génération de la description des clusters

Pour accélérer l'analyse, nous utilisons l'API de ChatGPT 4 pour générer automatiquement un titre et une description pour chaque cluster. Cela permet de comprendre les points communs au sein d'un cluster et de différencier rapidement les groupes. En examinant le nombre de réponses pour chaque cluster, nous obtenons rapidement l'opinion des élèves.

Une étape cruciale est la création du prompt pour ChatGPT. Bien que cette partie ne pose pas de difficulté majeure, il est essentiel de garantir une saisie de qualité pour obtenir des résultats fiables. Nous lui fournissons une brève description de ce que nous souhaitons, un titre, une description pour chaque cluster, ainsi que la question à laquelle les diplômés ont répondu. Nous spécifions également le format de la réponse attendue (un JSON) et les attributs nécessaires, tout en limitant à une trentaine de messages maximum par cluster pour éviter une surcharge d'informations.

Une fois le prompt créé, la requête est envoyée et la réponse est associée à chaque cluster. Pour l'exemple des formations utiles en STE, nous obtenons les descriptions suivantes :

- Cluster 0
  - Titre : Hydraulique, traitement des eaux et logiciels associés
  - Description : Ce cluster regroupe les commentaires qui mettent en avant l'importance des enseignements liés à l'hydraulique, au traitement des eaux et aux logiciels utilisés dans ces domaines. Les commentaires soulignent aussi la valeur des sorties terrain, des stages et des projets concrets pour se familiariser avec le monde du travail et les problématiques réelles

- Cluster 1
  - Titre : Compétences transversales et outils informatiques
  - Description : Ce cluster regroupe les commentaires qui mettent en avant l'importance des compétences transversales telles que la communication, la gestion de projet, le droit du travail ou l'anglais. Les commentaires mentionnent aussi l'utilité des outils informatiques tels que Excel, QGIS, PCSWMM ou la programmation. Les commentaires insistent sur le savoir-être et le fonctionnement des entreprises

- Cluster 2
  - Titre : Tous les enseignements ou les stages
  - Description : Ce cluster regroupe les commentaires qui ne se focalisent pas sur un enseignement ou un domaine en particulier, mais qui considèrent que tous les enseignements ou les stages sont utiles et pertinents pour l'exercice du métier et l'insertion professionnelle.

Résultat

Nous avons analysé trois questions à l'aide de cette méthode : Quels enseignements vous semblent les plus utiles pour l'exercice de votre métier et votre insertion professionnelle ? Quels enseignements, absents de votre formation, vous auraient été utiles ? Quels conseils pourriez-vous donner aux étudiants actuellement en formation pour bien choisir leur stage de fin d'étude et réussir leur insertion professionnelle ? Nous présentons les réponses les plus fréquentes à ces questions ci-dessous. Les calculs ont été effectués pour chaque filière, le pourcentage indique la proportion de commentaire appartenant au cluster.

1. Quels enseignements vous semblent les plus utiles pour l'exercice de votre métier et votre insertion professionnelle ?

Toutes les filières : Enseignements lié aux calculs et la conception (30%), Enseignements techniques variés (11%), Enseignements professionnels ou linguistique (14%) ...

IG : Application pratique et savoir-faire (46%), Enseignements informatique et transversale (15%), ...

GBA : Spécialisation et compétences techniques (35%), Diversité et application des enseignements (31%), ...

MAT : Compétences techniques spécifiques (31%), Eco-conception et travail technique (30%), ...

MEA : Système embarquée et électronique (29%), Programmation et projet technique (17%), ...

MI : Enseignement technique et informatique (23%), Enseignement transversal et gestion de projet (17%), ...

STE : Application spécifiques et compétences diverses (67%), Hydraulique et Hydrologie (32%), ...

Pour les enseignements les plus utiles la réponse varie beaucoup en fonction de la filière. En effet, les réponses majoritaires sont liées aux compétences techniques et spécifiques. Les enseignements plus généraux comme la communication es langues ou l'informatique sont aussi très fréquents.

2. Quels enseignements, absents de votre formation, vous auraient été utiles ?

Toutes les filières : Outils, technologie et méthodes modernes (49%), Compétence techniques et financières (35%), Aspects managériaux, entrepreneuriaux et personnels (12%), ...

IG : Enseignements liés au code et la sécurité (34%), Enseignements variés et complémentaires (31%), Enseignements plus approfondis ou plus théoriques (20%) ...

GBA : Innovation, environnement et vie professionnelle (69%), management et compétences transversales (21%) ...

MAT : Compétences industrielles et techniques (54%), Expérience pratique et la sensibilisation industrielle (8%), ...

MEA : Informatique, électronique et développement durable (57%), Conception et communication (26%), ...

MI : Sécurité industrielle et durabilité (48%), Compétence spécifique et complémentaire (26%), ...

STE : Besoin en matière spécifique (63%), Compétences techniques (33%), ...

Pour les enseignements absents c'est une fois encore des compétences spécifiques aux formations. Cependant un ressenti global serait d'avoir des enseignements plus approfondis.

3. Quels conseils pourriez-vous donner aux étudiants actuellement en formation pour bien choisir leur stage de fin d'étude et réussir leur insertion professionnelle ?

Toutes les filières : Importance de la passion et la curiosité (31%), Préparation et réflexion à long terme (19%), ...

IG : Stage aligné avec les objectifs professionnels (31%), Stage pour développer ses compétences (18%), ...

GBA : Stage en fonction du secteur d'activité, service et mission (34%), en fonction de ses envies, ses besoins (35%), ...

MAT : S'y prendre à l'avance et choisir son stage en fonction de son projet professionnel (36%), Bien se renseigner sur les entreprises et le métiers (25%), ...

MEA : Être ambitieux, proactif et ambitieux (28%), Faire ce qui plaît et sortir de la zone de confort (24%), ...

MI : Stage en France (38%), Projet professionnel et recherche d'entreprise (28%), ...

STE : Choisir son stage en fonction de ses intérêts et perspectives (45%), Utiliser le réseau des anciens élèves et enseignants (15%), ...

Un conseil que beaucoup de diplômés donnent est de choisir son stage en fonction de nos envie et perspective futur, que cela soit cohérent avec notre projet futur.

Réseaux de Neurones : PolyChat le “ChatGPT” de Polytech

Ne serait-il pas idéal de disposer d'un ChatGPT spécifique à Polytech, entraîné sur les données souhaitées, telles que les résultats des sondages? En discutant avec ce ChatGPT, il pourrait potentiellement fournir des réponses liées aux résultats des sondages. Par exemple, se connecter à PolyChat, poser la question " Quelle filière a le plus besoin de cours de Gestion? " et PolyChat nous répondrait. C'est l'idée à partir de laquelle nous sommes partis.

Nous avons effectué des recherches approfondies et avons discuté avec notre professeur en Deep Learning. Nous avons constaté que nous ne disposions ni de suffisamment de données ni de la puissance de calcul ou de l'investissement nécessaires pour créer notre propre version de ChatGPT. Cependant, nous n'avons pas abandonné et nous nous sommes orientés vers un réseau de neurones moins puissant mais fonctionnel, qui nous aiderait à analyser les données.

Nous avons construit un réseaux de neurons avec succès et nous avons entraîné 4 modèles que nous avons nommés :

- **PolyChatA**
- **PolyChatU**
- **PolyChatI**
- **PolyChatR**

Le but de chaque modèle est :

- **PolyChatA:** Répondre aux questions du type: “Quelle est la filière pour laquelle un cours de gestion/programmation/anglais/etc. était absent et aurait été utile ?
- **PolyChatU:** Répondre aux questions du type: “Quelle est la filière pour laquelle le cours de gestion/programmation/anglais/etc. était utile ?
- **PolyChatI:** Répondre aux questions du type: “Quelle est la filière pour laquelle le cours de gestion/programmation/anglais/etc. était inutile ?
- **PolyChatR:** Répondre aux questions du type: “Quel est la filière pour laquelle le cours de gestion/programmation/anglais/etc. devrait être renforcé ou approfondi ?”

Nous avons entraîné ces modèles sur les données textuelles des questions, car chaque réponse est étiquetée avec la filière de provenance.

Nous allons expliquer ci-dessous les fonctionnalités des modèles et comment nous les avons construits.

Tokenisation des données textuelles : NLTK et keras

Avant de concevoir n'importe quel réseau de neurones, il est essentiel de procéder à la tokenisation des données textuelles. Tokeniser les données consiste à diviser un texte en unités plus petites appelées tokens, généralement des mots, afin de les représenter de manière structurée. Cette étape est cruciale pour les réseaux de neurones car elle transforme le texte en séquences d'entiers, facilitant ainsi le traitement par les modèles qui travaillent avec des données numériques plutôt qu'avec des textes bruts. Dans notre cas, nous avons utilisé le tokeniseur de **NLTK** (Natural Language Toolkit), une bibliothèque de traitement du langage naturel, pour tokeniser l'ensemble des données textuelles. Ensuite, nous avons converti ces données tokenisées en séquences numériques à l'aide du tokeniseur de **Keras**, une bibliothèque de machine learning de **TensorFlow** développée par Google.

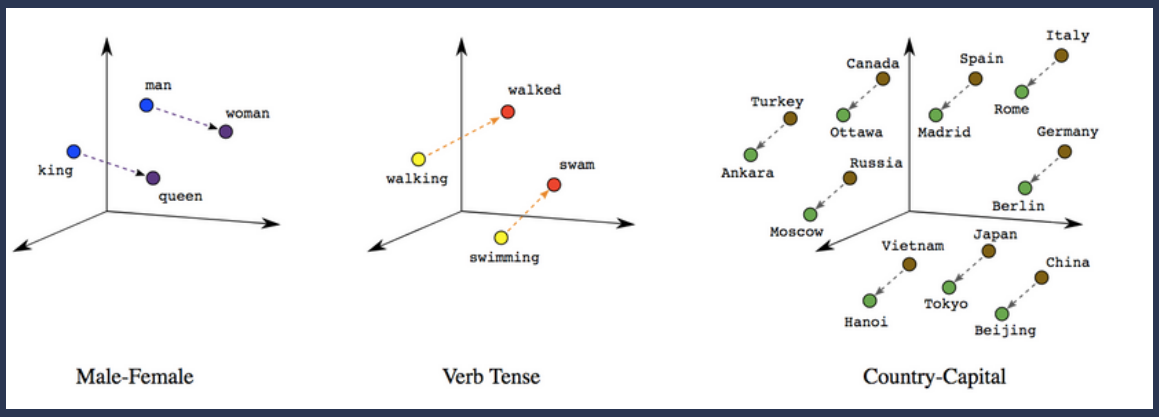
Cette étape a pour objectif de préparer les données textuelles en vue de les utiliser en tant qu'entrées pour le réseau de neurones.

Il est important d'utiliser le même tokeniseur pour traiter de nouvelles entrées textuelles une fois que les modèles sont entraînés. Ainsi, pour chaque modèle, nous avons veillé à enregistrer la configuration du tokeniseur.

Réseau avec TensorFlow

Notre objectif était de respecter l'objectif ultime idéaliste du Deep Learning, c'est-à-dire de construire un réseau qui puisse s'adapter à n'importe quelle question. Ainsi, le modèle final dépend uniquement des données et non du réseau. C'est avec ce but que nous avons construit notre réseau.

La première couche est une couche d'**Embedding**, qui sert à apprendre une représentation dense et de plus basse dimension pour les données d'entrée. L'embedding vise à capturer les relations sémantiques entre les mots, en les positionnant dans un espace vectoriel continu. Chaque mot est représenté par un vecteur d'embedding, et des mots similaires sont positionnés à proximité dans cet espace, comme nous le voyons sur cette figure :



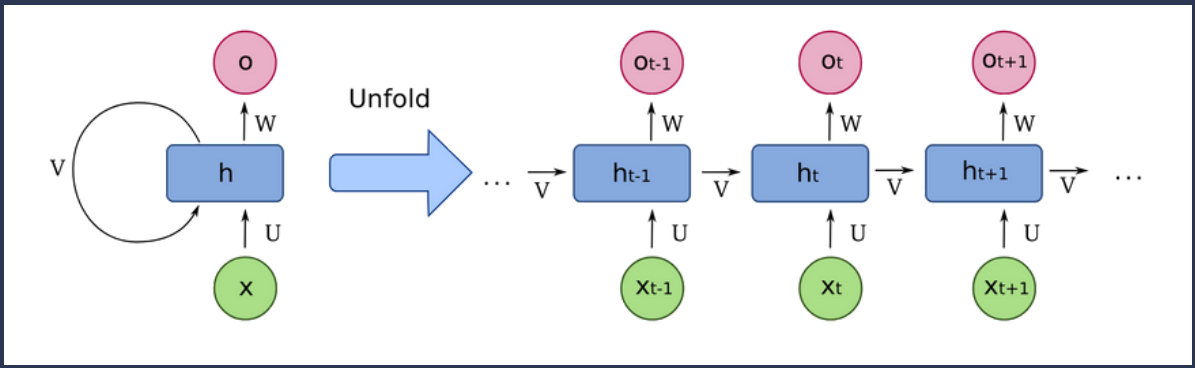
Visualisation de l'espace vectoriel de l'embedding

Les raison pour lesquelles ont a utilisé cette couche sont :

- **Réduction dimensionnelle**
- **Apprentissage des relations** : l'embedding capture des similitudes sémantiques entre les mots, renforçant la capacité du modèle à généraliser.
- **Économie de ressources** : En diminuant la dimensionnalité, la couche d'embedding réduit le nombre de paramètres à apprendre.

La prochaine couche et l'une des couches les plus importantes du réseau est la couche **LSTM (Long short-term memory).**

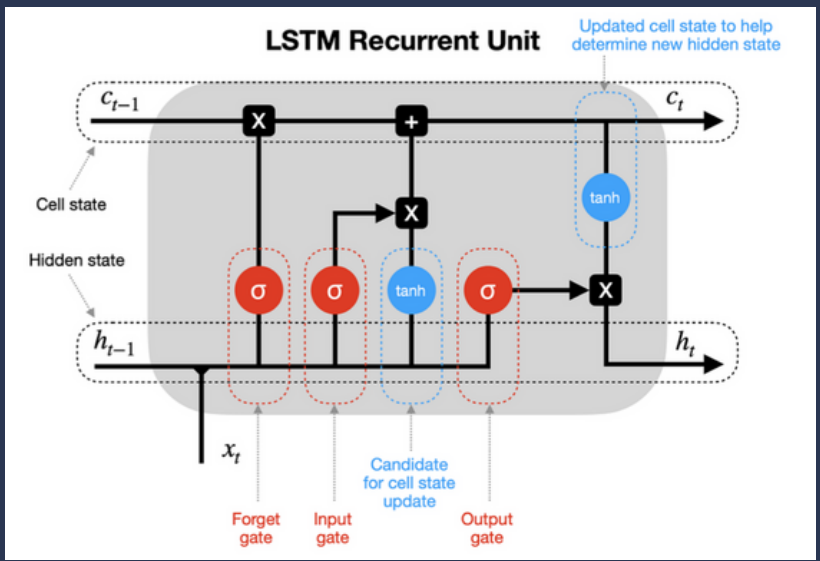
Tout d'abord, une couche LSTM est une **couche récurrente**. Les couches récurrentes dans sont spécialement conçues pour traiter des données séquentielles, en maintenant un état interne qui prend en compte les informations précédemment traitées dans la séquence. Cette caractéristique les rend idéales pour des tâches impliquant des séquences, telles que le traitement du langage naturel qui est notre cas. Voici un schéma plus clair:



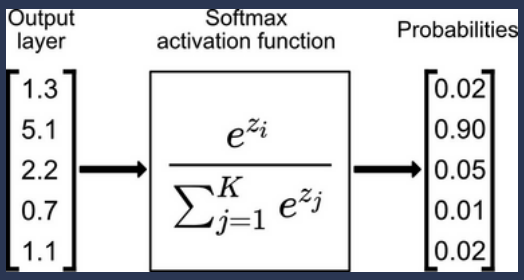
Architecture d'un RNN

Cependant, les couches récurrentes traditionnelles peuvent présenter des défis lors de l'apprentissage de dépendances à long terme, notamment la disparition du gradient, où les gradients deviennent trop petits pour avoir un impact significatif, et la saturation, où les activations des neurones atteignent des valeurs extrêmes.

D'où l'invention des **LSTM** : conçus pour surmonter les problèmes de dépendance à long terme rencontrés par les couches récurrentes traditionnelles. Les LSTM utilisent des portes pour réguler le flux d'information, permettant de mémoriser, d'oublier ou de mettre à jour des informations en fonction du contexte de la séquence, résolvant ainsi le problème de la disparition du gradient. Cette architecture spéciale améliore la capacité du modèle à apprendre des dépendances complexes dans les séquences de données, c'est-à- dire mieux prendre en compte le contexte du début de la phrase quand la couche traite la fin de la phrase. Voici un schéma pour visualiser les portes :



La dernière du réseau est une couche **Dense** avec une fonction d'activation **softmax**. Elle génère une distribution de probabilités sur les différentes classes de secteurs. En effet, la fonction softmax normalise les scores de chaque classe, transformant les sorties du modèle en probabilités. Ainsi, la classe avec la probabilité la plus élevée est sélectionnée comme la prédiction finale du modèle pour chaque instance. La fonction softmax très simplement :



Hyperparamètres

La détermination des hyperparamètres s'est faite en utilisant des recherches des meilleures pratiques et également à travers des évaluations empiriques.

Pour la fonction de perte, nous avons choisi la **Sparse Categorical Crossentropy** est une mesure couramment utilisée dans l'entraînement de modèles de classification multiclasse, particulièrement lorsque les labels sont fournis sous forme d'entiers, en évaluant la disparité entre la distribution de probabilités prédite par le modèle et la distribution réelle des classes. Sparse Categorical Crossentropy :

$$L_{\text{CE}} = -\sum_{i=1}^{n} t_i \log(p_i), \quad \text{for n classes,}$$

where t_i is the truth label and p_i is the Softmax probability for the i^{th} class.

Sparse Categorical Crossentropy

Pour l'optimiser, nous avons choisi **adam** qui combine des techniques de moment adaptatif et de descente de gradient stochastique, ajustant automatiquement le taux d'apprentissage pour accélérer la convergence du modèle tout en évitant les obstacles liés au choix manuel du taux d'apprentissage.

Nous avons choisi **30 epochs**, **100 unités LSTM** et un **batch size de 32** après plusieurs entraînements pour tester différentes valeurs.

Evaluations des modèles

On a divisé la base d'apprentissage en un ratio de 80/20, en entraînant le modèle sur 80 % des données et en le testant sur les 20 % restants. Comme l'objectif principal était de décrire les réponses, nous avons délibérément choisi d'optimiser le fit sur ces données.

PolyChatU :

- Nombre de paramètres: **162722**
- Accuracy sur les données: **0.9320 - 93%**

PolyChatR :

- Nombre de paramètres: **180226**
- Accuracy sur les données: **0.8846 - 88%**

PolyChatA :

- Nombre de paramètres: **180386**
- Accuracy sur les données: **0.8783 - 88%**

PolyChatI :

- Nombre de paramètres: **137410**
- Accuracy sur les données: **0.7835 - 78%**

En effet, certains modèles présentent des performances inférieures à d'autres, et cela s'explique par le fait que ceux qui affichent des résultats moins satisfaisants disposent de moins de réponses, entraînant ainsi une quantité moindre de données pour l'entraînement.

Résultats

Il est intéressant de poser des questions à ces modèles en prenant en compte les résultats de parties précédentes. Quelques questions que nous lui avons posé ainsi que leurs réponses :

1. Quelle est la filière pour laquelle le cours de programmation devrait être renforcé ou approfondi ?

Microelectronique Et Automatique (MEA)

2. Quel est la filière pour laquelle le cours de maths était utile ?

Mecanique Structures Industrielles (MSI - apprentissage)

3. Quel est la filière pour laquelle un cours de management était absent et aurait été utile ?

Genie Biologique et Agroalimentaires (GBA)

4. Quelle est la filière pour laquelle le cours de droit était inutile ?

Materiaux (MAT)

5. Quelle est la filière pour laquelle le cours de jeux d'entreprise était inutile ?

Microelectronique Et Automatique (MEA)

6. Quelle est la filière pour laquelle le cours de espagnol était inutile ?

Informatique et Gestion (IG)

7. Quelle est la filière pour laquelle le cours de maths devrait être renforcé ou approfondi ?

Genie Biologique et Agroalimentaires (GBA)

8. Quelle est la filière pour laquelle le cours de python devrait être renforcé ou approfondi ?

Microelectronique Et Automatique (MEA)

9. Quel est la filière pour laquelle le cours de finance était inutile ?

Informatique et Gestion (IG)

Nous pouvons encore lui poser des centaines de questions, et nous invitons les décideurs de Polytech à le faire. Nous retrouvons les mêmes résultats que ceux que nous avons trouvés dans les analyses précédentes. Évidemment, les modèles ne sont jamais parfaits, et il est toujours bon de croiser ces résultats avec d'autres analyses.

Tentative de construction d'un BERT

Nous avons cherché à améliorer l'embedding d'entrée de notre réseau ainsi que notre outil de manière générale. À cette fin, nous avons entrepris la construction d'un modèle BERT, qui constitue une composante de l'architecture des transformers telle que décrite dans "Attention Is All You Need" écrit par *Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin* et qui aujourd'hui est beaucoup utilisé pour l'embedding de texte. BERT est utilisé en pré-entraînant un modèle sur de vastes corpus textuels, ce qui permet au modèle d'apprendre des représentations contextuelles riches pour chaque mot. Ensuite, pour une tâche spécifique, le modèle BERT peut être fine tuné sur un ensemble de données ciblé, ajustant ainsi ses représentations pour mieux s'adapter à la tâche particulière, et ces représentations peuvent être utilisées comme embeddings contextuels pour le texte. Nous n'avons pas réussi à mettre en place un BERT dû à la complexité de la tâche.

Conclusion

Tout au long de cette analyse, nous avons pu mettre en œuvre l'ensemble du processus d'études de données textuelles. Nous avons effectué des extractions, des transformations de données, des analyses globales, et nous avons poursuivi avec des méthodes d'analyses plus approfondies et affinées. Nous avons présenté les résultats et les interprétations tout au long de ce rapport. De plus, nous avons fourni des outils et une application interactive qui pourront être réutilisés et améliorés lorsque Polytech recevra les réponses des prochaines enquêtes au fil des années à venir.

Annexes

Code

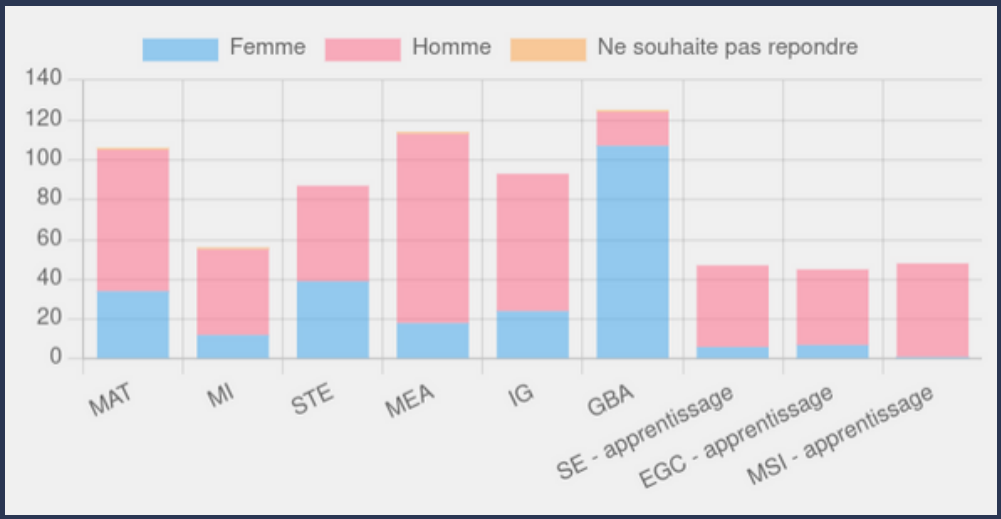
- **ETL, Outils utilisés, backend application, etc. :** https://github.com/alexdeloire/data_science_backend
- **Dashboard, chat, frontend :** https://github.com/alexdeloire/data_science_frontend
- **L'application est déployée ici :** https://poly-analyse.onrender.com/

Échantillon de Graphiques

Voici des exemples de graphiques pour 2023, pour les autres années nous invitons le lecteur à utiliser notre outil de décision interactif.

Nombre de réponses par filière

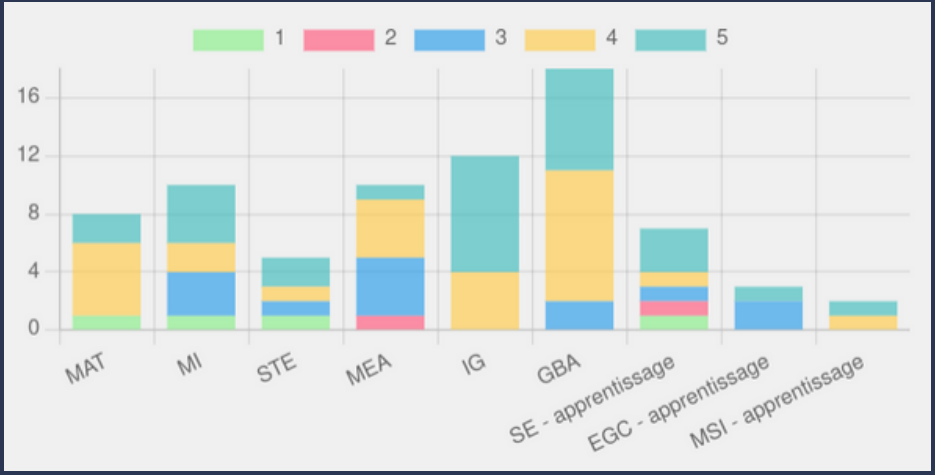
Répartition des sexes des répondants pour chaque filière



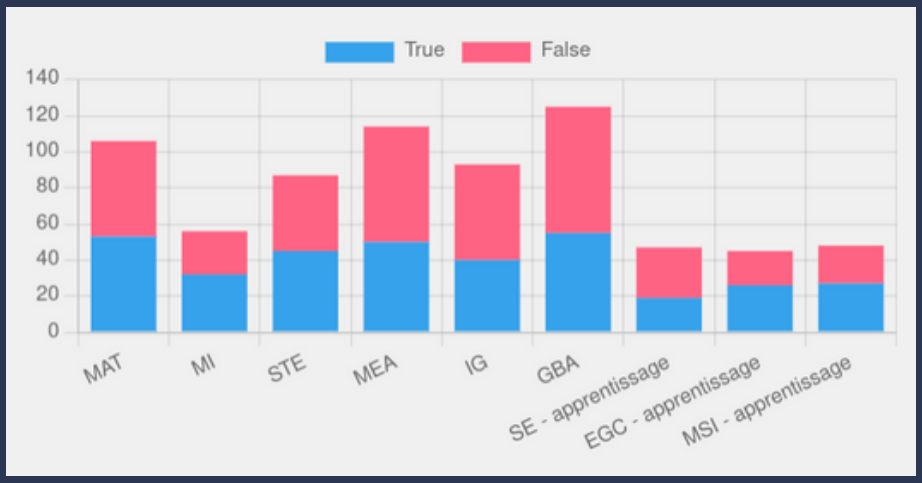
Répartition des réponses pour la question : Les éléments suivants vous semblent-ils avoir joué un rôle dans votre recrutement ? - la réputation de la filière de formation

Répartitions des répondants à la question (True: la personne a répondu, False: la personne n'a pas répondu) : Quels enseignements vous semblent les plus utiles pour l'exercice de votre métier et votre insertion professionnelle ?

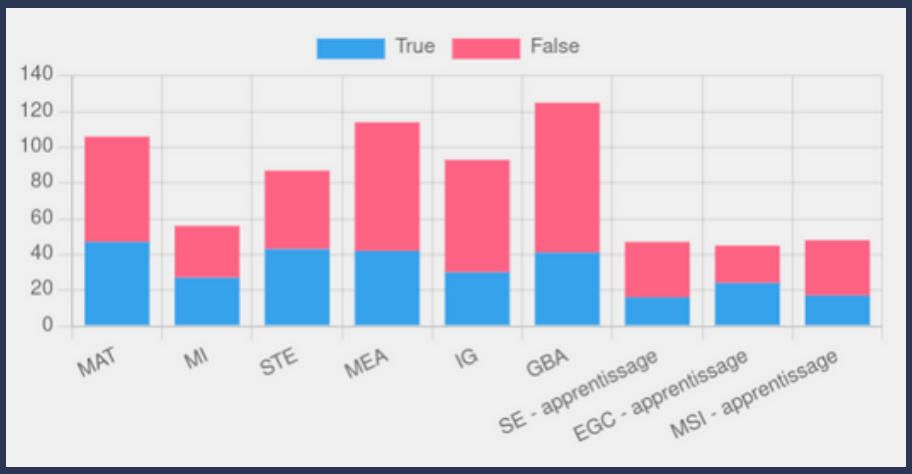
Répartition des réponses pour la question : Les éléments suivants vous semblent-ils avoir joué un rôle dans votre recrutement ? - la formation



Répartitions des répondants à la question (True: la personne a répondu, False: la personne n'a pas répondu) : Parmi les enseignements fournis par l'école, quels sont ceux qui mériteraient d'être approfondis ou renforcés ?



Répartitions des répondants à la question (True: la personne a répondu, False: la personne n'a pas répondu) : Quels enseignements, absents de votre formation, vous auraient été utiles ?



Répartitions des répondants à la question (True: la personne a répondu, False: la personne n'a pas répondu) : Quels enseignements, presents dans votre formation, vous paraissent inutiles ?