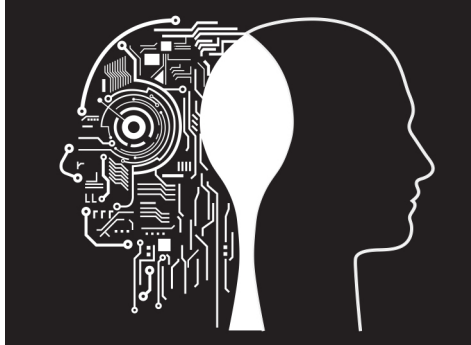


Τεχνητή νοημοσύνη

Θέμα 2



Ονοματεπώνυμο: Κερασιώτης Ιωάννης
Ονοματεπώνυμο: Ραφτόπουλος Ευάγγελος

AM: 03114951
AM: 03114743

Σκοπός της Άσκησης

Η άσκηση είναι η επέκταση της προηγούμενης. Στον κώδικα έχουν γίνει αρκετές τροποποιήσεις, τόσο στην διόρθωση λαθών, εισαγωγή νέων χαρακτηριστικών που βοηθούν στην επεξεργασία δεδομένων κατά της έξοδο. Πέρα από αυτά προσθέτοντας περισσότερες πληροφορίες στο ταξί προκειμένου να γίνεται η σωστότερη επιλογή οχήματος.

Δεδομένα Άσκησης

Πέρα από τα δεδομένα που δίνονταν στο πρώτο θέμα σχετικά με τους οδούς, την θέση των ταξί και του πελάτη, το πρόγραμμα δέχεται επιπλέον πληροφορίες. Συγκεκριμένα, στο αρχείο traffic.csv δίνονται πληροφορίες για την κίνηση στους δρόμους και στο αρχείο lines.csv τους οδούς, όπως ο αριθμός των λωρίδων. Αντίστοιχα υπήρχαν στα αρχεία taxis.csv και client.csv περιέχει περισσότερες πληροφορίες για τα ταξί και τον πελάτη αντίστοιχα, προκειμένου να γίνει η καλύτερη επιλογή.

Επεκτείναμε τις κλάσεις customer, taxi, node, ώστε να περιέχονται τα καινούρια δεδομένα και δημιουργήσαμε τις κλάσεις line, traffic και TimeAndAmount η οποία συνδέεται με την traffic για να παρέχει με εύκολο τρόπο τις αντιστοιχίες ώρες-κίνηση.

Υλοποίηση

Για το πιθανό ταίριασμα πελάτη-ταξί κάναμε χρήση prolog λαμβάνοντας υπόψη την γλώσσα, την χωρητικότητα και την απόσταση. Αυτή η είναι πρώτη χρήση που κάναμε του API JIProlog που μας βοηθάει να τρέχουμε κώδικα Prolog σε περιβάλλον Java ώστε να κάνουμε χρήση λογικού προγραμματισμού που θα μας βοηθήσει να δημιουργήσουμε κανόνες. Η δεύτερη χρήση είναι για τον υπολογισμό της ευρηστικής συνάρτησης για τον κάθε κόμβο (εκτός τους πρώτου φυσικά).

Τα στοιχεία που χρησιμοποιήθηκαν εδώ ήταν: Αν ο δρόμος ήταν της κατηγορίας highway ή όχι. Αν δεν ήταν η prolog επέστρεφε σαν ευριστική false, το οποίο το ερμηνεύαμε σαν την μέγιστη τιμή ενός double αριθμού. Δεν έγινε κάποιος έλεγχος για τις υποκατηγορίες του highway. Έπειτα γινόταν έλεγχος για την κατεύθυνση του δρόμου. Σε αυτό το σημείο σαν έλεγχο για την κατεύθυνση χρησιμοποιήσαμε την τιμή που έχουν στον άξονα X τα σημεία που ήταν ο κόμβος που βρισκόταν το ταξί και ο επόμενος κόμβος στον οποίο ήθελε να ταξιδέψει. Γνωρίζοντας ότι τα δεδομένα των κόμβων μας δίνονται ταξινομημένα από τον άξονα X και ότι η διεύθυνση είναι σχετική με την σειρά που μας δίνονται τα δεδομένα (-1 αν η διεύθυνση είναι αντίθετη από την σειρά που μας δίνονται τα δεδομένα, 1 το ανάποδο), αναπτύξαμε τον τρόπο να βρίσκουμε την διεύθυνση του δρόμου. Τέλος, σαν τιμή της ευριστικής χρησιμοποιήθηκε πάλι η Ευκλείδεια απόσταση. Προσπαθήσαμε να προσθέσουμε και την κίνηση μέσα στην ευριστική, χρησιμοποιώντας έναν πολλαπλασιαστή. Συγκεκριμένα αν ο δρόμος που θα ταξιδεύαμε είχε μεσαία ή υψηλή κίνηση πολλαπλασιάζαμε την ευριστική με 1.5 ή 2 αντίστοιχα. Τα γεγονότα αυτά μεταφέρονταν από το πρόγραμμα της Java σε αυτό της Prolog μέσω assertions που κάναμε δημιουργώντας γεγονότα, όπως belongsIn(nodeId, lineId), language(taxiId, language), traffic(lineID, startingHour, endingHour, Value), oneway(lineId, booleanOneway, booleanOnewayDirection) (booleanDirection false = wrong way), highway(lineId, boolean isHighway).

Αρα δεν υπάρχουν αρχεία με γεγονότα, αφού αυτά παράγονται κάθε φορά που τρέχουμε το πρόγραμμα!

Έξοδος Προγράμματος

Αυτή την φορά τροποποιήσαμε το πρόγραμμα ώστε να βγάζει ως αποτέλεσμα 2 αρχεία KML καθώς και απάντησεις στο standar output. Αρχικά εμφανίζει τα 5 πρώτα ταξί με βάση της απόσταση από τον πελάτη, δημιουργώντας το πρώτο αρχείο kml. Όπως παρακάτω:

The top five best results depending on distance we found were:
(visual representation in map1.kml)

| | | |
|----|---------|--------------------------------|
| 1: | Id: 230 | Distance: 0.02202831082311162 |
| 2: | Id: 120 | Distance: 0.02486894781253231 |
| 3: | Id: 210 | Distance: 0.0249914084432662 |
| 4: | Id: 200 | Distance: 0.02584569442605691 |
| 5: | Id: 170 | Distance: 0.027737490460401564 |

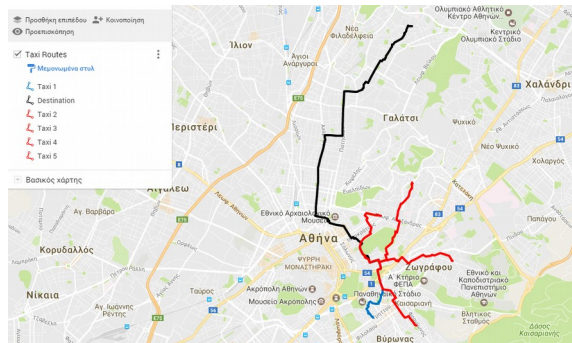
Έπειτα ταξινομεί τις 5 επιλογές με βάση τις προτιμήσεις του πελάτη, καθώς και τις παροχές των ταξί.

Top five taxi candidates rearranged by rating:
(visual representation in map2.kml)

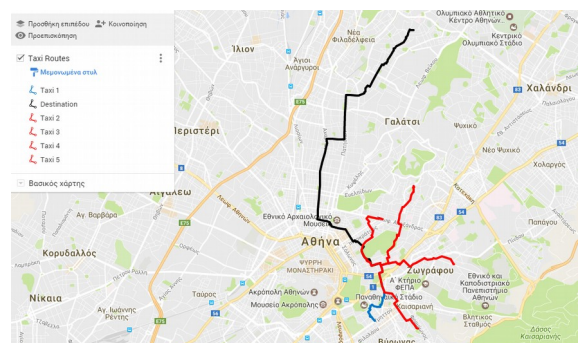
| | | |
|-----|---------|-------------|
| 1 : | Id: 230 | Rating: 9.8 |
| 2 : | Id: 210 | Rating: 9.2 |
| 3 : | Id: 120 | Rating: 8.0 |
| 4 : | Id: 200 | Rating: 8.0 |
| 5 : | Id: 170 | Rating: 7.1 |

Και στα δύο KML αρχεία οι μπλε διαδρομές είναι το ταξί που είναι βέλτιστη επιλογή, με κόκκινες με άλλες επιλογές και με μαύρο η απόσταση από τον τελικό προορισμό.

Παρακάτω θα δούμε το αποτέλεσμα για τα δεδομένα που δίνονται:



Map 1



Map 2

Που τυχάνει να συμπίπτουν.