Projet TER par l'équipe ARS

TER ARTIST-RUN-SPACES

KARAMI Aya

JEBALI Anas

GUYON NOAH

VITOFFODJI Adjimon

EL OUALYDY Mohamed-Amine

Février, 2025

1. Choix des outils pour optimiser l'efficacité du travail

Nous avons opté pour des outils libres, efficaces et multi plateformes afin d'optimiser le travail en équipe. Ces choix tiennent compte de la diversité des systèmes d'exploitation utilisés (Windows et MacOS), et visent à garantir une compatibilité maximale entre les postes de travail des membres du groupe.

Outils collaboratifs et leur usage

- ***GitHub** : utilisé pour le versionnement du code, le partage de documents, le suivi des tâches via les issues, et la collaboration à travers les branches et pull requests. Il centralise l'ensemble des ressources du projet.*

- ***Discord** : outil de communication principal, permettant des échanges rapides, l'organisation de réunions, et le partage informel d'idées, de fichiers ou de liens. Il favorise la réactivité et la coordination.*

Langages de programmation et d'analyse

- ***Python** : langage principal du projet, choisi pour sa simplicité, sa large communauté, et ses bibliothèques puissantes en traitement de texte et visualisation de données (Pandas, Scikit-learn, NLTK, SpaCy, BERTopic, etc.).*

- ***SQL (SQLite)** : utilisé pour structurer et interroger efficacement notre base de données. SQLite est une solution légère et adaptée à un projet de cette envergure, avec peu de relations complexes.*

Stockage et accessibilité des données

- ***SQLite** : base de données locale, facile à intégrer à nos scripts Python, offrant un accès rapide et structuré aux données des artist-run spaces.*

- ***GitHub** : point centralisé de stockage pour les scripts, les notebooks, les jeux de données nettoyés et les versions successives du rapport. Il garantit l'accessibilité, le suivi et le partage fluide des ressources du projet.*

Reproductibilité du projet

Nous avons adopté une approche rigoureuse de reproductibilité :

- *Scripts Python organisés et commentés, facilitant la lecture et la réutilisation.*

- *Données nettoyées accompagnées de métadonnées décrivant les transformations effectuées.*

- *Fichier environment.yml : permet de recréer l'environnement de travail à l'identique avec Conda, incluant toutes les dépendances nécessaires (bibliothèques de NLP, visualisation, clustering, etc.).*

- *Suivi de version via Git, avec une arborescence claire, des messages de commit explicites, et un README pour la prise en main du projet.*

Suivi des modifications

- *Git assure la traçabilité complète du projet.*

- *Chaque membre travaille sur sa propre branche avant de fusionner via une pull request, permettant un contrôle qualité et une validation collective.*

- *L'historique des modifications permet de revenir à toute version antérieure si nécessaire.*

Productivité et organisation

- *Chaque membre pousse régulièrement ses avancées sur GitHub.*

- *Nous utilisons Discord pour faire le point sur les tâches en cours, discuter des idées ou résoudre des problèmes techniques (ex : intégration de différentes parties du code).*

- *Une réunion interne est organisée au minimum une fois par semaine pour coordonner nos travaux.*

- *Nous rencontrons notre tuteur environ toutes les deux semaines, selon ses disponibilités. Ces échanges durent entre 1h et 2h30.*

- *En période d'alternance, nous travaillons majoritairement de manière autonome, tout en maintenant une communication continue au sein du groupe.*

- *Notre cohésion d'équipe est solide, et nous n'avons rencontré aucun problème de communication majeur depuis le début du projet.*

2. Modélisation et évaluation

Critères d'évaluation des modèles

*Les critères principaux pour évaluer le modèle **BERTopic** sont la diversité des topics et la couverture des topics.*

- ***Diversité des topics** : La diversité est mesurée en examinant la proportion de mots uniques dans l'ensemble des topics générés. L'objectif est de maximiser cette diversité pour éviter que le modèle ne génère des topics redondants.*

- ***Couverture des topics** : Cela indique combien de documents sont effectivement associés à chaque topic. Une couverture trop faible suggère qu'un ou plusieurs topics ne sont pas assez représentés dans les données analysées.*

Présentation des modèles choisis

*Le modèle **BERTopic** repose sur des techniques de topic modeling combinant des embeddings de phrases et des algorithmes de clustering pour identifier des sujets sous-jacents dans un corpus de textes. La méthode peut être exprimée par la relation suivante :*

$$d_i = \text{Topic}(t_j), \quad \text{pour } j = 1 \dots N$$

*où di est le document i et tj représente le topic j, avec N le nombre total de topics générés. Le modèle utilise des embeddings pré-entrainés, tels que ceux produits par **Sentence-BERT**, pour représenter chaque document dans un espace vectoriel, suivi de l'application de **HDBSCAN** pour le clustering des documents en topics distincts.*

Conditions (pré-supposés) d'application des méthodes

*Le modèle suppose que les documents peuvent être représentés sous forme de vecteurs denses (embeddings), ce qui nécessite l'utilisation de modèles comme **Sentence-BERT** ou des alternatives. Il repose également sur la présupposition que les données peuvent être efficacement groupées en topics à l'aide de **HDBSCAN**, qui nécessite des données bien séparées et bien distribuées. En outre, un pré-traitement adéquat des textes est essentiel, notamment la suppression des mots vides et la réduction des données bruitées.*

Vérification des pré-supposés

Pour vérifier que les présupposés sont respectés, il est nécessaire de :

- *Analyser la distribution des topics pour voir si les documents sont correctement répartis parmi différents sujets.*

- *Évaluer la qualité des embeddings pour s'assurer que les représentations vectorielles des textes capturent correctement les relations sémantiques.*

Commencer avec un modèle simple

*Nous avons choisi de commencer avec **BERTopic**, qui est un modèle relativement simple, car il repose sur des algorithmes éprouvés de **clustering** et des **embeddings** bien établis. Cette approche est adaptée aux données textuelles et offre une solution performante sans entrer dans des modèles plus complexes comme les réseaux neuronaux profonds.*

Justification du choix des modèles et techniques utilisés

*Le choix de **BERTopic** a été motivé par son efficacité prouvée dans le **topic modeling** avec des textes non étiquetés. L'utilisation d'un modèle de représentation comme **Sentence-BERT** permet de capturer les nuances sémantiques, et l'algorithme **HDBSCAN** de clustering assure une séparation robuste des topics sans avoir besoin de spécifier un nombre de topics à l'avance.*

Fixation des valeurs des hyper-paramètres

*Les hyper-paramètres importants dans **BERTopic** incluent le nombre de topics et les paramètres du clustering (comme **min_cluster_size**), qui contrôlent la taille des groupes formés. Nous avons ajusté ces paramètres en fonction de la qualité des résultats obtenus et de l'analyse de la distribution des topics. La valeur de **top_n_words** a également été fixée à 50 pour limiter le nombre de mots-clés par topic.*

Incorporation de la connaissance métier

*Nous avons intégré une **connaissance métier** dans la modélisation en sélectionnant des **topics initiaux** qui sont spécifiquement liés à l'art et aux espaces d'art (par exemple, "exposition", "atelier", "performance"). Cela permet d'orienter le modèle vers des sujets plus pertinents et d'améliorer la qualité des topics extraits.*

Privilégier l'interprétabilité sur les boîtes noires

*Bien que **BERTopic** repose sur des techniques de clustering qui peuvent être considérées comme une "boîte noire", il reste plus **interprétable** que des modèles plus complexes comme les réseaux neuronaux profonds. De plus, des outils de visualisation comme **t-SNE** ou **UMAP** permettent de rendre les résultats plus compréhensibles et accessibles.*

Inférence et intervalles de confiance

*Actuellement, **BERTopic** ne fournit pas directement des **intervalles de confiance** pour les topics. Cependant, une approche future pourrait inclure l'utilisation de techniques comme le **bootstrap** ou des tests de permutation pour évaluer la stabilité des topics identifiés.*

Vérification de la convergence des algorithmes

*Le modèle repose sur **HDBSCAN** pour le clustering, qui est bien adapté aux données denses. Une bonne **convergence** peut être vérifiée en examinant la distribution des topics ou en ajustant les paramètres du clustering pour obtenir des groupes de taille appropriée.*

Appropriateness de la technique pour la tâche envisagée

*Le modèle **BERTopic** est particulièrement adapté à la tâche envisagée, car il permet d'extraire des **topics** sous-jacents à partir de textes en grande quantité et non étiquetés, ce qui répond parfaitement aux objectifs du projet, qui consiste à analyser les espaces d'art à travers leurs caractéristiques textuelles.*

Réponse à la question de recherche

*Les **modélisations réalisées** avec **BERTopic** permettent de répondre efficacement à la question de recherche en fournissant une vue détaillée des **thèmes principaux** abordés dans les textes collectés sur les artist-run spaces, offrant ainsi une base solide pour l'analyse et la visualisation des différents sujets dans ces espaces.*