Σάββατο 30 Ιανουαρίου 2020

Τμήμα Ηλεκτρολόγων Μηχανικών & Τεχνολογίας Υπολογιστών
Πανεπιστήμιο Πατρών

Εργαστηριακή άσκηση στο μάθημα «Ανάκτηση Πληροφορίας»

Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών & Πληροφορικής

Χειμερινό Εξάμηνο 2020-2021

Φραγκίσκος Φούρλας (up1059336@upnet.gr)
ΑΜ:1059336 (ΗΜΤΥ)

Υπεύθυνος Καθηγητής: Χ. Μακρής
Επικουρικό: Α. Μπομπότας, Γ. Ρόμπολας

Σελίδα Παραδοτέου: github.com/frank20a/personalised-search

Εισαγωγή

Στόχος της παρούσας άσκησης είναι η κατασκευή μιας μηχανής αναζήτησης για κινηματογραφικές ταινίες. Η αναζήτηση των ταινιών βασίζεται κατά βάση στην μηχανή αναζήτησης Elasticsearch. Στην συνέχεια τα αποτελέσματα προσωπικοποιούνται όσον αφορά την ταξινόμηση για να ικανοποιήσουν τις προτιμήσεις του εκάστοτε χρήστη. Για τον σκοπό αυτό αξιοποιούνται τεχνικές τεχνητής νοημοσύνης και μηχανικής μάθησης.

Περιβάλλον Υλοποίησης

Η υλοποίηση της εργασίας έγινε σε γλώσσα Python 3 χρησιμοποιώντας κατάλληλες βιβλιοθήκες λογισμικού. Συγκεκριμένα χρησιμοποιήθηκε η έκδοση 3.8 της Python (μπορεί να μεταφορτωθεί από τον σύνδεσμο www.python.org/downloads) και οι εγγενής βιβλιοθήκες της re (Regular Expressions), csv, os, sys, json και pickle, αλλά επιπλέον και οι βιβλιοθήκες tensorflow, numpy, scikit-learn, genism, pandas και elasticsearch οι οποίες πρέπει να εγκατασταθούν. Η εγκατάσταση των επιπλέον βιβλιοθηκών της Python μπορεί να γίνει με την εντολή *pip install *library** από την γραμμή εντολών στον φάκελο εγκατάστασης της Python. Λίστα με τις βιβλιοθήκες και τις εκδόσεις τους υπάρχει στο αρχείο requirements.txt το οποίο μπορεί να χρησιμοποιηθεί από γνωστά IDEs ή με την εντολή *pip install -r requirements.txt* για αυτόματη εγκατάστασή τους.

Ακόμη, χρησιμοποιήθηκε η τελευταία έκδοση της Elasticsearch (η οποία μπορεί να μεταφορτωθεί από τον σύνδεσμο www.elastic.co/downloads/elasticsearch). Η αντιγραφή των δεδομένων του προβλήματος στην βάση δεδομένων της Elasticsearch θεωρείτε μέρος της εγκατάστασης. Τρέχοντας τον κώδικα του αρχείου import_from_csv.py, το πρόγραμμα θα αντιγράψει τα δεδομένα από τα αρχείο CSV στην Elasticsearch, χρησιμοποιώντας το bulk API μέσω των Helpers για ταχύτερη αντιγραφή. Θα πρέπει να γίνει εκτέλεση του αρχείου /bin/elasticsearch.bat στην θέση που εγκαταστάθηκε η Elasticsearch πριν την εκτέλεση του προγράμματος. Σημειώνεται πως με χρήση Regular Expressions οι κατηγορίες των ταινιών χωρίζονται και μπαίνουν σε λίστα και ο τίτλος ξεχωρίζεται από την χρονολογία δημοσίευσης σε διαφορετική μεταβλητή.

Το παραδοτέο μπορεί να εκτελεστεί με άνοιγμα του αρχείου *search.py* μέσω του αρχείου *run.bat*. Σημειώνετε πως για την υλοποίηση του 4ου ερωτήματος είναι απαραίτητη η προπόνηση ενός νευρωνικού δικτύου για κάθε χρήστη. Η δομή του προπονημένου δικτύου αποθηκεύεται σε αρχείο της μορφής *\bin\user_models\m*ID*.h5*. Τα μοντέλα προπονούνται και αποθηκεύονται την πρώτη φορά που ο συγκεκριμένος χρήστης πραγματοποιεί μια αναζήτηση και φορτώνονται από την μνήμη από εκεί και πέρα. Τα μοντέλα για τους χρήστες 15, 137, 220 και 353 έχουν είδη αποθηκευτεί. Για οποιονδήποτε άλλο χρήστη είναι φυσιολογική αναμονή έως και ενός λεπτού πριν την παρουσίαση αποτελεσμάτων.

Διαδικασία Λύσης

Ερώτημα 1

Αφού γίνει αντιγραφή των δεδομένων στην Elasticsearch μπορούμε να χρησιμοποιήσουμε τις εγγενείς συναρτήσεις της για να λάβουμε αποτελέσματα για μια αναζήτηση. Το πρόγραμμα αρχικά ζητά από τον χρήστη τον αριθμό ID του (ο οποίος προς το παρόν απλώς αποθηκεύεται και δεν χρησιμοποιείτε) και μια καταχώρηση αναζήτησης. Στη συνέχεια το πρόγραμμα εκτελεί μια αναζήτηση με την Elasticsearch ψάχνοντας για ταύτιση στα πεδία τίτλου, κατηγορίας και χρονολογίας και κρατάει την λίστα αποτελεσμάτων και το σκορ της μετρητικής BM25 της Elasticsearch. Η λίστα που επιστρέφεται (στο εξής «λίστα αποτελεσμάτων»), ζητείται να έχει το μέγιστο επιτρεπτό μέγεθος και έτσι τα αποτελέσματα μπορούν να ταξινομηθούν βάση των προτιμήσεων του χρήστη και να μην περιοριστούν απόλυτα από την μετρητική ομοιότητας.

Ερώτημα 2

Για την υλοποίηση αυτού του ερωτήματος, το πρόγραμμα βρίσκει όλες τις κριτικές από το αρχείο ratings.csv για κάθε ταινία στην λίστα αποτελεσμάτων. Στην συνέχεια αναθέτει στην κάθε ταινία δύο επιπλέον τιμές σκορ εκτός της μετρητικής ομοιότητας BM25: αφενός τον μέσο όρο των κριτικών της από όλους τους χρήστες και αφετέρου την κριτική του χρήστη που πραγματοποιεί την αναζήτηση. Η τελευταία σε πολλές περιπτώσεις δεν υπάρχει και η κάλυψη της είναι το αντικείμενο των επόμενων 2 ερωτημάτων.

Στην συνέχεια τα σκορ κανονικοποιούνται (ο BM25 ως προς το μέγιστο σκορ, οι κριτικές ως προς το 5) και στην συνέχεια υπολογίζεται ο μέσος όρος των κανονικοποιημένων σκορ με βάρος. Έπειτα από πειραματισμούς επιλέχθηκε βάρος 1 για όλες τις τιμές εκτός της κριτικής του χρήστη, όπου, εάν υπάρχει, το βάρος είναι 5 και, εάν γίνει εκτίμηση της, το βάρος είναι 2. Εάν για οποιονδήποτε λόγο δεν μπορεί να εκτιμηθεί η κριτική του χρήστη, δεν συμπεριλαμβάνεται στον τελικό μέσο όρο.

Τελικά, παρουσιάζονται N πρώτα αποτελέσματα της αναζήτησης, ταξινομημένα βάσει της νέας μετρητικής που περιλαμβάνει την μετρητική ομοιότητας BM25, τον μέσο όρο κριτικών και την κριτική του χρήστη, εφόσον αυτή υπάρχει.

Ερώτημα 3

Σε προσπάθεια βελτίωσης της παραπάνω μεθόδου προσπαθώ να γεμίζω τα κενά όπου ο χρήστης δεν έχει δει και άρα αξιολογήσει μία ταινία. Η μέθοδος επίλυσης αυτού του προβλήματος που αντιμετωπίζεται σε αυτό το ερώτημα περιλαμβάνει τον διαχωρισμό των χρηστών σε ομάδες ατόμων με παρόμοιες αξιολογήσεις. Η παραδοχή πίσω από την συγκεκριμένη τεχνική είναι πως άτομα που τους αρέσουν παρόμοιες ταινίες θα αξιολογούσαν παρόμοια ταινίες που δεν έχουν δει. Η παραπάνω μέθοδος είναι γνωστή ως clustering (συσταδοποίηση) στην τεχνητή νοημοσύνη και υπάρχουν αλγόριθμοι που την υλοποιούν. Στην συνέχεια, το πρόγραμμα του ερωτήματος 2, όταν δεν υπάρχει βαθμολογία για μια ταινία από τον εκάστοτε χρήστη, μπορεί να χρησιμοποιεί αντ' αυτής τον μέσο όρο των κριτικών της ταινίας στην συστάδα που ανήκει ο χρήστης.

Για την εκτέλεση της ιδέας χρησιμοποιήθηκε ο αλγόριθμος clustering K-Means από την βιβλιοθήκη scikit-learn και η συσταδοποίηση έγινε βάση του μέσου όρου κριτικών ανά κατηγορία ταινιών κάθε χρήστη. Για το σενάριο όπου δεν υπάρχει καμία κριτική χρήστη για κάποια κατηγορία το κενό γεμίζει με τον μέσο όρο όλων των κριτικών της κατηγορίας. Αυτό προέκυψε από την δοκιμή με την μέθοδο k-cross validation μεταξύ των τιμών -1 (θεωρητικά απουσία, πρακτικά πολύ κακή κριτική), 0 (ίδιο με -1), 2.5 (ουδέτερη τιμή) και μέσο όρο. Ο αριθμός των clusters επιλέχθηκε 12, έπειτα από δοκιμή για σφάλματα από 5 έως 20 clusters.

Σημειώνω στο σημείο αυτό, πως λόγω του μεγέθους των δεδομένων και της πολυπλοκότητας των αλγορίθμων που τα επεξεργάζονται, ο χρόνος εκτέλεσης του προγράμματος γίνεται ανυπόφορα μεγάλος. Για τον λόγο αυτό, δημιουργήθηκε κλάση User η οποία αποθηκεύει όλα τα απαραίτητα δεδομένα για κάθε χρήστη (συμπεριλαμβανομένου των IDs των χρηστών της συστάδας στην οποία ανήκει). Λίστα με τα δεδομένα όλων των χρηστών αποθηκεύεται ως αντικείμενο pickle στον δίσκο και εκτέλεση των αλγορίθμων απαιτείται μόνο όταν τέτοιο αρχείο δεν υπάρχει διαθέσιμο. Αυτή η τεχνική χρησιμοποιείται για πολλά δεδομένα που δεν χρειάζεται να υπολογίζονται ξανά σε κάθε εκτέλεση του προγράμματος. Αυτό καθιστά την εκτέλεση πρακτικά real-time.

Ερώτημα 4

Παρόλο που η τεχνική του ερωτήματος 3 καλύπτει πολλές φορές τα κενά στις κριτικές των χρηστών, δεν είναι σπάνιο κανένα μέλος μιας συστάδας να μην έχει βαθμολογήσει μια ταινία. Στην προκειμένη περίπτωση καλούμαι να χρησιμοποιήσω τεχνικές μηχανικής μάθησης για να προβλέψω η βαθμολογία που θα έδινε ο χρήστης.

Για να επιτευχθεί το παραπάνω οι τίτλοι των ταινιών κωδικοποιούνται με την τεχνική των Word Embeddings. Με την τεχνική αυτή ένα μοντέλο προπονείται να αποτυπώνει λέξεις σε διανυσματικό χώρο βάση των νοημάτων και σχέσεων της σε ένα μεγάλο κείμενο. Με τον τρόπο αυτό τα διανύσματα των λέξεων αποκτούν νόημα και πράξεις όπως $\kappa\acute{o}\rho\eta - \theta\upsilon\lambda\iota\kappa\acute{o} = \tau\acute{\varepsilon}\kappa\nu o$ αποκτούν νόημα. Η προπόνηση ενός τέτοιου μοντέλου από τους τίτλους των ταινιών κρίνεται ακατάλληλη για τρεις λόγους: Το μέγεθος των δεδομένων είναι σχετικά μικρό, δεν υπάρχει νοηματική συνοχή μεταξύ των λέξεων (εκφράσεις, φυσικός λόγος) και οι τίτλοι δεν είναι πλήρεις προτάσεις. Για τον λόγο αυτό χρησιμοποιήθηκε το προ-προπονημένο μοντέλο GloVe[1]. Οι τίτλοι των ταινιών αποτελούν το άθροισμα των διανυσμάτων των λέξεων που τους αποτελούν. Στο διάνυσμα αυτό επαυξάνεται το διάνυσμα των κατηγοριών της ταινίας που δημιουργείται από την one-hot κωδικοποίησή τους. Αυτό σημαίνει πως σε σταθερού μήκους διάνυσμα παρουσία μιας κατηγορίας μεταφράζεται σε 1 και απουσία της σε 0, σε συγκεκριμένη θέση.

Στην συνέχεια αυτό το τελικό διάνυσμα ταινίας πρέπει να αντιστοιχιστεί σε μια τιμή βαθμολογίας μεταξύ 0 και 5. Το πρόβλημα λοιπόν ανάγεται σε πρόβλημα regression. Για την επίλυσή του κατασκευάζεται νευρωνικό δίκτυο της παρακάτω μορφής[Σχήμα 1]. Η έξοδος του νευρωνικού δικτύου έχει γραμμική συνάρτηση ενεργοποίησης και μοναδικό κόμβο. Ως συνάρτηση σφάλματος για την εκμάθηση, χρησιμοποιεί το μέσο τετραγωνικό σφάλμα που κρίνεται κατάλληλο για γραμμικά προβλήματα regression. Χρησιμοποιώ τον KerasRegressor, ένα εργαλείο της scikit-learn για επίλυση προβλημάτων regression με δεδομένο νευρωνικό δίκτυο κατασκευασμένο με την βιβλιοθήκη keras.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 80)                9600
_________________________________________________________________
dense_1 (Dense)              (None, 50)                4050
_________________________________________________________________
dense_2 (Dense)              (None, 18)                918
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 19
=================================================================
Total params: 14,587
Trainable params: 14,587
Non-trainable params: 0
_________________________________________________________________
```

Σχήμα 1: Δομή νευρωνικού δικτύου

Σχολιασμός αποτελεσμάτων

Για τη σύγκριση και τον σχολιασμό των αποτελεσμάτων θα χρησιμοποιήσω τους χρήστες 353, 220 και 137. Οι χρήστες αυτοί έχουν ικανοποιητικό αριθμό αξιολογήσεων σε ταινίες και για αυτό κρίνονται κατάλληλοι για τον σκοπό αυτό. Ως καταχώρηση αναζήτησης χρησιμοποιείται η λέξη *star* που μπορεί να παραπέμπει στις γνωστές σειρές ταινιών Star Wars ή Star Trek.

Ήδη από το τρίτο ερώτημα τα αποτελέσματα της αναζήτησης είναι πολύ καλά όπως φαίνεται παρακάτω[Σχήμα 2]. Ο χρήστης 353 λαμβάνει πρώτα την ταινία Star Wars VI μιας και την έχει βαθμολογήσει καλύτερα από το Star Wars IV, αντίθετα με τους δύο άλλους οι οποίοι λαμβάνουν την εν λόγω ταινία πρώτη. Με τον ίδιο τρόπο ο χρήστης 137 λαμβάνει δεύτερη την Star Wars V ενώ ο 220 την Star Wars VI, ενώ δεν την έχει βαθμολογήσει και η επιλογή του εκτιμάται βάσει της συστάδας στην οποία ανήκει.

Όσον αφορά την επίλυση της τέταρτης ερώτησης, αυτή παρουσίασε πολλά προβλήματα. Αρχικά, δοκίμασα την τεχνική one-hot encoding για την διανυσματοποίηση των τίτλων, ταυτίζοντας κάθε λέξη της πρότασης με την θέση τους σε λεξικό. Αυτή η μέθοδος, αν και όχι αυτή που ζητείται, έδωσε ένα μέτριο μέσο σφάλμα 1.2, όταν χρησιμοποιήθηκε με το νευρωνικό δίκτυο. Στην συνέχεια, έγινε προσπάθεια να εκπαιδευτεί μοντέλο Word Embeddings με την βιβλιοθήκη Word2Vec της Gensim. Αυτό αποδείχθηκε για τους προαναφερθέντες λόγους ακατάλληλο και έδωσε αποτελέσματα ανεπαρκή και με μεγάλες αποκλίσεις.

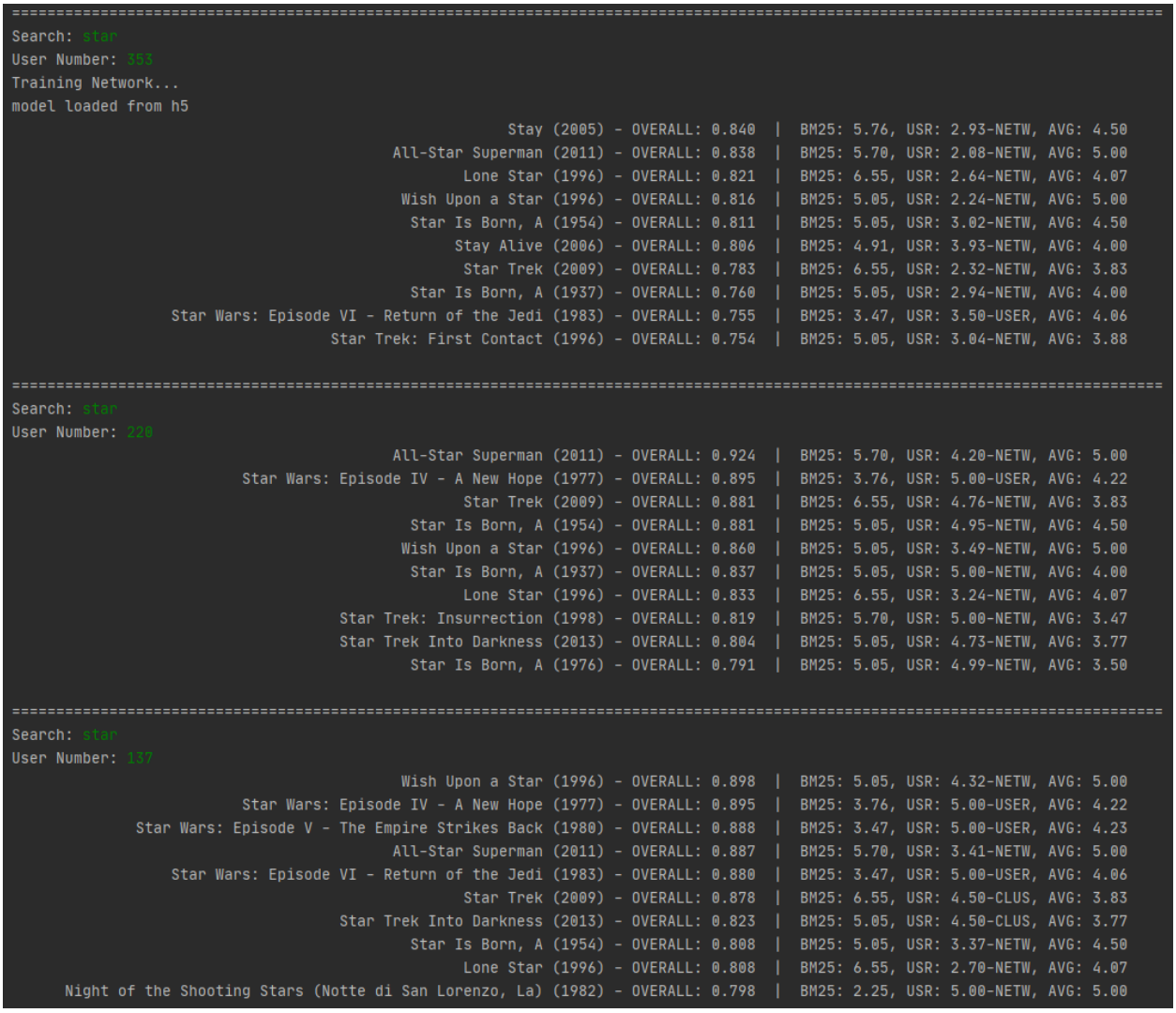
Ακόμα, κρίνω την χρήση νευρωνικού δικτύου άστοχη για το συγκεκριμένο πρόβλημα για τους εξής λόγους:

```
================================================================================================================================================
User Number: 353
Search: star
                  Star Wars: Episode VI - Return of the Jedi (1983) - OVERALL: 0.906  |  BM25: 3.47, USR: 3.50-0, AVG: 4.06
                          Star Wars: Episode IV - A New Hope (1977) - OVERALL: 0.815  |  BM25: 3.76, USR: 3.00-0, AVG: 4.22
               Star Wars: Episode V - The Empire Strikes Back (1980) - OVERALL: 0.809  |  BM25: 3.47, USR: 3.00-0, AVG: 4.23
                 Star Wars: Episode III - Revenge of the Sith (2005) - OVERALL: 0.690  |  BM25: 3.47, USR: 2.50-0, AVG: 3.63
                 Star Wars: Episode II - Attack of the Clones (2002) - OVERALL: 0.675  |  BM25: 3.47, USR: 2.50-0, AVG: 3.10
                    Star Wars: Episode I - The Phantom Menace (1999) - OVERALL: 0.582  |  BM25: 3.76, USR: 2.00-0, AVG: 3.20
                                       All-Star Superman (2011) - OVERALL: 0.468  |  BM25: 5.70, USR: 0.00-1, AVG: 5.00
                                               Lone Star (1996) - OVERALL: 0.454  |  BM25: 6.55, USR: 0.00-1, AVG: 4.07
                                                    Stay (2005) - OVERALL: 0.445  |  BM25: 5.76, USR: 0.00-1, AVG: 4.50
                                        Wish Upon a Star (1996) - OVERALL: 0.443  |  BM25: 5.05, USR: 0.00-1, AVG: 5.00

================================================================================================================================================
User Number: 220
Search: star
                          Star Wars: Episode IV - A New Hope (1977) - OVERALL: 0.917  |  BM25: 3.76, USR: 5.00-0, AVG: 4.22
                  Star Wars: Episode VI - Return of the Jedi (1983) - OVERALL: 0.635  |  BM25: 3.47, USR: 3.00-1, AVG: 4.06
                   Star Wars: Episode I - The Phantom Menace (1999) - OVERALL: 0.602  |  BM25: 3.76, USR: 3.00-0, AVG: 3.20
                                       All-Star Superman (2011) - OVERALL: 0.468  |  BM25: 5.70, USR: 0.00-1, AVG: 5.00
                                               Lone Star (1996) - OVERALL: 0.454  |  BM25: 6.55, USR: 0.00-1, AVG: 4.07
                                                    Stay (2005) - OVERALL: 0.445  |  BM25: 5.76, USR: 0.00-1, AVG: 4.50
                                        Wish Upon a Star (1996) - OVERALL: 0.443  |  BM25: 5.05, USR: 0.00-1, AVG: 5.00
                                               Star Trek (2009) - OVERALL: 0.442  |  BM25: 6.55, USR: 0.00-1, AVG: 3.83
                                         Star Is Born, A (1954) - OVERALL: 0.418  |  BM25: 5.05, USR: 0.00-1, AVG: 4.50
                                               Star Maps (1997) - OVERALL: 0.400  |  BM25: 6.55, USR: 0.00-1, AVG: 3.00

================================================================================================================================================
User Number: 137
Search: star
                          Star Wars: Episode IV - A New Hope (1977) - OVERALL: 0.917  |  BM25: 3.76, USR: 5.00-0, AVG: 4.22
               Star Wars: Episode V - The Empire Strikes Back (1980) - OVERALL: 0.911  |  BM25: 3.47, USR: 5.00-0, AVG: 4.23
                   Star Wars: Episode VI - Return of the Jedi (1983) - OVERALL: 0.906  |  BM25: 3.47, USR: 5.00-0, AVG: 4.06
                         Star Trek III: The Search for Spock (1984) - OVERALL: 0.748  |  BM25: 3.76, USR: 4.00-0, AVG: 3.31
                   Star Wars: Episode I - The Phantom Menace (1999) - OVERALL: 0.745  |  BM25: 3.76, USR: 4.00-0, AVG: 3.20
                                       All-Star Superman (2011) - OVERALL: 0.468  |  BM25: 5.70, USR: 0.00-1, AVG: 5.00
                                               Lone Star (1996) - OVERALL: 0.454  |  BM25: 6.55, USR: 0.00-1, AVG: 4.07
                                                    Stay (2005) - OVERALL: 0.445  |  BM25: 5.76, USR: 0.00-1, AVG: 4.50
                                        Wish Upon a Star (1996) - OVERALL: 0.443  |  BM25: 5.05, USR: 0.00-1, AVG: 5.00
                                               Star Trek (2009) - OVERALL: 0.442  |  BM25: 6.55, USR: 0.00-1, AVG: 3.83

================================================================================================================================================
```

Σχήμα 2: Αποτελέσματα αναζήτησης. Το 1 μετά το σκορ του χρήστη παραπέμπει σε εκτίμηση και όχι πραγματική βαθμολόγηση. Βαθμολόγηση 0.0-1 παραπέμπει σε αδυναμία προσέγγισης με την μέθοδο συσταδοποίησης.

- Αρχικά η προπόνηση ενός νευρωνικού δικτύου για κάθε χρήστη φαίνεται ανορθόδοξη λόγω του πολλού χρόνου που διαρκεί, του αποθηκευτικού χώρου που καταλαμβάνει αλλά κυρίως λόγο του ότι δεν υπάρχουν αρκετά δεδομένα από πολλούς χρήστες για να γίνει η εκπαίδευση.
- Ο μέσος όρος τον βαθμολογιών ανά χρήστη είναι μόνο 70, ενώ υπάρχουν και πολλοί χρήστες με λιγότερες από 10.
- Η ιδέα να δώσουμε το id του χρήστη σαν μια είσοδο σε ένα μοναδικό νευρωνικό απορρίφθηκε επίσης, εφόσον θεωρούμε πως η διαφορά μεταξύ του αριθμού πχ 1 και 2 για το δίκτυο είναι αμελητέα ενώ μπορεί να εκφράζει έναν εντελώς διαφορετικό χρήστη.

Παραθέτω παρακάτω τα αποτελέσματα της ίδιας αναζήτησης εμπλουτισμένα με τις προβλέψεις από το ερώτημα 4[Σχήμα 3]. Σημειώνω πως έχει γίνει μεγάλη αλλαγή στα βάρη που χρησιμοποιούνται κατά τον υπολογισμό του σκορ κάθε ταινίας[Πίνακας 1]. Αυτές οι επιλογές δηλώνουν αμφιβολία για την ποιότητα του αποτελέσματος του νευρωνικού δικτύου και



Σχήμα 3: Αποτελέσματα αναζήτησης με βοήθεια νευρωνικού δικτύου. Χαρακτηριστική είναι η περίπτωση του χρήστη 137 όπου και οι 3 μέθοδοι συμπληρώνουν η μία την άλλη.

προσδίδουν μεγάλη σημασία στις πραγματικές αξιολογήσεις του χρήστη. Παρατηρούμε στα τελευταία αποτελέσματα, παρουσία ταινιών που οφείλετε σε πρόταση του νευρωνικού δικτύου. Τα αποτελέσματα πια δεν είναι γεμάτα με τις πρώτες αξιολογίσεις των χρηστών (ταινίες Star Wars) αλλά με αποτελέσματα πιο ταιριαστά στην αναζήτηση. Παρόλα αυτά το νευρωνικό δίκτυο προτείνει στους χρήστες ταινίες Star Trek που μοιάζουν με τις ταινίες Star Wars τις οποίες έχουν αξιολογίσει θετικά. Παρόλ' αυτά καμία από τις δύο σειρές ταινιών δεν είναι κυρίαρχη στα αποτελέσματα αφού ο τίτλος τους δεν ταυτίζεται απόλυτα με την καταχώρηση αναζήτησης.

BM25	Average Rating	User Rating	Cluster Average	NN Prediction
4	6	15	8	3

Πίνακας 1

Τέλος, θα αναλύσω αποτελέσματα αναζητήσεων από τον χρήστη 15 ο οποίος έχει 1700 αξιολογήσεις ταινιών. Ξεκινώντας από την συνηθισμένη αναζήτηση «Star»[Σχήμα 4] αυτό που

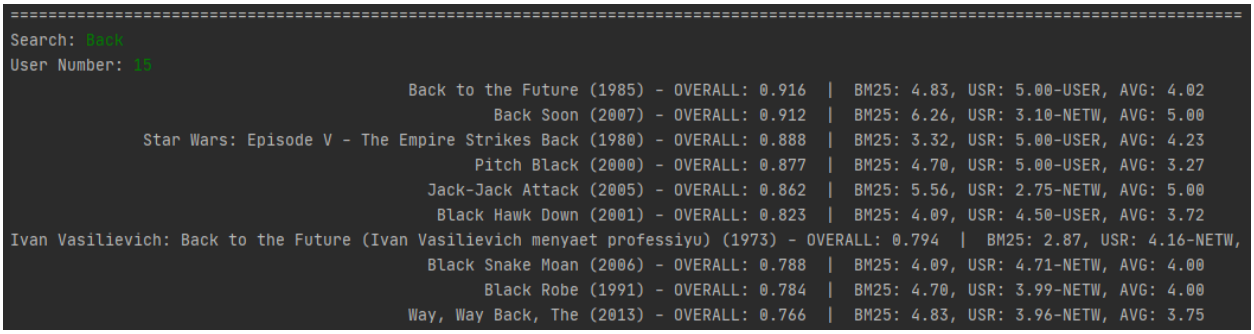
παρατηρούμε είναι η σημασία του πλήθους δεδομένων όσον αφορά την προσωποποιημένη αναζήτηση, μιας και τα αποτελέσματα αποτελούνται κυρίως από ταινίες που ο χρήστης έχει αξιολογήσει θετικά. Στην συνέχεια με την αναζήτηση «Toy»[Σχήμα 5] βλέπουμε το πρόβλημα που προκαλεί κάποιες φορές το Fuzziness της αναζήτησης της Elasticsearch μιας και μόνο ένα αποτέλεσμα περιέχει την λέξη που ζητήσαμε. Αυτό μπορεί ίσως να διορθωθεί με αύξηση του βάρους της μετρητικής BM25. Τέλος, η αναζήτηση μιας γενικής λέξης όπως «Back»[Σχήμα 6] βοηθάει να δούμε μια γενικότερη εικόνα των αποτελεσμάτων της αναζήτησης. Στην προκειμένη περίπτωση τα αποτελέσματα περιλαμβάνουν και προτιμήσεις του χρήστη όπως «Back to the Future» και «The Empire Strikes Back» αλλά και «Pitch Black» αξιοποιώντας θετικά το Fuzziness, όπως επίσης και προτάσεις από το σύστημα βάση αξιολογήσεων και της βαθμολογίας του νευρωνικού δικτύου.

```
================================================================================================================================================
Search: Star
User Number: 15
Training Network...
model loaded from h5
                                          Lone Star (1996) - OVERALL: 0.955  |  BM25: 6.55, USR: 5.00-USER, AVG: 4.07
                        Star Wars: Episode IV - A New Hope (1977) - OVERALL: 0.895  |  BM25: 3.76, USR: 5.00-USER, AVG: 4.22
             Star Wars: Episode V - The Empire Strikes Back (1980) - OVERALL: 0.888  |  BM25: 3.47, USR: 5.00-USER, AVG: 4.23
                                          Star Trek (2009) - OVERALL: 0.884  |  BM25: 6.55, USR: 4.50-USER, AVG: 3.83
                Star Wars: Episode VI - Return of the Jedi (1983) - OVERALL: 0.880  |  BM25: 3.47, USR: 5.00-USER, AVG: 4.06
                                               Stay (2005) - OVERALL: 0.860  |  BM25: 5.76, USR: 3.76-NETW, AVG: 4.50
               Star Wars: Episode III - Revenge of the Sith (2005) - OVERALL: 0.859  |  BM25: 3.47, USR: 5.00-USER, AVG: 3.63
                                     Wish Upon a Star (1996) - OVERALL: 0.821  |  BM25: 5.05, USR: 2.64-NETW, AVG: 5.00
                                    All-Star Superman (2011) - OVERALL: 0.814  |  BM25: 5.70, USR: 1.84-NETW, AVG: 5.00
                                      Star Is Born, A (1954) - OVERALL: 0.795  |  BM25: 5.05, USR: 3.09-NETW, AVG: 4.50
```

Σχήμα 4: Αποτελέσματα αναζήτησης χρήστη 15 με καταχώρηση «Star»

```
================================================================================================================================================
Search: Toy
User Number: 15
                                       Too Big to Fail (2011) - OVERALL: 0.862  |  BM25: 4.06, USR: 3.68-NETW, AVG: 4.50
                                  Back to the Future (1985) - OVERALL: 0.854  |  BM25: 1.74, USR: 5.00-USER, AVG: 4.02
                                           Boy (2010) - OVERALL: 0.816  |  BM25: 3.53, USR: 2.96-NETW, AVG: 4.75
                                 Toy Soldiers (1991) - OVERALL: 0.814  |  BM25: 4.51, USR: 2.98-NETW, AVG: 4.00
                                Coming to America (1988) - OVERALL: 0.784  |  BM25: 1.96, USR: 4.50-USER, AVG: 3.62
                                       Tom Jones (1963) - OVERALL: 0.779  |  BM25: 3.00, USR: 3.51-NETW, AVG: 4.46
                                       Boy Crazy (2009) - OVERALL: 0.778  |  BM25: 3.00, USR: 2.41-NETW, AVG: 5.00
                                     Top Secret! (1984) - OVERALL: 0.777  |  BM25: 3.00, USR: 4.00-USER, AVG: 3.96
                          Man Who Knew Too Little, The (1997) - OVERALL: 0.765  |  BM25: 1.89, USR: 4.50-USER, AVG: 3.29
                                  Regret to Inform (1998) - OVERALL: 0.757  |  BM25: 1.96, USR: 5.00-NETW, AVG: 4.25

================================================================================================================================================
Search: Toy Story
User Number: 15
                                       Toy Story (1995) - OVERALL: 0.878  |  BM25: 10.25, USR: 2.00-USER, AVG: 3.87
                                    Toy Story 3 (2010) - OVERALL: 0.869  |  BM25: 8.93, USR: 2.00-USER, AVG: 4.07
                                Toy Story of Terror (2013) - OVERALL: 0.866  |  BM25: 7.91, USR: 1.86-NETW, AVG: 4.00
         Internet's Own Boy: The Story of Aaron Swartz, The (2014) - OVERALL: 0.756  |  BM25: 4.30, USR: 2.45-NETW, AVG: 3.50
                                    Toy Story 2 (1999) - OVERALL: 0.611  |  BM25: 8.93, USR: 1.00-USER, AVG: 3.84
```

Σχήμα 5: Αποτελέσματα αναζήτησης χρήστη 15 με καταχώρηση «Toy» και «Toy Story»



Σχήμα 6: Αποτελέσματα αναζήτησης χρήστη 15 με καταχώρηση «Back»

Πηγές

[1] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation.