

Εργασία Ανάκτηση Πληροφορίας

Χειμερινό Εξάμηνο 2023-2024

Μέλη:

Μαρινέλα Κάλθι, 3984

Καλλιόπη Μαλλά, 3979

Εισαγωγή

Η εργασία αφορά την εφαρμογή των τεχνικών Ανάκτησης Πληροφορίας στις ομιλίες της Βουλής των Ελλήνων που καταγράφηκαν στο χρονικό πλαίσιο Ιουλίου 1989 μέχρι και Ιουλίου 2020. Στόχος είναι η οργάνωση και η επεξεργασία των δεδομένων που θα εξάγουν χρήσιμες πληροφορίες για τις ομιλίες αυτές. Δημιουργήθηκε λοιπόν μια web εφαρμογή αξιοποιώντας τη Flask και ένα πλαίσιο web Python. Αυτή η εφαρμογή χρησιμεύει ως μια πλατφόρμα για την αναζήτηση και την επεξεργασία πολιτικών δεδομένων. Αξιοποιώντας το σύστημα δρομολόγησης της Flask, σχεδιάστηκε μια φιλική προς τον χρήστη διεπαφή που τους επιτρέπει να υποβάλλουν ερωτήματα και να ανακτούν αποτελέσματα. Ακόμη ενσωματώνει μηχανισμούς διαχείρισης ασφαλμάτων για να εξασφαλίσει μια ομαλή εμπειρία χρήσης σε οποιεσδήποτε απρόβλεπτες περιστάσεις.

1.

Ανάγνωση: ReadCSV.py

Λειτουργία: `readCSV()`

Περιγραφή:

Αυτή η λειτουργία διαβάζει δεδομένα από ένα αρχείο CSV που περιέχει κοινοβουλευτικές διαδικασίες και εξάγει σχετικές πληροφορίες ενώ φιλτράρει άσχετες καταχωρίσεις δεδομένων.

Λεπτομέρειες υλοποίησης:

Χρησιμοποιεί τη βιβλιοθήκη pandas για την ανάγνωση δεδομένων από το αρχείο CSV. Φιλτράρει σειρές με το πολιτικό κόμμα «βουλή», οι οποίες θεωρούνται άσχετες. Διαβάζει λέξεις τερματισμού από ένα εξωτερικό αρχείο για περαιτέρω προ επεξεργασία.

Συνάρτηση: `read_stop_words()`

Περιγραφή:

Αυτή η λειτουργία διαβάζει λέξεις διακοπής από ένα αρχείο κειμένου και τις αποθηκεύει σε έναν πίνακα για μελλοντική χρήση στην προ επεξεργασία κειμένου.

Λεπτομέρειες υλοποίησης:

Ανοίγει και διαβάζει λέξεις διακοπής από το καθορισμένο αρχείο κειμένου.

Αφαιρεί χαρακτήρες νέας γραμμής από κάθε λέξη τερματισμού και τους προσαρτά στον πίνακα λέξεων τερματισμού.

Αρχικοποίηση : data_initiliazation.py

Συνάρτηση: init()

Περιγραφή:

Αυτή η λειτουργία προετοιμάζει την προ επεξεργασία και εξαγωγή σχετικών πληροφοριών από τα δεδομένα των κοινοβουλευτικών διαδικασιών. Αρχικοποιεί τα λεξικά για την αποθήκευση των επεξεργασμένων δεδομένων και των σχετικών μεταδεδομένων για μεταγενέστερη ανάλυση.

Λεπτομέρειες υλοποίησης:

Καλεί τη συνάρτηση readCSV() για ανάγνωση δεδομένων και λήψη λέξεων τερματισμού.

Επαναλαμβάνει τα δεδομένα των κοινοβουλευτικών διαδικασιών, προ επεξεργάζεται κάθε ομιλία και εξάγει σχετικές πληροφορίες.

Ενημερώνει τα λεξικά με επεξεργασμένα δεδομένα, συχνότητες λέξεων και μεταδεδομένα, όπως πληροφορίες μελών και κομμάτων.

Παρέχει ενημερώσεις προόδου κατά την επεξεργασία.

Προ επεξεργασία Δεδομένων: data_processing.py

Απαιτείται η προ επεξεργασία του ακατέργαστου κειμένου για την εξαγωγή ουσιαστικών πληροφοριών. Αυτό επιτυγχάνεται με:

1. Αφαίρεση σημείων στίξης και ειδικών χαρακτήρων: Για να παραμείνουν τα δεδομένα ουσιαστικά για μετέπειτα ανάλυση (όπως εξέταση ομοιότητας) πρέπει να αφαιρεθούν τα σημεία στίξης και οι ειδικοί χαρακτήρες. Κατά την υλοποίηση χρησιμοποιήθηκε η συνάρτηση `replace` της Python που αντικαθιστά τους ειδικούς χαρακτήρες και τα σημεία στίξης με κενά, αντικαθιστά τα κεφαλαία γράμματα με πεζά και τις τονικότητες. Ύστερα φιλτράρει τις κενές συμβολοσειρές, τα κενά και τις μη αλφαβητικές λέξεις.
2. Αφαίρεση stop words: Οι λέξεις τερματισμού, λέξεις που απαντώνται συνήθως όπως "και", "το", "είναι" κ.λπ., δεν έχουν σημαντικό σημασιολογικό νόημα και ενδέχεται να παραμορφώσουν τα αποτελέσματα της ανάλυσης. Τα stop words ανακτήθηκαν από τον σύνδεσμο <https://github.com/stopwords-iso/stopwords-el/blob/master/stopwords-el.txt> σε .txt αρχείο. Ο έλεγχος γίνεται με βρόγχο επανάληψης και εξέταση συνθήκης.
3. Διαδικασία Stemming: Χρησιμοποιείται μια ελληνική βιβλιοθήκη προέλευσης `greek_stemmer` για να μετατρέψουμε τις λέξεις στις κανονικές τους μορφές, μειώνοντας τις κλιτές ή παράγωγες λέξεις.

result.py

1. Λειτουργία:"retrieve_sitting(ερώτημα,δεδομένα, index_dictionary, words_dictionary, tags_dictionary)"

Περιγραφή:

Αυτή η συνάρτηση ανακτά τις κορυφαίες 5 πιο παρόμοιες κοινοβουλευτικές συνεδριάσεις σε ένα δεδομένο ερώτημα με βάση προϋπολογισμένες βαθμολογίες ομοιότητας.

Λεπτομέρειες υλοποίησης:

- Χρησιμοποιεί έναν υπολογισμό ομοιότητας εγγράφου-ερωτήματος από τη κλάση «similarity_formula».
- Ανακτά σχετικές πληροφορίες συνεδρίας, όπως αναγνωριστικό συνεδρίας, όνομα ομιλητή, πολιτικό κόμμα και ετικέτες (συχνές λέξεις στην ομιλία) από τα παρεχόμενα σύνολα δεδομένων.
- Επιστρέφει μια λίστα λεπτομερειών συνεδρίας ταξινομημένες κατά βαθμολογίες ομοιότητας.

2. Λειτουργία:"retrieve_sitting_information(sitting_id,data_frame,index_dictionary, tags_dictionary)"

Περιγραφή:

Αυτή η λειτουργία παρέχει λεπτομερείς πληροφορίες σχετικά με μια συγκεκριμένη κοινοβουλευτική σύνοδο που προσδιορίζεται από το αναγνωριστικό της συνόδου.

Λεπτομέρειες υλοποίησης:

- Ανακτά πληροφορίες συνεδρίας, συμπεριλαμβανομένων του ονόματος του ομιλητή, της ημερομηνίας συνεδρίας, του πολιτικού κόμματος, του περιεχομένου

ομιλίας και των ετικετών (συχνές λέξεις στην ομιλία) από τα παρεχόμενα σύνολα δεδομένων.

- Επιστρέφει μια λίστα λεπτομερειών συνεδρίας που αντιστοιχούν στο δεδομένο αναγνωριστικό περιόδου συνεδρίασης.

3. Λειτουργία: `retrieve_sittings_for_speaker(speaker,data_frame, index_dictionary, tags_dictionary, member_dictionary)"`

Περιγραφή:

Αυτή η λειτουργία ανακτά όλες τις κοινοβουλευτικές συνεδριάσεις που σχετίζονται με έναν συγκεκριμένο ομιλητή.

Λεπτομέρειες υλοποίησης:

- Χρησιμοποιεί το "member_dictionary" για την ανάκτηση αναγνωριστικών περιόδου συνεδρίασης που σχετίζονται με το συγκεκριμένο ομιλητή.
- Καλεί το "retrieve_sitting_information()" για κάθε αναγνωριστικό συνεδρίας για να ανακτήσει λεπτομερείς πληροφορίες συνεδρίας.
- Επιστρέφει μια λίστα λεπτομερειών περιόδου λειτουργίας που σχετίζονται με το καθορισμένο ομιλητή.

4. Λειτουργία: `retrieve_sittings_for_party(party,data_frame, index_dictionary, tags_dictionary, party_dictionary, member_dictionary)"`

Περιγραφή:

Αυτή η λειτουργία ανακτά 5 κοινοβουλευτικές συνόδους από ένα συγκεκριμένο πολιτικό κόμμα, η καθεμία από διαφορετικό μέλος.

Λεπτομέρειες υλοποίησης:

- Ανακτά αναγνωριστικά περιόδου σύνδεσης που σχετίζονται με τα μέλη του καθορισμένου πολιτικού κόμματος χρησιμοποιώντας το «party_dictionary» και το «member_dictionary».
- Καλεί το "retrieve_setting_information()" για κάθε αναγνωριστικό συνεδρίας για να ανακτήσει λεπτομερείς πληροφορίες συνεδρίας.

- Επιστρέφει μια λίστα λεπτομερειών συνεδρίας, διασφαλίζοντας ότι κάθε συνεδρία προέρχεται από διαφορετικό μέλος του κόμματος.

similarity_formula.py

➤ Λειτουργία: ``doc_query_similarity(words_dict, query)``

Περιγραφή:

Αυτή η συνάρτηση υπολογίζει την ομοιότητα μεταξύ των ερωτημάτων των χρηστών και των κοινοβουλευτικών ομιλιών με βάση τις βαθμολογίες TF-IDF.

Λεπτομέρειες υλοποίησης:

- Χρησιμοποιεί ένα λεξικό λέξεων για να ανακτήσει σχετικές ομιλίες που περιέχουν τουλάχιστον μία λέξη από το ερώτημα.
- Υπολογίζει τη βαθμολογία TF-IDF για κάθε έγγραφο που ταιριάζει με τους όρους του ερωτήματος.
- Υπολογίζει την ομοιότητα συνημίτονου μεταξύ του διανύσματος ερωτήματος και κάθε διανύσματος εγγράφου.
- Επιστρέφει ένα λεξικό που περιέχει τα αναγνωριστικά των κορυφαίων 5 πιο σχετικών ομιλιών μαζί με τις βαθμολογίες ομοιότητάς τους.

Σχέδιο βαθμολογίας TF-IDF:

Το σύστημα βαθμολόγησης TF-IDF χρησιμοποιείται για να σταθμίσει τη σημασία των όρων στις κοινοβουλευτικές ομιλίες σχετικά με τα ερωτήματα των χρηστών. Αποτελείται από τα ακόλουθα βήματα:

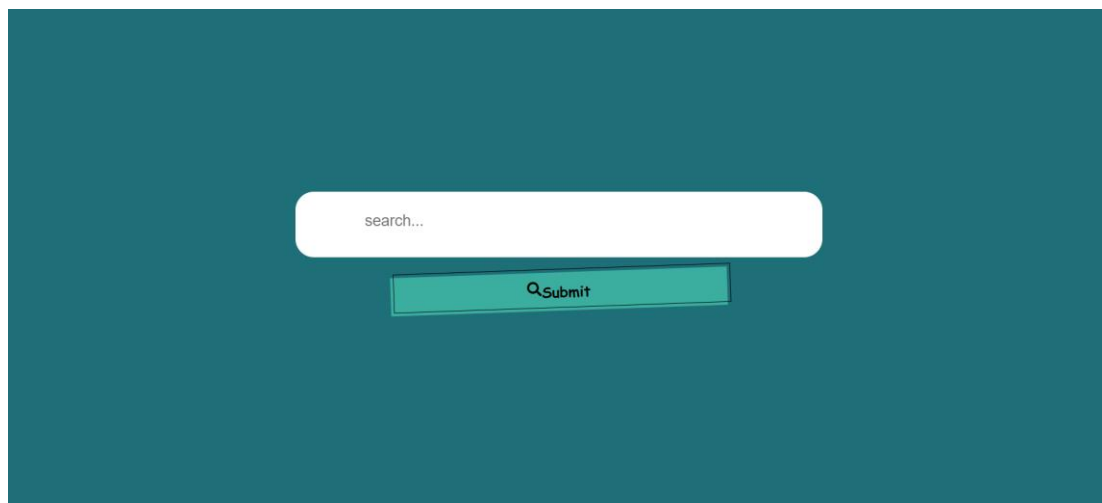
1. ****TF (Term Frequency)****: Μετρά τη συχνότητα των όρων σε κάθε έγγραφο.
2. ****IDF (Αντίστροφη Συχνότητα Εγγράφων)****: Καθορίζει τη σημασία των όρων σε όλα τα έγγραφα.
3. ****Βαθμολογία TF-IDF****: Συνδυάζει TF και IDF για να εκχωρήσει βάρη σε όρους σε μεμονωμένα έγγραφα.

Ομοιότητα συνημίτονου:

Η μέτρηση ομοιότητας συνημίτονου χρησιμοποιείται για την ποσοτικοποίηση της ομοιότητας μεταξύ του διανύσματος ερωτήματος και κάθε διανύσματος εγγράφου. Υπολογίζει το συνημίτονο της γωνίας μεταξύ δύο διανυσμάτων, υποδεικνύοντας την κατευθυντική τους ομοιότητα ανεξάρτητα από το μέγεθός τους.

Templates: index.html

Ο παρεχόμενος κώδικας HTML (περιλαμβάνει εσωτερικά τον κώδικα CSS) δημιουργεί μια φόρμα αναζήτησης με ένα πεδίο εισαγωγής και ένα κουμπί υποβολής. Η φόρμα επιτρέπει στους χρήστες να εισάγουν ένα ερώτημα αναζήτησης, το οποίο υποβάλλεται με το πάτημα του κουμπιού υποβολής. Το πεδίο εισόδου έχει κείμενο τοποθέτησης "search..." για να προτρέπει τους χρήστες να εισάγουν το ερώτημά τους.



search...

Submit

Templates: result.html

Ο παρεχόμενος κώδικας HTML (περιλαμβάνει εσωτερικά τον κώδικα CSS) εμφανίζει τα αποτελέσματα αναζήτησης σε μορφοποιημένη διάταξη πίνακα. Περιλαμβάνει:

- Ένα κουμπί επιστροφής που επιτρέπει στους χρήστες να πλοηγηθούν πίσω στην προηγούμενη σελίδα.
- Ένα τμήμα που υποδεικνύει το ερώτημα αναζήτησης του χρήστη.
- Έναν πίνακα που εμφανίζει τα αποτελέσματα της αναζήτησης με τις ακόλουθες στήλες: ID, ομιλητής, πολιτικό κόμμα, ετικέτες και βαθμολογία.
- Κάθε γραμμή του πίνακα αντιστοιχεί σε ένα έγγραφο παρόμοιο με το ερώτημα του χρήστη.
- Τα κουμπιά μέσα σε κάθε γραμμή επιτρέπουν στους χρήστες να βλέπουν περισσότερες λεπτομέρειες.

← You searched: {{uquery}}				
ID	Speaker	Political Party	Tags	Score
{% for id, speaker, party, tags, score in queryDetails %}				
{{id}}	{{speaker}}	{{party}}	{{tags}}	{{score}}
{% endfor %}				

Templates: party.html

Ο παρεχόμενος κώδικας HTML (περιλαμβάνει εσωτερικά τον κώδικα CSS) εμφανίζει πληροφορίες σχετικά με τα πολιτικά κόμματα και τις σχετικές συνεδριάσεις τους. Περιλαμβάνει:

- Ένα κουμπί επιστροφής που επιτρέπει στους χρήστες να επιστρέψουν στην προηγούμενη σελίδα.
- Ένα τμήμα που εμφανίζει το όνομα του πολιτικού κόμματος.
- Έναν πίνακα με στήλες για ID, Speaker και Tags, που αντιπροσωπεύει τις λεπτομέρειες των συνεδριάσεων που σχετίζονται με το πολιτικό κόμμα.
- Κάθε γραμμή του πίνακα αντιστοιχεί σε μια συνεδρίαση, με κουμπιά που επιτρέπουν στους χρήστες να βλέπουν περισσότερες λεπτομέρειες.

←		
Political Party		{{party_name}}
{% for id, speaker, tags in sittings %}		
ID	Speaker	Tags
{{id}}	{{speaker}}	{{tags}}
{% endfor %}		

Templates: sitting.html

Ο παρεχόμενος κώδικας HTML (περιλαμβάνει εσωτερικά τον κώδικα CSS) εμφανίζει λεπτομερείς πληροφορίες σχετικά με μια ομιλία, συμπεριλαμβανομένων του ομιλητή, του πολιτικού κόμματος, των ετικετών, της ημερομηνίας και του ίδιου του κειμένου της ομιλίας. Περιλαμβάνει:

- Ένα κουμπί επιστροφής που επιτρέπει στους χρήστες να πλοηγηθούν πίσω στην προηγούμενη σελίδα.
- Μια ενότητα που εμφανίζει τον ομιλητή, το πολιτικό κόμμα, τις ετικέτες και την ημερομηνία της ομιλίας.
- Το κύριο σώμα της σελίδας περιέχει το κείμενο της ομιλίας.
- Κάθε στοιχείο είναι δομημένο μέσα σε λίστες και πίνακες για σωστή οργάνωση και μορφοποίηση.

Speaker	Political party	Tags	Date
{{toPrint[0]}}	{{toPrint[2]}}	{{toPrint[4]}}	{{toPrint[1]}}

Speech

{{toPrint[3]}}

Templates: speaker.html

Ο παρεχόμενος κώδικας HTML (περιλαμβάνει εσωτερικά τον κώδικα CSS) παρουσιάζει πληροφορίες σχετικά με έναν συγκεκριμένο ομιλητή, συμπεριλαμβανομένου του ονόματός του, μαζί με λεπτομέρειες των συνεδριάσεών του, όπως το αναγνωριστικό, το πολιτικό κόμμα και τις σχετικές ετικέτες. Ακολουθεί μια ανάλυση:

- Η σελίδα περιλαμβάνει ένα κουμπί επιστροφής για πλοήγηση.
- Το όνομα του ομιλητή εμφανίζεται σε περίοπτη θέση.
- Χρησιμοποιείται ένας πίνακας για την οργάνωση των λεπτομερειών των συνεδριάσεων του ομιλητή, συμπεριλαμβανομένου του αναγνωριστικού, του πολιτικού κόμματος και των ετικετών που σχετίζονται με κάθε συνεδρίαση.
- Κάθε συνεδρίαση παρατίθεται σε μια γραμμή πίνακα, με επιλογές για την προβολή ολόκληρης της συνεδρίασης ή όλων των συνεδριάσεων ενός συγκεκριμένου πολιτικού κόμματος.

←		
Speaker		{{speaker_name}}
{% for id, party, tags in sittings %}		
ID	Political Party	Tags
<u>{{id}}</u>	<u>{{party}}</u>	{{tags}}
{% endfor %}		

Templates: Error.html

Αυτός ο κώδικας HTML δημιουργεί μια οπτικά ελκυστική σελίδα σφάλματος με προσαρμοσμένο σχέδιο. Η σελίδα σφάλματος είναι κεντραρισμένη και διαμορφωμένη με μεγάλα, στυλιζαρισμένα ψηφία "404" και είναι διακριτικά κινούμενη.

KeywordAnalyzer.py

1. Προεπεξεργασία ομιλιών:

- Η συνάρτηση `'preprocess_speeches'` διαβάζει ακατέργαστα δεδομένα ομιλίας από ένα αρχείο CSV χρησιμοποιώντας τη συνάρτηση `'ReadCSV'`.
- Προεπεξεργάζεται κάθε ομιλία με τη δημιουργία συμβόλων (tokenizing), την αφαίρεση των stopwords και την εξαγωγή σχετικών λέξεων-κλειδιών χρησιμοποιώντας την ενότητα `'data_processing'`.
- Τα προεπεξεργασμένα δεδομένα οργανώνονται σε λεξικά με βάση τα ονόματα των μελών και τα πολιτικά κόμματα, με ευρετήριο το έτος της ημερομηνίας συνεδρίασης.

2. Παραγωγή αναλύσεων λέξεων-κλειδιών:

- Η συνάρτηση `'write_keywords_to_file'` παράγει αναλύσεις λέξεων-κλειδιών τόσο για μεμονωμένα μέλη όσο και για πολιτικά κόμματα.
- Επαναλαμβάνει τα προεπεξεργασμένα λεξικά δεδομένων και υπολογίζει τις συχνότητες των λέξεων χρησιμοποιώντας την κλάση `'Counter'`.
- Οι 15 συχνότερες λέξεις-κλειδιά επιλέγονται και γράφονται σε ξεχωριστά αρχεία κειμένου για κάθε έτος.

3. Κύρια λειτουργία:

- Η συνάρτηση `'find_KeyWords'` χρησιμεύει ως σημείο εισόδου για τον κώδικα.
- Ορίζει παραμέτρους όπως η αύξηση για την επεξεργασία των ομιλιών και καλεί τις συναρτήσεις προεπεξεργασίας και ανάλυσης λέξεων-κλειδιών.

Τα εξαγόμενα αρχεία βρίσκονται στο folder "output_files" και ονομάζονται MemberKeywordsAnalysis.txt και PartyKeywordsAnalysis.txt

TopKSimilaritiesFinder.py

Αυτός ο κώδικας υλοποιεί τη λειτουργία εύρεσης των κορυφαίων k παρόμοιων ομιλητών με βάση την ομοιότητα των ομιλιών τους. Ας δούμε αναλυτικά τι κάνει ο κώδικας:

1. Προεπεξεργασία Δεδομένων

Τα δεδομένα διαβάζονται από ένα αρχείο CSV χρησιμοποιώντας τη συνάρτηση `ReadCSV`. Κάθε ομιλία χαρακτηρίζεται και επεξεργάζεται χρησιμοποιώντας συναρτήσεις από τη μονάδα επεξεργασίας δεδομένων. Οι λέξεις τερματισμού και τα άσχετα διακριτικά φιλτράρονται και τα δεδομένα οργανώνονται σε λεξικά.

2. Υπολογισμός ομοιότητας

Η διανυσματοποίηση TF-IDF εφαρμόζεται χρησιμοποιώντας το `TfidfVectorizer` από τη λειτουργική μονάδα `sklearn.feature_extraction.text`. Οι βαθμολογίες ομοιότητας συνημιτονικού υπολογίζονται με χρήση `syntetikonikou_similarity` από τη λειτουργική μονάδα `sklearn.metrics.pairwise`.

3. Εύρεση Top K Παρόμοιων Ομιλιών

Εφαρμόζεται ένας αλγόριθμος για να βρεθούν τα κορυφαία K παρόμοια ζεύγη με βάση τις βαθμολογίες ομοιότητας συνημιτόνου. Τα ζευγάρια επιλέγονται και ταξινομούνται με βάση τις βαθμολογίες ομοιότητάς τους.

4. Εγγραφή αποτελεσμάτων στο αρχείο

Τα κορυφαία 15 παρόμοια ζεύγη εγγράφονται σε ένα αρχείο κειμένου σε καθορισμένη μορφή. Κάθε ζευγάρι περιλαμβάνει τα ονόματα των ομιλητών, τα πολιτικά τους κόμματα και τη βαθμολογία ομοιότητας.

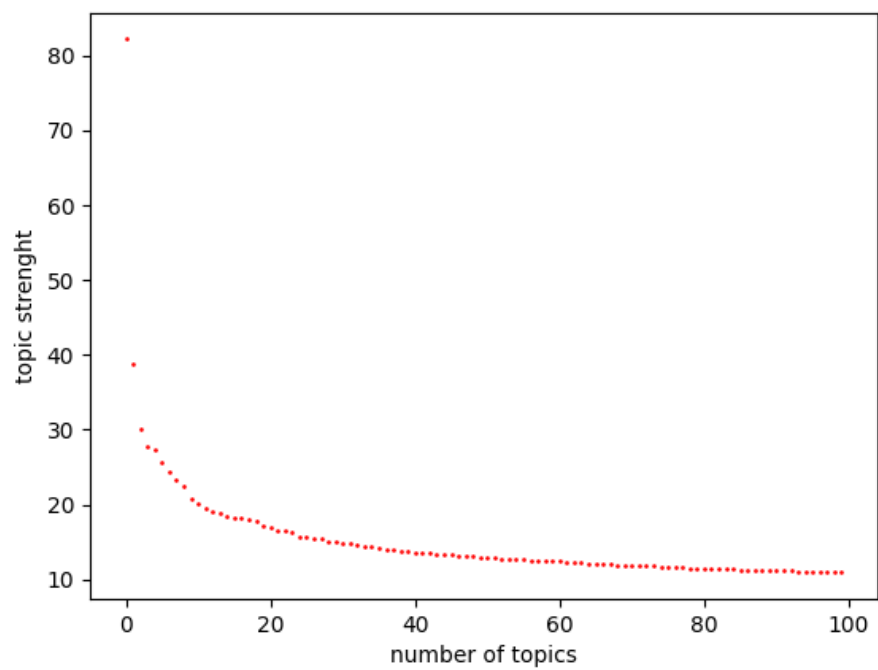
Το εξαγόμενο αρχείο βρίσκεται στο folder “output_files” και ονομάζεται `TopKSimilar.txt`

LSI.py

Χρησιμοποιήθηκε η τεχνική LSI, ώστε να βρούμε τις σημαντικότερες θεματικές περιοχές και να εκφράσουμε την κάθε ομιλία ως διάνυσμα σε κάποιον πολυδιάστατο χώρο.

Η διαδικασία εξαγωγής θεμάτων από τις ομιλίες χρησιμοποιώντας τον αλγόριθμο LSA (Latent Semantic Analysis) περιλαμβάνει τα εξής βήματα:

1. Αρχικά φορτώνουμε δεδομένα από ένα αρχείο CSV, επεξεργάζομαστε τις ομιλίες που περιέχονται σε αυτό, υπολογίζουμε το μήκος των δεδομένων και φορτώνουμε λέξεις-κλειδιά από ένα αρχείο κειμένου ("stopwords.txt") στη μεταβλητή `stop_words_array`.
2. Έπειτα επεξεργάζομαστε κάθε ομιλία από το αρχικό σύνολο δεδομένων, διαιρώντας την σε λέξεις, ελέγχοντας αν έχει περισσότερες από 100 λέξεις και αν ναι, εφαρμόζουμε την συνάρτηση προεπεξεργασίας `preprocess` στο κείμενο(από την κλάση `data_processing.py`), αφαιρώντας τις λέξεις-κλειδιά, και τέλος το προσθέτουμε σε μια λίστα επεξεργασμένων ομιλιών.
3. Στην συνέχεια χρησιμοποιούμε τη μέθοδο TF-IDF για τη μετατροπή των κειμένων σε διανύσματα χαρακτηριστικών και στη συνέχεια τη μείωση της διαστατικότητας τους με χρήση της μεθόδου SVD.
4. Τέλος υπολογίζουμε την αναπαράσταση των εγγράφων στον μειωμένο χώρο χρησιμοποιώντας το μοντέλο LSA . Το αποτέλεσμα είναι ένας πίνακας με τις νέες αναπαραστάσεις των εγγράφων στον μειωμένο χώρο . Τα δεδομένα αυτά εγγράφονται στο αρχείο `LSAtopics.txt`.
5. Επίσης χρησιμοποιήσαμε τον κώδικα `LSI_topics_strength.py` όπου ο κώδικας αυτός παράγει ένα γράφημα σε μορφή γραμμής που δείχνει την "ισχύ" ή "δύναμη" των θεμάτων σε ένα σύνολο δεδομένων.



Το εξαγόμενο αρχείο ονομάζεται LSItopics.txt