

Détection de Fausses Nouvelles par Traitement du Langage Naturel (NLP)

Réalisée par : Boulidam Abdellah

Encadrée par : Sara Baghdadi

May 20, 2025



- ▶ Contexte
- ▶ introduction au NLP
- ▶ Modèles Statistiques de NLP
- ▶ Données
- ▶ Prétraitement des Données
- ▶ Extraction des Features NLP
- ▶ Entraînement des Modèles
- ▶ Comparaison des Modèles
- ▶ conclusion

Pourquoi détecter les Fake News ?

- Les fausses informations se propagent six fois plus vite que les vraies sur Twitter.
- Elles menacent la démocratie, la confiance dans les institutions et la stabilité sociale.
- Elles peuvent entraîner de graves conséquences : violences, manipulations électorales, crises politiques.



Que propose ce projet ?

- Réaliser une étude comparative entre plusieurs approches pour la détection des Fake News :
 - Prétraitement du texte, incluant le nettoyage et la normalisation des données textuelles.
 - Extraction des features en utilisant des modèles NLP (TF-IDF, Doc2Vec).
 - Comparaison entre plusieurs modèles de classification :
 - Modèle SVM (Support Vector Machine).
 - Modèle de Régression Logistique.
 - Modèle Dense de Deep Learning.
- Mettre en place les trois modèles et les évaluer sur des métriques de performance appropriées (précision, rappel, F1-score, etc.).
- Comparer les résultats obtenus pour déterminer l'approche la plus performante et en analyser les avantages et les inconvénients.

Qu'est-ce que le NLP (Natural Language Processing) ?

Le NLP pour Natural Language Processing ou Traitement du Langage Naturel est une discipline qui porte essentiellement sur la compréhension, la manipulation et la génération du langage naturel par les machines. Ainsi, le NLP est réellement à l'interface entre la science informatique et la linguistique. Il porte donc sur la capacité de la machine à interagir directement avec l'humain.

Les modèles statistiques de NLP reposent sur des techniques basées sur des calculs de fréquence et des représentations vectorielles simples. Les plus populaires sont les suivants :

TF-IDF (Term Frequency-Inverse Document Frequency)



TF-IDF est une méthode de pondération qui mesure l'importance d'un mot dans un document par rapport à l'ensemble des documents.

$$TF\text{-}IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

TF (Term Frequency) :

$$TF(t, d) = \frac{f_{t,d}}{\sum_k f_{k,d}} \quad (2)$$

IDF (Inverse Document Frequency) :

$$IDF(t) = \log \frac{N}{n_t} \quad (3)$$

avec :

- $f_{t,d}$: fréquence du terme t dans le document d .
- N : nombre total de documents.

- **Word2Vec** est un algorithme de *word embedding* développé par Google.
- Il transforme les mots en **vecteurs denses continus** capturant leur signification sémantique.
- Deux architectures : **CBOW** (Continuous Bag of Words) et **Skip-Gram**.

Pourquoi utilisé en NLP ?

- Encode la **proximité sémantique** : mots similaires ont des vecteurs proches.
- Réduit la dimensionnalité par rapport aux représentations *one-hot*.
- Permet des opérations vectorielles : `roi - homme + femme` `reine`.

Puissance :

- Capture les **relations syntaxiques et sémantiques** implicites.
- Utilisé comme base dans plusieurs modèles avancés (e.g. LSTM, transformers).

Source des données :

- Corpus utilisé : **Fake and Real News Dataset** (Kaggle)
- Taille totale : environ 45 000 articles (mélange de fake et vrai)
- Chaque article contient :
 - Titre
 - Texte complet
 - Sujet (topic)
 - Date

Répartition :

- Environ 50% de fausses nouvelles, 50% de vraies nouvelles
- Corpus équilibré pour l'entraînement du modèle

- Conversion en minuscules
- Suppression de la ponctuation et des caractères spéciaux
- Suppression des stopwords (mots outils)
- Lemmatization (normalisation des mots)

Exemple :

- Texte brut : “The government isn’t telling the full truth!”
- Après prétraitement : “government tell full truth”

TF-IDF Vectorization

- Utilisation de `TfidfVectorizer(ngram_range=(1,2), min_df=50)`
- Prise en compte des **unigrammes** et **bigrammes** pour capturer plus de contexte lexical.
- Suppression des termes rares pour limiter le bruit et la dimensionnalité.
- Transformation des titres et textes en vecteurs numériques.

Similarité Cosinus TF-IDF

- Calcul de la similarité cosinus entre chaque paire **titre-texte** (même article).
- Objectif : mesurer la cohérence lexicale entre le titre et son contenu.
- Ce score est ajouté comme feature supplémentaire.

Doc2Vec Embedding

- Entraînement d'un modèle Doc2Vec sur l'ensemble des titres et textes.
- Extraction de représentations denses (embeddings) pour chaque titre et chaque texte.
- Calcul de la **similarité cosinus** entre ces deux vecteurs pour chaque article.

Fusion des Features

- Pour chaque article, construction d'un vecteur final :
 - TF-IDF du titre
 - TF-IDF du texte
 - CosSim (TF-IDF)
 - CosSim (Doc2Vec)
- Cette combinaison enrichit la représentation, en tenant compte à la fois du lexique et de la sémantique.

Objectif de cette phase



- Après l'extraction des features, l'étape suivante consiste à entraîner des modèles de classification pour détecter les fake news.
- Nous avons comparé des approches classiques et une approche par Deep Learning.

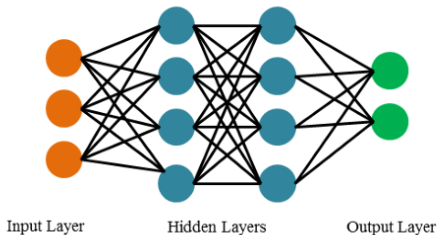
- **Régression Logistique** : modèle non linéaire simple pour la classification binaire.
- **SVM** : algorithme de classification qui maximise la marge entre les classes pour une meilleure généralisation.
- **Multi-Layer Perceptron (MLP)** : modèle de Deep Learning avec plusieurs couches cachées.

But : Comparer leurs performances sur les mêmes données vectorisées.

Architecture du Modèle Deep Learning (MLP)

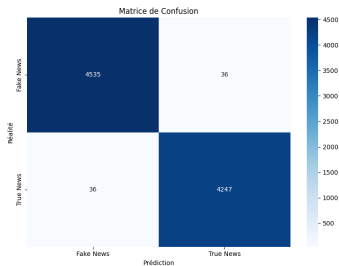
- **Entrée** : vecteur concaténé (TF-IDF + SimCos titre-texte + Doc2Vec)
- **Couches cachées** :
 - Dense (1024), ReLU, régularisation L2
 - Dense (512), ReLU
 - Dense (128), ReLU
 - Dense (64), ReLU
- **Dropout** (0.1) appliqué après chaque couche cachée
- **Sortie** : Dense (2), activation softmax

Optimiseur : Adam **Fonction de perte** :
categorical_crossentropy

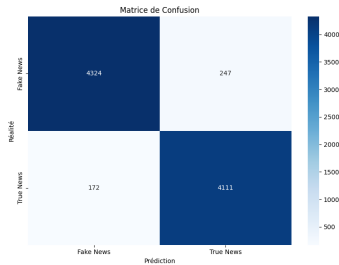


- Évaluation sur un même jeu de test
- Métriques utilisées : Accuracy, Precision (weighted avg), Recall (weighted avg), F1-score (weighted avg)

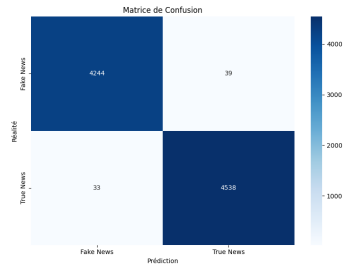
Modèle	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.99	0.99	0.99	0.99
SVM	0.95	0.95	0.95	0.95
Deep Learning	0.99	0.99	0.99	0.99



regression lineaire



SVM



DNN

Ce projet de détection de fake news par NLP démontre l'efficacité des techniques de traitement automatique du langage face à un enjeu sociétal majeur. En combinant prétraitement textuel, vectorisation (TF-IDF/Doc2Vec) et modèles de classification (Régression Logistique, SVM et Deep Learning), nous avons obtenu d'excellents résultats . Cette approche ouvre la voie à des applications concrètes pour lutter contre la désinformation qui menace notre démocratie.

Merci pour votre attention !

FAKE NEWS

Questions ?