

**2023 - 2024**

**PROJET DE FIN D'ÉTUDES**

**DIPLÔME NATIONAL D'INGÉNIEUR**

**SPECIALITÉ: INFORMATIQUE**

**Système de recommandation de  
produits en cristal 3D**

**Réalisé par: Yosri BEN SALEM**

**Encadré par:**

**Encadrant ESPRIM: Mme. Fehima ACHOUR**

**Encadrant Entreprise: M. Mohamed BOURAQUI**

Je valide le dépôt du rapport PFE relatif à l'étudiant nommé ci-dessous / I  
validate the submission of the student's report:

- Nom & Prénom /Name & Surname : BEN SALEM Yosri

Encadrant Entreprise/ Business site Supervisor

- Nom & Prénom /Name & Surname : .....

Cachet & Signature / Stamp & Signature

Encadrant Académique/Academic Supervisor

- Nom & Prénom /Name & Surname : .....

Signature / Signature

---

## Remerciements

Je tiens à exprimer mes sincères remerciements à **mes parents** pour leur soutien inconditionnel tout au long de mon parcours académique. Leur amour, leurs encouragements et leur dévouement ont été une source d'inspiration et de motivation essentielle pour moi.

Je souhaite également exprimer ma gratitude envers la **société Bouraoui Group** et son co-fondateur, **Mr. Mohamed Bouraoui**, pour leur généreux soutien Technique et leur confiance en mes capacités. Leur engagement en faveur de l'éducation et du développement des jeunes talents est exemplaire, et je suis honoré d'avoir été bénéficiaire de leur soutien.

En outre, je tiens à remercier chaleureusement l'ensemble des enseignants de l'**ESPRIM** pour leur expertise, leurs conseils précieux et leur dévouement à notre apprentissage. Leur passion pour l'enseignement et leur engagement à nous fournir une éducation de qualité ont grandement contribué à ma formation et à ma croissance personnelle et professionnelle.

Yosri Ben Salem

# Table des matières

<b>Acronymes</b>	<b>vii</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Cadre et contexte du projet</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Présentation Générale de l'entreprise . . . . .	3
1.3 Contexte du projet . . . . .	4
1.3.1 Problématique . . . . .	4
1.3.2 Etude de l'existant . . . . .	6
1.3.3 Solution proposée et objectifs . . . . .	8
1.4 Méthodologie du travail .....	10
1.5 Etat de l'art.....	12
1.5.1 Principe du NLP .....	12
1.5.2 Principaux domaines d'application du NLP .....	13
1.5.3 Classification du texte.....	14
1.5.4 Reconnaissance du texte .....	16
1.6 Conclusion.....	18
<b>2 Compréhension du métier</b>	<b>19</b>
2.1 Introduction .....	19
2.2 Description du domaine des produits 3D Crystal .....	19
2.3 Analyse des besoins.....	20
2.3.1 Besoins fonctionnels.....	20
2.3.2 Besoins non fonctionnels.....	21
2.4 Objectifs et Contraintes du projet.....	21
2.4.1 Objectifs du système de recommandation.....	22
2.4.2 Contraintes du projet .....	22
2.5 Conclusion.....	23
<b>3 Exploration des données</b>	<b>24</b>
3.1 Introduction .....	24
3.2 Présentation des sources de données .....	24
3.2.1 Base des produits 3D Crystal.....	24
3.2.2 Ensemble de questions-réponses préétablies .....	25
3.2.3 Données d'entrée utilisateur pour le NLP .....	26
3.3 Analyse des données .....	27
3.3.1 Analyse du fichier CSV des produits 3D Crystal.....	27

3.3.2	Étude des questions-réponses .....	28
3.3.3	Préparation des données textuelles pour la recommandation .....	30
3.3.4	Étapes du prétraitement .....	30
3.4	Conclusion.....	31
<b>4</b>	<b>Modélisation et développement</b>	<b>32</b>
4.1	Introduction . . . . .	32
4.2	Pipeline proposé . . . . .	32
4.3	Construction du modèle . . . . .	33
4.3.1	K plus proches voisins . . . . .	33
4.3.2	Classifieur zéro-shot . . . . .	37
4.4	Conclusion . . . . .	40
<b>5</b>	<b>Evaluation et Déploiement</b>	<b>41</b>
5.1	Introduction . . . . .	41
5.2	Critères d'évaluation théorique . . . . .	41
5.3	Cycle de vie du projet . . . . .	42
5.4	Méthodologie d'évaluation . . . . .	43
5.4.1	Évaluation du système de recommandation basé sur les questions.....	43
5.4.2	Évaluation de la classification de texte sans apprentissage (NLP).....	45
5.5	Déploiement .....	46
5.5.1	Technologies utilisées .....	47
5.5.2	Interfaces .....	49
5.5.3	Scénario d'application .....	52
5.5.4	Hébergement .....	57
5.6	Conclusion.....	58
	<b>Conclusion générale et perspectives</b>	<b>59</b>
	<b>Bibliographie</b>	<b>61</b>

# Liste des Figures

1.1	Logo Société Bouraoui Group [1]	3
1.2	Artpix 3D Logo [2]	6
1.3	Logo Crystal Clear Memories [3]	7
1.4	Logo 3dcystal.com [4]	7
1.5	Les phases principales de la Méthodologie CRISP-DM.....	12
1.6	Domaines d'application du NLP .....	13
3.1	Logo Application Ablestar Bulk Product Editor [6].....	25
4.1	Pipeline proposé .....	33
4.2	Fonctionnement d'approche KNN.....	34
4.3	Fonctionnement du prétraitement .....	35
4.4	Vectorisation des textes prétraité .....	35
4.5	Calcul des Similarités .....	35
4.6	Entraînement du modèle KNN .....	36
4.7	Recherche des k voisins .....	36
4.8	Renvoyer les produits recommandés .....	36
4.9	Zero-Shot Learning .....	37
4.10	Chargement du pipeline Zero-Shot.....	38
4.11	Fonction de classification du texte.....	39
4.12	Combinaison des étiquettes .....	39
4.13	Vectorisation des classes.....	39
5.1	Cycle de vie du projet.....	42
5.2	Fonction d'évaluation du modèle KNN.....	44
5.3	Fonction d'évaluation du modèle ZERO-SHOT.....	46
5.4	Logo Python .....	47
5.5	Logo Flask.....	47
5.6	Logo HTML.....	48
5.7	Logo CSS .....	48
5.8	Logo Javascript .....	48
5.9	Page d'accueil.....	49
5.10	Page de recommandation basée sur le traitement des questions/réponses	50
5.11	Page de recommandation basée sur le traitement du text NLP .....	51
5.12	Scénario d'application : questionnaire a choix multiple.....	52
5.13	Application de l'algorithme SVD pour la recommandation de pro- duits.....	53
5.14	Application de calcul du score de similarité .....	53

## LISTE DES FIGURES

---

5.15	Résultat de la classification du texte.....	54
5.16	Page du produit.....	54
5.17	Exemple du texte à traiter par le modèle de traitement du langage naturel.....	55
5.18	Application de l'algorithme "Zero-shot Classifier" pour la recommandation de produits .....	55
5.19	Résultat du traitement du langage naturel.....	56
5.20	Logo AWS Lightsail.....	57

# Liste des tableaux

1.1	Critique de l'existant . . . . .	8
	TABLE 1.2 – Comparaison des approches Similarité Cosinus et KNN	15
	TABLE 1.3 – Comparaison des approches SVD et RNN .....	15
	TABLE 1.4 – Comparaison des approches Zero-Shot et DistilBERT	17
	TABLE 1.5 – Comparaison du approches MultiOutputClassifier et CNN .....	17
	TABLE 3.1 – Tableau des questions-réponses.....	26



# Acronymes

- **ZSL** : Zero-Shot Learning
- **NLP** : Natural Language Processing
- **BERT** : Bidirectional Encoder Representations from Transformers
- **TF-IDF** : Term Frequency-Inverse Document Frequency
- **KNN** : K Nearest Neighbors
- **RNN** : Recurrent Neural Network
- **SVD** : Singular Value Decomposition
- **MNB** : Multinomial Naive Bayes
- **CRISP-DM** : Cross-Industry Standard Process for Data Mining
- **AI** : Artificial Intelligence
- **ML** : Machine Learning
- **DL** : Deep Learning
- **UML** : Unified Modeling Language
- **HTML** : Hypertext Markup Language
- **CSS** : Cascading Style Sheets
- **JS** : JavaScript
- **CSV** : Comma-Separated Values



# Introduction générale

Le secteur des produits 3D Crystal connaît une popularité croissante, offrant aux consommateurs la possibilité de créer des objets personnalisés et uniques. Cependant, avec la multitude d'options disponibles, les utilisateurs peuvent se retrouver confrontés à un défi de choix. Dans cette optique, nous proposons la mise en place d'un système de recommandation de produits 3D Crystal avec un volet de questions/réponses et de traitement du langage naturel (NLP). Ce système vise à simplifier le processus de sélection des produits en offrant aux utilisateurs des recommandations personnalisées basées sur leurs préférences et leurs besoins spécifiques.

L'objectif principal de ce projet est de développer un système de recommandation intelligent et efficace, capable de comprendre les préférences des utilisateurs à partir de leurs réponses à des questions spécifiques et d'utiliser des techniques de traitement du langage naturel pour interpréter et analyser ces réponses. Grâce à ce système, les utilisateurs pourront bénéficier d'une expérience de sélection de produits plus personnalisée et pertinente, tout en réduisant le temps et les efforts nécessaires pour trouver le produit idéal.

Pour atteindre cet objectif, nous adoptons une approche basée sur la méthodologie CRISP-DM (Cross-Industry Standard Process for Data Mining), qui nous permettra de gérer efficacement les différentes étapes du projet, de la compréhension du problème à la modélisation et à l'évaluation du système de recommandation.

Ce rapport présente une analyse détaillée de notre démarche, en mettant l'accent sur les différentes étapes du processus CRISP-DM adaptées à notre projet. Le chapitre 1 présente le contexte et les objectifs du projet. Il décrit la problématique de l'entreprise Crystal, passe en revue les solutions existantes et définit les objectifs de notre système de recommandation.

Le chapitre 2 est consacré à l'exploration approfondie des données mises à notre disposition. Il présente les différentes sources de données, notamment le catalogue de produits, les questions-réponses et les données textuelles. Ces données

---

feront l'objet d'analyses pour en comprendre la structure et les caractéristiques.

Le chapitre 3 se concentre sur la présentation des sources de données à notre disposition et les analyses effectuées pour en comprendre la structure et les caractéristiques en amont de leur modélisation.

Le chapitre 4 couvre la modélisation de notre système de recommandation. Il détaille les techniques retenues, en particulier celles issues de la recommandation collaborative, du filtrage par contenu et de l'apprentissage automatique.

Enfin, le chapitre 5 aborde la méthodologie d'évaluation mise en place et le déploiement pratique du système au sein de Crystal, avant de conclure sur les résultats obtenus, les limites identifiées et les perspectives d'amélioration futures dans la conclusion générale.

# Cadre et contexte du projet

---

## 1.1 Introduction

Dans ce premier chapitre, nous commencerons par présenter l'organisme en question, afin de fournir un contexte approprié pour la suite de l'étude. Ensuite, j'entreprendrai une étude approfondie de l'existant, mettant en évidence les critiques du système actuel. Une fois cette base établie, je procéderai à la définition des Langage de modélisation et méthodologie de développement. Enfin, j'aborderai l'état de l'art du projet.

## 1.2 Présentation Générale de l'entreprise



FIGURE 1.1 – Logo Société Bouraoui Group [1]

Fondée en 2021, la Société Bouraoui Group est une entreprise dynamique et innovante dédiée à la fourniture de services de haute qualité dans divers domaines, dont le logo est présenté dans la figure 1.1 ci-dessus. Avec une forte focalisation sur l'excellence, elle œuvre pour répondre aux besoins uniques et aux attentes de

sa clientèle. Son expertise s'étend à travers diverses industries, y compris l'architecture, le design et la technologie.

Au sein du Société Bouraoui Group, l'engagement est porté sur l'offre de solutions de pointe et la promotion des avancées créatives. Son équipe est composée de professionnels hautement qualifiés avec une vaste expérience dans leurs domaines respectifs. La société est fière de rester à la pointe des tendances de l'industrie et d'utiliser les dernières technologies pour fournir des résultats exceptionnels.

En tant qu'organisation centrée sur le client, nous accordons la priorité à la construction de relations solides et à la compréhension des exigences spécifiques de chaque projet. Que ce soit pour la visualisation architecturale, modélisation 3D ou d'autres services de photographie, nous abordons chaque tâche avec une attention méticuleuse aux détails et une passion pour dépasser les attentes des clients.

## **1.3 Contexte du projet**

### **1.3.1 Problématique**

La problématique du projet consiste à améliorer le taux de conversion d'un site proposant plus de 1000 produits, notamment des créations en cristal 3D gravées, en utilisant les technologies de la machine learning. Le site fait face à des difficultés pour convertir les visiteurs en clients, ce qui limite sa croissance et sa rentabilité. L'objectif est donc d'intégrer des techniques de machine learning afin d'optimiser l'expérience utilisateur en fournissant des recommandations de produits pertinentes et personnalisées. En exploitant les données existantes sur les comportements d'achat des utilisateurs et en utilisant des algorithmes d'intelligence artificielle, il sera possible de comprendre les préférences individuelles des clients, d'analyser les tendances du marché et de proposer des recommandations précises et ciblées. Ces recommandations, basées sur des modèles prédictifs, permettront de susciter l'intérêt des visiteurs en leur présentant des créations en cristal 3D gravées qui correspondent parfaitement à leurs goûts et à leurs aspirations. Cette approche personnalisée facilitera leur processus de prise de décision et augmentera ainsi le nombre de conversions.

Pour répondre à cette problématique, plusieurs défis doivent être relevés :

1. Collecte et analyse des données : L'un des défis majeurs sera de collecter et d'analyser les données sur les comportements d'achat des utilisateurs. Cela implique la mise en place de mécanismes de suivi et de collecte de données pertinents, ainsi que le développement d'algorithmes de traitement de données efficaces pour extraire les informations nécessaires à la création de recommandations personnalisées.
2. Modélisation des préférences des utilisateurs : une étape clé pour être capable de formuler des recommandations personnalisées pertinentes. Cela nécessite la mise en place de modèles d'apprentissage automatique (machine learning) capables d'analyser les données collectées sur les utilisateurs et d'en déduire leurs centres d'intérêt récurrents.
3. Développement d'algorithmes de recommandation efficaces : La création d'algorithmes de recommandation précis et performants constitue un autre défi. Il faudra explorer différentes approches, telles que les filtres collaboratifs, les méthodes basées sur le contenu ou les techniques de deep learning, pour trouver la méthode la plus adaptée aux caractéristiques des produits en cristal 3D gravés et aux spécificités du marché.
4. Intégration des recommandations dans l'expérience utilisateur : Une fois les recommandations générées, il sera crucial de les intégrer de manière transparente dans l'expérience utilisateur du site. Cela nécessitera une réflexion sur le design de l'interface, l'emplacement des recommandations et la manière de les présenter de manière attrayante, afin d'inciter les utilisateurs à explorer les produits suggérés et à passer à l'achat.
5. Évaluation et ajustement continu : Enfin, un défi constant sera de mesurer l'efficacité des recommandations et de les ajuster en fonction des retours des utilisateurs et des performances observées. Une évaluation régulière des résultats obtenus permettra d'identifier les améliorations à apporter et d'optimiser en permanence les algorithmes de recommandation pour maximiser le taux de conversion.

### 1.3.2 Etude de l'existant

#### 1.3.2.1 Analyse de l'existant

En termes d'analyse de l'existant, les exemples mentionnés ci-dessous montrent trois des concurrents en ligne qui offre des services similaires :

### Artpix 3d



FIGURE 1.2 – Artpix 3D Logo [2]

ArtPix 3D offre une large sélection de cadeaux personnalisés en cristal gravé en 3D, avec la possibilité de prévisualiser les modèles avant l'achat. Leur engagement envers la satisfaction client, la livraison gratuite et la garantie de retour de 14 jours sont des points positifs. Cependant, en ce qui concerne la présentation des produits, le site manque peut-être d'un design plus moderne et attrayant. Les produits sont listés de manière assez standard, ce qui peut rendre la recherche un peu fastidieuse pour les clients qui souhaitent trouver rapidement ce qu'ils cherchent.



## Crystal Clear Memories



FIGURE 1.3 – Logo Crystal Clear Memories [3]

Crystal Clear Memories est un site web spécialisé dans la vente de cristaux gravés en 3D, offrant aux clients la possibilité de personnaliser leurs propres photos sur ces cristaux. Cependant, le site ne semble pas proposer un système de recommandation de produits 3D gravés en cristal spécifique. Le site offre une large gamme de produits, avec une sélection de cadeaux telles que des tours, des briques, des cubes, des cœurs et des modèles de prestige. Cependant, il manque un mécanisme qui recommanderait spécifiquement des produits 3D gravés en cristal en fonction des préférences et des besoins des clients.

## 3DCrystal.com



FIGURE 1.4 – Logo 3dcystal.com [4]

3DCrystal.com présente plusieurs points forts dans son activité de vente de cadeaux en cristal gravés en 3D. Il offre une large gamme de produits personnalisables, permettant aux clients de transformer leurs précieux souvenirs en pièces d'art uniques. La variété de formes et de tailles disponibles, comme les rectangles, les cœurs et les modèles de prestige, offre aux clients un choix diversifié. Cependant, certaines critiques peuvent être formulées à l'égard de 3DCrystal.com. Certains clients pourraient souhaiter un système de recommandation plus spécifique pour

les produits gravés en cristal, basé sur leurs préférences et leurs besoins. De plus, l'absence d'informations détaillées sur les matériaux utilisés et les processus de fabrication peut laisser certains clients en quête de transparence.

#### 1.3.2.2 Critique de l'existant

En analysant l'existant dans le domaine des cadeaux personnalisés en cristal gravé en 3D, il est possible de relever certains éléments qui seront présentés dans le tableau ci-dessous :

Site Web	Design et Attractivité	Personnalisation	Prévisualisation	NLP
Artpix 3D	Moderne et Attrayant	oui	oui	<b>non</b>
Crystal Clear Memories	Moyen	oui	non	<b>non</b>
3DCrystal.com	Moyen	oui	non	<b>non</b>
Notre Application	Moderne et Attrayant	oui	yes	<b>oui</b>

TABLE 1.1 – Critique de l'existant

Les principaux sites concurrents de personnalisation de produits 3D comme Artpix 3D, Crystal Clear Memories et 3DCrystal.com présentent des fonctionnalités sommaires comme des possibilités de personnalisation et parfois un outil de prévisualisation. Cependant, leur interface reste basique sur le plan du design et ils ne disposent d'aucun système de recommandation basé sur le traitement automatique du langage naturel. Or, notre étude montre que la capacité à comprendre des requêtes exprimées en langage courant et à suggérer des produits personnalisés est un facteur déterminant de qualité pour les utilisateurs. C'est pourquoi notre application se démarque par son interface épurée et moderne, mais aussi et surtout par son système de recommandation intelligent reposant sur les techniques avancées de NLP. Grâce à cette fonctionnalité inédite sur le marché, nous sommes en mesure de proposer une expérience utilisateur de bien meilleure facture, en personnalisant les recommandations à chaque besoin exprimé de manière simple et naturelle.

#### 1.3.3 Solution proposée et objectifs

Pour combler le manque d'un système de recommandation spécifique aux produits 3D gravés en cristal, une solution intéressante et efficace serait de combiner

un chatbot interactif basé sur des questions-réponses avec des algorithmes de recommandation intelligents. Lorsqu'un client visite le site, le chatbot se présente et engage une conversation avec lui. Il poserait des questions ciblées pour comprendre les préférences de l'utilisateur en termes de centres d'intérêts, de cible, d'émotions, de style etc. Ces questions seraient présentées de manière conviviale et engageante, permettant aux clients de répondre facilement et de manière ludique.

En fonction des réponses fournies par l'utilisateur, le chatbot utiliserait des algorithmes de recommandation pour générer des suggestions personnalisées. Les algorithmes pourraient prendre en compte les préférences déclarées par l'utilisateur à partir d'analyse du langage naturel du texte saisi par l'utilisateur et prédire les labels des préférences pour les cadeaux. Cela permettrait de proposer des recommandations pertinentes et adaptées aux goûts de chaque client.

Ci-dessous, vous trouverez une liste d'objectifs relatifs à la mise en place :

- Collecter les préférences des utilisateurs : L'objectif principal est de collecter de manière précise et complète les préférences des utilisateurs en termes d'occasion, intérêt, émotions, audience, etc. Cela permettra de comprendre leurs goûts et leurs besoins spécifiques.
- Générer des recommandations personnalisées : l'objectif est d'utiliser les réponses fournies par les utilisateurs pour générer des suggestions de produits personnalisés. Les recommandations doivent être pertinentes et adaptées aux préférences de chaque client.
- Améliorer l'expérience d'achat : L'objectif est d'améliorer l'expérience d'achat des clients en leur fournissant des recommandations personnalisées qui correspondent à leurs goûts et à leurs besoins. Cela facilitera leur processus de recherche et de prise de décision, en leur proposant des produits pertinents et attrayants.
- Optimiser les recommandations au fil du temps : L'objectif est d'améliorer la précision des recommandations en utilisant des algorithmes d'apprentissage automatique qui apprennent des interactions précédentes avec les clients. Le système doit être capable de s'adapter et de fournir des recommandations de plus en plus précises à mesure qu'il collecte davantage de données et de feedback des utilisateurs.

- Encourager la découverte de nouveaux produits et augmenter les ventes : L’objectif est de stimuler l’exploration des options disponibles par les clients en leur proposant des produits qu’ils n’auraient peut-être pas considérés initialement. Les recommandations peuvent inclure des produits similaires ou complémentaires qui pourraient susciter l’intérêt des clients et les inciter à découvrir de nouvelles possibilités. En fournissant des recommandations personnalisées et attrayantes, le site vise également à augmenter les ventes en proposant des produits pertinents aux clients.

## 1.4 Méthodologie du travail

Pour mener à bien ce projet de développement d’un système de recommandation, il était nécessaire de choisir une méthodologie de travail rigoureuse. Les principales méthodologies existantes sont le cycle de vie en cascade, les méthodes agiles (Scrum, Kanban) et la méthodologie CRISP-DM.

Le cycle en cascade est trop rigide et ne permet pas d’adaptation facile aux évolutions de projet. Les méthodes agiles sont quant à elles plus adaptées au développement de logiciels interactifs mais moins à l’exploration et l’analyse de données, cœur de notre projet.

Nous avons donc opté pour la méthodologie CRISP-DM, qui fournit une approche structurée tout en laissant une grande flexibilité dans la gestion du cycle de vie du projet. En effet, CRISP-DM sépare clairement les étapes clés sans imposer d’ordre strict, permettant ainsi des allers-retours et des optimisations continues entre les phases.

CRISP-DM (Cross-Industry Standard Process for Data Mining) est un modèle de processus standardisé largement utilisé dans le domaine de l’exploration de données et de l’analyse prédictive [5]. Il fournit une approche structurée pour guider les équipes dans la réalisation de projets d’exploration de données, de la compréhension des besoins commerciaux à la mise en œuvre des solutions.

CRISP-DM se compose de six phases principales, qui sont les suivantes :

- **Compréhension du domaine** : Cette phase consiste à définir les objectifs commerciaux et les besoins en matière de données, ainsi qu'à comprendre le domaine d'application dans lequel le projet d'exploration de données est mené.
- **Compréhension des données** : Ici, on rassemble les données nécessaires au projet, on les explore, on les nettoie et on les prépare pour l'analyse.
- **Préparation des données** : Dans cette phase, les données sont transformées et préparées pour l'analyse ultérieure. Cela peut inclure des activités telles que la sélection des variables, la création de nouvelles caractéristiques, la normalisation des données, etc
- **Modélisation** : Cette phase implique le développement de modèles prédictifs ou descriptifs en utilisant différentes techniques d'exploration de données, telles que la régression, les arbres de décision, les réseaux neuronaux, etc.
- **Évaluation** : Une fois que les modèles ont été construits, ils sont évalués en utilisant des critères appropriés. Cela peut inclure des mesures de précision, de rappel, de F-mesure, de matrice de confusion, etc. L'objectif est de sélectionner le modèle le plus performant.
- **Déploiement** : Enfin, le modèle sélectionné est mis en œuvre dans l'environnement de production. Cela peut impliquer le développement d'applications logicielles, l'intégration dans les systèmes existants ou la création de rapports et de visualisations pour une utilisation pratique.

La figure ci-dessous présente les phases principales de la méthodologie CRISP-DM sous forme d'un schéma :

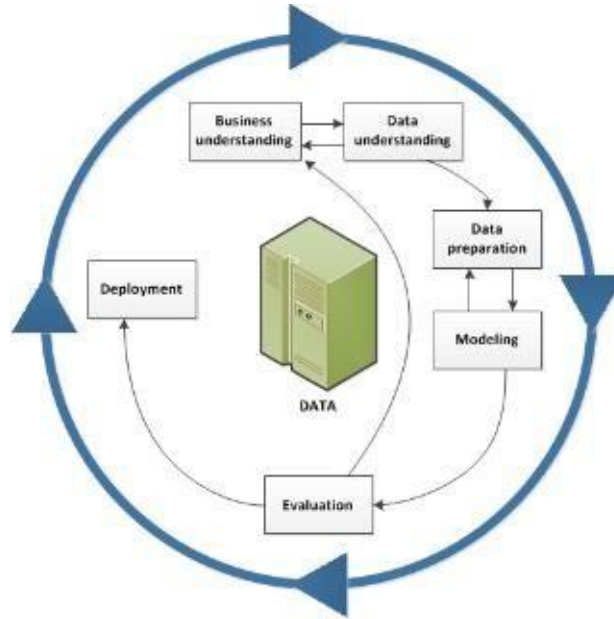


FIGURE 1.5 – Les phases principales de la Méthodologie CRISP-DM

## 1.5 Etat de l'art

Dans cette section, nous examinons les différentes approches et techniques utilisées dans le domaine de l'apprentissage automatique pour résoudre des problèmes liés au traitement du langage naturel (NLP).

### 1.5.1 Principe du NLP

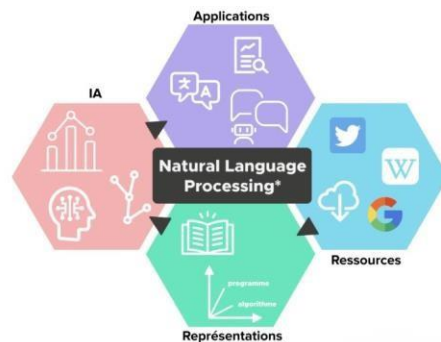
La discipline du NLP, également connue sous le nom de Traitement du Langage Naturel, se concentre principalement sur la compréhension, la manipulation et la création du langage naturel par les machines. De cette manière, le NLP représente véritablement un pont entre la science informatique et la linguistique. Il aborde donc la possibilité pour la machine d'interagir directement avec l'être humain.

Dans l'ensemble, on peut identifier deux éléments clés à tout problème de NLP :

**La section « linguistique »** implique de préparer et de convertir les informations en entrée en un jeu de données exploitable. Cela peut inclure des tâches telles que la tokenisation, l'étiquetage des parties du discours (POS tagging), et l'analyse syntaxique.

**La section "apprentissage automatique" ou "Science des données"** concerne l'utilisation de modèles d'apprentissage automatique ou de Deep Learning pour analyser ce jeu de données. Ces modèles peuvent inclure des réseaux de neurones, des modèles de transformer comme BERT et GPT, et d'autres algorithmes sophistiqués qui permettent d'extraire des informations et de réaliser des prédictions à partir des données textuelles.

### 1.5.2 Principaux domaines d'application du NLP



\*Traitement Automatique du Langage Naturel

FIGURE 1.6 – Domaines d'application du NLP

Le NLP a permis des avancées significatives dans divers domaines qui sont présentés dans la figure ci-dessus. Il est utilisé pour traduire automatiquement des textes, analyser les sentiments des auteurs, cibler les consommateurs en marketing, développer des chatbots pour interagir avec les clients, classer et organiser des textes, extraire des informations de documents, corriger automatiquement l'orthographe et produire des résumés de texte, etc. Ces applications démontrent le potentiel du NLP pour automatiser et améliorer les tâches liées au langage naturel,

offrant ainsi des opportunités dans de nombreux domaines professionnels.

Dans notre projet de recommandation de cadeaux basé sur le traitement du texte, nous exploitons ces avancées du NLP pour analyser les préférences des utilisateurs, comprendre leurs choix et leur fournir des recommandations de cadeaux personnalisées. En utilisant des techniques de traitement du langage naturel, nous sommes en mesure d'extraire des informations pertinentes des descriptions et des préférences des utilisateurs, ce qui nous permet de proposer des cadeaux adaptés à leurs goûts et à leurs besoins. Le NLP joue un rôle crucial dans notre projet en permettant une compréhension fine du langage naturel et en facilitant la recommandation de cadeaux personnalisés pour une expérience utilisateur optimale.

### 1.5.3 Classification du texte

La classification de texte est une tâche importante dans le domaine du NLP, qui consiste à attribuer des étiquettes prédéfinies à des documents textuels en fonction de leur contenu. Différentes approches peuvent être utilisées pour résoudre ce problème, telles que :

- **SVM (Support Vector Machine)** : Les SVM sont une technique d'apprentissage automatique couramment utilisée pour la classification de texte. Ils cherchent à trouver un hyperplan optimal qui sépare les différentes classes de documents.
- **Réseaux de neurones** : Les réseaux de neurones, en particulier les réseaux de neurones convolutifs (CNN) et les réseaux de neurones récurrents (RNN), ont également été largement utilisés pour la classification de texte. Ces modèles sont capables d'apprendre des représentations complexes du texte et peuvent capturer des dépendances à long terme.
- **Ensembles de classifieurs** : Les ensembles de classifieurs, tels que les forêts aléatoires et les méthodes de vote pondéré, peuvent également être utilisés pour améliorer les performances de classification de texte en combinant les prédictions de plusieurs classifieurs individuels.

les tableaux ci-dessous présente une comparaison de quatre approches couramment utilisées en apprentissage automatique : la similarité cosinus, les k plus



## 1.5 Etat de l'art

proches voisins (KNN), la décomposition en valeurs singulières (SVD) et les réseaux de neurones récurrents (RNN), selon plusieurs critères :

Caractéristique	Similarité Cosinus	KNN
Approche	Mesure le cosinus de l'angle entre deux vecteurs, généralement utilisé pour les calculs de similarité et de distance.	Classifie en fonction des k plus proches voisins dans l'espace des caractéristiques.
Entraînement	Aucun entraînement requis, calcule simplement les similarités entre les vecteurs.	Nécessite un entraînement pour stocker des points de données étiquetés pour des comparaisons futures.
Exigences en données	Faibles, peut fonctionner avec de petites quantités de données pour les mesures de similarité.	Moyennes, nécessite des données étiquetées suffisantes pour trouver des voisins significatifs.
Performance	Bonne pour les tâches nécessitant des mesures de similarité, mais peut avoir des difficultés avec une grande dimensionnalité.	Bonne pour les données de faible à moyenne dimensionnalité, mais peut avoir des difficultés avec une grande dimensionnalité.
Complexité	Faible, simple à mettre en œuvre et à calculer les similarités.	Moyenne, nécessite un ajustement des paramètres (valeur de k) et la sélection de la métrique de distance.
Latence	Faible, calcul rapide des similarités cosinus.	Variable, dépend du nombre de voisins et de la dimensionnalité des données.
Cas d'utilisation	Idéal pour les recherches de similarité rapide et les tâches de recommandation avec une faible dimensionnalité.	Adapté pour la classification et la recommandation avec des ensembles de données de petite à moyenne taille.

TABLE 1.2 – Comparaison des approches Similarité Cosinus et KNN

Caractéristique	SVD	RNN
Approche	Factorise la matrice en vecteurs et valeurs singulières pour réduire la dimensionnalité et capturer les caractéristiques latentes.	Utilise des données séquentielles et des connexions récurrentes pour capturer les dépendances au fil du temps.
Entraînement	Nécessite la décomposition de la matrice, généralement à l'aide d'algorithmes comme TruncatedSVD.	Nécessite un entraînement intensif avec des données séquentielles étiquetées pour apprendre des motifs et des dépendances.
Exigences en données	Élevées, nécessite une grande quantité de données pour capturer efficacement les structures latentes.	Élevées, nécessite une grande quantité de données séquentielles pour un entraînement efficace
Performance	Excellente pour capturer les structures latentes et réduire la dimensionnalité, améliorant les performances sur des tâches complexes.	Excellente pour les données séquentielles, capturant les dépendances temporelles et les motifs complexes.
Complexité	Élevée, implique une factorisation matricielle complexe et intensément calculée.	Très élevée, implique des architectures de réseaux complexes et un entraînement intensif.
Latence	Modérée à élevée, en fonction de la taille des matrices impliquées.	Variable, dépend de la taille et de la complexité du réseau, mais généralement élevée en raison de la nature récurrente.
Cas d'utilisation	Excellent pour la réduction de la dimensionnalité et l'extraction des caractéristiques latentes dans les systèmes de recommandation	Le meilleur pour les tâches de recommandation séquentielles et l'analyse des données temporelles.

TABLE 1.3 – Comparaison des approches SVD et RNN

### 1.5.4 Reconnaissance du texte

La reconnaissance du texte est une tâche courante dans le domaine du NLP, qui vise à extraire des informations significatives à partir de données textuelles non structurées. Plusieurs approches ont été utilisées pour résoudre ce problème, notamment :

- **Approche basée sur les règles :** Cette approche consiste à définir des règles spécifiques pour extraire des informations du texte. Cependant, cette méthode est limitée par sa dépendance à des règles spécifiques et peut ne pas être capable de s'adapter à des données nouvelles ou inconnues.
- **Approche basée sur l'apprentissage automatique :** Cette approche utilise des techniques d'apprentissage automatique pour entraîner des modèles capables d'apprendre automatiquement à extraire des informations du texte. Parmi les techniques les plus couramment utilisées, on trouve les modèles de séquence tels que les réseaux de neurones récurrents (RNN) et les modèles de Markov cachés (HMM).
- **Approche basée sur les réseaux de neurones :** les réseaux de neurones, en particulier les réseaux de neurones profonds, ont récemment connu un grand succès dans le domaine de la reconnaissance de texte. Ces modèles sont capables d'apprendre des représentations complexes du texte et peuvent obtenir de très bonnes performances sur une variété de tâches de NLP.

Les tableaux ci-dessous présente une comparaison de quatre approches d'apprentissage automatique pour le traitement du langage naturel : le zero-shot learning, DistilBERT, les classificateurs bayésiens MultinomialNB et les réseaux de convolution (CNN), selon plusieurs critères tels que leur fonctionnement, leurs besoins en données et entraînement, leurs performances et leurs cas d'usage.

## 1.5 Etat de l'art

Caractéristique	Zero-Shot	DistilBERT
Approche	Utilise des modèles pré-entraînés pour effectuer des prédictions sans besoin d'entraînement spécifique sur la tâche cible.	Utilise un modèle BERT allégé pour des tâches NLP, pré-entraîné sur une grande quantité de texte.
Entraînement	Pas besoin de réentraînement spécifique.	Peut nécessiter un affinement (fine-tuning) sur des données spécifiques.
Exigences en données	Faibles, peut fonctionner avec peu ou pas de données spécifiques à la tâche.	Moyennes, nécessite des données pour l'affinement.
Performance	Bonne pour des tâches générales et des domaines avec peu de données.	Très bonne performance pour diverses tâches NLP après affinement.
Complexité	Élevée, implique une factorisation matricielle complexe et intensément calculée.	Moyenne à élevée, nécessite des connaissances en ajustement de modèles pré-entraînés.
Latence	Faible, facile à mettre en œuvre avec des bibliothèques telles que Hugging Face.	Moyenne, plus rapide que BERT complet mais toujours exigeant en calcul.
Cas d'utilisation	Idéal pour des tâches de classification rapide où peu ou pas de données étiquetées sont disponibles.	Bon pour des tâches nécessitant une compréhension fine du langage naturel après ajustement.

TABLE 1.4 – Comparaison des approches Zero-Shot et DistilBERT

Caractéristique	MultiOutputClassfier	CNN
Approche	Combinaison de classificateurs Bayésiens pour chaque sortie, adapté pour des tâches multi-étiquettes.	Réseau de neurones convolutifs appliqué aux embeddings de texte.
Entraînement	Entraînement nécessaire sur des données étiquetées.	Entraînement nécessaire sur des données étiquetées.
Exigences en données	Élevées, nécessite beaucoup de données étiquetées.	Élevées, nécessite beaucoup de données étiquetées.
Performance	Performance variable, dépend de la qualité et de la quantité des données d'entraînement.	Bonne performance, surtout pour des tâches nécessitant une compréhension complexe des relations dans le texte.
Complexité	Moyenne, nécessite une bonne compréhension des méthodes de classification Bayésienne.	Élevée, nécessite des connaissances en réseaux de neurones et en traitement des embeddings.
Latence	Faible, les modèles bayésiens sont généralement rapides.	Variable, dépend de la complexité du réseau et des embeddings.
Cas d'utilisation	Adapté pour des tâches multi-étiquettes avec des données d'entraînement suffisantes.	Utilisé pour des tâches complexes de compréhension du texte et des relations entre les mots.

TABLE 1.5 – Comparaison des approches MNB et CNN

Pour mettre en place notre système de recommandation personnalisée, nous avons étudié plusieurs algorithmes de classification de texte et de recherche de similarité.

Tout d'abord, concernant la reconnaissance du texte, nous avons comparé les performances d'un modèle de classification multilabel DistilBERT contre une approche de classification zero-shot. Bien que DistilBERT ait obtenu de bonnes métriques sur les jeux de données standards, sa capacité de généralisation à de nouvelles catégories ouvertes était limitée. À l'inverse, la classification zero-shot s'est avérée mieux adaptée pour classer les réponses des utilisateurs dans nos catégories prédéfinies, sans nécessiter d'exemples d'apprentissage spécifiques.

Pour la recherche de similarité, nous avons évalué les algorithmes KNN, SVM et SVD sur un échantillon de notre jeu de données. Si SVD a présenté de meilleures performances en termes de précision, sa complexité calculatoire ne permettrait pas de traiter efficacement des requêtes en temps réel sur l'ensemble de notre base. Au contraire, KNN avec la similarité cosinus s'est révélé suffisamment performant pour notre problématique, tout en étant plus léger et efficient à mettre en œuvre.

Au final, notre choix s'est donc porté sur une architecture hybride combinant une classification zero-shot du texte et un modèle KNN de recherche de similarité, afin de proposer des recommandations personnalisées en temps réel à partir des retours spontanés des utilisateurs.

## 1.6 Conclusion

Grâce à ce chapitre, nous avons pu saisir le contexte général de notre projet et analyser les critiques de l'existant ainsi que la méthodologie à suivre. Le prochain chapitre se concentrera sur la première étape clé du processus CRISP-DM, à savoir la compréhension du métier.

# Compréhension du métier

---

## 2.1 Introduction

Ce chapitre présente la phase de départ de notre application. Tout d'abord, nous décrivons le domaine d'activité de l'entreprise et ses produits afin de cadrer le périmètre fonctionnel du futur système. Puis, nous listons les besoins fonctionnels et non fonctionnels de notre système. Ensuite, nous identifions les principaux objectifs métier et contraintes auxquels devra répondre le système de recommandation pour assurer sa réussite.

## 2.2 Description du domaine des produits 3D Crystal

Les entreprises des produits "3D Crystal" sont spécialisée dans la conception et la fabrication de produits en résine époxy décorés de motifs et figures en 3D. Leurs produits incluent :

- Objets de décoration domestique (figurines, bougeoirs, porte-clés etc.)
- Badges et insignes pour événements et conventions
- Accessoires

Les produits sont personnalisés sur demande grâce à l'impression 3D. Les clients peuvent choisir parmi une bibliothèque de motifs prédéfinis ou soumettre leur propre design.

La personnalisation rend chaque pièce unique. Cependant, cela complique aussi la tâche marketing pour suggérer des produits aux clients. Actuellement, les recommandations se font de manière aléatoire sur le site web.

## 2.3 Analyse des besoins

Dans cette section, nous allons présenter les utilisateurs de notre application tout en identifiant les besoins fonctionnels et non fonctionnels.

### 2.3.1 Besoins fonctionnels

Cette partie décrit les exigences que le système doit satisfaire d'une façon informelle. Les fonctionnalités que nous proposons dans notre application sont les suivantes :

#### 1. Intégration des données provenant de Shopify

- Shopify est une plateforme e-commerce populaire qui permet aux entreprises de créer et de gérer leurs boutiques en ligne. Le système doit être capable de récupérer les données des produits disponibles à partir de Shopify, y compris les descriptions, les images, les prix, etc.

#### 2. Gestion des questions pour analyser les préférences de l'utilisateur

- Le système doit permettre la création et la gestion des questions posées aux utilisateurs pour comprendre leurs préférences.
- Le système doit être capable de stocker les réponses des utilisateurs pour les utiliser dans l'algorithme de recommandation.

#### 3. Développement de l'algorithme de recommandation basé sur les réponses aux questions :

- Le système doit mettre en place un algorithme qui analyse les réponses aux questions pour comprendre les préférences de l'utilisateur.
- Il doit utiliser ces informations pour générer des recommandations personnalisées, en mettant l'accent sur les produits correspondant aux préférences exprimées.

#### 4. Affichage des recommandations personnalisées :

- Le système doit fournir une interface utilisateur conviviale pour afficher les recommandations personnalisées aux utilisateurs.
- Les recommandations doivent être présentées de manière attrayante, avec des informations claires sur les produits recommandés et leur pertinence par rapport aux préférences de l'utilisateur.

- L'interface utilisateur doit permettre aux utilisateurs d'interagir avec les recommandations, par exemple en ajoutant des produits au panier ou en les sauvegardant pour une consultation ultérieure.

### 5. Évaluation et ajustement de l'algorithme de recommandation :

- Le système doit inclure des mécanismes pour évaluer l'efficacité et la pertinence des recommandations fournies.
- Il doit permettre aux administrateurs ou aux développeurs d'analyser les résultats et d'apporter des ajustements à l'algorithme pour améliorer les performances.
- L'évaluation peut inclure des mesures telles que le taux de conversion, la satisfaction des utilisateurs ou d'autres indicateurs pertinents.

### 2.3.2 Besoins non fonctionnels

#### Scalabilité

- Le système doit être capable de gérer un grand nombre d'utilisateurs simultanément, sans compromettre les performances.

#### Convivialité

- L'interface utilisateur doit être conviviale et intuitive, permettant aux visiteurs du store de naviguer facilement et de comprendre les recommandations fournies.

## 2.4 Objectifs et Contraintes du projet

Les objectifs décrivent de manière générale ce que le système de recommandation devra permettre de réaliser. Ainsi, Les contraintes identifient les limitations et difficultés susceptibles d'impacter le développement du projet.

### 2.4.1 Objectifs du système de recommandation :

Afin de répondre aux enjeux métier et aux attentes de l'entreprise, le système devra remplir les objectifs suivants :

- Améliorer l'expérience utilisateur en proposant des produits personnalisés répondant à ses centres d'intérêt tels que déclarés lors de la réponse aux questions de recommandations ou de la description de son intérêt.
- Booster le taux de conversion en orientant davantage les visiteurs vers des produits susceptibles de les intéresser.
- Soutenir la prise de décision clients en leur présentant une sélection plus restreinte de 3 à 5 produits seulement, adaptée à leur profil. L'objectif est de ne pas les submerger sous les choix.

### 2.4.2 Contraintes du projet

Afin de répondre de manière pertinente aux objectifs définis, le développement du système de recommandation devra prendre en compte un certain nombre de contraintes, à la fois techniques, métier et liées aux parties prenantes. Les principales contraintes identifiées sont :

#### **Contraintes techniques :**

- Traitement en temps réel des données utilisateur / base de données des produits
- Nécessité de traitements légers pour une expérience utilisateur fluide sur mobile comme sur ordinateur (temps de réponse optimale).
- Se connecter avec le système Shopify existant pour la récupération des données de stock, vente et catalogue produit.

#### **Contraintes métier :**

- Renouvellement fréquent de l'offre produits avec l'ajout de nouveaux designs et collections toutes les semaines.
- Stock parfois limité ou écoulable pour certains produits à durée de vie courte (ex : saisons, événements).



### **Enjeux business :**

- Fidéliser la clientèle sur un marché attractif mais très concurrentiel.
- Booster la notoriété de la marque grâce à des ventes accrues et du référencement naturel.
- Maintenir la rentabilité de l'entreprise en augmentant le panier moyen et le ROI.

### **Enjeux utilisateur :**

- Proposer une expérience d'achat sur-mesure et ludique à chaque visite.
- Répondre de façon proactive à leurs besoins par des suggestions personnalisées.

## **2.5 Conclusion**

Ce deuxième chapitre nous a permis de poser le contexte du projet et de définir de manière globale l'objectif et contraintes du futur système de recommandation. Le chapitre suivant portera désormais sur l'analyse détaillée des données disponibles.

# Exploration des données

---

## 3.1 Introduction

Ce chapitre présente l'analyse approfondie des principales sources de données nécessaires au développement d'un système de recommandation pour les produits en cristal 3D.

## 3.2 Présentation des sources de données

Cette section présente les trois sources des données disponibles pour notre système.

### 3.2.1 Base des produits 3D Crystal

La base de données produite constitue une source essentielle pour le développement du système de recommandation. Elle rassemble les détails techniques de plus de 1000 références uniques répertoriées sur notre site de vente en ligne des produits "3D Crystal". Afin de constituer cette base, nous avons eu recours à l'application Ablestar Bulk Product Editor dédiée à la plateforme Shopify dont le logo est affiché dans la figure 3.1 de la page suivante. Cet outil nous a permis d'interroger en masse le catalogue de produits hébergé sur le site Shopify de AB Crystal Collection de manière systématique et automatisée.

Plus précisément, Ablestar Bulk Product Editor offre la possibilité d'extraire efficacement les données de l'ensemble de fiches produites, y compris les méta-données structurées. Nous avons ainsi pu récupérer l'inventaire complet composé initialement de plus de 1000 références uniques sous la forme d'un unique fichier au format CSV.

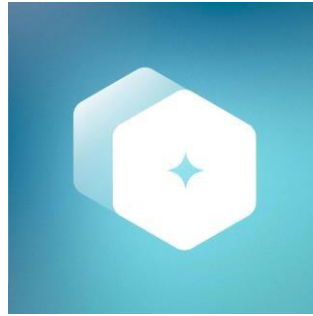


FIGURE 3.1 – Logo Application Ablestar Bulk Product Editor [6]

Ce fichier regroupe l’inventaire détaillé des produits qui sera exploité par la suite pour le développement et l’alimentation du système de recommandation. Il constitue une source de données clé pour sa mise en place.

#### 3.2.2 Ensemble de questions-réponses préétablies

Afin de mieux comprendre les préférences des utilisateurs, une série de questions leur est posée lors de leur navigation sur le site. L’analyse de ces questionnaires personnalisés fournira des informations essentielles pour le système de recommandation. Le site 3D Crystal s’appuie sur un ensemble de questions types, élaborées en amont, afin de collecter les avis et préférences des visiteurs de manière structurée. Ces questions préétablies, au nombre de 5, portent sur des thématiques suivantes :

1. **Emotions**
2. **Occasion**
3. **Centre d’intérêts**
4. **Audience**
5. **Style personnel**

Le tableau ci-dessous présente de façon structurée les thématiques abordées dans le questionnaire, les questions correspondantes ainsi que les réponses possibles pour chaque question :

### 3.2 Présentation des sources de données

Thème	Question	Réponses
Emotions souhaitées	What emotions or feelings would you like show in your gift?	Love and romance, Happiness and joy, Peace and tranquility, Inspiration and motivation, Sentimental and nostalgic
Occasion du cadeau	What is the occasion/event you a gift for?	Birthday, Anniversary, House Warming, Holiday, Celebration
centre d'intérêt du receveur	Could you please tell me about the hobbies and interests of receiver ?	Description libre (texte d'utilisateur)
Public cible	Is the gift targeted	Child, Teen, Adult, Senior
Style personnel	How would you describe your personal style and aesthetic preferences?	Design Créatif, Style rétro, Style nature, Tenues illustrées, Style artisanal, Style , Passion animale

TABLE 3.1 – Table des questions-réponses

#### 3.2.3 Données d'entrée utilisateur pour le NLP

Deux sources principales de données textuelles sont collectées auprès des utilisateurs :

1. **Réponses aux questionnaires** Les réponses standardisées aux questions permettent de comprendre directement certaines préférences comme le style, les thématiques d'intérêt, les émotions souhaitées pour le cadeau, etc.
2. **Descriptions libres du profil et des besoins** Nous offrons un champ du texte libre au utilisateur pour décrire ses choix préféré afin d'extraire ces informations :
  - **Description du profil du receveur** : L'utilisateur est invité à renseigner librement les centres d'intérêt, la personnalité et le style esthétique préféré de la personne pour laquelle le cadeau est destiné.
  - **Expression des besoins et attentes pour le cadeau** : Un champ permet à l'utilisateur de décrire plus précisément le type de cadeau recherché, les éléments importants à prendre en compte, etc.
  - **Réponses textuelles à certaines questions** : Pour certaines questions, des réponses sous forme de texte libre peuvent être fournies.

## 3.3 Analyse des données

### 3.3.1 Analyse du fichier CSV des produits 3D Crystal

Le fichier CSV regroupant l'inventaire produits est une source clé pour le développement du système de recommandation. Son exploration approfondie est nécessaire pour paramétrer l'algorithme de recommandation.

#### 3.3.1.1 Structure des données

Le fichier CSV contient l'inventaire de l'ensemble des produits référencés sur le site de 3D Crystal.

Les principales colonnes identifiées sont :

1. **Product\_id** : identifiant unique de chaque produit.
2. **Title** : titre ou nom du produit.
3. **Description** : description textuelle détaillée du produit.
4. **Tags** : étiquettes/mots-clés associés au produit (catégories, matériaux, motifs...)
5. **Images** : liens vers les images du produit en haute résolution.
6. **Website\_url** : lien web vers la page détaillée du produit sur le site.
7. **Shopify\_url** : lien e-commerce vers la fiche produit dans la boutique Shopify.
8. **Metafields** : autres métadonnées structurées sur le produit (prix, stock, options...)

Le fichier contient actuellement un millier de lignes, chaque ligne décrivant un produit de manière structurée via les attributs listés. L'exploration préliminaire a permis de constater que certaines colonnes comme les tags ou la description contiennent des données manquantes pour certains produits. Ce jeu de données fournit les informations clés sur chaque produit qui seront utilisées par le système de recommandation, notamment pour le calcul de similarité entre produits.

#### 3.3.1.2 Évaluation de la qualité des données

L'évaluation de la qualité des données est essentielle pour garantir la fiabilité et la cohérence des informations contenues dans le jeu de données. Dans cette section, nous présentons une analyse approfondie de la qualité des données, en examinant plusieurs aspects clés.

##### **Taux de données manquantes par variable :**

- Certains attributs comme les tags sont manquants pour <10% des produits
- La description est complète à >90%

##### **Détection de valeurs aberrantes :**

- Certains Produits avec des prix négatifs ou nuls (Produits de test)

##### **Cohérence entre attributs :**

- Correspondance entre le nombre d'images et de liens : On a besoin d'une seule image et d'un seul lien, donc on conserve l'image principale et on élimine les autres images supplémentaires.
- Adéquation des tags avec la description : Chaque produit a au moins un seul tag unique par rapport aux autres produits

##### **Valeurs anormales**

- Descriptions trop longues ou non informatives : Toutes les descriptions sont générées par les modèles d'intelligence artificielle GPT pour s'assurer que toutes les descriptions ont à peu près la même longueur et le même niveau de détail que le titre des produits.

##### **Risques identifiés**

- Recommandations non pertinentes avec les tags manquants
- Perte de sens avec les descriptions trop vagues
- Les produits non filtré par catégorie

#### 3.3.2 Étude des questions-réponses

Après avoir analysé les données produites et compris la structure du catalogue, intéressons-nous à présent à la seconde source clé d'informations pour le système de recommandation : les réponses des utilisateurs aux questionnaires. Ces données

subjectives fournies directement par les clients sont essentielles pour personnaliser l'expérience et affiner les recommandations. Leur étude approfondie est donc nécessaire.

#### 3.3.2.1 Format des questions

L'ensemble des questions proposées dans le questionnaire utilisent un format à choix multiples, avec des réponses prédéfinies.

Cette approche présente plusieurs avantages :

- Elle simplifie et fluidifie l'expérience utilisateur en limitant le temps de réponse.
- Elle garantit l'homogénéité et la standardisation des réponses collectées.
- Elle facilite le traitement automatisé des données issues des questionnaires.
- Elle permet d'obtenir directement des profils utilisateurs formalisés sous forme codée.
- Elle limite les non-réponses grâce à l'exhaustivité des propositions de réponses.

Ce format de questions à choix multiples a donc été privilégié afin d'optimiser à la fois la qualité des données recueillies et l'ergonomie du questionnaire.

#### 3.3.2.2 Objectifs et rôle dans le système de recommandation

Les réponses fournies ont plusieurs finalités :

- Mieux cerner les attributs déterminants pour la satisfaction de l'utilisateur.
- Identifier des segments de clientèle aux attentes similaires.
- Personnaliser le contenu et l'ordre de présentation des recommandations.
- Affiner en continu le modèle de recommandation grâce aux nouvelles réponses collectées.
- Améliorer l'expérience utilisateur en la rendant plus contextuelle.

L'analyse des questions-réponses est donc essentielle pour orienter le développement du système de manière pertinente.

#### 3.3.3 Préparation des données textuelles pour la recommandation

Après La réception des entrées texte par l'utilisateur il faut d'abord prétraiter les Données pour le nettoyage afin d'analyser les choix des utilisateurs (occasion, émotions, intérêts, personnalité)

##### 3.3.3.1 Objectifs du prétraitement

Le prétraitement vise à structurer ces données texte pour l'analyse par NLP afin d'en extraire :

- Les termes-clés décrivant les préférences implicites
- Les relations sémantiques entre produits et thématiques
- La polarité (positif/négatif) sur certains attributs

##### 3.3.4 Étapes du prétraitement

Ces étapes visent à préparer les textes bruts pour l'analyse en appliquant notamment :

- Conversion en minuscules pour uniformiser et faciliter le traitement (exemple : "Cadeau" devient "cadeau")
- Suppression des accents et des caractères spéciaux qui peuvent poser problème (exemple : "café" devient "cafe")
- Elimination des retours à la ligne et des espaces multiples qui séparent artificiellement les termes
- Correction orthographique des mots mal écrits en se basant sur un dictionnaire (exemple : "interressé" devient "intéressé")
- Conservation des mots même incorrects si présents dans le vocabulaire produit
- Suppression des mots vides comme "le", "la", "des" n'apportant pas de sens



## **3.4 Conclusion**

Cette phase essentielle de compréhension du domaine et des données à disposition jette les bases pour la modélisation et le développement du système de recommandation répondant de manière optimale aux besoins exprimés. Les étapes suivantes consisteront à examiner la sélection et l'implémentation des algorithmes d'apprentissage automatique les plus adaptés.

# Modélisation et développement

---

## 4.1 Introduction

Après l'étape importante d'exploration et d'analyse des données présentée dans le chapitre précédent, nous allons à présent entamer la phase de modélisation et de développement du système de recommandation. Cette étape clé va nous permettre de formaliser les besoins métiers exprimés afin de concevoir une architecture technique répondant aux objectifs fixés. Ce chapitre a pour objectif de poser les bases de la conception et du développement proprement dit du système de recommandation.

## 4.2 Pipeline proposé

Le pipeline de modélisation et de développement pour notre système de recommandation de produits 3D Crystal suit un processus structuré en plusieurs étapes clés. Il débute par le chargement des données, englobant les informations sur les produits, les questions-réponses associées, et les descriptions textuelles. Ces données brutes subissent ensuite un prétraitement rigoureux, incluant le nettoyage du texte, la tokenisation, et l'application de techniques de réduction comme la suppression des mots vides et la lemmatisation. La phase de construction du modèle se divise en deux approches principales : la classification des produits, utilisant les méthodes KNN et SVD, et la classification du texte, exploitant les techniques de Zero-Shot Learning et DistilBERT. L'entraînement des modèles s'effectue sur des données divisées en ensembles d'entraînement, de validation et de test, suivi d'une étape d'optimisation pour affiner les hyper paramètres. L'évaluation rigoureuse des modèles permet de mesurer leurs performances et d'analyser les erreurs. L'intégration des modèles dans le système de recommandation se fait soit par similarité entre les questions-réponses et les produits, soit par traitement NLP du texte, aboutissant

à des recommandations personnalisées. Enfin, le déploiement du modèle optimisé conclut le processus, assurant son intégration dans l'environnement opérationnel. Ce pipeline complet, illustré dans la figure ci-dessus, offre une vue d'ensemble des étapes cruciales de notre approche de modélisation pour la recommandation de produits 3D crystal.

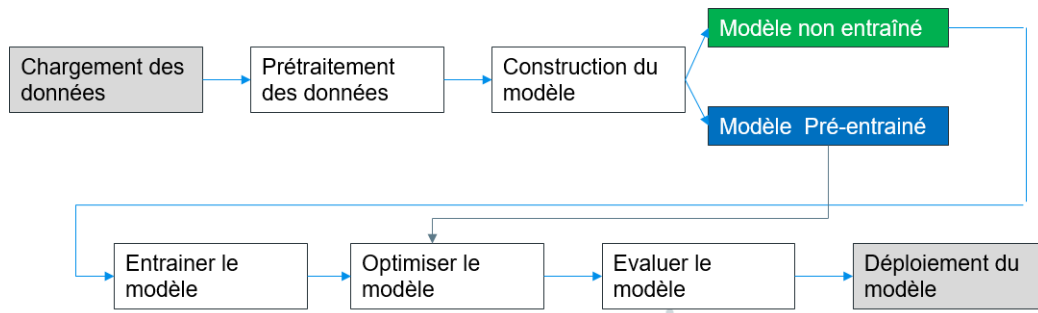


FIGURE 4.1 – Pipeline proposé

## 4.3 Construction du modèle

Après une comparaison théorique des modèles dans la phase initiale du projet, nous avons entamé la construction de deux modèles pour la recommandation de produits en cristal 3D : un basé sur l'approche des K plus proches voisins (KNN) pour la prédiction basée sur les réponses aux questions, et l'autre basé sur l'apprentissage zéro-shot pour la reconnaissance de descriptions textuelles entrées par l'utilisateur.

### 4.3.1 K plus proches voisins

#### 4.3.1.1 Fonctionnement du modèle

KNN classe un objet selon l'étiquette majoritaire parmi ses k plus proches voisins dans l'espace des caractéristiques, en se basant sur une mesure de distance (généralement euclidienne). Plus k est grand, plus la décision est robuste au bruit mais moins le modèle est réactif aux données locales. La figure dans la page suivante schématisé le principe du KNN :

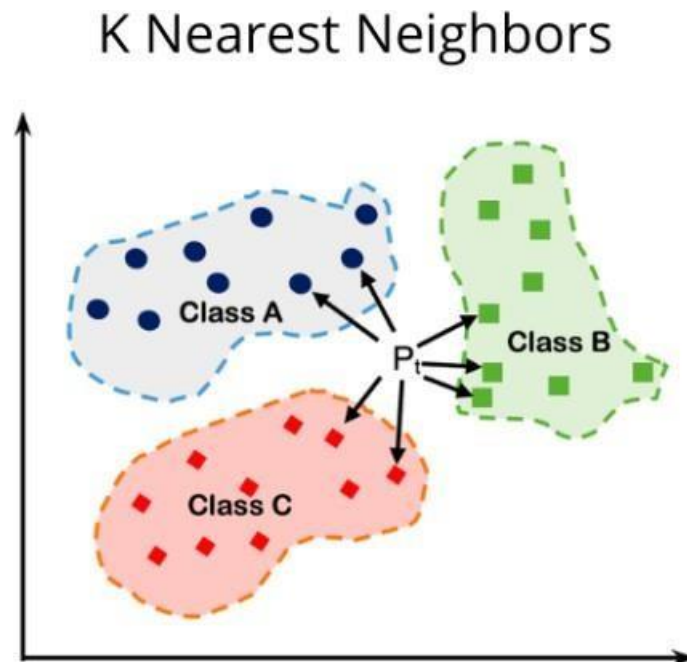


FIGURE 4.2 – Fonctionnement d’approche KNN

##### 4.3.1.2 Modélisation KNN

Notre objectif est de recommander des produits personnalisés aux visiteurs de notre site en se basant sur leurs réponses à des questions sur leurs centres d’intérêt et préférences. Pour cela, nous avons choisi d’implémenter un algorithme des K plus proches voisins.

Ce modèle non supervisé recherche dans un espace de représentation vectoriel les K produits les plus similaires à une nouvelle requête donnée. Il s’agit donc d’un algorithme de référence efficace pour la recommandation personnalisée.

Les principales étapes de notre modèle KNN sont les suivantes :

1. Prétraitement et vectorisation du texte : les descriptions de produits sont nettoyées puis représentées sous forme de vecteurs avec TF-IDF. Les figures ci-dessous présentent le code de la fonction du prétraitement et vectorisation des textes :

### 4.3 Construction du modèle

```
def preprocess_text(text):
    # Remove HTML tags
    text = re.sub(r'<[^>]*>', '', text)
    # Remove HTML entities
    text = re.sub(r'&[^\s]*;', '', text)
    # Remove non-alphabetic characters
    text = re.sub(r'[^a-zA-Z]', ' ', text)
    # Convert to lowercase
    text = text.lower()
    # Lemmatization
    lemmatizer = WordNetLemmatizer()
    words = text.split()
    lemmatized_words = [lemmatizer.lemmatize(word) for word in words]
    text = ' '.join(lemmatized_words)
    # Filter stop words
    stop_words = set(stopwords.words('english'))
    words = text.split()
    filtered_words = [word for word in words if word not in stop_words]
    text = ' '.join(filtered_words)
    return text
```

FIGURE 4.3 – Fonctionnement du prétraitement

```
vectorizer = TfidfVectorizer(max_features=5000)
tfidf_matrix = vectorizer.fit_transform(data["preprocessed_text"])
```

FIGURE 4.4 – Vectorisation des textes prétraité

TF-IDF (Term Frequency-Inverse Document Frequency) est une technique de vectorisation du texte. Elle permet de représenter l'importance d'un mot dans un document par rapport à une collection de documents.

- TF (Term Frequency) : fréquence d'un terme dans un document
- IDF (Inverse Document Frequency) : mesure de l'importance du terme dans l'ensemble du corpus

2. Calcul des Similarités : une matrice de similarités cosinus est calculée entre tous les vecteurs de produits. La figure ci-dessous présente le code du calcul du similarité :

```
nn_model = NearestNeighbors(metric='cosine', n_neighbors=3, algorithm='brute', n_jobs=4)
```

FIGURE 4.5 – Calcul des Similarités

- metric='cosine' : utilise la similarité cosinus pour mesurer la distance entre les vecteurs

### 4.3 Construction du modèle

---

- `n_neighbors=3` : recherche les 3 voisins les plus proches
  - `algorithm='brute'` : calcule la distance avec tous les points (efficace pour les petits à moyens ensembles de données)
  - `n_jobs=4` : utilise 4 cœurs de processeur pour accélérer le calcul
3. Entraînement du modèle : le modèle KNN est entraîné sur cette matrice pour apprendre la similarité entre produits. La figures ci-dessous présente le code d'entraînement du modèle :

```
nn_model.fit(tfidf_matrix)
```

FIGURE 4.6 – Entraînement du modèle KNN

4. Recommandation : les réponses de l'utilisateur sont prétraitées et comparées aux produits existants pour trouver les K plus proches voisins. La figures ci-dessous présente le code du comparaison :

```
_, indices = nn_model.kneighbors(input_vector, n_neighbors=3)
```

FIGURE 4.7 – Recherche des k voisins

5. Présentation : les produits recommandés sont renvoyés à l'utilisateur. Le code correspondant présenté dans la figure suivant :

```
recommended_products = data.iloc[indices[0]]  
titles = recommended_products["Title"].tolist()  
urls = recommended_products["URL (web)"].tolist()  
images = recommended_products["Image 1"].tolist()
```

FIGURE 4.8 – Renvoyer les produits recommandés

Ainsi, en décrivant leurs préférences, les visiteurs se voient proposer des suggestions personnalisées de produits susceptibles de les intéresser.

### 4.3.2 Classifieur zéro-shot

#### 4.3.2.1 Fonctionnement du modèle

Les classifieurs zéro-shot permettent de classer des données dans des catégories non vues à l'entraînement, sans exemples étiquetés pour ces nouvelles classes.

La figure 4.9 ci-dessous explique l'architecture du modèle

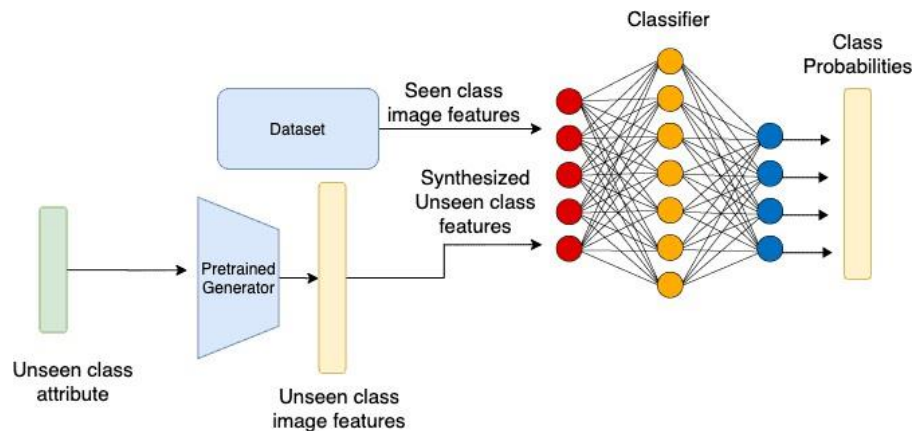


FIGURE 4.9 – Zero-Shot Learning

Notre système met en œuvre cette approche en deux étapes clés :

#### 1. Apprentissage des plongements lexicaux

Nous entraînons un modèle de plongement de mots non supervisé (Word2Vec par exemple) sur un grand corpus généraliste. Cet apprentissage projette chaque mot dans un espace vectoriel où les similarités sémantiques sont préservées. Il capture les régularités distributionnelles sans supervision.

#### 2. Classification zéro-shot Pour chaque nouveau texte d'entrée :

- Nous encodons son contenu et les labels candidats dans l'espace des plongements.
- Un calcul de similarité cosinus est effectué entre chaque représentation de label et le texte.
- Le label dont le vecteur de plongement est le plus proche du texte (cosinus maximal) est assigné.

Cette approche très peu supervisée permet de catégoriser les réponses de l'utilisateur dans les classes existantes de façon non supervisée. Les prédictions

sont ensuite transformées en sacs de mots avec TF-IDF, à l'instar des descriptions de produits. La similarité cosinus entre ces représentations vectorielles permet enfin de recommander les items les plus pertinents de manière contextualisée.

Notre système exploite judicieusement les capacités zéro-shot et la représentation vectorielle du langage pour formuler des recommandations adaptées.

#### 4.3.2.2 Modèle de classification zero-shot

Pour traiter les entrées textuelles libres des utilisateurs via la route NLP, nous avons opté pour une approche de classification zero-shot.

Cette technique consiste à classer un texte dans des catégories ouvertes sans avoir besoin d'exemples d'entraînement pour chaque catégorie. Elle repose sur les propriétés sémantiques et contextuelles des mots plutôt que sur des correspondances exactes.

Notre pipeline de classification zero-shot est basé sur le modèle pré-entraîné BERT (*Bidirectional Encoder Representations from Transformers*), en particulier sa variante BART développée par Facebook. Nous avons chargé le modèle avec la fonction pipeline de la bibliothèque transformers dont le code est présenté dans la figure ci-dessous :

```
# Chargement du pipeline de classification zero-shot avec un modèle plus petit
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli")
```

FIGURE 4.10 – Chargement du pipeline Zero-Shot

Les principales étapes sont les suivantes :

1. Le texte de l'utilisateur est prétraité pour normaliser le format. Le même code du prétraitement avec l'approche KNN sera appliqué.
2. Le texte est classé en parallèle dans chaque catégorie prédéfinie. La fonction de classification est présentée dans la figure ci-dessous :



### 4.3 Construction du modèle

---

```
# Fonction de classification du texte
def classify_text(user_input, category):
    """
    Cette fonction utilise le modèle de classification zero-shot pour classer le texte de l'utilisateur
    dans une catégorie donnée. Elle renvoie l'étiquette la plus probable pour cette catégorie.
    """
    labels = options[category]
    result = classifieur(user_input, labels)
    return result['labels'][0], category
```

FIGURE 4.11 – Fonction de classification du texte

3. Les étiquettes de classe prédites sont combinées de manière pondérée.

Le figure suivant explique le principe du fonctionnement :

```
predicted_labels = [label for label, _ in results]
processed_classes = ' '.join(predicted_labels)

weighted_processed_classes = f"{predicted_labels[2]} {predicted_labels[2]} {predicted_labels[4]}
{predicted_labels[3]} {predicted_labels[1]} {predicted_labels[0]}"
```

FIGURE 4.12 – Combinaison des étiquettes

4. Cette classification est vectorisée pour être comparée aux produits. Le code suivant illustre la vectorisation :

```
predicted_labels = [label for label, _ in results]
processed_classes = ' '.join(predicted_labels)

weighted_processed_classes = f"{predicted_labels[2]} {predicted_labels[2]} {predicted_labels[4]}
{predicted_labels[3]} {predicted_labels[1]} {predicted_labels[0]}"
```

FIGURE 4.13 – Vectorisation des classes

5. Le modèle KNN recherche les produits les plus similaires. le même fonction neighbors utilisé dans l'approche précédente sera appliqué sur le vecteur normalisé.

Ainsi, sans exemples d'apprentissage, le modèle BERT capture la sémantique des questions pour effectuer une classification pertinente vers les catégories de produits.

## 4.4 Conclusion

Ce chapitre a jeté les bases de la conception et du développement du système de recommandation en alignant les besoins métiers avec une architecture technique appropriée. En suivant une approche méthodique et itérative, nous avons pu créer un système de recommandation robuste et fiable, capable de répondre aux attentes des utilisateurs et de contribuer aux objectifs de l'entreprise. Dans le chapitre suivant, nous évaluerons les performances du système de recommandation ainsi que le déploiement du modèle dans une application.

# Evaluation et Déploiement

---

## 5.1 Introduction

Ce chapitre se concentre sur l'évaluation et le déploiement des résultats obtenus dans notre projet de recommandation, en mettant en avant l'importance de mesurer les performances du modèle développé et de l'intégrer dans un environnement réel pour une utilisation pratique, tout en considérant les aspects techniques, la formation des utilisateurs et l'évaluation continue des résultats déployés.

## 5.2 Critères d'évaluation théorique

Au cours de notre processus d'évaluation nous mettons en compte ces critères :

### **Performance et temps de réponse**

- Assurer que le système de recommandation répond rapidement aux requêtes des utilisateurs, en optimisant les algorithmes et les opérations de traitement des données.
- Utiliser des techniques de mise en cache pour accélérer les réponses et minimiser les temps de chargement.
- Effectuer des tests de charge et d'évolutivité pour garantir que le système peut gérer une charge importante sans compromettre ses performances.

### **Convivialité et expérience utilisateur**

- Concevez une interface utilisateur intuitive et conviviale pour faciliter la navigation et l'interaction avec le système de recommandation.
- Personnaliser les recommandations en fonction des préférences et des comportements des utilisateurs pour améliorer leur expérience.

### 5.3 Cycle de vie du projet

---

- Fournisse des options de filtrage et de tri pour permettre aux utilisateurs de trouver facilement les recommandations qui les intéressent.

#### Évolutivité et extensibilité du système

- Concevez le système de recommandation de manière modulaire et extensible, en utilisant des architectures flexibles et évolutives.
- Prévoyez des mécanismes pour ajouter de nouveaux types de recommandations ou étendre les fonctionnalités existantes sans compromettre la stabilité du système.

#### Performance Technique

- Infrastructure adaptée : Utiliser des ressources matérielles (RAM et CPU Performants) et logicielles (Visual Studio - Google Collab) adaptées pour exécuter nos algorithmes efficacement, en tenant compte de la charge attendue et des besoins en calcul intensif.
- Temp d'apprentissage : Utilisez des modèles pré-entraînés puissants tels que BERT ([facebook/bart-large](https://huggingface.co/facebook/bart-large)) pour bénéficier d'une meilleure performance de classification

### 5.3 Cycle de vie du projet

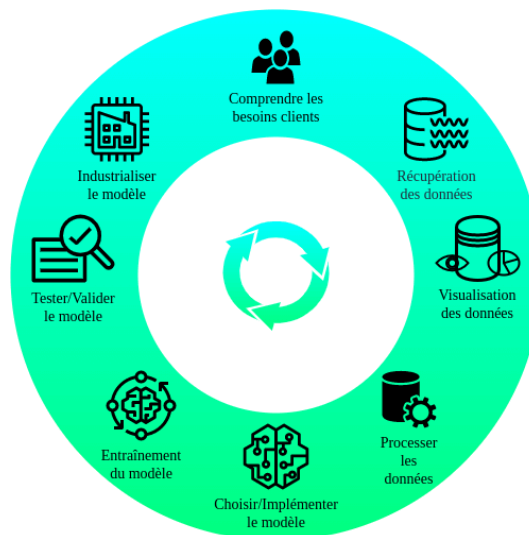


FIGURE 5.1 – Cycle de vie du projet

L'évaluation joue un rôle essentiel dans le cycle de vie d'un projet de data science qui est présenté dans la figure 5.1 dans la page précédente, surtout pour "Comprendre les besoins clients". Elle mesure l'efficacité, la performance et la validité des modèles développés, et assure leur conformité aux objectifs du projet et aux exigences des parties prenantes.

Elle quantifie objectivement la capacité du modèle à résoudre les problèmes identifiés et à fournir des résultats fiables. Cela nécessite la sélection de mesures adéquates, telles que l'exactitude, la précision, le rappel, la F-mesure, etc., en fonction de la nature du problème et des objectifs spécifiques. Pour assurer une évaluation efficace, il est crucial d'employer des méthodologies rigoureuses afin de garantir des résultats fiables et généralisables.

Dans la section suivante, nous détaillerons la méthodologie employée pour mener à bien cette évaluation.

## 5.4 Méthodologie d'évaluation

L'évaluation de notre système de recommandation et de classification de texte sans apprentissage est cruciale pour mesurer sa qualité et sa performance. Notre approche combine la classification zero-shot pour l'analyse du langage naturel (NLP) et un système de recommandation basé sur les réponses à des questions spécifiques. Cette méthodologie nous permet d'évaluer la précision de nos prédictions et la pertinence de nos recommandations sans nécessiter d'exemples d'entraînement spécifiques pour chaque classe ou produit.

### 5.4.1 Évaluation du système de recommandation basé sur les questions

Pour évaluer notre système de recommandation basé sur les questions, nous avons mis en place la méthodologie suivante :

1. Jeu de données de test : Nous avons créé un ensemble de données de test (test\_questions\_data) comprenant des entrées utilisateur simulées avec les champs 'émotions', 'occasion', 'intérêts', 'audience', et 'personnalité', ainsi que les produits attendus pour chaque entrée.

### 2. Mesures de performance :

- Précision : Nous calculons le ratio entre le nombre de produits correctement recommandés et le nombre total de recommandations.
- Temps de traitement : Nous mesurons le temps moyen nécessaire pour générer des recommandations.

### 3. Processus d'évaluation :

- Pour chaque cas de test, nous appelons notre fonction de recommandation.
- Nous comparons les produits recommandés avec les produits attendus.
- Nous calculons la précision et le temps de traitement moyen.

La figure ci-dessous montre le code de notre fonction :

```
def evaluate_questions(test_data):
    correct_predictions = 0
    total_predictions = 0
    total_time = 0

    for test_case in test_data:
        start_time = time.time()

        # Appel direct à la fonction de recommandation
        recommended_products = questions(test_case['input'])

        end_time = time.time()
        total_time += (end_time - start_time)

        expected_products = test_case['expected_products']

        correct_predictions += len(set(recommended_products) & set(expected_products))
        total_predictions += len(recommended_products)

    precision = correct_predictions / total_predictions if total_predictions > 0 else 0
    average_time = total_time / len(test_data)

    return {
        'precision': precision,
        'average_time': average_time
    }
```

FIGURE 5.2 – Fonction d'évaluation du modèle KNN

### 5.4.2 Évaluation de la classification de texte sans apprentissage (NLP)

Pour évaluer notre système de classification de texte sans apprentissage, nous utilisons l'approche suivante :

1. Jeu de données de test : Nous avons créé un ensemble de données de test (test\_nlp\_data) contenant des entrées textuelles d'utilisateurs et les catégories attendues pour chaque entrée.
2. Mesures de performance :
  - Précision (Accuracy) : Pourcentage de prédictions correctes sur l'ensemble des prédictions.
  - Précision pondérée (Weighted Precision) : Moyenne de la précision pour chaque classe, pondérée par le nombre d'échantillons de cette classe.
  - Rappel pondéré (Weighted Recall) : Moyenne du rappel pour chaque classe, pondérée par le nombre d'échantillons de cette classe.
  - Score F1 pondéré : Moyenne harmonique de la précision et du rappel, pondérée par le nombre d'échantillons de chaque classe.
  - Temps de traitement : Temps moyen nécessaire pour classer une entrée.
3. Processus d'évaluation :
  - Pour chaque cas de test, nous appelons notre fonction de classification NLP.
  - Nous comparons les catégories prédites avec les catégories attendues.
  - Nous calculons les métriques de performance (précision, rappel, score F1) et le temps de traitement moyen.

La figure ci-dessous montre le code de notre fonction :

```
def evaluate_questions(test_data):
    correct_predictions = 0
    total_predictions = 0
    total_time = 0

    for test_case in test_data:
        start_time = time.time()

        # Appel direct à la fonction de recommandation
        recommended_products = questions(test_case['input'])

        end_time = time.time()
        total_time += (end_time - start_time)

        expected_products = test_case['expected_products']

        correct_predictions += len(set(recommended_products) & set(expected_products))
        total_predictions += len(recommended_products)

    precision = correct_predictions / total_predictions if total_predictions > 0 else 0
    average_time = total_time / len(test_data)

    return {
        'precision': precision,
        'average_time': average_time
    }
```

FIGURE 5.3 – Fonction d’évaluation du modèle ZERO-SHOT

## 5.5 Déploiement

La dernière étape du méthodologie CRISP-DM est le déploiement, où nous utiliserons diverses technologies pour créer une application web conviviale et facile à utiliser. Cette section examine les technologies utilisées pour le déploiement et les interfaces de notre application. Cela nous permettra de proposer une solution intégrée et efficace pour la reconnaissance automatique des plaques minéralogiques.



### 5.5.1 Technologies utilisées

#### . Python



FIGURE 5.4 – Logo Python

Python est un langage de programmation interprété, multiparadigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet [7].

#### . Flask



FIGURE 5.5 – Logo Flask

Flask est un micro framework open-source de développement web en Python. Il est classé comme microframework car il est très léger. Flask a pour objectif de garder un noyau simple mais extensible [8].

### . HTML



FIGURE 5.6 – Logo HTML

Le HyperText Markup Language, généralement abrégé HTML ou, dans sa dernière version, HTML5, est le langage de balisage conçu pour représenter les pages web [9].

### . CSS



FIGURE 5.7 – Logo CSS

Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium [10].

### . JavaScript



FIGURE 5.8 – Logo Javascript

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives [11].

Avec les langages HTML et CSS, JavaScript sont au cœur des langages utilisées par les développeurs web.

### 5.5.2 Interfaces

Cette partie représente la phase de mise en œuvre de l'application. Nous exposons alors les différentes interfaces de notre solution proposée.

#### . Page d'accueil

Lorsque vous utilisez notre application Web, vous serez redirigé vers notre page d'accueil qui propose deux interfaces utilisateur. Ces interfaces vous permettent d'interagir avec notre système de recommandation de cadeaux en cristal 3D en utilisant soit un questionnaire interactif, soit une description du destinataire du cadeau.

La page d'accueil est représentée dans l'illustration ci-dessous :

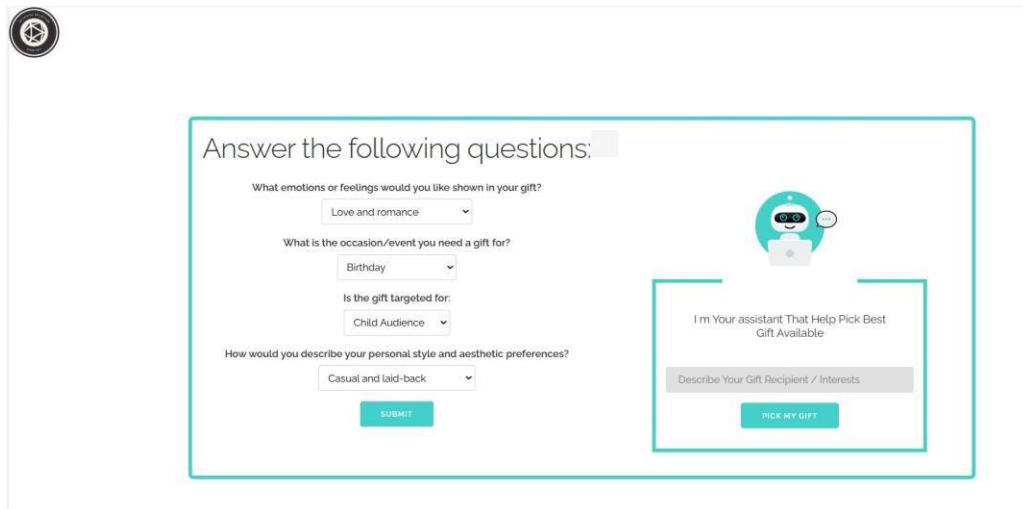


FIGURE 5.9 – Page d'accueil

#### . Page de recommandation basée sur le traitement des questions/réponses

Une fois l'utilisateur fourni ses réponses aux questions, notre système entre en action. Il compare ses informations avec les titres et les descriptions des produits que nous avons préalablement stockés dans notre base de données.

## 5.5 Déploiement

Grâce à des algorithmes de similarité, nous identifions les trois produits les plus pertinents et les plus similaires à ses préférences qui seront affichés dans la page.

La page de recommandation est représentée dans l'illustration ci-dessous :

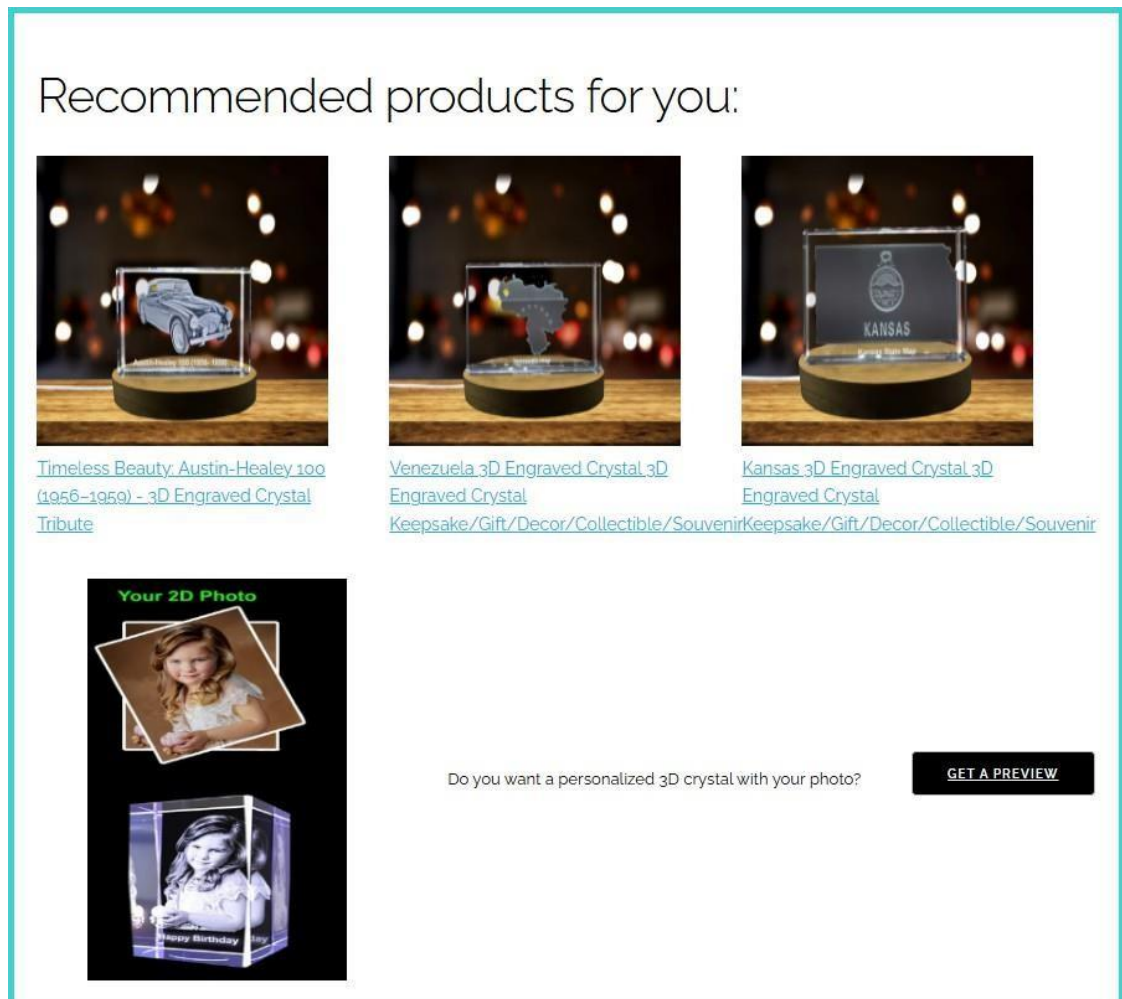


FIGURE 5.10 – Page de recommandation basée sur le traitement des questions/réponses

### . Page de recommandation basée sur le traitement du text NLP

Notre page de recommandation basée sur le traitement du texte NLP vous offre une solution avancée pour trouver les produits qui correspondent à vos besoins. Après avoir soumis votre texte, notre système utilise des modèles de classification pré-entraînés pour prédire les catégories les plus appropriées à votre demande. Les classes prédites sont ensuite affichées, ce qui vous permet de comprendre comment votre texte a été interprété et catégorisé. Enfin, la page vous présente les trois meilleurs produits recommandés, soigneusement sélectionnés en fonction de leur pertinence par rapport à votre demande initiale.

La page de recommandation est représentée dans l'illustration ci-dessous :

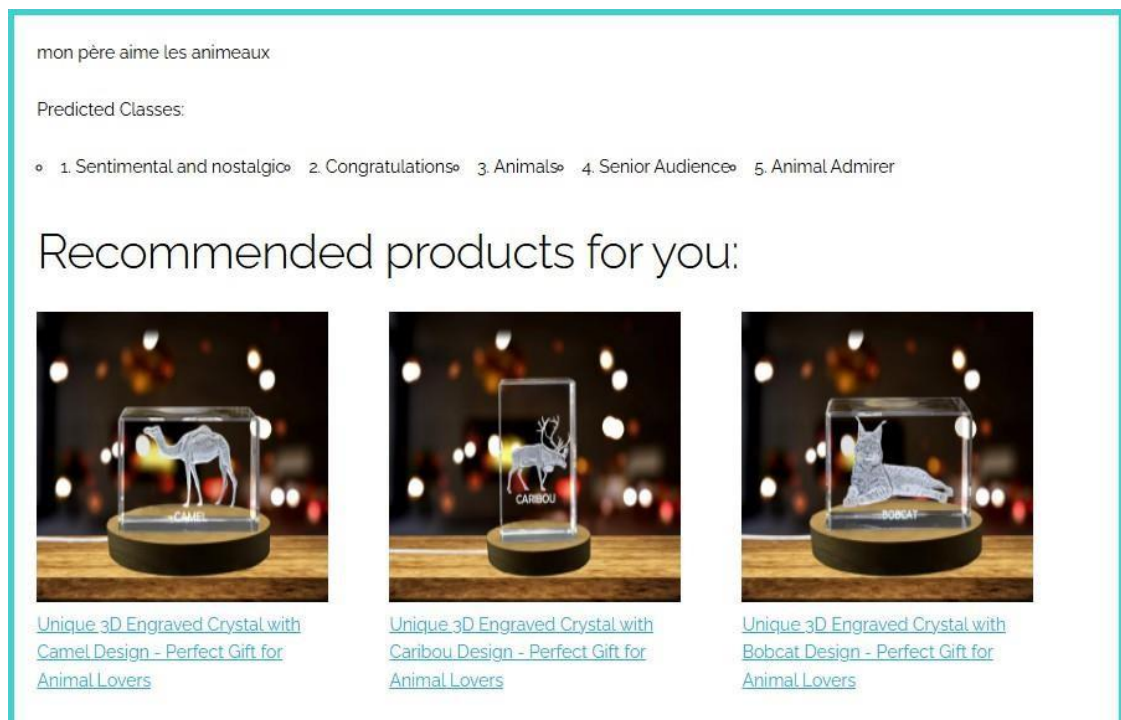


FIGURE 5.11 – Page de recommandation basée sur le traitement du texte NLP

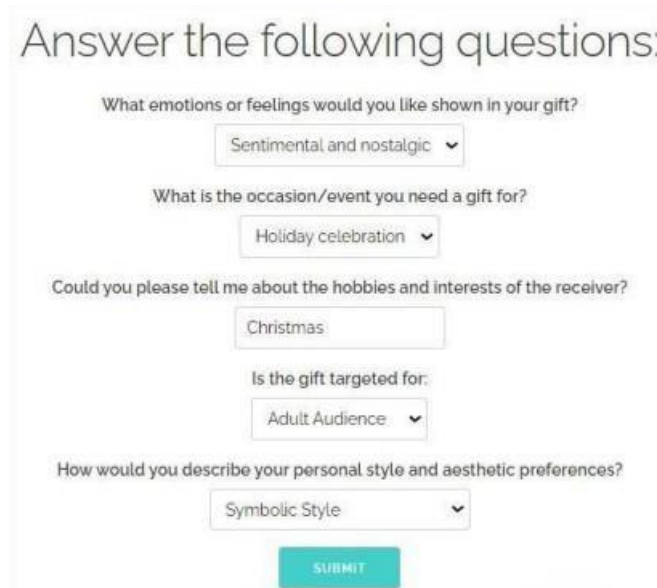
### 5.5.3 Scénario d'application

Cette section décrit un scénario d'application concret mettant en œuvre les capacités du notre système de recommandation. Elle vise à démontrer de manière tangible comment les techniques apprises peuvent être appliquées dans un cas d'usage réel et donner à voir les interactions possibles avec les utilisateurs via une interface de dialogue.

- **Classification du texte** : Cette étape consiste à analyser le contenu textuel fourni par l'utilisateur en répondant à 5 questions prédéfinies (ex : une description de ses centres d'intérêts, goûts, critères de recherche) afin d'en extraire les informations clés et de le classer dans une ou plusieurs catégories prédéfinies.

Les figures ci-dessous explique les éléments clés de l'application ainsi qu'un exemple d'interface pour chaque étape pour un **utilisateur cherche un cadeau en cristal dans l'occasion du Christmas** :

1. L'utilisateur est invité à répondre à un court questionnaire à choix multiples permettant d'identifier ses centres d'intérêt. Dans notre Cas on a choisis ces réponses :



Answer the following questions:

What emotions or feelings would you like shown in your gift?  
Sentimental and nostalgic ▼

What is the occasion/event you need a gift for?  
Holiday celebration ▼

Could you please tell me about the hobbies and interests of the receiver?  
Christmas

Is the gift targeted for:  
Adult Audience ▼

How would you describe your personal style and aesthetic preferences?  
Symbolic Style ▼

SUBMIT

FIGURE 5.12 – Scénario d'application : questionnaire à choix multiple

2. Ses réponses sont ensuite analysées à l'aide d'un modèle de classification du text (KNN dans notre cas) pour en extraire les informations clés et les associer à des catégories prédéfinies. La figure ci-dessous montre le code du modèle d'entrainement utilisé :

```
# Preprocess user answers
preprocessed_answers = preprocess_text(" ".join(answers.values()))

# Create TF-IDF matrix
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(data["preprocessed_text"])

# Apply SVD
k = 50 # Number of latent factors
svd = TruncatedSVD(n_components=k, random_state=42)
svd_matrix = svd.fit_transform(tfidf_matrix)

# Calculate cosine similarity between user answers and all products
user_answers_vector = vectorizer.transform([preprocessed_answers])
user_svd_vector = svd.transform(user_answers_vector)
similarities = cosine_similarity(user_svd_vector, svd_matrix)
```

FIGURE 5.13 – Application de l'algorithme SVD pour la recommandation de produits

3. La catégorie obtenant le score de similarité le plus élevé définit le profil de l'utilisateur.

```
# Get indices of top 3 similar products based on cosine similarity
top_3_indices = similarities.argsort()[0][-3:][::-1]
```

FIGURE 5.14 – Application de calcul du score de similarité

4. Le système recommande les 3 produits correspondant le mieux à ce profil parmi l'offre disponible. Et dans notre cas d'exemple le système affiche les trois produits dans la figure de la page suivante.

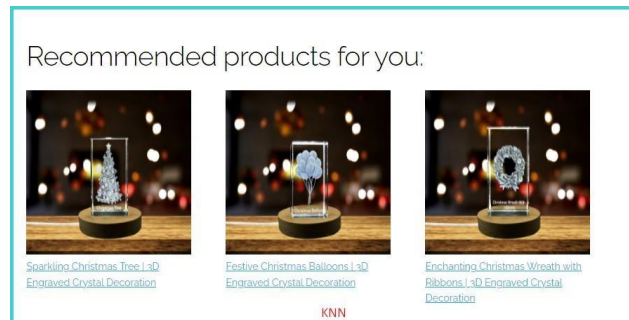


FIGURE 5.15 – Résultat de la classification du texte

5. Enfin, l'utilisateur peut naviguer un des produits recommandé :



FIGURE 5.16 – Page du produit

Le modèle a calculé la similarité des réponses avec les titres, descriptions et mots clés des produits disponibles dans la base des données et afficher des produits correspond au thème du décoration du noel.

- **Traitement des langues naturelles** : Cette étape consiste à analyser le texte saisi par l'utilisateur décrivant ses besoins (ex : "**I need gift for my father . He love Raising Pets at Home**") afin d'en extraire le sens et le contexte.



Les figures ci-dessous expliquent les éléments clés de l'application ainsi qu'un exemple d'interface pour chaque étape :

1. L'utilisateur décrit le destinataire du cadeau et ses intérêts et sa personnalité dans un zone du texte libre. Un exemple est présenté dans la figure ci-dessous :

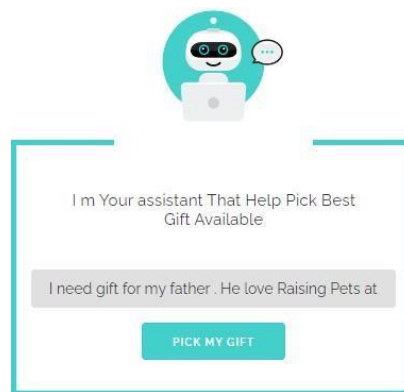


FIGURE 5.17 – Exemple du texte à traiter par le modèle de traitement du langage naturel

2. Le texte d'entrée est traité par un modèle de deep learning. Celui-ci identifie les entités, concepts, thèmes et intentions principaux du texte. La figure ci-dessous montre le code du modèle d'entraînement utilisé :

```
# Nouveau texte à classifier
new_paragraph = "I need gift for my father . He love Raising Pets at Home"

# Convertir les valeurs du dictionnaire en liste
labels = []
for values in options.values():
    labels.extend(values)

# Classification du nouveau texte
result = classifier(new_paragraph, labels)
```

FIGURE 5.18 – Application de l'algorithme "Zero-shot Classifier" pour la recommandation de produits

Sur la base de cette analyse linguistique, le modèle attribue au texte d'entrée un score de similarité pour chacune des catégories prédéfinies (matériau, occasion, public visé, style etc.).

3. Les 5 catégories obtenant les scores les plus élevés sont ensuite affichées à l'utilisateur avec pour chacune les 3 produits correspondants les mieux notés. Et dans notre cas d'exemple le système affiche les classes et produits suivants :

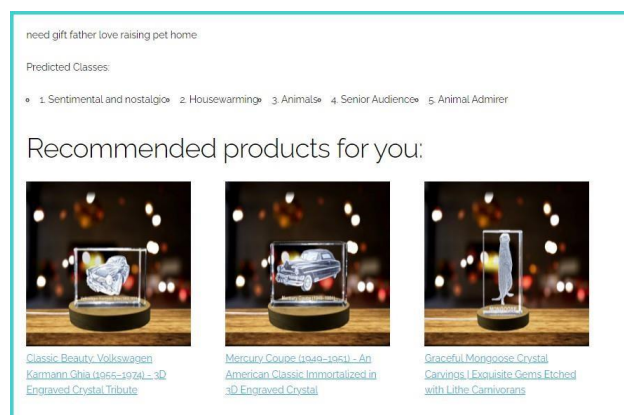


FIGURE 5.19 – Résultat du traitement du langage naturel

Le modèle a analysé le texte d'entrée "need gift father love raising pet home" pour prédire les 5 classes les plus pertinentes : 1) Sentimental et nostalgique, en identifiant les notions de love pour le père, 2) Housewarming, reliant raising au foyer, 3) Animals, via les mots pet et home évoquant les animaux de compagnie, 4) Senior Audience car le public visé est le père, et 5) Animal Admirer de par l'amour des animaux présent dans le texte. Sur cette base, les produits recommandés sont des sculptures de voitures anciennes ou d'animaux en cristal, à la fois symboliques de la nostalgie, du foyer et de l'animalité, correspondant bien aux thématiques de sentimentalité, maison/famille et animaux captées par le modèle de manière implicite dans le court texte.

### 5.5.4 Hébergement

Après avoir développé notre solution de recommandation, nous avons choisi de déployer et d'héberger notre application sur les Serveurs Virtuels Privés (VPS) proposés par le service AWS Lightsail.

AWS Lightsail est un service d'hébergement dans le cloud qui offre une grande facilité de déploiement et de gestion des applications web. Il fournit des instances de serveurs virtuels privés (VPS) qui peuvent être utilisées pour héberger divers types d'applications, y compris les applications Flask [12]. Leur logo est représenté dans la figure ci-dessous :



FIGURE 5.20 – Logo AWS Lightsail

Les VPS d'AWS Lightsail offrent plusieurs avantages :

- Facilité de déploiement : Lightsail permet de créer et de configurer rapidement des VPS adaptés à nos besoins, en fournissant des options préconfigurées pour différents types de serveurs.
- Gestion simplifiée : Lightsail offre une interface conviviale pour gérer nos VPS, y compris le redimensionnement des ressources, la surveillance des performances et la sauvegarde des données.
- Cout Adorable : Lightsail propose des tarifs compétitifs pour les VPS, ce qui le rend accessible même pour les petites applications

Cette solution nous permet de bénéficier d'une flexibilité accrue, car notre application n'est pas liée à une infrastructure spécifique et peut être facilement déplacée vers d'autres environnements de déploiement si nécessaire.

## 5.6 Conclusion

En conclusion de ce chapitre consacré à l'évaluation et au déploiement de notre solution, nous avons réalisé une évaluation approfondie de notre système, en mettant l'accent sur sa performance, sa précision et sa convivialité. Grâce à ces tests rigoureux, nous avons pu améliorer et optimiser notre application pour offrir une expérience utilisateur optimale. De plus, nous avons choisi de déployer notre application en utilisant les technologies principales, notamment Flask comme framework de développement et Heroku comme plateforme d'hébergement.

# Conclusion générale et perspectives

Dans cette étude, nous avons présenté un projet visant à développer un système de recommandation pour les produits 3D Crystal. Nous avons commencé par introduire le cadre et le contexte du projet, en mettant en évidence la problématique existante et en décrivant la solution proposée ainsi que les objectifs à atteindre.

Ensuite, nous avons examiné en détail le domaine des produits 3D Crystal et analysé les besoins des utilisateurs. Nous avons identifié les acteurs impliqués, les besoins fonctionnels et non fonctionnels, ainsi que les objectifs et les contraintes du projet.

L'étape suivante a été consacrée à l'exploration des données disponibles. Nous avons examiné les sources de données, notamment la base de produits 3D Crystal, les questions-réponses préétablies et les données d'entrée utilisateur pour le traitement du langage naturel (NLP). Nous avons effectué une analyse approfondie de ces données, y compris leur prétraitement, afin de les préparer pour la recommandation.

Dans la section de modélisation et de développement, nous avons présenté deux approches de recommandation : une basée sur les questions-réponses et une basée sur le NLP. Nous avons décrit les méthodes utilisées, y compris le prétraitement des données et la transformation en sacs de mots. De plus, nous avons exploré les principes du NLP et les algorithmes associés, en mettant en évidence l'utilisation d'un classifieur zéro-shot.

La phase d'évaluation et de déploiement a été abordée dans la section suivante. Nous avons défini les critères d'évaluation théorique, établi le cycle de vie du projet et présenté une méthodologie d'évaluation. Nous avons également discuté du déploiement du système, en détaillant les technologies utilisées, les interfaces mises

---

En place et les aspects liés à l'hébergement.

Bien que nous ayons développé un système de recommandation fonctionnel pour les produits 3D Crystal, il existe encore des possibilités d'expansion et d'amélioration. Par exemple, nous pourrions envisager d'intégrer des fonctionnalités supplémentaires telles que des recommandations basées sur les évaluations des utilisateurs, des recommandations contextuelles en fonction des tendances actuelles ou des recommandations croisées avec d'autres produits connexes.

En conclusion, ce projet de développement d'un système de recommandation pour les produits 3D Crystal représente une contribution significative dans le domaine de l'intelligence artificielle et du traitement du langage naturel. Les différentes étapes, de l'analyse des besoins à la modélisation et au déploiement, ont été abordées de manière approfondie, offrant ainsi une base solide pour la mise en œuvre du système. Les résultats obtenus à partir de l'évaluation théorique et les perspectives d'amélioration future montrent le potentiel et l'importance de ce projet dans le domaine des recommandations personnalisées.

# Bibliographie

## Web Bibliography

1. *Société Bouraoui Group* Google. Disponible sur : <https://thebouraouigroup.com/>.
2. *Artpix 3D* Google. Disponible sur : <https://artpix3d.com/>.
3. *Crystal Clear Memories* Google. Disponible sur : <https://www.crystalclearmemories.com/>.
4. *3dcystal.com* Google. Disponible sur : <https://www.3dcystal.com/>.
5. *Cross Industry Standard Process for Data Mining*. Wikipedia. Disponible sur : [https://fr.wikipedia.org/wiki/Cross\\_Industry\\_Standard\\_Process\\_for\\_Data\\_Mining](https://fr.wikipedia.org/wiki/Cross_Industry_Standard_Process_for_Data_Mining).
6. *Ablestar Bulk Product Editor* Shopify. Disponible sur : <https://apps.shopify.com/bulk-product-editor>.
7. *Python (langage)*. Wikipedia. Disponible sur : [https://fr.wikipedia.org/wiki/Python\\_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)).
8. *Flask (framework)*. Wikipedia. Disponible sur : [https://fr.wikipedia.org/wiki/Flask\\_\(framework\)](https://fr.wikipedia.org/wiki/Flask_(framework)).
9. *Hypertext Markup Language*. Wikipedia. Disponible sur : [https://fr.wikipedia.org/wiki/Hypertext\\_Markup\\_Language](https://fr.wikipedia.org/wiki/Hypertext_Markup_Language).
10. *Feuilles de style en cascade*. Wikipedia. Disponible sur : [https://fr.wikipedia.org/wiki/Feuilles\\_de\\_style\\_en\\_cascade](https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade).
11. *JavaScript*. Wikipedia. Disponible sur : <https://fr.wikipedia.org/wiki/JavaScript>.
12. *Cloud Application Platform*. Amazon Lightsail. Disponible sur : <https://aws.amazon.com/lightsail/>.

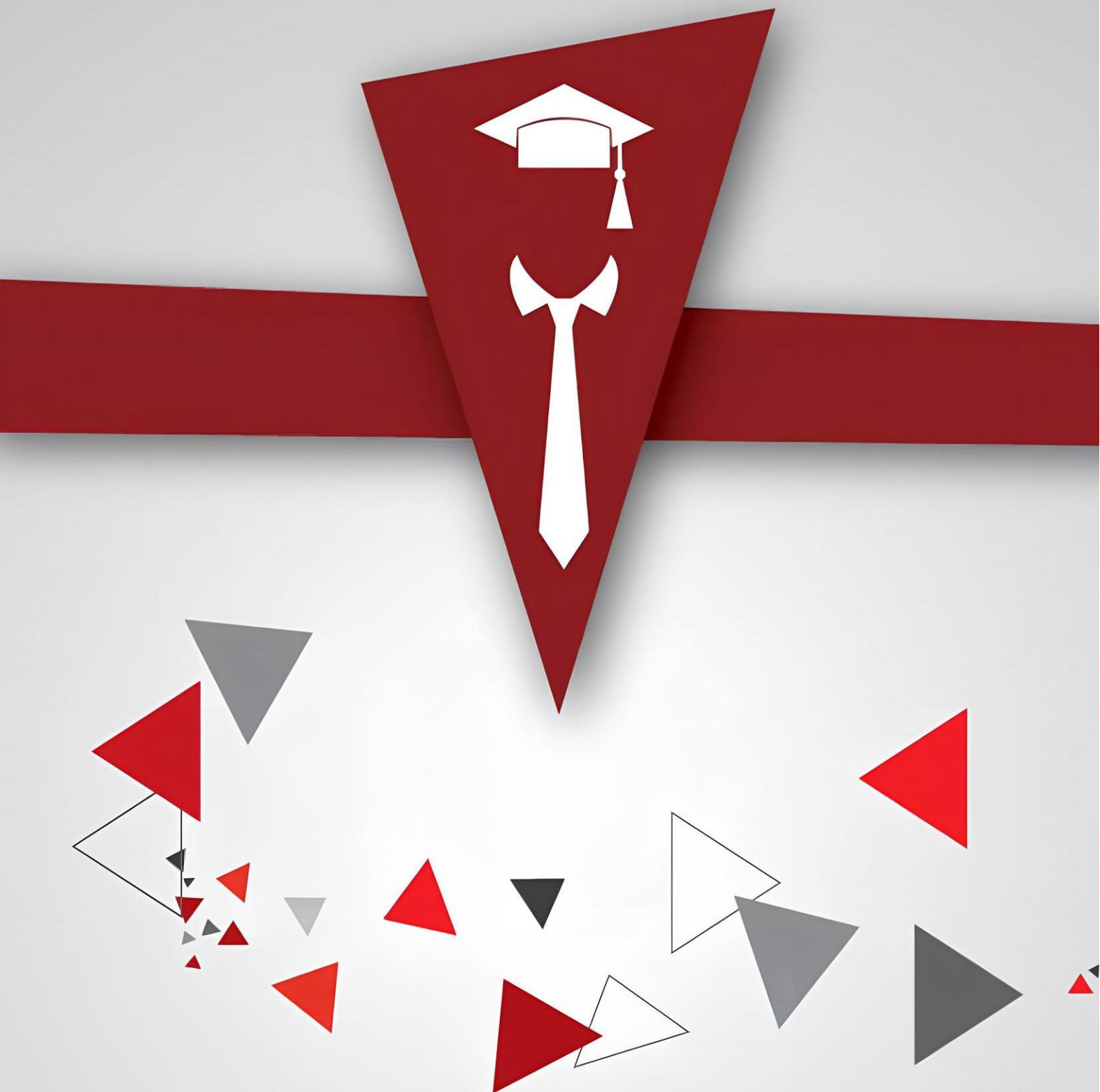
---

## Articles

13. *Qu'est-ce que l'apprentissage zero-shot?* IBM. Disponible sur : <https://www.ibm.com/fr-fr/topics/zero-shot-learning>.
14. *Zero-Shot Classification Using Transformers : Unlocking the Power of AI for Text-Based Tasks*. DataScience-ProF. Medium. Disponible sur : <https://tinyurl.com/medium-zero-shot-DSC>.
15. A. Mishra. *A Simple Explanation of Zero-Shot Learning and its Applications*. arXiv :1810.04805. Disponible sur : <https://arxiv.org/pdf/1810.04805.pdf>.
16. *NLP (Glove, BERT, TF-IDF, LSTM) Explained*. Kaggle. Disponible sur : <https://www.kaggle.com/code/andreshg/nlp-glove-bert-tf-idf-lstm-explained>.



---



## **ECOLE SUPÉRIEURE PRIVÉE D'INGÉNIEURS DE MONASTIR**

**[www.esprim.tn](http://www.esprim.tn) - E-mail : [contact@esprim.tn](mailto:contact@esprim.tn)**

**Adresse :** Boulevard de l'environnement BP 104, Monastir République – 5060 Tunisie

**Tél. :** +216 94 840 200 - **Fax. :** +216 73 500 200