Projet Machine Learning

HAI817 - 2022/2023

Détection automatique des fake news à partir de données textuelles
(Fake News Detection)

Projet en groupe (4 à 5 étudiants)

Ce projet s'inscrit dans le contexte de l'apprentissage supervisé, i.e. les données possèdent des labels. Il vise à trouver les modèles les plus performants pour prédire si des articles de presse sont vrais ou faux. Les articles contiennent des assertions (une assertion est une proposition que l'on avance et que l'on soutient comme vraie) faites, par exemple, par des hommes politiques.

1. **Les Données**

Le jeu de données utilisé provient de *CLEF2022* (Conference and Labs of the Evaluation Forum), une conférence internationale visant à évaluer des systèmes de recherche d'information multilingue et multimodale. Il est accessible sur Moodle dans la section "Projet". Il contient des articles de presse collectés à partir de sites de fact-checking (tels que www.politifact.org ou www.snopes.org). Le contenu exact du jeu de données est décrit en détails ici: https://zenodo.org/record/6555293#.Y-Pkoy8w3rE et dans l'article suivant (facilement trouvable sur Google) :

Köhler, J., Shahi, G. K., Struß, J. M., Wiegand, M., Siegel, M., Mandl, T., & Schütz, M. (2022). Overview of the CLEF-2022 CheckThat! lab task 3 on fake news detection. Working Notes of CLEF.

Attention : il est important de lire attentivement la description du jeu de données afin de bien comprendre à quoi correspondent les différents attributs.

2. **Ingénierie des données**

Le jeu de données contient, pour chaque article de presse, le **titre** ainsi que **le texte intégral de l'article.**

Pour préparer des données textuelles, il existe de nombreux pré-traitements (élimination des stop words, lemmatisation, n-grammes, etc.) vus en cours et disponibles dans les notebooks (e.g. ingénierie des données textuelles).

Après le pré-traitement des données vient l'étape de l'ingénierie des données (*feature engineering*). Comme il s'agit de données textuelles, une étape indispensable sera de choisir une méthode de représentation du texte *(e.g. bag of words, tf-idf)* de manière à ce qu'il soit compréhensible et utile pour un modèle de classification.

Les **notebooks** sont là pour vous aider. N'hésitez pas à les consulter.

Pour aller plus loin:

- **Modélisation par sujets (topic modelling):** Appliquer des algorithmes de modélisation par sujets puis analyser les données à travers ces sujets. *Exemple: Y a-t-il des sujets pour lesquels les fake news sont plus nombreuses?.* Une fois les sujets modélisés, évaluez l'utilité des sujets

modélisés en tant que features en entrée des modèles de classification (cf section 3)

- **Reconnaissance d'entités (entity recognition):** Appliquer des méthodes de reconnaissance d'entités et intégrer les entités extraites aux features fournies en entrée aux modèles de classification. Analyser l'impact des entités dans la tâche de classification. *Exemple: Les articles ont-ils plus tendance à être faux lorsqu'ils mentionnent des entités? Les entités en tant que features sont-elles utiles aux modèles de classification?* Une fois les entités extraites, évaluez leur utilité en tant que features en entrée des modèles de classification (cf section 3)

3. Les tâches de classification

Une fois que vous aurez conçu vos features à partir des données, l'étape suivante consistera en le choix d'un classifieur adapté. N'oubliez pas que comme vous ne connaissez pas les données il est indispensable de tester plusieurs classifieurs pour voir celui ou ceux qui ont de meilleures performances (e.g. notebooks premières classifications, classification de données textuelles) pour au final définir une chaîne de traitement complète adaptée à vos données.

Nous nous intéressons à trois tâches de classification :

1. {VRAI} vs. {FAUX} (deux classes)
2. {VRAI ou FAUX} vs. {AUTRE} (deux classes)
3. {VRAI} vs. {FAUX} vs. {MIXTE} vs. {AUTRE} (quatre classes)

Dans les trois cas, il faudra classer les assertions en groupes selon les labels. Pensez bien à vérifier que les instances sont labellisées selon les catégories indiquées et éventuellement apportez les modifications nécessaires.

Attention, selon la tâche de classification choisie, vos données d'apprentissage risquent de ne pas être équilibrées, i.e. il peut y en avoir plus dans une classe que dans l'autre. Quelle solution proposeriez-vous ? Idée : Pensez à l'*upsampling* et/ou au *downsampling*.

Vous pouvez utiliser les **modèles de classification** vus en cours, tels que les arbres de décision, les SVMs, le Naïve Bayes, les K-NN, les random forests, etc. Ne vous censurez pas, vous pouvez utiliser d'autres approches de classification (par exemple, les réseaux de neurones), si vous le souhaitez.

N'oubliez pas de bien évaluer vos modèles. L'accuracy n'est pas suffisante. Pensez à la matrice de confusion, au rappel, à la précision, à la F-mesure.

Pour aller plus loin:

Sélection de variable (feature selection): Pour chacune des trois tâches de classification, en plus de vos modèles de classification, préparez une liste de features discriminantes en ordre décroissant. Pour cela, vous pouvez vous appuyer sur des méthodes de sélection de variables (ou de features). Le plus important est de tirer les conclusions. Qu'en concluez-vous en comparant les listes obtenues pour les deux tâches ?

4. **Analyse des erreurs, validation et comparaisons des modèles**

La partie ‘analyse’ du projet consiste à comparer empiriquement les différents choix que vous avez pu faire dans la partie sélection des features, des prétraitements, des modèles utilisés, de l'échantillonnage, etc. par rapport à leur impact sur la qualité de la classification.

Cette analyse devra être présentée de manière synthétique et lisible à l’aide d’un tableau comparatif et/ou des courbes. Il est important d’essayer de “comprendre" les raisons des résultats obtenus en fonction des choix effectués (par exemple : Pourquoi ce modèle se comporte mieux ou moins bien qu’un autre ? Pourquoi la suppression des stop words améliore ou au contraire n’améliore pas les résultats ? etc). Cette prise de recul sera particulièrement prise en compte lors de l’évaluation.

5. **Organisation et rendu**
    - Le travail s'effectuera en groupes de **4 à 5 étudiants**.
    - Le rendu final sera soumis sous la forme d’un fichier compressé (gzip) identifié par **les noms des membres** du groupe **à déposer sur Moodle avant la fin du module** et consiste en :
      (1) Un rapport de **8 pages maximum**
      (2) Le notebook en pdf et ipynb de vos codes de l'ensemble des traitements automatiques
    - Attention à bien mettre le prénom, nom et numéro d’étudiant de chaque personne du groupe dans les documents rendus
    - Tout devoir ne respectant pas les consignes ne sera pas évalué