Rapport de Projet : Travel Order Resolver

1. Introduction

Ce rapport présente le projet **Travel Order Resolver**, dont l'objectif est de développer un système capable de traiter des commandes de voyage en langage naturel pour générer des itinéraires optimaux. Ce projet inclut des modules pour la reconnaissance vocale, le traitement du langage naturel (NLP), la classification de texte, et la recherche du chemin optimal.

---

2. Objectifs

1. Extraire les lieux de départ et d'arrivée à partir de commandes en langage naturel.
2. Filtrer les phrases non pertinentes.
3. Calculer le chemin le plus court entre deux gares.
4. Intégrer la reconnaissance vocale pour permettre l'entrée audio.
5. Fournir une interface utilisateur pour tester le système.

---

3. Méthodologie

3.1 Reconnaissance des entités nommées (NER)

- **Objectif :** Identifier les gares de départ et d'arrivée dans une phrase.
- **Approche :**
    - Entraînement de **BERT** sur des données générées aléatoirement.
    - Amélioration des résultats avec **CamemBERT**, adapté pour le français.
- **Résultats attendus :** Amélioration de la précision grâce à la spécialisation linguistique.

3.2 Classification des textes

- **Objectif :** Filtrer les phrases parasites.
- **Approche :**
    - Utilisation de **CamemBERT** pour entraîner un modèle de classification sur des données générées.
- **Résultats attendus :** Réduction des faux positifs et négatifs.

3.3 Recherche du chemin optimal

- **Objectif :** Calculer le chemin le plus court entre deux gares.
- **Algorithmes utilisés :**
    - **A*** et **Dijkstra**.
- **Données :**
    - Graphe initial basé sur le dataset fourni (timetables.csv).
    - Amélioration par un graphe personnalisé avec des gares supplémentaires et des temps de trajet estimés.

3.4 Reconnaissance vocale

- **Objectif :** Permettre des commandes vocales.
- **Outils :**
    - **VOSK** pour une première implémentation.
    - Passage à **Whisper** pour des résultats plus précis.
- **Résultats attendus :** Transcriptions robustes et compatibles avec le module NLP.

3.5 Interface utilisateur

- **Technologie :** Développement avec **Streamlit**.
- **Fonctionnalités supplémentaires :**
    - Importation de fichiers texte contenant plusieurs phrases.
    - Remplacement optionnel du NER par un modèle LLM (Mixtral).

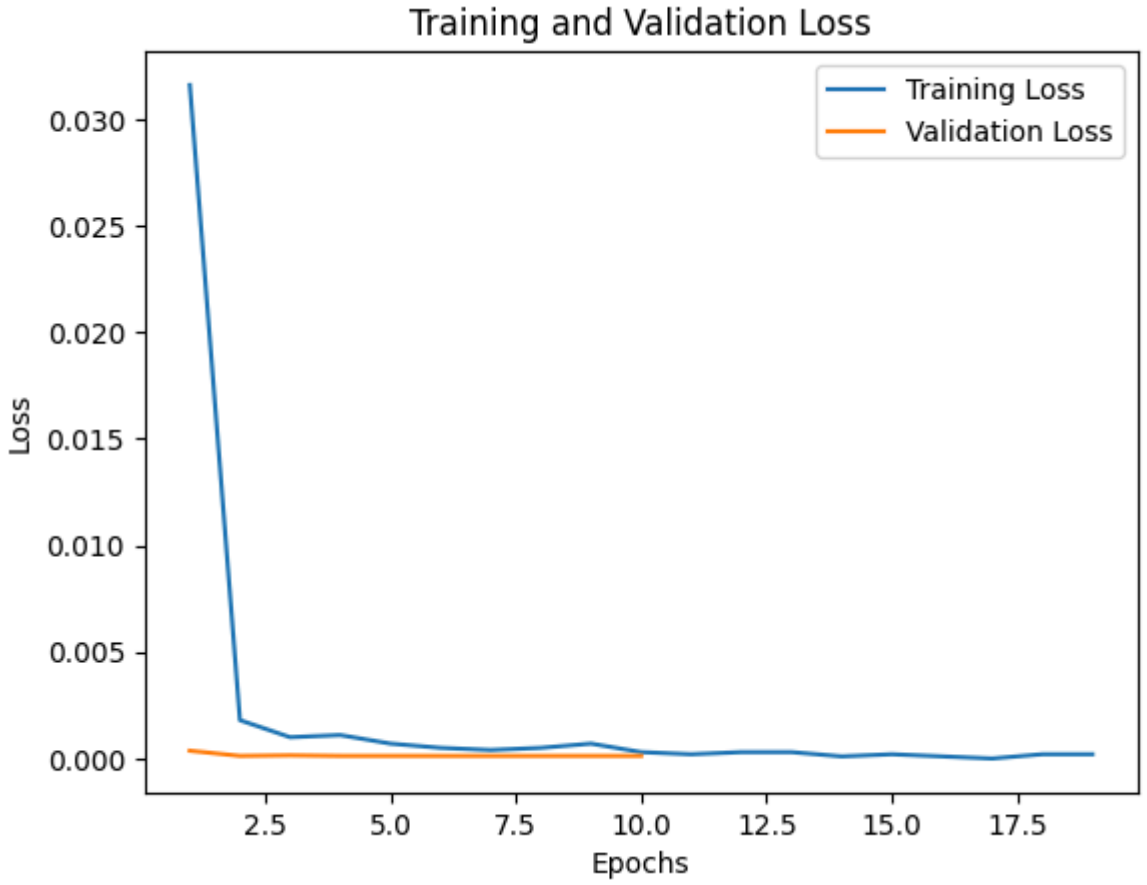
---

4. Résultats

4.1 NER

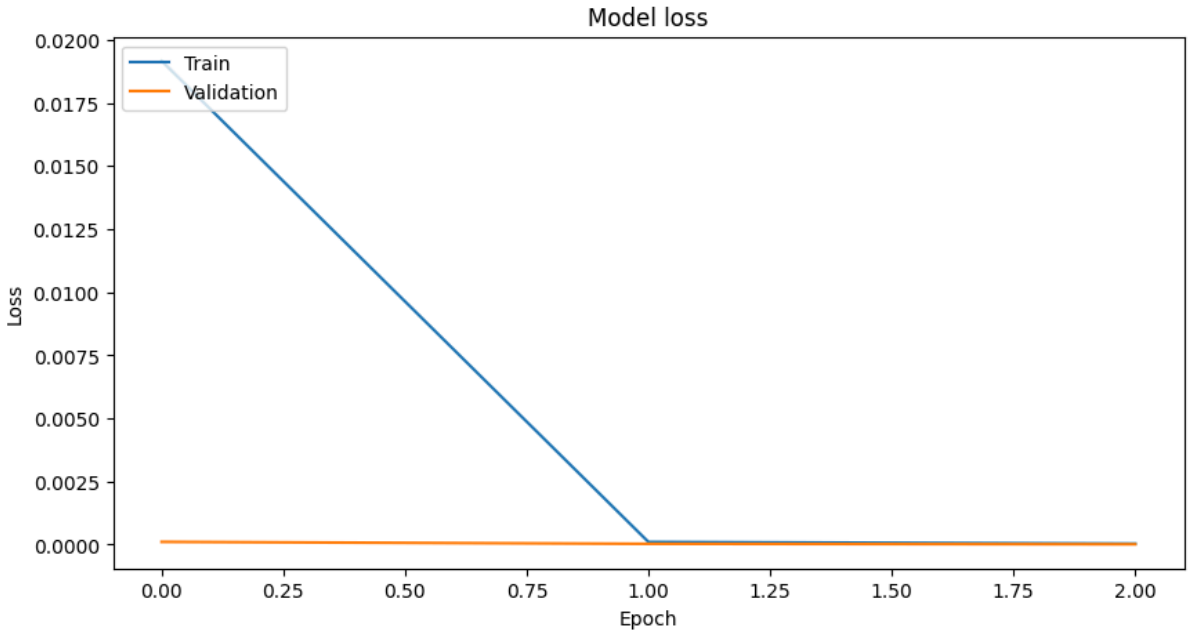
- Précision obtenue avec BERT : 98%. (fscore : 98.07)

- Amélioration avec CamemBERT : 99%. (fscore : 99.89)



4.2 Classification

- Taux de classification correct : 99%.



4.3 Pathfinder

- Temps de calcul moyen pour le chemin optimal : 1s.
- Comparaison entre A* et Dijkstra : 0.05s (différence négligeable)

4.4 Reconnaissance vocale (Française)

- Taux de précision avec VOSK : 95%.
- Taux de précision avec Whisper : 99%.

---

5. Discussion

5.1 Problèmes rencontrés

- Difficultés dans l'entraînement des modèles sur des données limitées.
- Défis liés à l'intégration des différents modules.
- Manque de données pour avoir tous les trajets de train possible

5.2 Solutions apportées

- Génération de données synthétiques pour l'entraînement.
- Optimisation de l'infrastructure backend pour la gestion des requêtes.
- Création d’un datasets a l’aide d’autre datasets (résultat plus concluant mais quelques trajets restes impossible)

---

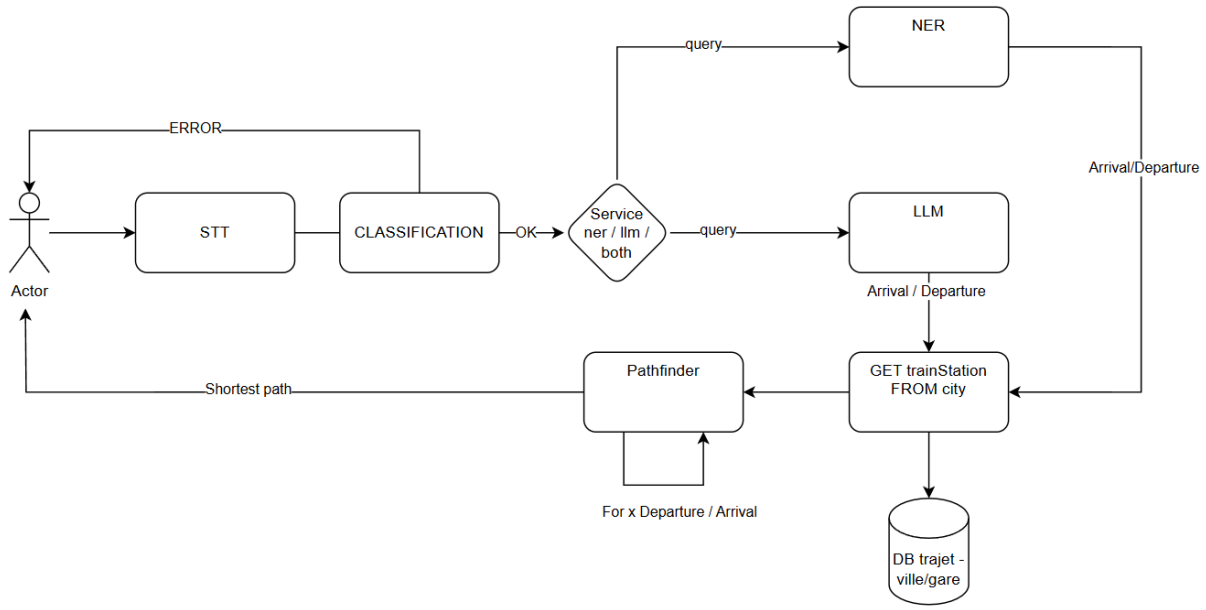
6. Conclusion et Perspectives

- Synthèse des résultats.
- Pistes d'amélioration : support pour d'autres langues, prise en compte des horaires de train.

---

7. Annexes

7.1 Architecture du projet



7.2 Architecture du Backend

- Alembic: Database migrations.
- Docker: Docker configuration files.
- Infrastructure: Infrastructure-related code.
- Main.py: Entry point for the backend application.
- Model: Data models.
- Repositories: Data access layer.
- Services: Business logic layer.

7.3 Datasets

Quelques exemples de datasets utilisé

NER

1er dataset :

```
token,label,sentence_id,ner_tag

Je,O,0,0

veux,O,0,0
```

2eme dataset :

```
id,Tokens,ner_tags,ner_labels
0,"['Y', 'a-t-il', 'des', 'réductions', 'pour', 'les', 'billets', 'de',
'Orléans', 'à', 'Bezier', '?']","[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 3,
0]","['O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B-START', 'O', 'B-END',
'O']"
1,"['Quels', 'sont', 'les', 'trains', 'les', 'plus', 'rapides', 'de',
'Dijon', 'à', 'Tours', '?']","[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 3,
0]","['O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B-START', 'O', 'B-END',
'O']"
2,"['Je', 'veux', 'aller', 'de', 'Tourcoing', 'à', 'La',
'Rochelle']","[0, 0, 0, 0, 1, 0, 3, 4]","['O', 'O', 'O', 'O',
'B-START', 'O', 'B-END', 'I-END']"
```

CLASSIFICATION

```
question,label
J'ai besoin d'un vol de Angoulême à Issy-les-Moulineaux.,1
Le musée est fermé aujourd'hui,0
```

PATHFINDER / GRAPH

1er dataset :

```
trip_id,Departure,Arrival,Duration
OCESN003100F140147152,Gare de Le Havre,Gare de Paris-St-Lazare,138
OCESN003190F040047309,Gare de Dieppe,Gare de Paris-St-Lazare,145
```

2eme dataset :

```
ID,Departure,Arrival,Duration,Departure_City,Arrival_City
dbc9373f,La Douzillère,Loches,12,JOUE-LES-TOURS,LOCHES
6d400d6c,La Douzillère,Chambourg,10,JOUE-LES-TOURS,CHAMBOURG-SUR-INDRE
```