Γλωσσική Τεχνολογία

Εργασία 2016-2017

Σπύρος Καφτάνης (5542)

Μέρος Β

Στο μέρος αυτό υλοποιήθηκε ένα σύστημα κατηγοριοποίησης κειμένων σε προκαθορισμένες θεματικές κατηγορίες. Η συλλογή κειμένων που χρησιμοποιήθηκε βρίσκεται στη διεύθυνση http://qwone.com/~jason/20Newsgroups/.

Από τα κείμενα που βρίσκονται εκεί χρησιμοποιήθηκε ένα μικρό μέρος τους, έτσι ώστε η διαδικασία εκτέλεσης να γίνει γρηγορότερη παρ' όλα αυτά ο χρήστης είναι ελεύθερος να προσθέσει και να αφαιρέσει όσα αρχεία επιθυμεί είτε σαν train data είτε σαν test data.

Ο κώδικας του προγράμματος βρίσκεται στο αρχείο part_two.py ενώ τα δεδομένα στον φάκελο 20news-bydate ο οποίος πρέπει να βρίσκεται στο ίδιο directory. Στον φάκελο αυτό υπάρχουν τα δεδομένα εκπαίδευσης και τα δεδομένα δοκιμής. Μαζί με τον κώδικα της εφαρμογής υπάρχουν τα αρχεία allDifferentWords.txt και allDifferentTestWords.txt. Αρχεία αυτά περιέχουν όλες τις διαφορετικές λέξεις των training και testing δεδομένων αντίστοιχα. Τα αρχεία αυτά έχουν παραχθεί από το part_two.py για τα δεδομένα που βρίσκονται στον φάκελο των δεδομένων κατά την παράδοση. Κατά την εκτέλεση του part_two.py, ο χρήστης ερωτάται εάν θέλει να παράξει ξανά τα δύο αυτά αρχεία, κάτι που έχει νόημα μόνο όταν έχουν προστεθεί ή έχουν αφαιρεθεί αρχεία (διαφορετικά θα παραχθούν οι ίδιες λέξεις). Οι λέξεις αυτές χρησιμοποιούνται για τον υπολογισμό του tf-idf. Χρησιμοποιήθηκαν αρχεία μιας και η διαδικασία παραγωγής των διαφορετικών λέξεων είναι αρκετά αργή ιδιαίτερα σε όχι τόσο ισχυρά μηχανήματα.

Η μόνη βιβλιοθήκη που χρησιμοποιήθηκε είναι η ntlk η οποία βοηθάει στη διαδικασία του tokenizing και του stemming.

Αρχικά το πρόγραμμα διαβάζει τα training και τα testing αρχεία, τα κάνει tokenize με τη χρήση του RegexpTokenizer κανονικοποιώντας τα παράλληλα (μετατρέποντας όλα τα γράμματα σε μικρά δηλαδή). Στη συνέχεια, αφού πλέον έχουμε ένα διάνυσμα με όλα τα tokens για κάθε κείμενο αφαιρούμε τις stop words, δηλαδή της λέξεις που δεν βοηθούν τον ταξινομητή στο να αποδώσει νόημα (άρθρα, σύνδεσμοι κτλ). Οι λέξεις αυτές παρέχονται από την ntlk. Μετά την αφαίρεση των άχρηστων λέξεων, γίνεται το stemming, δηλαδή η μετατροπή του κάθε token σε μία λέξη που θα αποδίδει το θέμα του, έτσι ώστε λέξεις σε παρόμοιες μορφές ή χρόνους να έχουν την ίδια σημασιολογική απόδοση. Το stemming γίνεται με τη βοήθεια του PorterStemmer της nltk.

Στη συνέχεια γίνεται η ερώτηση η οποία αναφέρθηκε πιο πάνω για τον αν ο χρήστης επιθυμεί τη δημιουργία των αρχείων με όλες τις διαφορετικές λέξεις. Οι λέξεις αυτές (είτε αλλάξει το dataset είτε όχι) φορτώνονται τελικά στη μνήμη προσωρινά

(all_words_in_all_docs), έτσι ώστε να υπολογιστούν οι συντελεστές που χρειάζονται για η μετατροπή του διανύσματος με τα stems σε ένα διάνυσμα με tf-idf αριθμούς. Από το all_words_in_all_docs υπολογίζεται ο όρος: Πλήθος των αρχείων που περιέχουν τον όρο t για κάθε όρο t, ο οποίος χρειάζεται για τον υπολογισμό του tfidf.

Αφού ολοκληρωθεί η παραπάνω διαδικασία για τα testing και για τα training data, στη συνέχεια γίνεται η μετατροπή των stems σε αριθμούς. Η tf-idf τιμή υπολογίζεται ως:

Tf-idf = TF*IDF

όπου για κάθε όρο t

TF = (αριθμός όπου η λέξη t εμφανίζεται στο κείμενο της)/(Συνολικό πλήθος λέξεων στο κείμενο)

και

IDF = log(Συνολικός αριθμός κειμένων/Πλήθος αρχείων που περιέχον τον όρο t)

Όταν όλα τα κείμενα μετατραπούν με ένα σταθερό διάνυσμα αριθμών είναι πλέον δυνατή υλοποίηση υποσυστήματος σύγκρισης διανυσμάτων με τη χρήση μετρικών. Σημειώνουμε εδώ ότι τα testing διανύσματα με τα training διανύσματα έχουν ίδιο μήκος επειδή χρησιμοποιούν σαν βάση όλες τις λέξεις όλων των κειμένων.

Το σύστημα σύστημα διανυσμάτων κάνει ερωτήσεις στον χρήστη για το ποιο testing data θέλει να ταξινομήσει και με ποια από τις δύο μετρικές (cosine ή jaccard), και αυτό απαντά τον αριθμό που δίνει η μετρική ομοιότητας, το που κατηγοριοποιήθηκε και τη πραγματική του κατηγορία.

Ένα παράδειγμά μίας σωστής και μία λάθος ταξινόμησης φαίνεται παρακάτω:

```
Ποια μετρική θέλετε να χρησιμοποιήσετε [1:Cosine, 2:Jaccard -1:Τερματισμός]:1
Ποιο αρχείο θέλετε να κατηγοριοποιήσετε [0-199]:61
Cosine Similarity =  0.126311322046
Κατηγοριοποιήθηκε σαν:  comp.graphics
Στη πραγματικότητα ανήκε στη κατηγορία:  sci.med
Ποια μετρική θέλετε να χρησιμοποιήσετε [1:Cosine, 2:Jaccard -1:Τερματισμός]:1
Ποιο αρχείο θέλετε να κατηγοριοποιήσετε [0-199]:62
Cosine Similarity =  0.135537858449
Κατηγοριοποιήθηκε σαν:  sci.med
Στη πραγματικότητα ανήκε στη κατηγορία:  sci.med
```

Επειδή τα training δεδομένα που έχουν κρατηθεί είναι λίγα (για γρηγορότερη εκτέλεση και παρουσίαση του προγράμματος) τα αποτελέσματα δεν είναι τόσο εντυπωσιακά, αλλά αυτά αλλάξει εύκολα εάν προσθέσουμε περισσότερα χωρίς να ξεχάσουμε να ανανεώσουμε τα αρχεία που περιέχουν όλες τις λέξεις από τα δεδομένα. Για πολύ καλύτερα αποτελέσματα έγιναν δοκιμές και με το alternative_dataset το οποίο περιέχει περισσότερα αρχεία.