Data Mining

Système de recommandation d'images

Juliette TARDY et Estelle ZHENG
4ETI

Objectif du Projet

L'objectif principal de ce projet est de développer un **système de recommandation d'images** de **chiens** et de **chats,** basé sur les préférences des utilisateurs. Pour cela, un ensemble d'activités automatisées a été mis en place, incluant la **collecte**, l'**annotation**, l'**analyse** et la **visualisation** des images. Enfin, un système de recommandation basé sur le contenu a été développé, et des tests ont été effectués pour valider le bon fonctionnement de ce système. Le système est constitué de plusieurs étapes :

1. **Acquisition.ipynb** : Collecte des images de chiens et de chats à partir de Wikipédia.
2. **Etiquetage.ipynb** : Étiquetage automatique des images avec différentes caractéristiques (orientation, qualité, type d'animal, nature, luminosité) et génère le fichier `étiquetage.json`.
3. **Interaction.ipynb** : Création du profil utilisateur sur un échantillon de 20 photos et génère un fichier `profil.json`
4. **Analyse.ipynb** : Analyse des données et des préférences utilisateur pour identifier les éléments clés.
5. **Recommandation.ipynb** : Génération des recommandations basées sur les préférences utilisateur.
6. **Visualisation.ipynb** : Visualisation des données sous forme de graphiques.

1. Collecte de données

La collecte des données a été automatisée à l'aide d'une requête **Wikidata**, qui a permis de télécharger **58 images de chats** et **58 images de chiens** provenant de **Wikipedia**, une base de données liée ouverte. Ces images sont toutes sous licence libre et peuvent être utilisées sans restriction. Les images sont stockées dans un dossier dédié **"/images"** et renommées par un numéro pour faciliter leur traitement.

2. Étiquetage et annotation

Les images ont été étiquetées selon 5 catégories à l'aide d'une approche automatisée :

1. **Orientation de l'image** (portrait, paysage, carré)
2. **Race de l'animal** (chien, chat)
3. **Qualité de l'image** (basse, moyenne, haute)
4. **Présence d'éléments naturels** (en nature, pas de nature)
5. **Luminosité de l'image** (sombre, lumineuse)

Extraction des métadonnées EXIF

Les métadonnées associées à chaque image ont été extraites. En fonction de la **largeur** et de la **hauteur** de l'image, **l'orientation** a été définie comme "portrait, paysage ou carré". De plus, le **nombre de pixels** a permis d'estimer la **qualité** de l'image comme "basse, moyenne, haute".

Prédiction de la race des animaux

Pour chaque photo, l'animal représenté est précisé si c'est un chien ou un chat. Bien que cette information aurait pu être récupérée via la requête Wikipédia, nous avons choisi d'utiliser une intelligence artificielle. **MobileNetV2** est un modèle de classification d'images basé sur le deep learning qui repose sur un réseau de neurones convolutifs (CNN). Grâce à son apprentissage supervisé sur un large ensemble de données annotées sur 1 000 catégories, il est capable de reconnaître les motifs visuels distinctifs. Cette méthode

présente l'avantage d'être plus flexible et de pouvoir être utilisée en particulier sur des photos ne provenant pas de Wikipedia et dont l'information ne pourrait pas être obtenue via cette plateforme.

Détection de la présence de végétation

Un algorithme a été développé pour analyser les images et déterminer si elles contiennent une proportion significative de végétation, les classant ainsi comme représentant un animal dans la nature ou non. Nous avions utilisé un algorithme de **clustering** KMeans **RGB** pour extraire les couleurs dominantes, puis vérifié si l'un des clusters correspondait à une couleur verte. Cependant, pour mieux distinguer les nuances de vert, nous avons préféré utiliser l'espace de couleur **HSV**. Chaque image est redimensionnée et convertie en HSV, puis filtrée selon des seuils pour la teinte, la saturation et la valeur. Si la proportion de pixels verts dépasse 15 %, l'image est classée comme étant dans un environnement naturel. Ce système permet d'automatiser le classement des images en fonction de leur contenu végétal.

Analyse de la luminosité

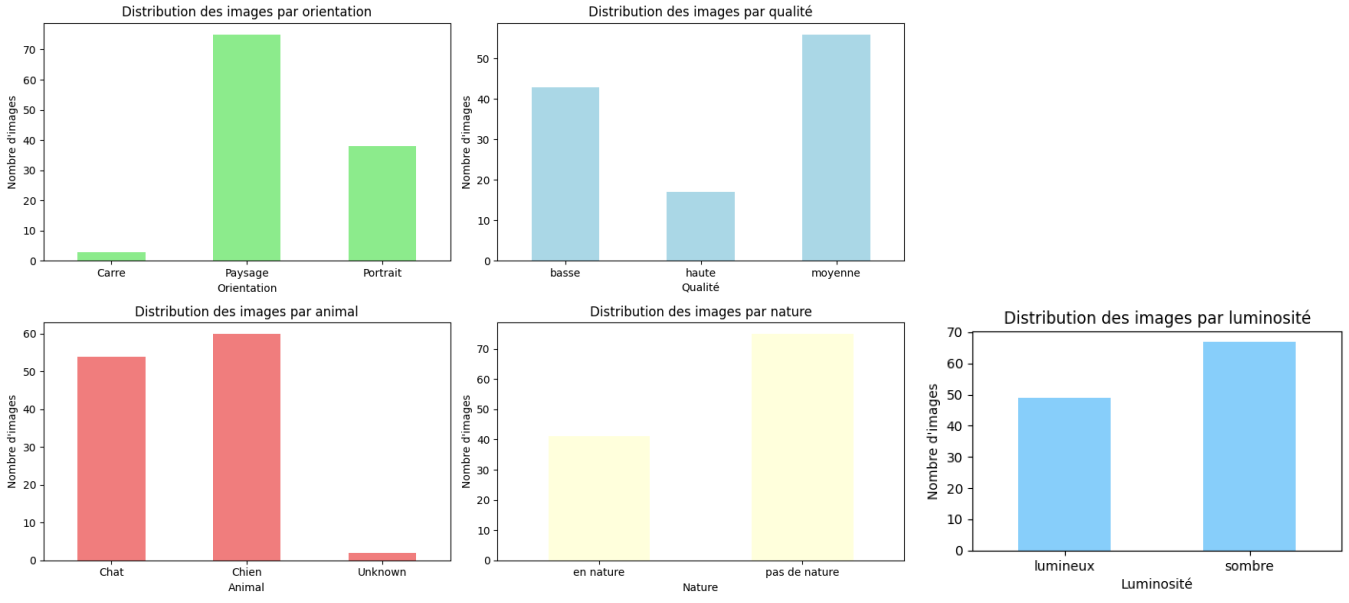
La luminosité des images a été déterminée en les convertissant en **niveaux de gris**, puis en calculant la moyenne des pixels de l'image. Une image en niveaux de gris contient des pixels dont les valeurs varient entre **0** (noir) et **255** (blanc). Si la moyenne des niveaux de gris dépasse un seuil de **128**, l'image est considérée comme lumineuse. Sinon, elle est classée comme sombre.


```
{
    "nom_image": "1.jpg",
    "orientation": "Paysage",
    "qualite": "moyenne",
    "animal": "Chien",
    "nature": "pas de nature",
    "luminosite": "sombre"
},
```


Extrait du fichier etiquetage.json

3. Visualisation des données

Afin de mieux comprendre la distribution des caractéristiques des images, la visualisation des données a été réalisée à l'aide de **Matplotlib** et **Pandas**. L'objectif étant d'identifier des tendances et des relations entre différentes variables. Les fréquences de chaque caractéristique ont été calculées en utilisant la fonction **groupby()**, permettant de regrouper les images par leurs attributs et de créer des graphiques en barres .



4. Construction d’un profil utilisateur

Le profil de chaque utilisateur est construit à partir de ses interactions avec le système. Un **échantillon d’images** (20 images prises au hasard parmi les 116 disponibles) est proposé à l’utilisateur sous forme de grille. Celui-ci peut marquer les images comme favorites ou non. Ces choix sont enregistrés dans un fichier **profil.json**, qui contient :

- **images** : Liste des 20 images proposées
- **data** : Balises associées aux images
- **result** : Indicateurs de mise en favori ou non

Un profil utilisateur est ainsi construit sur la base de ses préférences, qui peuvent être utilisées pour affiner les recommandations.

☑ Favorite

☐ Favorite

Extrait affichage de la grille pour la construction du profil utilisateur

5. Analyses de données

Dans cette section, nous analysons un profil utilisateur en fonction des préférences associées à un ensemble d’images favorites. Ces informations permettent d’adapter le système de recommandation en fonction des tendances observées dans les données d’interaction des utilisateurs.

Analyse des critères influents

Nous avons utilisé l'algorithme **Random Forest** de **Scikit-learn**, une méthode d'apprentissage supervisé qui repose sur la construction d'un ensemble d'arbres de décision pour la **classification**. Chaque arbre de décision dans la forêt est entraîné sur un sous-ensemble des données, ce qui améliore la robustesse du modèle en réduisant le risque de sur-apprentissage. Cet apprentissage des données permet de **prédire un résultat** si une image sera ajoutée aux favoris ou non et également d’identifier les **caractéristiques** les plus influentes qui affectent ses choix d'images.

Dans notre test, on construit un profil utilisateur qui met en favoris toutes les images présentant des chiens entourés de nature. L’analyse révèle ainsi les critères les plus influents. Cette analyse est proportionnellement fidèle au profil de l’utilisateur que nous avons construit.

```
Importances des caractéristiques (Random Forest) :
   - orientation: 0.085
   - qualite: 0.125
   - animal: 0.142
   - nature: 0.609
   - luminosite: 0.039
```

Analyse des tendances et tags des images favorites

Dans cette section, nous avons analysé les tendances de l'utilisateur en examinant les images qu'il a sélectionnées comme favorites. Pour cela, nous avons étudié les **fréquences** des caractéristiques les plus populaires et identifié les **tags** les plus appréciés, afin de dégager les principales tendances parmi les images favorites. On retrouve bien le résultat prévu pour notre utilisateur test, avec les tags préférés “chien” et “en nature”.

```
Fréquence des animaux dans les images favorites :
animal
Chien      5
Chat       1
Unknown    1
Name: count, dtype: int64

Fréquence en nature ou non dans les images favorites :
nature
en nature    7
Name: count, dtype: int64
```

```
Tag préféré par catégorie :
- Orientation : Paysage
- Qualité : moyenne
- Animal : Chien
- Nature : en nature
- Luminosité : sombre
```

6. Système de recommandation

Apprentissage non supervisé : c’est un type de méthode d'apprentissage automatique qui travaille sur des données non étiquetées, sans connaître les réponses ou catégories attendues. Il cherche à découvrir des structures cachées, comme des regroupements ou des modèles, en se basant uniquement sur les similarités ou les relations entre les données.

Le modèle de recommandation repose sur l’analyse des préférences de l’utilisateur en fonction des caractéristiques visuelles des images. Dans notre cas avec un seul profil, nous avons adopté un **filtrage basé sur le contenu**, où les images similaires à celles que l’utilisateur a marquées comme favorites sont recommandées.

L’algorithme se base uniquement sur les caractéristiques des données pour créer des regroupements (**clusters**, ici 4) en minimisant la distance entre les points à l’intérieur d’un même cluster. Cela permet d'identifier des similitudes sans qu'on ait besoin de dire à l'avance quelles sont les catégories ou les classes des éléments. Chaque image se voit attribuer une **étiquette de cluster** correspondant au groupe auquel elle appartient. Pour chaque image favorite de l'utilisateur, on détermine à quel cluster elle appartient, ce qui permet d'identifier les groupes d'images similaires. Pour chaque cluster identifié, l'algorithme sélectionne des images à recommander en excluant celles déjà présentes dans la liste des images fournies lors de la construction de l’utilisateur. Il choisit ensuite jusqu'à 3 images au hasard parmi les images restantes du cluster.

L'utilisation de bibliothèques comme **Pandas** et **Scikit-learn** facilite la gestion des données et l'implémentation de ces modèles, en permettant de manipuler facilement des **DataFrames** pour organiser les données et d’utiliser des algorithmes de machine learning pour effectuer les recommandations.

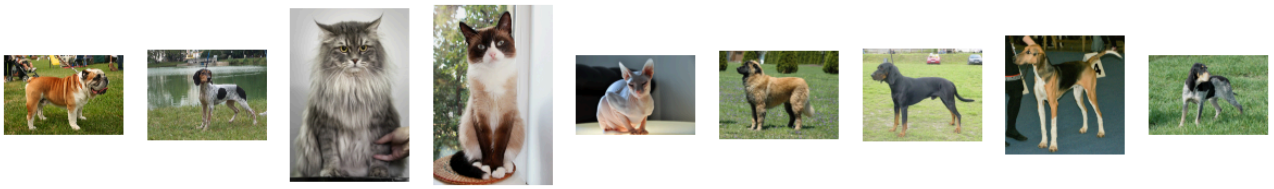
7. Tests

Nous avons identifié les types de tests pour évaluer la performance et la fiabilité du système :

Tests fonctionnels : Ces tests visent à vérifier le bon fonctionnement de chaque module du système, dont le **test de précision** qui calcule de la proportion de bonnes prédictions sur le total et la **matrice de confusion** qui analyse des faux positifs et faux négatifs. Néanmoins, après plusieurs essais infructueux,

nous n”avons pas poursuivi cette approche, bien qu’elle aurait pu permettre une meilleure caractérisation des performances du système.

Tests utilisateurs : Ces tests ont été réalisés avec succès. Pour notre utilisateur test qui a des préférences pour les chien en nature. Le système a ensuite généré des recommandations d’images en fonction de ces choix. On observe que parmi les 9 images recommandées, 6 représentent des chiens et 7 sont en milieu naturel. Ces résultats indiquent que le système prend bien en compte les étiquettes liées aux préférences de l'utilisateur et propose des recommandations pertinentes.



8. Auto-évaluation et limites

Auto-évaluation : Le système est fonctionnel, mais présente des limites dues à la nature des données utilisées. Par exemple, certaines catégories trop **génériques**, comme “l'orientation” et la “qualité de l'image”, représentent un poids important dans l’analyse des critères et peuvent influencer de manière excessive les recommandations.

Limites : En raison des outils limités à notre disposition pour l'**analyse et le traitement des images**, il est difficile d'effectuer une classification précise. L'enrichissement des caractéristiques visuelles influençant la préférence d’une photo à une autre, améliorerait les résultats. A l’aide de tags plus détaillés et pertinents comme : "âge (chaton, chiot)", “longueur du pelage”, “forme des oreilles”, “attitude de l’animal”). De plus, un approfondissement des caractéristiques aurait permis de faire des associations entre les différents tags.

Améliorations possibles : Pour offrir des recommandations plus adaptées, il serait utile d'intégrer un système de **filtrage collaboratif**, qui repose sur l’analyse des comportements d’utilisateurs ayant des préférences similaires. Afin de réaliser ce système, un **clustering préalable des utilisateurs** pourrait être mis en place. Ce processus permettant ainsi d'affiner davantage les recommandations et d'optimiser l'expérience utilisateur.

Conclusion

Le projet a permis de développer un système de recommandation d'images basé sur les préférences des utilisateurs, en utilisant des techniques de **data mining** et d'**apprentissage automatique**. Ce système est capable de recommander des images selon les caractéristiques, telles que l'animal, la végétation, la luminosité, l'orientation et la qualité. Les tests ont confirmé la fonctionnalité du système. Bien que les résultats soient encourageants, des améliorations sont possibles pour rendre le système plus précis et performant.

Ce projet a constitué une première approche enrichissante pour la compréhension des concepts fondamentaux de l'analyse de données. Il a permis de se familiariser avec l'utilisation d'algorithmes d'apprentissage automatique et de techniques d'exploration de données, tout en posant les bases pour aborder des projets plus complexes dans le domaine de la **data science**.