Introduction à la fouille de textes

Master 1, tous documents autorisés, 2h

1. Identification de tâches

Les systèmes de *résumé automatique* "par extraction" produisent le résumé d'un texte en "extrayant" (sélectionnant) un certain nombre des phrases qu'il contient. Les phrases sélectionnées, mises bout à bout, constituent le résumé. Pour réaliser un tel système, une stratégie habituelle est d'essayer de se ramener à une tâche plus classique de fouille de textes, pour laquelle on pourra utiliser un programme d'apprentissage automatique. On décrit ici différentes approches (plus ou moins naïves) pour procéder ainsi. On suppose pour cela que le texte est découpé en *phrases*.

1. Une première approche consiste à considérer chaque phrase indépendamment des autres et à chercher à savoir si elle doit être conservée dans le résumé ou pas. Les critères pris pour ce choix seront des informations comme : la taille de la phrase, sa position dans le texte, le nombre de verbes, de noms communs ou d'entités nommées qu'elle contient, etc. Chaque phrase est donc transformée en un "vecteur" avec ces différentes valeurs. Donner la nature de la tâche permettant la sélection des phrases, ses données d'entrée et de sortie, et les ressources ou pré-traitements nécessaires pour transformer les phrases en vecteurs.

L'inconvénient de cette méthodes est que les phrases sont considérées indépendamment les unes des autres et qu'elle ne prend pas en compte le sens des phrases et du texte à résumer. On ne peut pas non plus en procédant ainsi fixer à l'avance un "taux de compression" du texte (imposer par exemple qu'on extrait 1 phrase sur 2, ou 1 phrase sur 10...).

2. Pour remédier à ces limites, une autre approche consiste à chercher d'abord à regrouper entre elles les phrases sur une base sémantique (par exemple, en fonction des mots qu'elles ont en commun). A quelle tâche fera-t-on appel dans ce cas ?

Une fois les phrases regroupées entre elles en "paquets", il restera à sélectionner les phrases les plus représentatives de chaque paquet en respectant le taux de compression, puis à les remettre dans l'ordre initial (on ne demande pas de caractériser ces dernières étapes).

2. Questions sur la précision, le rappel, etc.

Pour les questions indépendantes suivantes, justifier votre réponse sans faire de calcul.

1. Si on veut que le plus possible de documents obtenus quand on fait une requête à un système de recherche d'information soient pertinents, peu importe si certains autres sont manqués par le système : favorise-t-on donc la précision ou le rappel ?

2. Face au résultat d'un système de recherche d'information opérant sur un corpus inconnu, avec une requête portant sur un sujet qu'il connaît bien, un utilisateur est-il capable de mesurer la précision et le rappel du système ?

3. Soit deux programmes distincts de recherche d'information, notés 1 et 2. On sait que, pour une requête donnée, le programme 1 a une meilleure précision que le programme 2 et que les deux ont la même F-mesure. Que peut-on dire sur leur rappel pour cette requête ?

4. On trouve un moyen de faire baisser le silence d'un programme de recherche d'information : améliore-t-on ainsi sa précision ou son rappel ?

5. Des documents pertinents pour une requête sont ajoutés à un corpus, mais on obtient pourtant toujours exactement les mêmes résultats avec un certain moteur de recherche pour cette requête : quel effet cet ajout de documents a-t-il eu sur la précision et le rappel du système ?

3. Recherche d'information et classification

On considère les petits textes suivants, constitués chacun d'une unique phrase :
texte 1 : Le chien aboie après l'oiseau.
texte 2 : Un chien est un animal.
texte 3 : Cet animal aboie.
texte 4 : Un oiseau est un animal qui vole.
texte 5 : Un oiseau est un animal qui a des plumes.
texte 6 : L' oiseau perd ses plumes.

1. Les textes sont segmentés et mis en minuscule, la ponctuation est éliminée. Les mots suivants sont considérés comme des mots vides : {a, après, cet, des, est, l', le, qui, ses, un}. Donner la liste des mots qui constituent l'espace de représentation de ces textes et donner la représentation vectorielle booléenne de chacun d'eux dans cet espace.

2. Quels mots de l'espace ont la plus faible et la plus forte valeur d'IDF quand on considère ces 6 textes (on demande de justifier la réponse mais pas de calculer la valeur exacte) ?

3. On reste dans la représentation booléenne. On suppose qu'on soumet à un moteur de recherche vectoriel la requête constituée de la nouvelle phrase suivante : "Le chien aboie dans son sommeil, il rêve qu'il a des plumes et vole comme un oiseau." Donner la représentation de cette phrase dans l'espace défini question 1. En prenant comme mesure de proximité la mesure de Jaccard, donner l'ordre dans lequel le moteur de recherche vectoriel va présenter les 6 textes.
indication : quand la représentation est booléenne, la mesure de Jaccard entre deux textes t_1 et t_2 vaut $\frac{|t_1 \cap t_2|}{|t_1 \cup t_2|}$ (nombre de mots distincts présents à la fois dans t_1 et t_2 divisé par le nombre de mots distincts présents soit dans t_1 soit dans t_2).

4. En fait, les textes 1 à 3 parlent principalement de "chien" tandis que les textes 4 à 6 parlent surtout d'"oiseau". On suppose que les 6 textes appartiennent donc à ces deux classes distinctes et servent d'exemples classés, tandis que la phrase jouant le rôle de requête dans la question 3 sert de donnée à classer. Dans quelle classe le programme NaiveBayes va-t-il ranger cette phrase ?

4. Recherche d'information et classification

On donne les coordonnées, dans un espace vectoriel simplifié à deux dimensions seulement (notées x et y), de 6 vecteurs (ou points) représentant (par exemple) des textes :

donnée	x	y
1	1	1
2	1	3
3	3	3
4	3	4
5	5	3
6	3	2

1. Dessiner sur un graphique (feuille quadrillée) les points représentant ces données.
2. Parmi ces points, lesquels sont à une distance nulle si on les considère comme des vecteurs et si on utilise une mesure de proximité cosinus ? Imaginer les coordonnées de deux autres points qui seraient aussi à une distance nulle des deux vecteurs précédents.
3. On suppose que le premier vecteur joue le rôle de requête, et les 5 autres le rôle de textes d'un corpus. Dans quel ordre un moteur de recherche vectoriel fondé sur la mesure de proximité cosinus va-t-il classer les textes en réponse à la requête ? (il n'y a besoin de faire aucun calcul, les proximités peuvent s'évaluer à l'oeil nu).
4. On suppose que seuls les 3 textes les plus proches sont considérés comme pertinents par le moteur et que seuls les textes représentés par les données 3, 4 et 5 sont réellement pertinents. Calculer la précision, le rappel et la F-mesure du moteur de recherche vectoriel.
5. Reprendre les questions 3 et 4 en prenant cette fois comme distance entre deux points (ou deux vecteurs) la distance euclidienne (elle peut aussi être évaluée sans calcul).
6. On suppose maintenant que les données 1, 2 et 6 appartiennent à une classe C1, alors que les 3 autres appartiennent à une classe C2. Ces 6 données étiquetées sont fournies en exemple à des programmes d'apprentissage automatique :
   - un programme d'arbre de décision : dessiner (sans calculs) l'arbre qui sera construit
   - un SVM : dessiner sur le graphique le meilleur séparateur possible
7. On fournit quelques autres données avec leur classe :

donnée	x	y	classe
7	1	4	C2
8	2	4	C2
9	3	1	C1
10	2	2	C2
11	1	2	C1

   Comment les deux algorithmes précédents vont-ils étiqueter ces nouvelles données ? Même question pour l'algorithme des 3 plus proches voisins.
8. Donner les 3 matrices de confusion des 3 algorithmes de la question précédente sur ces nouvelles données.