SUP'COM
INDP2
2020-2021
A. BEN AZZA & S. BEN JEBARA

Travaux dirigés en fondements de l'intelligence artificielle

Filtrage des emails spam/non-spam

De nombreux services de messagerie proposent aujourd'hui des filtres anti-spam capables de classer les emails et de les placer, soit dans la boîte de réception, soit dans la boîte du courrier indésirable selon qu'ils soient 'non-spam' ou 'spam'. Dans cet exercice, un classificateur sera créé pour savoir si un email donné, x, est spam ($y = 1$) ou non-spam ($y = 0$). A cet effet, un email doit être d'abord converti en un vecteur de caractéristiques $x \in R^p$ qui constituent l'entrée du classifieur.

Le document est organisé comme suit. La section 1 explique les pré-traitements à effectuer sur le texte contenu dans l'email et en fournit un exemple avant et après pré-traitement. La section 2 liste le vocabulaire de la langue anglaise considérée tandis que la section 3 explique la méthodologie pour extraire des descripteurs qui seront exploités pour la classification. Finalement, la dernière section dresse la liste des tâches à faire pour répondre aux objectifs de ce travail.

1 Pré-traitement des emails

Avant de commencer la tâche d'apprentissage automatique, il est utile de découvrir les données disponibles. La figure 1 montre un exemple de courrier électronique.

> Anyone knows how much it costs to host a web portal ?
>
Well, it depends on how many visitors youre expecting. This can be
anywhere from less than 10 bucks a month to a couple of $100. You
should checkout http://www.rackspace.com/ or perhaps Amazon EC2 if?
youre running something big.?
To unsubscribe yourself from this mailing list, send an email to:
groupname-unsubscribe@egroups.com

Figure 1: Exemple de contenu d'un email.

On peut noter qu'en plus du texte, cet email contient une URL, une adresse email (à la fin), des chiffres et le symbole dollar des montants. Ces entités spécifiques peuvent exister dans de nombreux emails. Bien que le contenu de chaque entité spécifique soit différent d'un email à un autre, ils seront 'normalisées' afin que toutes les URL soient traitées de la même manière, que tous les nombres soient traités de la même manière, etc. Par exemple, on remplace chaque URL dans l'email par une chaîne unique de caractères 'httpaddr' pour indiquer qu'une URL était présente. Ceci a pour effet de laisser le classificateur de spam prendre une décision de classification en fonction de la présence d'une URL, plutôt que de la présence d'une URL spécifique. Ainsi, cette normalisation améliore généralement les performances d'un classificateur de spam, car les 'spammeurs' randomisent souvent les URL, et donc les chances de voir une URL particulière dans un nouveau morceau de spam devient très petite.

Ci-dessous sont listézs les étapes suivantes de pré-traitement et de normalisation qui sont mises en oeuvre dans cet exercice.

- Minuscules: l'email en entier est converti en minuscules.
- Suppression du HTML: toutes les balises HTML sont supprimées des emails de sorte que seul le contenu reste.
- Normalisation des URL: toutes les URL sont remplacées par le texte 'httpaddr'.
- Normalisation des adresses email: toutes les adresses email sont remplacées par le texte 'emailaddr'.
- Normalisation des nombres: tous les nombres sont remplacés par le texte 'nombre'.
- Normalisation des symboles de monnaies: tous les signes de monnaie sont remplacés par le texte 'dollar'.
- Racine des mots: certains mots sont réduits à leur forme radicale. Par exemple les mots 'discount', 'discounts', 'discounted' et 'discounting' sont tous remplacés par 'discount'. Dans certains cas, on supprime aussi les caractères supplémentaires à la fin du mot. Donc, 'include', 'includes', 'included', et 'including' sont tous remplacés par 'include'.
- Suppression des non-mots: les non-mots et la ponctuation sont supprimés. Tous les espaces blancs (tabulations, sauts de ligne, espaces) sont également coupés et remplacés par un seul caractère d'espace.

Le résultat de ces étapes de pré-traitement sur l'email de la figure précédente est illustré à la nouvelle figure 1. Ce nouveau formulaire des données se révèle être beaucoup plus facile à utiliser pour effectuer l'extraction des descripteurs.

anyon know how much it cost to host a web portal well it depend on how
mani visitor your expect thi can be anywher from less than number buck
a month to a coupl of dollarnumb you should checkout httpaddr or perhap
amazon ecnumb if your run someth big to unsubscrib yourself from thi
mail list send an email to emailaddr

Figure 2: Contenu de l' email de la figure 1 après pré-traitement.

2 Liste du vocabulaire

Après le pré-traitement des emails, la liste de mots existants dans chaque email est disponible. L'étape suivante consiste à choisir les mots qui seront utilisés pour la classification. Pour cet exercice, uniquement les mots les plus fréquemment utilisés sont choisis. Leur ensemble (liste de vocabulaire) est fourni dans le fichier *vocab.txt*. Un extrait est illustré sur la figure 2. Les chiffres de la première colonne représentent les indices de chaque mot du dictionnaire qui, eux, sont fournis dans la colonne 2. Les mots existant ont été sélectionnés en choisissant tous les mots qui apparaissent au moins 100 fois dans le corpus de spam, résultant en une liste de 1899 mots. En pratique, une liste de vocabulaire avec environ 10 000 à 50 000 mots est souvent utilisée.

1	aa
2	ab
3	abil
...	
86	anyon
...	
916	know
...	
1898	zero
1899	zip

Figure 3: Liste des mots du dictionnaire ainsi que leurs indices.

La figure 2 montre le mappage de l'exemple de l'email pré-traité de la figure 1. Il s'agit plus précisément de remplacer chaque mot par son indice du dictionnaire. Par exemple, le premier mot 'anyon' a été remplacé par son indice 86 dans le vocabulaire du dictionnaire.

```
86 916 794 1077 883
370 1699 790 1822
1831 883 431 1171
794 1002 1893 1364
592 1676 238 162 89
688 945 1663 1120
1062 1699 375 1162
479 1893 1510 799
1182 1237 810 1895
1440 1547 181 1699
1758 1896 688 1676
992 961 1477 71 530
1699 531
```

Figure 4: Mappage de l'exemple de l'email pré-traité de la figure 1.

3 Extraction des descripteurs

Le contenu du vecteur des descripteurs retenus pour cette étude est de dimension p (p est la taille du dictionnaire). Chaque position i comprend une valeur booléenne qui devient égale à 1 si le mot du dictionnaire de la position i apparait dans l'email. Dans le cas contraire, la position i prend la valeur 0. Un exemple de vecteur de descripteur est le suivant:

$$x = [0...10...110...1]^T \in R^p.$$

Les étapes précédentes ont été effectuées sur la base de données 'SpamAssassin' [1]. Les vecteurs des descripteurs d'apprentissage sont fournis dans le fichier *spamTrain.mat* qui contient 4000 exemples d'emails spam et non-spam. Le fichier *spamTest.mat* contient 1000 autres exemples pour le test. Le vecteur X de

[1] *http : //spamassassin.apache.org/publiccorpus/*

chaque fichier comprend les vecteurs des descripteurs de chaque email tandis que le vecteur Y contient la décision (1 pour spam et 0 pour non-spam).

4 Travail à faire

1. Lire le contenu de emails fournis dans les fichiers *spamSample.txt* et *nonspamSample.txt*. Etes vous d'accord sur leur classification manuelle spam pour le premier et non-spam pour le deuxième?
2. Consulter le contenu du fichier *vocab.txt* du dictionnaire.
3. A partir des données d'apprentissage *spamTrain.mat*, créer le modèle du classifieur selon la technique SVM. Pour cela, utiliser la fonction *fitcsvm.m*.
4. Appliquer ce modèle sur les données de test en utilisant la fonction *predict.m*.
5. Evaluer les performances en se basant sur la matrice de confusion qui peut être calculée selon la fonction *confusionmat.m*.
6. Commenter les performances du classificateur en modifiant le choix du noyau et ses paramètres.