PALAHIN Dmytro

Reconnaisseur de langue

La pertinence de la recherche

Ce projet est consacré à la recherche et au développement de diverses méthodes et à leur implémentation logicielle pour classer des textes et les comparer à certains modèles. Des fichiers texte en français et en anglais ont été choisis comme tels textes de test.

La construction de tels classificateurs de texte (Anglais : Document classification) appartient à la tâche de la linguistique computationnelle, qui consiste à attribuer un document à l'une de plusieurs catégories en fonction du contenu du document.

Cette direction est pertinente et peut être utilisée dans diverses applications pour déterminer automatiquement la langue du texte et sa corrélation avec l'un des exemples fournis pour un traitement ultérieur, une traduction, une aide contextuelle, etc. Par conséquent, la précision de la construction d'un classificateur de texte est un élément important de la recherche.

Le projet a mené une étude de cinq méthodes différentes qui diffèrent les unes des autres. Ensemble, l'utilisation de toutes les méthodes peut permettre une classification plus précise et de haute qualité des textes de différentes tailles et améliorer le système de classification.

Informations sur le programme

Le programme développé vous permet de charger deux fichiers pour la formation (Français et Anglais texte) et déterminer la langue du fichier de test.

Vous pouvez voir fichier "Code", où il y a un programme qui prend des fichiers comme arguments et lui-même les exécute. Par exemple, c'est ces 3 fichiers pour les arguments "En_text.txt", "Fr_text.txt", "Fr_test.txt" . Voir également pour exécuter fichier avec un nom "!_Compile_Instruction.txt".

Fichier chemin vers lequel est écrit aux *lignes 41 et 43* et déterminer la langue du fichier de test dont le chemin est écrit à la *ligne 45*. Au cours de l'étude, des fichiers de différents volumes (gros, moyen et petit) ont été testés. Dont les noms sont donnés aux *lignes 53 - 57*.

Le programme lors de la lecture des trois fichiers ci-dessus forme et écrit trois fichiers modifiés qui ne contiennent que des lettres minuscules alphabétiques (y compris les lettres françaises spéciales remplacées) et servir au traitement ultérieur du programme. Ces 3 fichiers mentionnés ci-dessus doivent être écrits sous les noms suivants "MODIFIED_En_text", "MODIFIED_Fr_text", "MODIFIED_TEST". Les principaux résultats de la classification de fichier de Test pour *5 méthodes différentes* sont affichés sur l'écran et les résultats intermédiaires dans un fichier créé par notre programme et nommé "RESULTS_INTERMEDIATE_COUNT" et les résultats finaux "RESULTS_FINAL".

1. Analyse fréquentielle des lettres pour la classification des textes

L'analyse fréquentielle des lettres dans les textes est l'une des méthodes générales de classification. Cette méthode est basée sur le fait que la particularité de chaque langue est l'utilisation de lettres de l'alphabet avec une intensité différente (probabilité). Par exemple, en français, les lettres les plus courantes sont : «e, n, r, i, l, ….», pour l'anglais : «e, t, a, n, r, ….». En utilisant cette caractéristique des langages, qui peut être étendue à d'autres langages, il est possible de construire un classifieur approprié. Pour utiliser cette approche, plusieurs étapes ont dû être franchies.

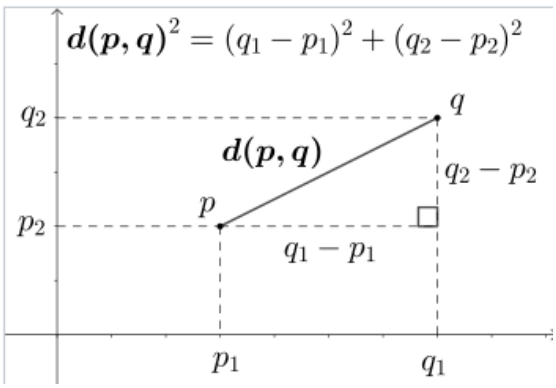
La première étape consiste à lire les fichiers texte correspondants - français, anglais et texte de test, qui seront vérifiés.

La deuxième étape est la préparation des textes pour un traitement ultérieur: la suppression de tous les caractères pouvant être attribués au service (points, virgules, chiffres, et divers autres caractères non alphabétiques, ainsi que la conversion des caractères français spéciaux tels que « é » en la lettre « e » correspondante, etc.).

La troisième étape est la compilation d'un dictionnaire de fréquences des textes correspondants (Français, Anglais et Test), dans lequel chaque lettre de l'alphabet (total 26) correspond à la valeur de probabilité (fréquence) d'occurrence dans les textes.

La quatrième étape consiste à calculer la métrique Euclidienne entre les probabilités d'occurrence de chaque lettre dans « Français et Test texte » et aussi dans « Anglais et Test texte ». La dimension de cette métrique sera égale à la dimension de l'alphabet - 26 :

$$d(p,q) = \sqrt{(p_1-q_1)^2 + (p_2-q_2)^2 + \cdots + (p_n-q_n)^2} = \sqrt{\sum_{k=1}^{n}(p_k-q_k)^2}.$$



La dernière étape de cette méthode est une analyse comparative des deux métriques. Où la métrique (racine carrée de la somme de la différence des carrés) sera plus petit et correspondra très probablement à la langue du texte.

Cette méthode est suffisamment fiable, ce qui correspond aux expériences réalisées. Une caractéristique de cette méthode est la nécessité de collecter les statistiques appropriées des lettres dans les textes. Si elle est insuffisante, cette méthode peut donner une certaine erreur. Des études ont montré que même avec des tailles de fichiers suffisamment petites (seulement environ 50-70 caractères), cette méthode fonctionne très bien. À mesure que la taille des fichiers augmente, la fiabilité des résultats augmente.

2. Méthode du produit des probabilités de lettres dans textes

Cette méthode permet d'analyser des dimensions critiques. Ensembles de test, juste quelques lettres (3, 4, 5, …), ce qui est impossible à faire en utilisant l'analyse fréquentielle.

Le texte est une séquence de lettres. Dans le modèle le plus simple, nous supposons que chaque lettre de cette séquence apparaît indépendamment des précédentes, c'est-à-dire que le texte est considéré comme une chaîne d'événements aléatoires indépendants. En raison de l'indépendance, la probabilité de rencontrer une séquence donnée de lettres dans une langue donnée est égale au produit des probabilités (fréquences) d'occurrence de lettres dans cette langue.

Connaissant les fréquences de lettres pour chacune des deux langues candidates, on peut trouver la probabilité que la phrase entière apparaisse. S'il n'y a pas d'informations a priori sur l'origine de la phrase, les langues candidates sont considérées comme égales. Cela permet de comparer les probabilités d'occurrence d'une phrase dans différentes langues, en les présentant de manière plus familière, en pourcentage.

Un algorithme aussi simple fonctionnait bien même sur un texte court (seulement trois lettres), ce qui peut être utile pour des problèmes de classification particuliers.

3. Utilisation du classificateur « Naive Bayes » pour classer des textes

La prochaine méthode intéressante est le classifieur « Naive Bayes » qui est largement utilisé pour classer les textes afin de déterminer une critique de film (positive ou négative), déterminer le texte du spam dans les e-mails, qui est l'auteur du texte (par exemple, une femme ou un homme), etc.

Ce classificateur est basé sur la formule bien connue de Bayes :

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)}$$

où P est la probabilité, c - classe (dans notre cas, ce sont 2 classes : Français ou Anglais), d est un document (peut être des mots, des lettres, des digrammes, etc.), qui fait l'objet d'une analyse.

Dans notre cas, des lettres ont été sélectionnées pour analyse dans les textes étudiés, pour lesquels les probabilités de leur occurrence pour la première méthode de fréquence ont déjà été trouvées, et les probabilités conditionnelles sont également définies, par exemple P(d|c) – probabilité conditionnelle d'occurrence d'une certaine lettre de l'alphabet (total 26) pour une certaine classe de texte (total 2 : Français ou Anglais).

Ainsi, toutes les probabilités conditionnelles d'apparition de lettres dans le texte du Test ont été calculées en supposant que qu'il peut faire référence à la langue Française ou à la langue Anglaise. Pour la somme des probabilités conditionnelles, là où elle était supérieure, une décision a été prise sur la langue du texte.

4. Analyse fréquentielle de l'apparition des Bigrammes pour la classification de texte

Lors de la classification des textes, un point important est leurs propriétés distinctives. Plus ces propriétés diffèrent, plus le processus de classification est probable. De ce point de vue, il est intéressant d'étudier l'identification des Bigrammes dans les textes. Par exemple, si le mot apparaît “merci”, alors l'ensemble de digrammes sera la combinaison suivante : “me”, “e”, “rc”, “ci” etc. pour d'autres mots. Chaque langue aura sa propre probabilité (fréquence) de ces bigrammes.

Cette méthode est également équipée à plusieurs étapes, dont la lecture des fichiers texte, leur retour préalable, similaire à la méthode №1, compiler des Bigrammes, compiler un dictionnaire de Bigrammes à partir d'un ensemble 26 * 26 = 676, chaque ensemble, la détermination de la fréquence d'occurrence correspondante dans les textes, et en dernière étape, le calcul de la métrique Euclidienne et son analyse.

Cette méthode a également montré d'assez bons résultats sur divers ensembles de données textuelles (grandes et petites).

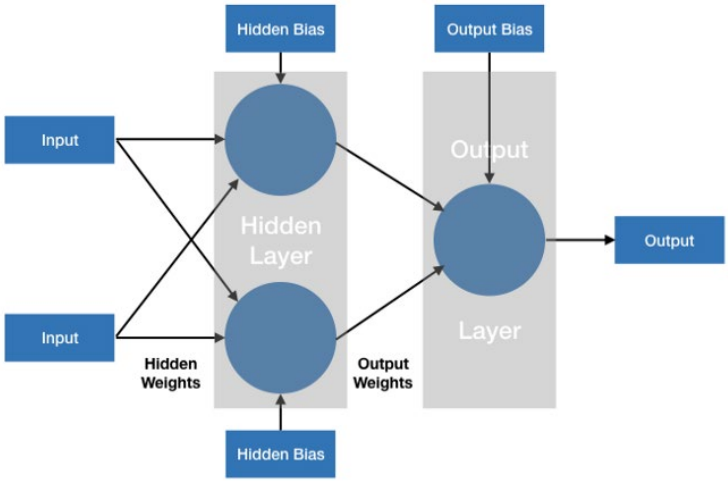
5. Construire un réseau de neurones pour la classification de texte

Les outils modernes de traitement des données informatiques utilisent activement les réseaux de neurones pour le traitement. Dans ce projet, le réseau de neurones le plus simple pour la classification de texte est proposé, qui a montré d'excellents résultats et peut être mis à niveau pour des recherches ultérieures.

Comme des données d'entrée "Input" deux vecteurs de dimension 26 ont été utilisés, le soi-disant "neural network features" et "x traine", qui ont été obtenus dans la première méthode de fréquence (un ensemble de probabilités d'occurrence de lettre pour chacun des textes). Chaque vecteur a une solution connue "y" qui est "0" pour le Français et "1" pour l'Anglais.

En tant que réseau de neurones, il est proposé d'utiliser le modèle le plus simple représenté sur la figure, où il est possible d'ajuster le nombre de couches internes (lors d'études expérimentales, le nombre 15 est le plus efficace).

La tâche de former ce modèle est de déterminer les coefficients de pondération les plus optimaux des connexions entre les nœuds, qui donnent la plus petite erreur entre les résultats connus "y" pour chaque vecteur d'entrée et les données calculées reçues sur le nœud de sortie "Output". Au stade initial, ces coefficients de pondération sont déterminés au hasard entre 0,0 et 1,0.



Les valeurs des nœuds internes sont calculées par la formule :

$$hiddenLayer_j = sigmoid\left(hiddenLayerBias_j + \sum_1^k training\_input_k * hiddenWeight_k^j\right)$$

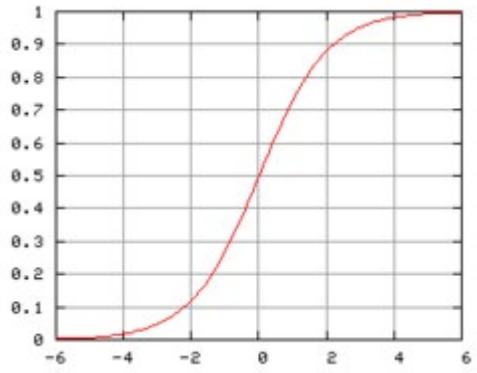
où **j** est le nombre de nœuds internes, et **k** est le nombre de sorties.

La valeur des nœuds de sortie (nous avons 1 nœud) est calculée par la formule :

$$outputLayer_j = sigmoid\left(outputLayerBias_j + \sum_1^k hiddenLayer_k * outputWeight_k^j\right)$$

où **j** est le nombre de nœuds de sortie, et **k** est le nombre de nœuds internes.

Le sigmoïde représenté sur la figure a été utilisé comme fonction d'activation :



Lors du fonctionnement du réseau de neurones, deux étapes de calculs sont envisagées : propagation direct et propagation inverse. La propagation inverse prend en compte l'erreur quadratique moyenne (MSE) entre la valeur de sortie connue "y" et la valeur calculée résultante. La valeur de cette erreur "accorde" les poids du réseau de neurones au résultat attendu le plus correct.

Lors de la formation d'un réseau de neurones, le paramètre du nombre d'étapes de calcul des coefficients de pondération effectuées est utilisé – « EPOCH », ainsi que le coefficient du niveau d'études " learning rate ". Les études expérimentales ont montré les résultats les plus acceptables : EPOCH = 10000, learning rate = 1. Avec de tels paramètres du réseau neuronal, un apprentissage assez rapide se produit avec une probabilité élevée de classification de texte - plus de 90%. Avec une augmentation du nombre d'époques, ce chiffre ne fera qu'augmenter.

Les directions prometteuses dans le développement du projet sont :

- Utilisation de diverses fonctions d'activation ;
- Modification du nombre de nœuds internes ;
- Modification du nombre de couches ;
- L'utilisation d'autres "neural network features", par exemple, le vecteur bigramme obtenu dans l'une des méthodes de ce projet ;
- Visualisation du processus de formation et sélection des paramètres optimaux du réseau neuronal lors de l'utilisation d'un ensemble de données pour la validation.