Pràctica2: Identificació de gènere utilitzant bag of words i machine learning.

1 Introducció

El camp d'investigació anomenat "Text/Document Classification" és un dels camps del processat del llenguatge amb més aplicacions pràctiques. Aquest camp tracta de classificar documents donades una sèrie de categories predeterminades utilitzant aprenentatge automàtic (Machine Learning).

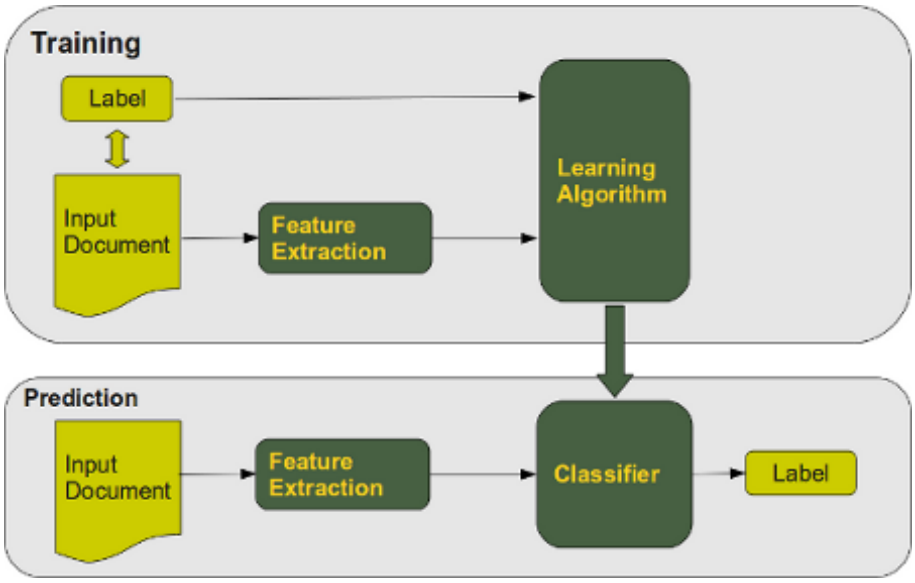

Figura 1: Text Classification Flow

Donat un document de input, s'extreuen una sèrie de característiques del mateix (o features), que caracteritzen la categoría a la que pertany (o label), i el diferencien de les altres opcions, les quals son conformen vectors de features que, a més de les classes a les que pertany cada document d'entrenament, son donats com input a un algoritme de machine learning, el qual aprendrà de les dades i será capaç, donat un exemple que no hagi vist mai, de predir a quina classe pertany.

A la figura es veu clarament que hi ha dues etapes diferenciades: primer s'extreuen features dels documents d'input, els quals estan etiquetats amb la seva categoría correcte (label) i se'ls donen com input a l'algoritme. Aquest conjunt d'exemples "resolts" conformen el conjunt de entrenament, o training set. Després, donat un exemple sense la seva etiqueta, s'extreuen les seves features i l'algoritme, amb el coneixement que ha extret del training set, fa una predicció.

2 De text a vectors de features

En aquesta classe de problemes, la elecció de les features correctes és la part més crítica de tot el sistema. Features ben seleccionades tindran gran capacitat predictiva, mentre que features mal triades seran incapaces de prediure res correctament.

El nostre input serà text plà, del qual extreurem una série de features. Posarem un exemple senzill per a entendre millor els conceptes:

Input: "El Joan, ha resultat ser el millor professor de tots els temps, el seu carisma només és superat per el seu sex-appeal."
Etiqueta: Veritat
Possibles etiquetes: Veritat, Mentida
Features a extreure: [numero de comes, numero de majúscules, numero de noms propis]
Vector de features: [2, 2, 1]

Després del procés d'extracció de features, el input és representat amb el vector de features que es mostra. És molt important, que tots els vectors de features tinguin les features representades en el mateix ordre: si el vector de l'exemple te a la dimensió 0 el numero de comes, a la dimensió 1 el numero de majúscules i a la dimensió 2 el numero de noms propis, es imprescindible que la resta de instáncies (que representen la resta dels inputs), representin els seus vectors de features de la mateixa manera, si no és així, l'algoritme de machine learning no estarà comparant dades comparables per aprendre.

3 Cas d'estudi: Author Profiling

En aquesta secció s'introdueix el cas d'estudi al qual ens enfrentarem. El camp de l'author profiling és un subconjunt del camp de "Text classification", on l'objectiu és prediure trets demogràfics de l'escriptor d'un text. És a dir, intentarem predir si l'autor de un text és un home o una dona, si és jove o vell o si és de Barcelona o Madrid. El principi bàsic darrere l'author profiling, és que persones que comparteixen trets demogràfics, també comparteixen patrons lingüistics que es poden detectar automáticament i utilitzar per predir.

Ens centrarem en el cas d'identificació de gènere: donat un text, l'ha escrit un home o una dona?

3.1 Bag of words classification

Per a classificar els autors per el seu gènere, utilitzarem una estratègia de bossa de paraules o bag of words. El que farem, serà agafar les N paraules més freqüents del corpus i per cada document, computarem el percentatge de les paraules que son cadascuna de les N més freqüents.

És a dir, si agafem les 5 paraules més freqüents i son, per exemple: i, és, jo, tu, ella, per cada text del corpus, els vectors de features seran així: [% paraules que son i, % paraules que son és, % paraules que son jo, % paraules que son tu, % paraules que son ella]. Aquesta tècnica, tot i ser de les més senzilles, pot obtenir molt bons resultats.

4 Realització de la pràctica

Per grups de 2 o 3. Es proporcionarà un zip amb un corpus compost per 1260 posts de columnes d'opinió en anglès. La meitat dels textos escrits per homes i l'altre meitat per dones. Als noms de fitxer hi ha la categoría a la qual pertanyen.

Les tasques que s'hauran de fer son les següents:

1. Implementar un codi que donat el corpus, extregui les N paraules més freqüents.
2. Implementar un codi que calculi la freqüència de cada una de les N paraules extretes al pas anterior, a cada un dels fitxers del corpus. L'output d'aquest pas ha de ser un vector de features per cada fitxer, on a cada dimensió hi ha la freqüència d'aparició de cada paraula de les N extretes abans. Molt important que encara que no aparegui cap vegada una paraula a un text, a la dimensió pertinent, aparegui un zero (tots els vectors de features han de tenir la mateixa llargada i les freqüències de cada paraula han de apareixer sempre en el mateix ordre.
3. Passar els vectors de features i les seves respectives labels a diversos classificadors i computar les precissions obtingudes.
4. Canviar els valors de N, el classificador i veure com varía la precissió.

4.1 Com fer tot això?

Primer pas: triar un llenguatge de programació. No hi ha imposició directe, pero la manera més fácil de fer-ho és en Python (de fet, si mirau el NLTK pot arribar a ser tot molt fàcil).

Segon pas: Extreure les N paraules més freqüents (no hauria de ser dificil).

Tercer pas: Calcular freqüències de aquestes paraules per text. Aquesta part no és dificil, pero haureu de pensar una estructura de dades per emmagatzemar tot això.

Quart pas: Escriure el output en un format concret i utilitzar un toolkit de machine learning. RECOMANACIONS: utilitzar WEKA `http://www.cs.waikato.ac.nz/ml/weka/`, el qual reb com input les features en format ARFF, `http://www.cs.waikato.ac.nz/ml/weka/arff.html`, té una gran quantitat d'algoritmes de machine learning disponibles i té interficie gràfica. Per tant, donat els vostres vectors de features i les seves labels correctes, haureu de escriure un fitxer ARFF, donar-li al weka i anar provant els diferents algoritmes que té implementats fent 10-fold cross-validation (tècnica definida breument a `https://www.cs.auckland.ac.nz/~pat/706_98/ln/node119.html`).

També podeu optar per utilitzar scikit learn, el qual no té interficie, i és una mica més complicat d'utilitzar, `http://scikit-learn.org/stable/`. La meva recomanació és que ho feu tot en Python, que creeu el fitxer ARFF i que jugueu amb el weka per a treure els resultats.

Cinquè pas: Variar el valor de la N, canviar el algoritme de classificació i veure com varía la precissió del sistema.

4.2 Entrega i Avaluació

S'ha d'entregar el codi i un informe on s'expliqui com s'ha implementat tot, i es mostrin els resultats, analitzant com varia la precissió segons la N i el classificador. És important que tant el codi com el informe estiguin ben presentats i explicats.

Per avaluar, es seguiran els següents criteris:

- Extracció de paraules freqüents (10%)
- Càlcul de features (35%)
- Generació output ARFF o input scikit learn (15%)
- Resultats i anàlisi dels mateixos, variant N, variant classificadors i explicant com evoluciona (40%)
- ****PUNTS EXTRA: Implementació de features extra que complementin el bag of words o que competeixin amb aquest approach. Les persones que implementin features noves i analitzin com vará la precissió afegint-les seran recompensats i es guanyaran el meu respecte. NOTA: si no s'aconsegueix fer tota una implementació de features noves, pero es raona un conjunt de features alternatiu, també es tindrà en compte.

L'informe i el codi s'han de lliurar el dia 26-03-2017 a les 23:55h.