Université D'Alger I Benyoucef Benkhedda

Faculté des Sciences-Département Mathématiques et Informatique

Master : Analyse et Sciences de Données

(ASD)

Matière :

Machine Learning (ML)

Travaux de Laboratoire N° : 1

Responsable de la matière
Dr. Khaled Lounnas

Année universitaire 2023/2024

1. Objectif :

L'objectif de ce premier travaux pratique est de mettre en œuvre un modèle de machine learning simple pour classer des emails en deux catégories : spam ou non spam (ham). L'approche utilisée repose sur la bibliothèque scikit-learn en Python, qui fournit des outils puissants pour la construction de modèles de machine learning.

Ce TP offre une introduction pratique à la construction d'un modèle de machine learning pour la classification des textes en utilisant des outils de base de scikit-learn. Il peut être étendu en ajoutant des fonctionnalités plus avancées et en expérimentant avec d'autres modèles selon les besoins spécifiques du projet.

2. Étapes détaillées :

- **Création de la Base de Données :**

  Une petite base de données d'emails est créée, comprenant des exemples de spam et de non spam (ham). Chaque exemple est associé à une étiquette (label) indiquant la catégorie à laquelle il appartient.

- **Séparation des Données :**

Les données sont séparées en deux parties : texte des emails (`texts`) et étiquettes correspondantes (`labels`).

- **Extraction des Caractéristiques avec CountVectorizer :**

- La classe `CountVectorizer()` de scikit-learn est utilisée pour transformer le texte des emails en vecteurs numériques, ce qui est requis par les modèles de machine learning.

Question :

1. Quelle est la différence entre unigramme, bigramme et trigramme dans le contexte de `CountVectorizer` ?
2. Faites varier les valeurs de N-gram de (1,1),(1,2),(1,3) Interpréter les résultats obtenus?
3. Qu'est-ce que le paramètre `stop_words` dans `CountVectorizer`?
4. Remplacer dans la fonction CountVectorizer()l 'analyseur par word ,char,char_wb que remarquer-vous par raport aux résultats obtenues ?

- **Division des données :**

- Les données sont divisées en un ensemble d'entraînement (`X_train`, `y_train`) et un ensemble de test (`X_test`, `y_test`) grâce à la fonction `train_test_split` pour évaluer la performance du modèle.

Question :

1. Expliquez-en quoi consiste cette fonction et son rôle dans le processus d'apprentissage machine.
2. Nommez et expliquez brièvement les paramètres clés que vous pouvez spécifier lors de l'utilisation de cette fonction.
3. Quel est l'objectif de cette division dans le contexte de l'apprentissage machine?
4. Quelles informations sont fournies par les variables de retour de cette fonction, et comment peuvent-elles être utilisées?
5. Pourquoi et comment utiliseriez-vous le paramètre `random_state` lors de la division des données?
6. Expliquez l'utilisation de la valeur de retour de `train_test_split`.
7. Ajuster la taille de l'ensemble de test en utilisant ce paramètre qu'elle son influence sur les performances de modèle test_size={10,20,30,35,40,50}, commenter?

- **Création et Entraînement du Modèle :**

- Un modèle de classification est choisi, en l'occurrence le modèle Naive Bayes multinomial (`MultinomialNB`), qui est adapté pour les données représentées en comptages (comme c'est le cas avec `CountVectorizer`).

- Le modèle est entraîné sur l'ensemble d'entraînement.

Question :

1. En ajustant le paramètre `norm` dans `Tfidfvectorizer`, comment cela affecte-t-il la classification des emails?
2. Testez le modèle sur des options telles que 'l1', 'l2', ou 'max'

3. Comparez les performances du classifieur Naive Bayes multinomial avec d'autres modèles de classification disponibles dans scikit-learn. Quels sont les avantages et les inconvénients relatifs?

- **Prédictions et Évaluation :**

  - Le modèle est utilisé pour faire des prédictions sur l'ensemble de test (`X_test`).

  - La précision du modèle est évaluée en comparant les prédictions avec les étiquettes réelles.

  - La matrice de confusion et le rapport de classification fournissent des détails supplémentaires sur la performance du modèle.

- **Affichage des Résultats :**

  - Les résultats, y compris la précision du modèle, la matrice de confusion et le rapport de classification, sont affichés pour évaluer la capacité du modèle à distinguer entre les emails spam et non spam.

- **Impacte de la taille des données sur la détection des messages**

Question :

1. Après avoir conçu le modèle sur la base de donnée proposée, refaire le même travail en utilisant une autre base des données, commenter ?.

NB :

1. Veuillez ne pas hésiter à explorer les fonctionnalités des fonctions existantes dans le script en saisissant le nom de la fonction ainsi que la bibliothèque où elle est référencée Exemple : train_test_split sklearn

2. Je vous encourage à ajuster les paramètres du modèle ou même à le remplacer entièrement par des approches telles que SVM, KNN, etc.