ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Υλοποίηση Chatbot με Χρήση Large Language Models

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΘΑΝΑΣΙΟΣ Σ. ΤΣΙΜΠΗΣ

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2024

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Υλοποίηση Chatbot με Χρήση Large Language Models

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΘΑΝΑΣΙΟΣ Σ. ΤΣΙΜΠΗΣ

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 15η Οκτωβρίου 2024.

.....	.....	.....
Δημήτριος Ασκούνης	Ιωάννης Ψαρράς	Ευάγγελος Μαρινάκης
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Επίκουρος Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2024

.....
Αθανάσιος Σ. Τσίμπης

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Τα συστήματα chatbot έχουν ολοένα και αυξανόμενο αντίκτυπο στην κοινωνία, απλοποιώντας σημαντικά πληθώρα εργασιών και παρέχοντας βελτιωμένη εμπειρία χρήστη για τις υπηρεσίες και τις ιστοσελίδες στις οποίες ενσωματώνονται. Τα σύγχρονα chatbot αξιοποιούν αλγορίθμους μηχανικής μάθησης και επεξεργασίας φυσικής γλώσσας, πετυχαίνοντας ισχυρές επιδόσεις.

Το Εργαστήριο Συστημάτων Υποστήριξης Αποφάσεων και Διοίκησης του Εθνικού Μετσόβιου Πολυτεχνείου συμμετέχει σε διάφορα ερευνητικά έργα, από τα οποία έχει προκύψει ένα σημαντικό πλήθος εγγράφων που περιέχουν πληροφορίες σχετικές με τα έργα αυτά. Σκοπός της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη ενός συστήματος chatbot, το οποίο θα επιτρέψει την διευκόλυνση της αναζήτησης πληροφορίας για το σύνολο των παραπάνω εγγράφων. Χρησιμοποιώντας το chatbot αυτό, οι χρήστες του θα μπορούν να μάθουν ποιος εργάστηκε σε συγκεκριμένα τμήματα των έργων καθώς και πως πραγματοποιήθηκε η εκάστοτε εργασία πολύ πιο εύκολα και γρήγορα.

Το σύστημα chatbot που προτείνεται ακολουθεί το μοτίβο σχεδίασης RAG. Πιο συγκεκριμένα, το chatbot αποτελείται από ένα τμήμα παραγωγής απάντησης το οποίο επιτρέπει την αξιοποίηση των υψηλών δυνατοτήτων ενός μεγάλου γλωσσικού μοντέλου και από ένα τμήμα ανάκτησης πληροφορίας που παρέχει στο τμήμα παραγωγής απάντησης δεδομένα σχετικά με το εκάστοτε ερώτημα. Το τμήμα ανάκτησης αξιοποιεί μοντέλα μηχανικής μάθησης και μια διανυσματική βάση, προκειμένου να αποθηκεύσει κατάλληλα την υπάρχουσα πληροφορία και να είναι σε θέση να ανακτήσει σχετικά με τα ερωτήματα τμήματα πληροφορίας, πραγματοποιώντας τεχνικές σύγκρισης ομοιότητας. Αξιοποιώντας τα δύο παραπάνω τμήματα, το τελικό σύστημα είναι ικανό να προσφέρει σχετικές και ορθές απαντήσεις στον χρήστη του.

Στα πλαίσια της παρούσας εργασίας, διερευνήθηκαν τρόποι με τους οποίους μπορεί να γίνει η όσο το δυνατόν ορθότερη εξαγωγή της πληροφορίας που περιέχεται στα έγγραφα που αποτελούν το σύνολο δεδομένων μας. Πραγματοποιήθηκαν επίσης πειράματα αξιολόγησης του τμήματος ανάκτησης του chatbot μας αλλά και του συνολικού συστήματος chatbot, όπου χρησιμοποιήθηκαν κατάλληλες μετρικές για κάθε περίπτωση αξιολόγησης. Παρουσιάζονται επιπλέον τα αποτελέσματα των πειραμάτων αξιολόγησης και σχετικές παρατηρήσεις.

Τέλος, παρατίθενται τα συμπεράσματα σχετικά με τις δυνατότητες και τους περιορισμούς του προτεινόμενου συστήματος chatbot, καθώς και τρόποι με τους οποίους εκείνο μπορεί πιθανά να επεκταθεί στο μέλλον.

Λέξεις-Κλειδιά

Chatbot, Μεγάλα Γλωσσικά Μοντέλα, RAG, Μηχανική Μάθηση, Επεξεργασία Φυσικής Γλώσσας, Διανυσματικές Βάσεις Δεδομένων, Σημασιολογική Αναζήτηση, Ανάκτηση Πληροφορίας

Abstract

Chatbot systems have an ever-increasing impact on society, significantly simplifying a multitude of tasks and providing an improved user experience in services and websites they are integrated into. Modern chatbots utilize machine learning and natural language processing algorithms, achieving strong performance.

The Decision Support Systems and Management Laboratory of the National Technical University of Athens participates in various research projects, from which a significant number of documents have emerged, containing information related to these projects. The goal of this thesis is the development of a chatbot system, which will facilitate the information search for the aforementioned documents. By using this chatbot, its users will be able to learn who worked on particular parts of a project and also how the action was implemented, much faster and easier.

The proposed chatbot system follows the RAG design pattern. More specifically, the chatbot consists of a response generation component which allows the utilization of the high capabilities of a large language model and a retrieval component that provides relevant data for each question to the response generation component. The retrieval component utilizes machine learning models and a vector database, in order to properly store the existing information and to be able to retrieve information passages relevant to the questions, by using similarity comparison techniques. By utilizing the above two components, the final system is capable of offering relevant and correct answers to its user.

In the scope of this work, ways that enable the extraction of the information contained in the documents that make up our dataset as correctly as possible, were investigated. Various experiments that evaluate the retrieval component and the chatbot system as a whole have also been carried out, using the appropriate metrics for each evaluation case. Furthermore, the evaluation experiments results are presented, along with the relevant observations.

Finally, the conclusions regarding the capabilities and limitations of the proposed chatbot system as well as possible ways with which it can be extended in the future, are listed.

Keywords:

Chatbot, Large Language Models, RAG, Machine Learning, Natural Language Processing, Vector Databases, Semantic Search, Information Retrieval

Ευχαριστίες

Καταρχάς θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Δημήτριο Ασκούνη για την ευκαιρία που μου δόθηκε να εκπονήσω την παρούσα διπλωματική εργασία στο Εργαστήριο Συστημάτων Υποστήριξης Αποφάσεων και Διοίκησης. Θα ήθελα επίσης να ευχαριστήσω τον Χρήστο Ντάνο, τον Λουκά Ηλία και τον Ανδρέα Μπότσικα για τον χρόνο που αφιέρωσαν, καθώς και για την καθοδήγηση και τις συμβουλές που μου παρείχαν καθ' όλη τη διάρκεια της εκπόνησης της διπλωματικής αυτής εργασίας.

Τέλος θα ήθελα να ευχαριστήσω την οικογένεια και τους φίλους μου, για την συνεχή υποστήριξη και την συμπαράσταση τους κατά τη διάρκεια των σπουδών μου.

Αθανάσιος Τσίμπης

Πίνακας Περιεχομένων

Πίνακας Εικόνων

Κατάλογος Πινάκων

1. Εισαγωγή

Τα chatbot είναι εργαλεία λογισμικού, τα οποία έχουν ως στόχο την παροχή βοηθητικών υπηρεσιών προς τον χρήστη τους, προσομοιάζοντας μια ανθρώπινη συνομιλία. Την σημερινή εποχή, τα συστήματα chatbot έχουν ιδιαίτερα σημαντικό αντίκτυπο στην κοινωνία, έχοντας ενσωματωθεί σε πληθώρα υπηρεσιών και προσφέροντας πολλές φορές σημαντικές διευκολύνσεις προς τους χρήστες τους. Τα σύγχρονα chatbots χρησιμοποιούν συνήθως τεχνητή νοημοσύνη καθώς και τεχνικές επεξεργασίας φυσικής γλώσσας προκειμένου να ανταπεξέλθουν στη δημιουργία μιας χρήσιμης απάντησης στις ερωτήσεις του χρήστη.

Τα τελευταία χρόνια έχουν δημιουργηθεί chatbots που χρησιμοποιούν γενετική τεχνητή νοημοσύνη, όπως για παράδειγμα το ChatGPT, τα οποία έχουν την ικανότητα να καταλαβαίνουν πολύ καλύτερα την ανθρώπινη γλώσσα σε σχέση με προηγούμενες υλοποιήσεις. Τα chatbot αυτά μπορούν να απαντήσουν σε πολύ πιο σύνθετες ερωτήσεις αλλά και να προσαρμοστούν στο ύφος της εκάστοτε συνομιλίας, πετυχαίνοντας τόσο μια καλύτερη εμπειρία για τον χρήστη όσο και την πιο πλούσια παροχή χρήσιμης πληροφορίας [1].

Το Εργαστήριο Συστημάτων Υποστήριξης Αποφάσεων και Διοίκησης του Εθνικού Μετσόβιου Πολυτεχνείου συμμετέχει σε διάφορα ερευνητικά έργα, από τα οποία έχει προκύψει μια σημαντική ποσότητα δεδομένων. Τα έγγραφα αυτά είναι συχνά αξιοσημείωτου μεγέθους και η εύρεση πληροφορίας σε αυτά είναι πολλές φορές μια χρονοβόρα διαδικασία.

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι η κατασκευή ενός chatbot το οποίο θα μπορεί βασιζόμενο στις πληροφορίες που περιέχονται στα παραπάνω έγγραφα να παρέχει πληροφορίες προς τον τελικό του χρήστη. Για παράδειγμα, ο χρήστης θα μπορεί να μάθει πληροφορίες σχετικά με τα έργα που αντιστοιχεί το εκάστοτε έγγραφο, όπως το ποιος δούλεψε σε κάθε κομμάτι ενός έργου αλλά και το πώς πραγματοποιήθηκε η εκάστοτε εργασία.

Χρησιμοποιώντας το chatbot αυτό, οι χρήστες θα μπορούν να ανακτούν την επιθυμητή πληροφορία πολύ πιο γρήγορα, καθώς δεν θα χρειάζεται πλέον να την αναζητούν μεταξύ ενός μεγάλου αριθμού εγγράφων μεγέθους αρκετών σελίδων. Με αυτόν τον τρόπο, βελτιώνεται η διαδικασία αναζήτησης πληροφορίας σχετικά με τα έργα αυτά αλλά και η συνολική εμπειρία του χρήστη κατά τη διάρκεια της αναζήτησης.

Η δομή της υπόλοιπης εργασίας είναι διαχωρισμένη στα εξής παρακάτω κεφάλαια:

Κεφάλαιο 2: Θεωρητικό Υπόβαθρο
Κεφάλαιο 3: Chatbots
Κεφάλαιο 4: Σύνολο δεδομένων
Κεφάλαιο 5: Μεθοδολογία
Κεφάλαιο 6: Αποτελέσματα
Κεφάλαιο 7: Μελλοντική Εργασία / Συμπεράσματα

2. Θεωρητικό Υπόβαθρο

Τα συστήματα chatbot είναι συχνά αρκετά πολύπλοκα και χρησιμοποιούν τεχνικές από διάφορα πεδία ώστε να πετύχουν τον στόχο τους. Με το πέρασμα των χρόνων, η ανακάλυψη ολοένα και πιο αποδοτικών αλγορίθμων και τεχνικών επιτρέπουν στα συστήματα αυτά να βελτιώνουν τις επιδόσεις τους, αυξάνοντας τις διαθέσιμες επιλογές του σχεδιαστή ενός τέτοιου συστήματος. Στο κεφάλαιο αυτό παρουσιάζουμε έννοιες, τεχνικές και αλγορίθμους που κρίνονται χρήσιμοι για την ορθή κατανόηση της παρούσας εργασίας αλλά και την ενημέρωση του αναγνώστη για τις μεθόδους που χρησιμοποιούν τα συστήματα αυτά.

2.1. Μηχανική Μάθηση

Στην παρούσα ενότητα, γίνεται μια εισαγωγή στο πεδίο της μηχανικής μάθησης. Ταυτόχρονα, παρουσιάζονται ορισμένοι αλγόριθμοι που εντάσσονται στο πεδίο της μηχανικής μάθησης.

2.1.1 Ορισμός

Η μηχανική μάθηση αποτελεί ένα συνεχώς αναπτυσσόμενο πεδίο. Έχει συμβάλλει σημαντικά στα πεδία της πληροφορικής και της τεχνητής νοημοσύνης αλλά και πολλών άλλων επιστημών. Έχει επηρεαστεί επίσης και από διάφορες άλλες επιστήμες και τα αποτελέσματα που οι τελευταίες έχουν προσφέρει. Όσον αφορά τον στόχο της, η μηχανική μάθηση αποσκοπεί στο να επιτρέψει στην τεχνητή νοημοσύνη ή γενικότερα σε ένα πρόγραμμα να μάθει, με χρήση συγκεκριμένων τεχνικών ή την παροχή κάποιας πληροφορίας, να βελτιώνει τα αποτελέσματα που παράγει. Ο στόχος αυτός μπορεί να παρομοιαστεί με το πως ένας άνθρωπος αξιοποιεί τις εμπειρίες του ούτως ώστε να προσαρμόσει την συμπεριφορά του [2], [3].

2.1.2 Είδη Μηχανικής Μάθησης

Η διαδικασία με την οποία ένα μοντέλο μηχανικής μάθησης μπορεί να αποκτήσει γνώση από τις πληροφορίες που επεξεργάζεται δεν είναι μοναδική. Όπως ένας άνθρωπος είναι σε θέση να μάθει με πολλούς διαφορετικούς τρόπους από το περιβάλλον του, έτσι και ένα μοντέλο μπορεί να αποκτήσει γνώση με εφαρμογή διαφορετικών τεχνικών. Οι αλγόριθμοι και οι τεχνικές που χρησιμοποιούνται στο πεδίο της μηχανικής μάθησης μπορούν να διακριθούν σε τρεις κύριες κατηγορίες που θα εξηγήσουμε παρακάτω:

<u>Επιβλεπόμενη Μάθηση</u>

Κατά την επιβλεπόμενη μάθηση, χρησιμοποιείται ένα σύνολο δεδομένων εισόδων και των αντίστοιχων αναμενόμενων εξόδων που χρειάζεται να έχει ο αλγόριθμος / μοντέλο. Χρησιμοποιώντας το σύνολο αυτό, οι αλγόριθμοι επιβλεπόμενης μηχανικής μάθησης μπορούν σταδιακά να προσαρμόσουν τις παραμέτρους τους, ώστε να παρά-

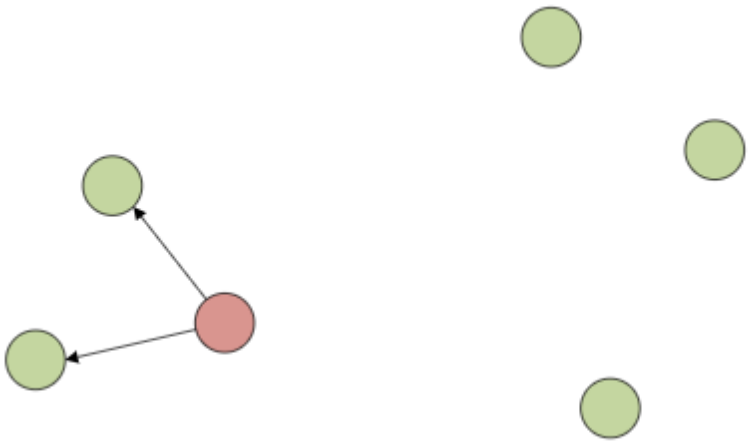
γουν εξόδους ανάλογες με αυτές που επιθυμούμε. Καθώς το σύνολο εκπαίδευσης περιέχει τις πραγματικές τιμές που καλείται ο αλγόριθμος / μοντέλο να μάθει, η κατηγορία μάθησης αυτή αναφέρεται ως επιβλεπόμενη

Προκειμένου να αξιολογήσουν την επίδοση τους, οι αλγόριθμοι αυτοί χρησιμοποιούν μια συνάρτηση σφάλματος, όπως για παράδειγμα την απόκλιση της παραγόμενης εξόδου από τον αλγόριθμο από αυτή της αναμενόμενης εξόδου του συνόλου δεδομένων. Όταν η συνάρτηση αυτή ικανοποιεί ορισμένες συνθήκες, τότε θεωρούμε πως ο αλγόριθμος έχει προσαρμοστεί επαρκώς στο σύνολο δεδομένων στο οποίο εκπαιδεύτηκε.

Συνεπώς, αν έχουμε στη διάθεση μας ένα ικανό μέγεθος πληροφορίας και χρόνου εκπαίδευσης, τα μοντέλα αυτά έχουν την δυνατότητα να συσχετίσουν μετά το πέρας της εκπαίδευσης αρκετά ικανοποιητικά τις μελλοντικές άγνωστες εισόδους με την αναμενόμενη έξοδο [2], [4].

Ο αλγόριθμος των κ-πλησιέστερων γειτόνων

Στο σημείο αυτό κρίνεται σκόπιμο να παρουσιάσουμε έναν σημαντικό αλγόριθμο επιβλεπόμενης μάθησης, που ονομάζεται **αλγόριθμος κ-πλησιέστερων γειτόνων**. Σε ένα πρόβλημα ομαδοποίησης μπορούμε να εφαρμόσουμε τον αλγόριθμο αυτό ως εξής: έχουμε αρχικά ένα σύνολο δειγμάτων όπου το καθένα από αυτά συνοδεύεται από μια τιμή που αντιπροσωπεύει την ομάδα στην οποία ανήκει. Στη συνέχεια, για να προσδιορίσουμε την ομάδα ενός νέου δείγματος, την οποία δεν γνωρίζουμε, αξιοποιούμε τον εξής κανόνα: προσδιορίζουμε πρώτα **κ** δείγματα από το αρχικό σύνολο των οποίων οι αναπαραστάσεις σε έναν διανυσματικό χώρο είναι οι εγγύτερες στην αναπαράσταση του υπό εξέταση δείγματος, με βάση μια μετρική απόστασης. Ο κανόνας αυτός καλείται *κανόνας του πλησιέστερου γείτονα* για κ ίσο με 1 ή *κανόνας των κ-πλησιέστερων γειτόνων*, για μεγαλύτερες τιμές του κ [5], [6].



Εικόνα 1: Ο κανόνας των κ-Πλησιέστερων Γειτόνων, για κ=2

Έχοντας διακρίνει τα κ πλησιέστερα δείγματα ως προς το υπό εξέταση δείγμα, των οποίων την κλάση γνωρίζουμε, μπορούμε έπειτα να επιλέξουμε ως πρόβλεψη της ομάδας του υπό εξέταση δείγματος, αυτή στην οποία ανήκει η πλειοψηφία των κ πλη-

σιέστερων του δειγμάτων. Η παράμετρος κ μπορεί να λάβει ακέραιες τιμές μεγαλύτερες η ίσες του 1 και προσδιορίζει, όπως εξηγήσαμε, το πλήθος των πλησιέστερων δειγμάτων που χρειάζεται να εντοπίσει και να αξιοποιήσει ο αλγόριθμος.

Μη Επιβλεπόμενη Μάθηση

Στην μη επιβλεπόμενη μάθηση το σύνολο δεδομένων με το οποίο εκπαιδεύεται ο αλγόριθμος / μοντέλο αποτελείται μόνο από εισόδους και σε αντίθεση με την επιβλεπόμενη μάθηση, δεν περιέχονται σε αυτό και οι αναμενόμενες έξοδοι. Ο αλγόριθμος / μοντέλο που ακολουθεί αυτή τη μέθοδο μάθησης αποσκοπεί στο να αναγνωρίσει μοτίβα καθώς και διαφορές και ομοιότητες μεταξύ της πληροφορίας που περιέχεται στο σύνολο δεδομένων, χωρίς την ανάγκη να υπάρχει κάποιος μηχανισμός που να επιβλέπει την διαδικασία [2], [4].

Ενισχυτική Μάθηση

Η ενισχυτική μάθηση αποτελεί μια διαφορετική μέθοδο μάθησης όπου το μοντέλο μαθαίνει αλληλεπιδρώντας με το περιβάλλον του και όχι με χρήση πρότερης πληροφορίας. Προκειμένου το μοντέλο να πετύχει την επιθυμητή λειτουργία στο τέλος της διαδικασίας, εφαρμόζουμε κάποιο σύστημα κριτικής το οποίο επιβραβεύει το μοντέλο μας όταν λαμβάνει τις κατάλληλες αποφάσεις και αντίστροφα το αποθαρρύνει όταν λαμβάνει λανθασμένες [2], [4].

2.1.3 Νευρωνικά Δίκτυα

Τα νευρωνικά δίκτυα αποτελούν ένα είδος μοντέλων μηχανικής μάθησης. Η λειτουργία τους είναι βασισμένη σε μεγάλο βαθμό στον τρόπο με τον οποίο λειτουργεί ο ανθρώπινος εγκέφαλος. Πιο συγκεκριμένα, ένα νευρωνικό δίκτυο απαρτίζεται από τεχνητούς νευρώνες, οι οποίοι αποτελούν μονάδες που εκτελούν ορισμένες απλές διαδικασίες και είναι συνδεδεμένοι μεταξύ τους με συχνά σύνθετο τρόπο. Όπως παρατηρείται και στον ανθρώπινο εγκέφαλο, ένα νευρωνικό δίκτυο είναι ικανό να αποκτήσει γνώσεις από το περιβάλλον του μαθαίνοντας μέσω μιας συγκεκριμένης διαδικασίας, γνωστή ως αλγόριθμος μάθησης. Κατά τη διάρκεια της μάθησης, προκειμένου να αποθηκεύσει την γνώση που αποκτά, το νευρωνικό δίκτυο αξιοποιεί τις συνδέσεις μεταξύ των νευρώνων του. Αλλάζοντας τα βάρη που τις συνοδεύουν ή αναπροσαρμόζοντας τις συνδέσεις αυτές με κατάλληλο τρόπο, ώστε να αντικατοπτρίζουν την γνώση που έλαβε ως είσοδο, το νευρωνικό δίκτυο προσπαθεί να πετύχει τον στόχο που ορίζεται από τον αλγόριθμο μάθησης [4].

Όπως αναφέραμε παραπάνω, οι νευρώνες που χρησιμοποιεί ένα νευρωνικό δίκτυο επεξεργάζονται τα δεδομένα που αυτό δέχεται ως είσοδο. Αποτελούν ένα βασικό κομμάτι για πληθώρα αρχιτεκτονικών με την οποία κατασκευάζονται τέτοια δίκτυα. Ένας τρόπος με τον οποίο μπορούμε να ορίσουμε τη δομή ενός νευρώνα [4] παρουσιάζεται συνοπτικά παρακάτω:

- Διασυνδέσεις: Οι διασυνδέσεις αποτελούν το σύνολο των εισόδων που συνδέονται με τον εκάστοτε νευρώνα. Όπως είδαμε, κάθε τέτοια σύνδεση συνοδεύε-

ται από ένα βάρος, το οποίο είναι μια αριθμητική τιμή που εκφράζει την σημαντικότητα της εισόδου που αντιστοιχεί στη διασύνδεση, κάτι που επιτυγχάνεται μέσω του πολλαπλασιασμού του βάρους με την αντίστοιχη είσοδο.

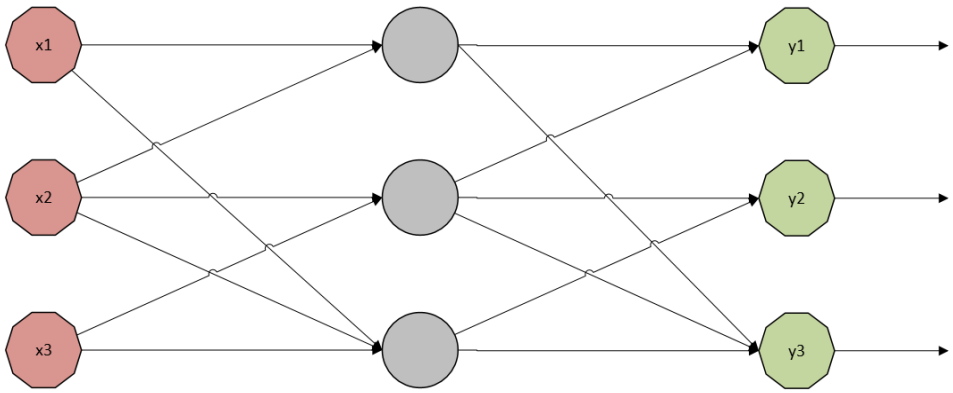
- Αθροιστής: Όπως υποδηλώνει και η ονομασία του, ο αθροιστής αθροίζει το σύνολο των γινομένων των εισόδων του νευρώνα με τα αντίστοιχα βάρη των διασυνδέσεων του.

- Συνάρτηση Ενεργοποίησης: Η συνάρτηση ενεργοποίησης εφαρμόζεται ούτως ώστε να περιορίσει και να καθορίσει την μορφή της εξόδου του νευρώνα ενός νευρωνικού δικτύου. Παραδείγματα γνωστών συναρτήσεων ενεργοποίησης αποτελούν η συνάρτηση κατωφλίου, η σιγμοειδής συνάρτηση, η συνάρτηση υπερβολικής εφαπτομένης ή η συνάρτηση ReLU.

Στο σημείο αυτό κρίνεται σκόπιμο να παρουσιάσουμε εν συντομία ορισμένες αρχιτεκτονικές νευρωνικών δικτύων που έχουν χρησιμοποιηθεί ευρέως. Οι νευρώνες στις αρχιτεκτονικές αυτές οργανώνονται σε ομάδες που ονομάζουμε επίπεδα.

Νευρωνικά Δίκτυα Πρόσθιας Τροφοδότησης

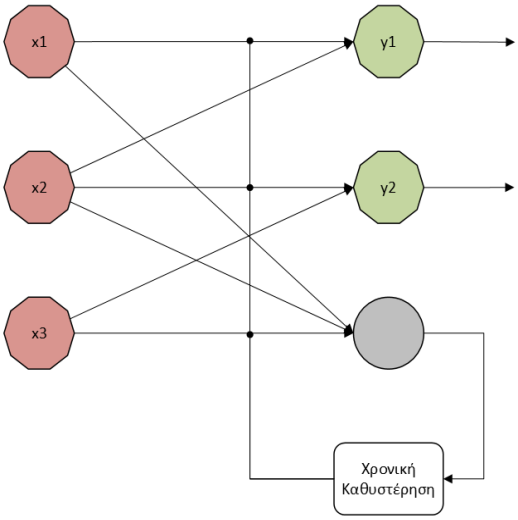
Ένα νευρωνικό Δίκτυο το οποίο παρουσιάζει ένα επίπεδο εισόδου και ένα επίπεδο εξόδου, όπου μόνο το επίπεδο εισόδου είναι συνδεδεμένο με το επίπεδο εξόδου, αποκαλείται Δίκτυο Πρόσθιας Τροφοδότησης Ενός Επιπέδου. Ονομάζεται έτσι, καθώς το δίκτυο συνδέεται μόνο από την είσοδο προς την έξοδο και το μοναδικό επίπεδο στο οποίο υπολογίζουμε κάποια πληροφορία είναι αυτό της εξόδου. Αν μεταξύ των επιπέδων εισόδου και εξόδου παρεμβάλλονται επιπλέον και άλλα επίπεδα , τα οποία αποκαλούμε κρυφά, τότε ένα τέτοιο δίκτυο ονομάζεται Δίκτυο Πρόσθιας Τροφοδότησης Πολλών Επιπέδων. Στα δίκτυα αυτά το 1ο από τα κρυφά επίπεδα δέχεται ως είσοδο τα σήματα που εξάγονται από το επίπεδο εισόδου. Αντίστοιχα, κάθε επιπλέον κρυφό επίπεδο δέχεται ως είσοδο την έξοδο του προηγούμενου του, και τέλος το στρώμα εξόδου λαμβάνει ως είσοδο τα σήματα που εξάγονται από το τελευταίο κρυφό επίπεδο [4], [7].



Εικόνα 2: Παράδειγμα Δικτύου Πρόσθιας Τροφοδότησης με Ένα Επίπεδο Εισόδου, Ένα Κρυφό Επίπεδο και Ένα Επίπεδο Εξόδου

Αναδρομικά Νευρωνικά Δίκτυα

Εν αντιθέσει με τα Νευρωνικά Δίκτυα πρόσθιας τροφοδότησης, σε ένα Αναδρομικό Νευρωνικό Δίκτυο, εκτός από την ύπαρξη συνδέσεων των εισόδων με κατεύθυνση προς τις εξόδους του δικτύου, είναι δυνατό να υπάρχουν συνδέσεις όπου τροφοδοτούνται οι έξοδοι ενός επιπέδου του δικτύου προς τις εισόδους του. Οι αναδρομές αυτές των εξόδων παρουσιάζουν μια χρονική καθυστέρηση ούτως ώστε να επιτραπεί στις παλαιές εισόδους του δικτύου να επηρεάσουν την συμπεριφορά του τελευταίου για την τρέχουσα είσοδο του [4], [7].



Εικόνα 3: Παράδειγμα Αναδρομικού Νευρωνικού Δικτύου

Αναδρομικά Νευρωνικά Δίκτυα Μακράς και Βραχείας Μνήμης

Τα Αναδρομικά Νευρωνικά Δίκτυα Μακράς και Βραχείας Μνήμης αποτελούν μια φημισμένη αρχιτεκτονική δικτύων, τα οποία αναπτύχθηκαν στοχεύοντας στο να έχουν την δυνατότητα να διατηρούν δεδομένα στη μνήμη τους για αρκετή χρονική διάρκεια ώστε να καταπολεμηθούν ορισμένες αδυναμίες που παρουσίαζαν τα αρχικά Αναδρομικά Νευρωνικά Δίκτυα. Για να πετύχει το παραπάνω, ένα τέτοιο δίκτυο απαρτίζεται από στοιχεία που ονομάζονται κελιά μνήμης και από ένα σύνολο από πύλες.

Αναλυτικότερα, το κελί μνήμης αποτελεί το στοιχείο μακράς μνήμης του δικτύου και διατηρεί την πληροφορία του μεταξύ χρονικών στιγμών. Όσον αφορά τις πύλες, μπορούμε να έχουμε 3 είδη πυλών: την πύλη λήθης, που είναι υπεύθυνη για το αν η πληροφορία που είναι αποθηκευμένη τη τρέχουσα χρονική στιγμή στο κελί μνήμης θα αντιγραφεί και στην επόμενη χρονική στιγμή ή θα διαγραφθεί, την πύλη εισόδου, η οποία αποφασίζει αν η πληροφορία που περιέχεται στο κελί μνήμης θα υποστεί κάποια ενημέρωση ή δε θα μεταβληθεί τη τρέχουσα χρονική στιγμή και τέλος μια πύλη εξόδου, που αποφασίζει το αν θα πραγματοποιηθεί μεταφορά της πληροφορίας του κελιού μνήμης προς την μνήμη βραχείας διάρκειας του δικτύου [7], [8].

2.2 Επεξεργασία Φυσικής Γλώσσας

Στην ενότητα αυτή παρουσιάζουμε μια εισαγωγή στο πεδίο της Επεξεργασίας Φυσικής Γλώσσας και ορισμένες εφαρμογές που έχει η τελευταία στην επιστήμη των υπολογιστών. Στη συνέχεια, αναλύουμε την έννοια της συμβολοποίησης ως προς τον στόχο της και τον τρόπο με τον οποίο αυτή επιτυγχάνεται. Έπειτα, δίνουμε έναν σύντομο ορισμό των Γλωσσικών Μοντέλων και των Ενσωματώσεων Λέξεων, αναφέροντας ορισμένα παραδείγματα και σχετικές πληροφορίες για καθένα από αυτά.

2.2.1 Εισαγωγή και Ορισμός

Η Επεξεργασία Φυσικής Γλώσσας αποτελεί ένα πεδίο με εφαρμογές σε αρκετά επιστημονικά πεδία. Ειδικότερα, όσον αφορά την επιστήμη των υπολογιστών, με τον όρο επεξεργασία φυσικής γλώσσας εννοούμε συνήθως τον τρόπο με τον οποίο ένας υπολογιστής χειρίζεται και επεξεργάζεται την ανθρώπινη γλώσσα ώστε να μπορέσει να καταλήξει σε συμπεράσματα, να αποκτήσει γνώση μέσα από αυτήν αλλά και να επικοινωνήσει με το περιβάλλον του. Προφανώς, υπάρχουν αρκετές περιπτώσεις πέρα από τις παραπάνω όπου κρίνεται χρήσιμο ένας υπολογιστής να μπορεί να επεξεργαστεί φυσική γλώσσα [7].

Οι εφαρμογές που έχουν αναπτυχθεί τα τελευταία χρόνια καθιστούν φανερό ότι το πεδίο αυτό έχει ιδιαίτερη σημασία στην επιστήμη των υπολογιστών, κάτι που επιβεβαιώνεται παρατηρώντας και την συνεχόμενη εισαγωγή νέων τεχνικών και αλγορίθμων, οι οποίοι καθιστούν την επεξεργασία φυσικής γλώσσας από υπολογιστικά συστήματα ολοένα και πιο αποδοτική σε ένα αυξανόμενο πλαίσιο δραστηριοτήτων. Ενδεικτικά, κάποιες εφαρμογές Επεξεργασίας Φυσικής Γλώσσας των οποίων οι επιδόσεις ολοένα και αυξάνονται με χρήση τεχνητής νοημοσύνης, παρουσιάζονται εν συντομία παρακάτω:

<u>Ανάκτηση Πληροφορίας</u>

Σε πολλές περιπτώσεις σε συστήματα αποθήκευσης δεδομένων έχουμε διαθέσιμη μια μεγάλη ποσότητα πληροφορίας και δεδομένων, αλλά κάποιος χρήστης δεν χρειάζεται πάντα όλη την διαθέσιμη πληροφορία. Οι εφαρμογές ανάκτησης πληροφορίας έχουν ως στόχο την αναζήτηση των πιο σχετικών δεδομένων με βάση ένα ερώτημα. Το ερώτημα αυτό δίνεται από τον χρήστη / τελικό δέκτη προς ένα τέτοιο σύστημα. Έπειτα, το σύστημα λαμβάνει το ερώτημα αυτό, το επεξεργάζεται και επιστρέφει στον χρήστη την πληροφορία την οποία θεωρεί πως είναι η πιο σχετική όσον αφορά το ερώτημα του χρήστη [9]. Η χρήση μοντέλων και αλγορίθμων μηχανικής μάθησης έχουν συμβάλει στην αισθητή βελτίωση τέτοιων συστημάτων.

<u>Σημασιολογική Ομοιότητα Κειμένου:</u>

Ως τη διαδικασία σημασιολογικής ομοιότητας κειμένου ορίζουμε το πρόβλημα όπου ένα υπολογιστικό σύστημα καλείται να αξιολογήσει την ομοιότητα μεταξύ δύο διαφορετικών κειμένων. Προκειμένου να εκφράσει το ποσοστό της ομοιότητας μεταξύ των κειμένων, το σύστημα υπολογίζει κάποια μετρική που αντιπροσωπεύει το ποσοστό αυτό και την επιστρέφει στον χρήστη [10]. Όπως θα δούμε αργότερα, και αυτή η εργασία έχει σημειώσει σημαντική βελτίωση με χρήση μοντέλων μηχανικής μάθησης.

Απάντηση Ερωτήσεων

Τα συστήματα απάντησης ερωτήσεων, όπως υποδηλώνει και η ονομασία τους, δέχονται μια ερώτηση από τον χρήστη τους και έχουν ως στόχο την απάντηση αυτής της ερώτησης, συνήθως με μορφή κειμένου. Για την παραγωγή της απάντησης ενός τέτοιου συστήματος είναι δυνατόν να χρησιμοποιηθούν αρκετές τεχνικές, καθεμία από τις οποίες παρουσιάζει ορισμένα πλεονεκτήματα σε σχέση με τις υπόλοιπες.

2.2.2 Συμβολοποίηση (Tokenization)

Ως συμβολοποίηση ορίζουμε την διαδικασία κατά την οποία χωρίζουμε ένα κείμενο σε μικρές μονάδες τις οποίες καλούμε σύμβολα. Τα σύμβολα αυτά είναι συνήθως λέξεις, όμως είναι αρκετές φορές δυνατόν να είναι επιθυμητό να χωρίσουμε μια λέξη σε παραπάνω από ένα σύμβολο ή να θεωρήσουμε περισσότερες από μία ως ένα μόνο σύμβολο.

Υπάρχουν διάφορες μέθοδοι συμβολοποίησης, αλλά μπορούμε να διακρίνουμε δύο μεγάλες κατηγορίες [11]:

- Μέθοδοι όπου χρησιμοποιούμε ορισμένους κανόνες για να διαχωρίσουμε το κείμενο, όπως για παράδειγμα τον διαχωρισμό μεταξύ κενών και σημείων στίξης. Μια τέτοια μέθοδος έχει το πλεονέκτημα του ότι συνήθως παρουσιάζει μικρό υπολογιστικό κόστος.

- Μέθοδοι όπου χωρίζουμε τις λέξεις του κειμένου σε υπό-λέξεις. Πιο συγκεκριμένα, στις μεθόδους αυτές έχουμε συνήθως αρχικά την δημιουργία ενός λεξιλογίου, μέσω της χρήσης ενός αλγορίθμου για την εξαγωγή λεκτικών μονάδων από ένα σύνολο κειμένων, και στη συνέχεια χρησιμοποιούμε το λεξιλόγιο που προέκυψε ώστε να διασπάσουμε μελλοντικά κείμενα σε σύμβολα. Αυτές οι μέθοδοι χρησιμοποιούνται συχνά σε εφαρμογές όπου δεν είναι επιθυμητό να προκύψουν τμήματα στο κείμενο τα οποία δεν μπορούν να αντιστοιχηθούν σε κάποιο σύμβολο.

Στο σημείο αυτό κρίνεται σκόπιμο να παρουσιάσουμε έναν αλγόριθμο συμβολοποίησης, ο οποίος χρησιμοποιείται σε μοντέλα μηχανικής μάθησης που θα αναλύσουμε στις ενότητες 2.3 και 2.4, ώστε να γίνει πιο κατανοητή η διαδικασία που ακολουθείται από έναν τέτοιο αλγόριθμο.

Συμβολοποίηση Βασισμένη σε Κωδικοποίηση Byte Pair

Η **κωδικοποίηση Byte Pair** αποτελεί έναν ιδιαίτερα δημοφιλή αλγόριθμο συμπίεσης δεδομένων. Στην κωδικοποίηση Byte Pair μια σειρά χαρακτήρων συμπιέζεται εφαρμόζοντας τον εξής κανόνα: υπολογίζουμε το πλήθος εμφάνισης κάθε ζεύγους συνεχόμενων χαρακτήρων και αντικαθιστούμε το ζεύγος το οποίο εμφανίζεται τις περισσότερες φορές με ένα νέο χαρακτήρα που δεν εμφανίζεται στην αρχική σειρά των χαρακτήρων. Η εφαρμογή του κανόνα αυτού επαναλαμβάνεται έως ότου να μην είναι πλέον δυνατό να αντικαταστήσουμε κάποιο ζεύγος χαρακτήρων, δηλαδή όλα τα ζεύγη να εμφανίζονται μια μόνο φορά στη τελική σειρά χαρακτήρων.

Η **συμβολοποίηση** που βασίζεται στην **κωδικοποίηση Byte Pair**, στηρίζεται στον κανόνα του αντίστοιχου αλγορίθμου συμπίεσης που εξηγήσαμε παραπάνω. Πιο συγκεκριμένα, ο αλγόριθμος συμβολοποίησης αυτός χρησιμοποιεί ένα αρχικό σύνολο λέξεων που έχει προκύψει, για παράδειγμα, από την διαδικασία διαχωρισμού ενός κειμένου σε επίπεδο λέξεων. Αρχικά το σύνολο των λέξεων αυτό διαχωρίζεται σε επίπεδο χαρακτήρων. Έπειτα, υπολογίζεται το πλήθος εμφάνισης κάθε ζεύγους χαρακτήρων στο σύνολο των διαχωρισμένων πλέον λέξεων και στη συνέχεια, ακολουθώντας παρόμοια λογική με τον κανόνα του αλγορίθμου συμπίεσης, το ζεύγος με την μεγαλύτερη συχνότητα εμφάνισης αντικαθίσταται με ένα νέο σύμβολο που αποτελεί την συνένωση των χαρακτήρων του ζεύγους που επιλέχθηκε. Η διαδικασία αυτή μπορεί να επαναληφθεί όσες φορές είναι επιθυμητό προκειμένου να εξάγουμε ένα ικανοποιητικό πλήθος τέτοιων συμβόλων. Μετά το πέρας της εκτέλεσης του πρώτου σταδίου που μόλις αναφέραμε, ο αλγόριθμος έχει ανανεώσει το σύνολο των χαρακτήρων, που αποτελεί το λεξιλόγιο του, έχοντας προσθέσει όλα τα νέα σύμβολα που προέκυψαν. Έτσι, ο αλγόριθμος αυτός μπορεί να χρησιμοποιηθεί αργότερα για την πιο αποτελεσματική εφαρμογή συμβολοποίησης νέων ακολουθιών κειμένου αξιοποιώντας τους νέους κανόνες που έχουν προστεθεί στο λεξιλόγιο του, αφού πρώτα μια τέτοια ακολουθία διασπαστεί διαδοχικά σε επίπεδο λέξεων και ύστερα χαρακτήρων, ώστε να είναι δυνατή η εφαρμογή των κανόνων [12].

Εκτός της παραπάνω υλοποίησης έχουν προταθεί και χρησιμοποιηθεί πολλοί ακόμα αλγόριθμοι συμβολοποίησης και παραλλαγές για καθένα από αυτούς. Ανάλογα με την εργασία για την οποία επιθυμούμε να τους αξιοποιήσουμε, μπορούμε να επιλέξουμε αυτόν τον οποίο αναμένουμε ή παρατηρούμε ότι αυξάνει την ποιότητα των αποτελεσμάτων μας.

2.2.3 Γλωσσικά Μοντέλα

Στην επιστήμη της Επεξεργασίας Φυσικής Γλώσσας, ένα Γλωσσικό μοντέλο ορίζεται ως ένα μοντέλο το οποίο είναι ικανό να αντιστοιχήσει την πιθανότητα εμφάνισης ενός στοιχείου κειμένου ως το επόμενο μέρος ενός γλωσσικού κειμένου ή γενικότερα να αντιστοιχήσει μια πιθανότητα συσχέτισης σε στοιχεία κειμένου με βάση κάποιο κριτήριο. Τα μοντέλα αυτά έχουν χρησιμοποιηθεί για πληθώρα εφαρμογών με σημαντική επιτυχία, όπως για παράδειγμα στην αυτόματη συμπλήρωση κειμένων, στην διόρθωση ορθογραφικών λαθών και στην απάντηση ερωτήσεων. Ωστόσο, δεν είναι δυνατόν να κάνουν πάντα σωστές προβλέψεις λόγω της σύνθετης λειτουργίας των φυσικών γλωσσών που καθιστούν δύσκολη την επιλογή μόνο μιας λέξης ή φράσης ως την βέλτιστη επιλογή ή απάντηση στο εκάστοτε πρόβλημα που αντιμετωπίζουμε [7], [11].

Στη συνέχεια, αξίζει να παρουσιάσουμε ένα ευρέως γνωστό είδος γλωσσικών μοντέλων, τα n-γραμμικά μοντέλα λέξεων (n-grams), ώστε να γίνει πιο κατανοητός ο τρόπος με τον οποίο λειτουργεί ένα γλωσσικό μοντέλο.

Γενικά, είναι λογικό να υποθέσουμε ότι η πιθανότητα εμφάνισης μιας λέξης, έστω x, ως επόμενη σε ένα κείμενο t θα μπορούσε να υπολογιστεί ως η πιθανότητα του να έχουμε εμφάνιση κάθε λέξης του κειμένου t με τη σειρά που εμφανίζονται στο κείμενο και τέλος να έχουμε εμφάνιση της λέξης x. Όμως η διαδικασία αυτή είναι ακριβή

υπολογιστικά όσο αυξάνεται το μέγεθος του κειμένου αλλά και το σύνολο των πιθανών λέξεων που μπορούμε να συναντήσουμε [7].

Τα n-γραμμικά μοντέλα επιλύουν το παραπάνω πρόβλημα, δεχόμενα ως παραδοχή ότι η πιθανότητα μια λέξη να είναι η επόμενη σε ένα κείμενο μπορεί να υπολογιστεί με ικανοποιητική προσέγγιση, χρησιμοποιώντας μόνο τις n-1 τελευταίες λέξεις που παρουσιάζονται ήδη στο κείμενο. Όσο μεγαλώνουμε το παράθυρο προηγούμενων λέξεων για τον υπολογισμό της πιθανότητας, πετυχαίνουμε ολοένα και καλύτερα αποτελέσματα αυξάνοντας όμως τις υπολογιστικές απαιτήσεις του μοντέλου [7], [11].

2.2.4 Ενσωματώσεις λέξεων

Ως ενσωματώσεις λέξεων ορίζουμε ένα διάνυσμα μικρών σχετικά διαστάσεων (της τάξης των μερικών εκατοντάδων έως μερικών χιλιάδων) όπου οι περισσότερες τιμές του δεν είναι μηδενικές και αντιστοιχεί σε μια αναπαράσταση μιας λέξης ή ενός συμβόλου. Τα διανύσματα αυτά έχουν την ιδιότητα του ότι διατηρούν την σχετικότητα τους, μοιάζουν δηλαδή μεταξύ τους, σε περίπτωση που οι λέξεις στις οποίες αντιστοιχούν έχουν κάποια συσχέτιση και εκείνες μεταξύ τους. Έχει επίσης παρατηρηθεί ότι τα διανύσματα αυτά είναι ικανά να εκφράσουν και πιο σύνθετες και περίπλοκες συνδέσεις μεταξύ λέξεων πέρα από την εξάρτηση τους ως προς την θέση τους σε ένα κείμενο [7], [13].

Έχουν αναπτυχθεί αρκετοί αλγόριθμοι οι οποίοι υπολογίζουν ενσωματώσεις λέξεων πετυχαίνοντας πολύ ικανοποιητικά αποτελέσματα. Σε αυτό το σημείο κρίνεται σκόπιμο να παρουσιάσουμε έναν από τους πιο γνωστούς αλγορίθμους αυτής της κατηγορίας:

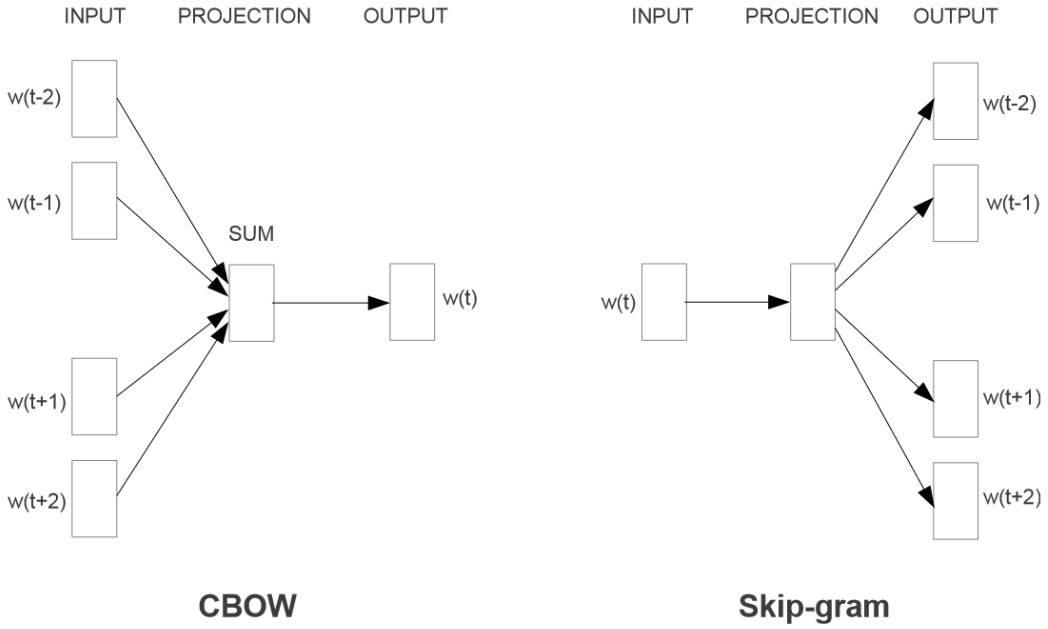
Word2Vec

Στην εργασία τους, οι Mikolov et al.[13] παρουσιάζουν δύο είδη αρχιτεκτονικών μοντέλων με τις οποίες μπορεί κάποιος να υπολογίσει ενσωματώσεις λέξεων:

- **Μοντέλα Συνεχόμενων Σάκων Λέξεων**: Τα μοντέλα συνεχόμενων σάκων λέξεων λαμβάνουν ως είσοδο μια σειρά από λέξεις, κωδικοποιημένες σε κατάλληλη μορφή. Στη συνέχεια κάθε τέτοια κωδικοποίηση κατευθύνεται στο επόμενο επίπεδο, που καλείται επίπεδο προβολής, όπου καθεμιά από αυτές πολλαπλασιάζεται με έναν κοινό πίνακα και έπειτα υπολογίζεται ο μέσος όρος των διανυσμάτων που προέκυψαν. Τέλος, χρησιμοποιείται ένα επίπεδο εξόδου όπου η έξοδος του επιπέδου προβολής μετατρέπεται και πάλι σε ένα διάνυσμα διαστάσεων ίδιων με αυτές των διανυσμάτων του επιπέδου εισόδου. Προκειμένου τα μοντέλα αυτά να εκπαιδευτούν, προστίθεται ένα επίπεδο για τη μετατροπή της εξόδου τους σε μια κατανομή πιθανοτήτων και ορίζεται ως στόχος εκπαίδευσης η σωστή πρόβλεψη μιας λέξης, που βρίσκεται στην μέση μιας πρότασης. Πιο συγκεκριμένα, δίνεται ως είσοδος στο μοντέλο μια ακολουθία *κ* λέξεων που προηγούνται και *κ* λέξεων που έπονται της λέξης-στόχου, που αποτελούν τα συμφραζόμενα τις τελευταίας, και ζητείται η πρόβλεψη της λέξης στόχου ως έξοδος του. Μετά το πέρας της εκπαίδευσης τους, μπορούμε να εξάγουμε από τα μοντέλα αυτά ενσωματώσεις λέξεων για τις λέξεις στις

οποίες εκπαιδεύτηκαν. Η ποιότητα των ενσωματώσεων εξαρτάται από την επιλογή της παραμέτρου κ αλλά και από το πλήθος των δεδομένων εκπαίδευσης.

- **Μοντέλα Συνεχόμενων Skip-Gram**: Τα μοντέλα συνεχόμενων Skip-Gram παρουσιάζουν αρχιτεκτονική παρόμοια με αυτή των μοντέλων συνεχόμενων σάκων λέξεων. Ωστόσο, διαφέρουν μεταξύ τους όσον αφορά τον τρόπο με τον οποίο εκπαιδεύονται. Ειδικότερα, στα μοντέλα συνεχόμενων Skip-Gram κατά τη διάρκεια της εκπαίδευσης δίνεται ως είσοδος μια λέξη και ζητείται η σωστή πρόβλεψη κ λέξεων που προηγούνται και κ λέξεων που έπονται της λέξης που δόθηκε ως είσοδος, που ουσιαστικά αποτελεί την αντίστροφη διαδικασία από αυτή που περιγράψαμε στα μοντέλα συνεχόμενων σάκων λέξεων.



Εικόνα 4: Η Αρχιτεκτονική Συνεχόμενων Σάκων Λέξεων (Αριστερά) και Η Αρχιτεκτονική Συνεχόμενων Skip Gram (Δεξιά) [13]

Οι δύο αυτές αρχιτεκτονικές αναφέρονται συχνά απλά και ως **Word2Vec**, η οποία αποτελεί την ονομασία ενός συνόλου λογισμικού που αναπτύχθηκε από τους Mikolov et al. [13] και χρησιμοποιεί τις δύο αρχιτεκτονικές αυτές για την εύρεση ενσωματώσεων λέξεων.

2.2.5 Μετρικές Αποστάσεων

Αφού υπολογίσουμε της ενσωματώσεις λέξεων, χρειάζεται να εφαρμόσουμε κάποιο κριτήριο σύγκρισης μεταξύ των διανυσμάτων που έχουν προκύψει, ούτως ώστε να μπορέσουμε να εξάγουμε πληροφορία για τις συσχετίσεις μεταξύ των λέξεων που αυτά αντιπροσωπεύουν. Για τον σκοπό αυτό, χρησιμοποιούνται ευρέως τρεις μετρικές [14], [15]. Ειδικότερα έχουμε:

- **Ευκλείδεια Απόσταση**: Η ευκλείδεια απόσταση αποτελεί μια από τις πιο διάσημες και πιο απλές μετρικές με τις οποίες μπορούμε να χρησιμοποιήσουμε. Η ευκλείδεια απόσταση εκφράζει την απόσταση μεταξύ δύο διανυσμάτων σε έναν

χώρο διαστάσεων. Πιο συγκεκριμένα για δύο διανύσματα **x** και **y** με k συνιστώσες, υπολογίζεται ως εξής:

$$d(\vec{x},\vec{y})_{euc} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + \cdots + (x_k - y_k)^2}$$

(2. 1)

Αξίζει να σημειωθεί ότι η ευκλείδεια απόσταση εξαρτάται σημαντικά από το εύρος των τιμών που μπορούν να λάβουν οι συνιστώσες των διανυσμάτων. Προφανώς, όσο μικρότερη είναι η τιμή της ευκλείδειας απόστασης τόσο περισσότερο όμοια μπορούμε να θεωρήσουμε τα διανύσματα x και y.

- **<u>Ομοιότητα Συνημιτόνου</u>**: Μια άλλη χρήσιμη μετρική αποτελεί η ομοιότητα συνημιτόνου. Για δύο διανύσματα **x** και **y** με k συνιστώσες που σχηματίζουν μεταξύ τους γωνία φ, η μετρική αυτή εκφράζεται ως το πηλίκο του εσωτερικού γινομένου των δύο διανυσμάτων με το γινόμενο των μέτρων των διανυσμάτων αυτών. Δηλαδή μπορούμε να υπολογίσουμε την ομοιότητα συνημιτόνου όπως φαίνεται παρακάτω:

$$d(\vec{x},\vec{y})_{cos} = \frac{\vec{x}\cdot\vec{y}}{\|\vec{x}\|\,\|\vec{y}\|} = cos\varphi$$

(2. 2)

Σε αντίθεση με την ευκλείδεια απόσταση, η ομοιότητα συνημιτόνου εκφράζει μόνο το κατά πόσο τα δύο διανύσματα έχουν παρόμοια κατεύθυνση, καθώς εξαρτάται μόνο από την γωνία μεταξύ των δύο διανυσμάτων. Επιπλέον, η μετρική αυτή μπορεί να λάβει τιμές στο διάστημα [-1,1], με τις τιμές προς το -1 να εκφράζουν ότι τα δύο διανύσματα έχουν σχεδόν αντίθετες κατευθύνσεις, ενώ τιμές κοντά στο 1 να εκφράζουν ότι τα δύο διανύσματα έχουν σχεδόν ίδιες κατευθύνσεις.

- **<u>Εσωτερικό Γινόμενο</u>**: Μια Τρίτη μετρική που μπορούμε να χρησιμοποιήσουμε είναι το εσωτερικό γινόμενο μεταξύ των διανυσμάτων. Ειδικότερα, αν θεωρήσουμε και πάλι τα διανύσματα x και y k διαστάσεων και την γωνία φ που αυτά σχηματίζουν, το εσωτερικό τους γινόμενο εκφράζεται ως:

$$\vec{x}\cdot\vec{y} = \sum_{n=0}^{k} x_n y_n = \|\vec{x}\|\,\|\vec{y}\| cos\varphi = \|\vec{x}\|\,\|\vec{y}\|\, d(\vec{x},\vec{y})_{cos}$$

(2. 3)

Όπως μπορούμε να παρατηρήσουμε και από την εξίσωση υπολογισμού η μετρική αυτή εξαρτάται τόσο από την γωνία όσο και από τα μέτρα των διανυσμάτων.

2.3 Μετασχηματιστές

Στην ενότητα αυτή παρουσιάζουμε αρχικά την δομή μιας οικογένειας μοντέλων μηχανικής μάθησης, που ονομάζονται μετασχηματιστές. Έπειτα, παραθέτουμε ορισμένες πληροφορίες για έναν ιδιαίτερα διάσημο μετασχηματιστή, τον BERT, ο οποίος χρησιμοποιείται ευρέως σε πολλές διαφορετικές εργασίες. Στη συνέχεια εξηγούμε τη δομή και ορισμένες τεχνικές που εφαρμόζονται από μετασχηματιστές των οποίων οι έξοδοι είναι χρήσιμοι για εργασίες αναζήτησης ομοιότητας μεταξύ τμημάτων κειμένου.

2.3.1 Αρχιτεκτονική των Μετασχηματιστών

Στην εργασία τους το 2017, οι Vaswani et al.[16] παρουσιάζουν τους **Μετασχηματιστές**, που αποτελούν μια σύγχρονη αρχιτεκτονική με την οποία μπορούμε να κατασκευάσουμε ένα νευρωνικό δίκτυο. Σε αντίθεση με αρχιτεκτονικές που είχαν παρουσιαστεί τα προηγούμενα χρόνια, οι οποίες συνδύαζαν μεθόδους που χρησιμοποιούν αναδρομικά μοντέλα και τεχνικές προσοχής, οι μετασχηματιστές είναι βασισμένοι αποκλειστικά στον όρο της προσοχής, πετυχαίνοντας τόσο καλύτερες επιδόσεις όσο και μικρότερα υπολογιστικά κόστη από τις παλαιότερες αρχιτεκτονικές.

Ο Μηχανισμός Προσοχής

Γενικά, οι τεχνικές προσοχής αποσκοπούν στο να επιτρέψουν σε ένα νευρωνικό δίκτυο το οποίο επεξεργάζεται μια είσοδο, να δίνει περισσότερη βάση στα σημεία της εισόδου που θεωρούνται ως τα πιο σημαντικά για το εκάστοτε σημείο της εξόδου που εκείνο προσπαθεί να υπολογίσει. Σε αρχιτεκτονικές δικτύων που έχουν προταθεί πριν από τους μετασχηματιστές, συχνά χρησιμοποιούνται μηχανισμοί προσοχής ώστε το δίκτυο που χρησιμοποιούμε να μπορέσει να μάθει ποια μέρη της εισόδου χρειάζεται να χρησιμοποιήσει περισσότερο και ποια λιγότερο [7]. Μια διαδεδομένη τεχνική που προτάθηκε αρχικά στην εργασία των Bahdanau et al. [17] αποτελεί την **προσθετική προσοχή**, η οποία με την χρήση ενός νευρωνικού δικτύου πρόσθιας τροφοδότησης, υπολογίζει μια συνάρτηση που χρησιμοποιείται στην φάση αποκωδικοποίησης και πρακτικά εκφράζει την σχετικότητα κάθε σημείου της εισόδου με τα σημεία της εξόδου του δικτύου.

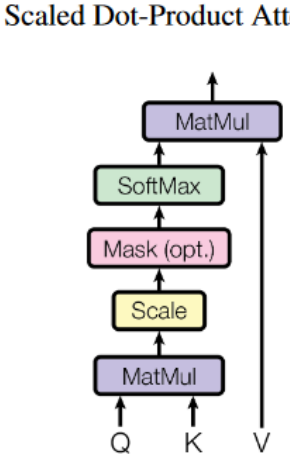
Ωστόσο, στους μετασχηματιστές χρησιμοποιείται μια παραλλαγή της τεχνικής της προσοχής εσωτερικού γινομένου, που ονομάζεται **προσοχή κλιμακωτού εσωτερικού γινομένου**. Η τεχνική αυτή προτιμήθηκε από τους Vaswani et al. [16] καθώς επιτρέπει την εφαρμογή τεχνικών που μειώνουν τις απαιτήσεις σε χώρο και χρόνο σε σχέση με την προσθετική προσοχή. Στην τεχνική αυτή έχουμε τρία είδη διανυσμάτων:

- Το διάνυσμα-ερώτημα **q**, που αποτελεί εκείνο για το οποίο ο μηχανισμός χρησιμοποιεί την προσοχή.

- Το διάνυσμα-κλειδιά **k**, που αποτελούν εκείνα στα οποία ο μηχανισμός θα δώσει βάση για τον υπολογισμό της προσοχής. Τα διανύσματα αυτά έχουν ίδια διάσταση με το διάνυσμα-ερώτημα.

- Τα διανύσματα-τιμές **v**, που χρησιμοποιούνται στο τελικό βήμα για τον υπολογισμό του διανύσματος συμφραζομένων.

Αξίζει να σημειωθεί ότι καθένα από τα παραπάνω είδη διανυσμάτων προκύπτει πολλαπλασιάζοντας την είσοδο με ένα διαφορετικό για κάθε είδος πίνακα, ώστε να επιλυθούν προβλήματα μεροληψίας ενός διανύσματος προς την δικιά του αναπαράσταση [7].



Εικόνα 5: Μηχανισμός Προσοχής Κλιμακωτού Εσωτερικού Γινομένου [16]

Στην παραπάνω εικόνα (Εικόνα 5) φαίνεται ο τρόπος με τον οποίο μπορούμε να υπολογίσουμε την έξοδο ενός τέτοιου μηχανισμού. Αρχικά υπολογίζουμε τα εσωτερικά γινόμενα του διανύσματος-ερώτημα με καθένα από τα διανύσματα-κλειδιά και τα κλιμακώνουμε πολλαπλασιάζοντας τα με μια σταθερά $\frac{1}{\sqrt{c}}$, όπου c είναι το πλήθος των διαστάσεων του διανύσματος-ερώτημα και των διανυσμάτων-κλειδιών. Έπειτα εφαρμόζουμε την συνάρτηση Softmax [18], η οποία φαίνεται στην εξίσωση (2. 4), και πολλαπλασιάζουμε τα βάρη που προκύπτουν με το διάνυσμα-τιμή που αντιστοιχεί στο καθένα από αυτά. Το σύνολο των τελικών αποτελεσμάτων καλείται συχνά διάνυσμα συμφραζομένων. Η διαδικασία αυτή μπορεί να εκτελεστεί παράλληλα για πολλά διανύσματα κλειδιά με χρήση πράξεων σε επίπεδο πινάκων, αν ομαδοποιήσουμε κατάλληλα τα διανύσματα εισόδου και εξάγουμε τα τρία είδη διανυσμάτων σε μορφή πινάκων [7].

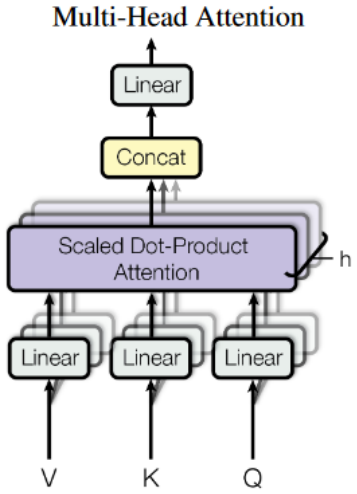
$$\text{softmax}(\vec{v})_n = \frac{e^{v_n}}{\sum_{m=1}^{k} v_m} \tag{2. 4}$$

Σημειώνουμε επίσης ότι το βήμα της κλιμάκωσης χρησιμοποιείται ώστε να καταπολεμηθεί το φαινόμενο όπου έχουμε μείωση της απόδοσης όταν τα διανύσματα-ερωτήματα και τα διανύσματα-κλειδιά έχουν μεγάλες διαστάσεις [16].

Προσοχή Πολλαπλών Κεφαλών

Αν εφαρμόσουμε τον μηχανισμό προσοχής που αναφέραμε παραπάνω, λαμβάνουμε ένα διάνυσμα συμφραζομένων το οποίο εξαρτάται στο σύνολο του από όλη την ακολουθία. Ωστόσο, οι Vaswani et al. [16] παρατήρησαν ότι μπορούμε να εξάγουμε πολύτιμες πληροφορίες για τις σχέσεις μεταξύ των εισόδων, που συχνά αποκρύπτονται όταν γίνεται απλή εφαρμογή του μηχανισμού, αν αντιθέτως εφαρμόσουμε παράλληλα τον μηχανισμό για διαφορετικές γραμμικές προβολές καθενός από τα τρία είδη διανυσμάτων, χωρίς να έχουμε σοβαρές επιπτώσεις στο κόστος του υπολογισμού.

Θεωρώντας πως έχουμε ομαδοποιήσει τα διανύσματα σε τρεις πίνακες, ένα για κάθε είδος διανύσματος, κάθε κεφαλή προσοχής προβάλλει γραμμικά τις τρεις κατηγορίες διανυσμάτων με διαφορετικό τρόπο από τις υπόλοιπες, χρησιμοποιώντας για τον σκοπό αυτό ορισμένους πίνακες βαρών που διαφέρουν για κάθε κεφαλή. Στη συνέχεια, κάθε κεφαλή προσοχής υπολογίζει το διάνυσμα συμφραζομένων που αντιστοιχεί στα διανύσματα της. Έπειτα, συνενώνουμε τα διανύσματα συμφραζομένων που προέκυψαν από κάθε κεφαλή προσοχής και τέλος πολλαπλασιάζουμε το τελικό διάνυσμα αυτό με έναν πίνακα, ώστε να προκύψει ένα διάνυσμα ίδιων διαστάσεων με αυτό που θα είχαμε αν εφαρμόζαμε την μέθοδο με μία μόνο κεφαλή προσοχής. Η διαδικασία που εξηγήσαμε φαίνεται αναλυτικά στο παρακάτω σχήμα (Εικόνα 6) [7], [16].

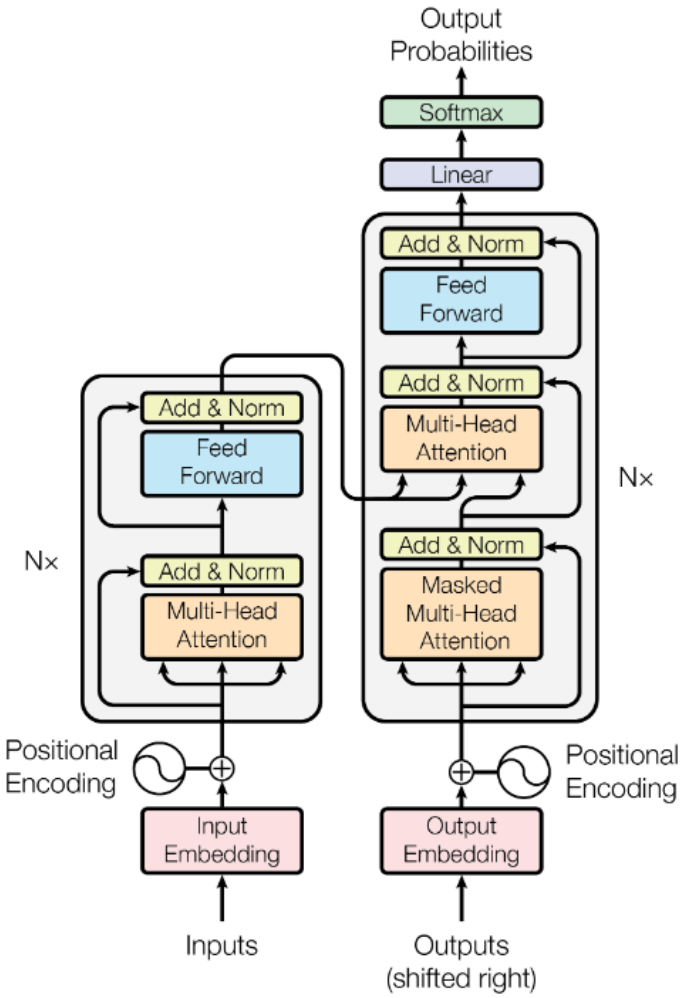


Εικόνα 6: Τεχνική Προσοχής Πολλαπλών Κεφαλών [16]

Κωδικοποίηση και Αποκωδικοποίηση

Η μετασχηματιστές αποτελούνται από δύο κύρια μέρη. Το πρώτο από αυτά αποτελεί τον **Κωδικοποιητή**, που είναι υπεύθυνος για την μετατροπή των εισόδων που δέχεται ο μετασχηματιστής, σε μορφή κατάλληλη για επεξεργασία. Το δεύτερο μέρος ενός μετασχηματιστή ονομάζεται **Αποκωδικοποιητής** και είναι με τη σειρά του υπεύθυνο για την επεξεργασία της εξόδου του κωδικοποιητή και για την παραγωγή της τελικής εξόδου του μετασχηματιστή.

Το μέρος του κωδικοποιητή αποτελείται από όμοια επίπεδα τα οποία σχηματίζονται από δύο μικρότερα τμήματα. Αρχικά έχουμε έναν μηχανισμό προσοχής πολλαπλών κεφαλών στον οποίον προστίθεται μια υπολειπόμενη σύνδεση, ώστε να καταπολεμηθούν τυχόν φαινόμενα εξαφάνισης κλίσεων [7], [19] και η τελική έξοδος του τμήματος κανονικοποιείται ούτως ώστε να μειωθεί ο χρόνος που απαιτείται για την εκπαίδευση του δικτύου [20]. Συνεπώς, η έξοδος του τμήματος αυτού αποτελεί το κανονικοποιημένο άθροισμα της αρχικής εισόδου και της εξόδου του μηχανισμού προσοχής. Έπειτα, η έξοδος του παραπάνω τμήματος δίνεται ως είσοδος στο δεύτερο τμήμα, που αποτελείται από ένα πλήρως συνδεδεμένο δίκτυο πρόσθιας τροφοδότησης του οποίου η εφαρμογή γίνεται σε κάθε θέση με ανεξάρτητο τρόπο. Όπως και στο πρώτο τμήμα, έχουμε και εδώ μια υπολειπόμενη σύνδεση και κανονικοποίηση των τελικών εξόδων. Η έξοδος του δεύτερου τμήματος αποτελεί και την τελική έξοδο ενός επιπέδου. Ο κωδικοποιητής ενός μετασχηματιστή έχει συνήθως αρκετά επίπεδα της παραπάνω δομής, όπου το καθένα δέχεται ως είσοδο την έξοδο του προηγούμενου του.



Εικόνα 7: Η Αρχιτεκτονική Ενός Μετασχηματιστή [16]

Όσον αφορά το μέρος του αποκωδικοποιητή, και αυτό αποτελείται από μια σειρά όμοιων μεταξύ τους επιπέδων, όπου το καθένα λαμβάνει ως είσοδο την έξοδο του προηγούμενου του, με εξαίρεση προφανώς το πρώτο επίπεδο. Η δομή ενός επιπέδου του αποκωδικοποιητή αποτελείται από τα ίδια τμήματα τα οποία σχηματίζουν ένα επίπεδο του κωδικοποιητή, με την προσθήκη ενός ακόμα τμήματος προσοχής πολλαπλών κεφαλών, που χρησιμοποιεί την έξοδο του τελευταίου επιπέδου του κωδικοποιητή του μετασχηματιστή. Επίσης, έχουμε και εδώ μια υπολειπόμενη σύνδεση και εφαρμογή κανονικοποίησης, για κάθε τμήμα του επιπέδου. Αξίζει επιπλέον να σημειωθεί ότι ένα επίπεδο αποκωδικοποιητή εφαρμόζει απόκρυψη των θέσεων που δεν έχουν ακόμα προβλεφθεί από τον μετασχηματιστή προκειμένου να μην χρησιμοποιηθούν στην διαδικασία για την εύρεση της τρέχουσας θέσης, θέτοντας τις αντίστοιχες τιμές σε $-\infty$ κατά την διάρκεια των υπολογισμών στο πρώτο τμήμα προσοχής πολλαπλών κεφαλών. Όταν μια έξοδος προβλεφθεί, τότε οι έξοδοι ανανεώνονται με χρήση ολίσθησης προς τα δεξιά ώστε να μπορέσουμε να τις χρησιμοποιήσουμε για την πρόβλεψη των επόμενων εξόδων.

Όπως φαίνεται και στην Εικόνα 7, η έξοδος του αποκωδικοποιητή, η οποία αποτελεί ένα διάνυσμα και ισοδυναμεί με την έξοδο του τελευταίου επιπέδου του αποκωδικοποιητή, μετασχηματίζεται γραμμικά και έπειτα εφαρμόζεται η συνάρτηση Softmax, ώστε να μετατρέψουμε το διάνυσμα αυτό σε μια σειρά από πιθανότητες οι οποίες

εκφράζουν το κατά πόσο ο μετασχηματιστής πιστεύει ότι η έξοδος αντιστοιχεί σε καθένα από τα σύμβολα που υπάρχουν στο λεξιλόγιο του.

Πριν χρησιμοποιηθούν από τα δύο κύρια μέρη του μετασχηματιστή, τα τμήματα της εισόδου και τα υπολογισμένα τμήματα της εξόδου μέχρι την τρέχουσα χρονική στιγμή μετατρέπονται στις αντίστοιχες ενσωματώσεις τους, που καθορίζονται κατά τη διάρκεια του σταδίου της μάθησης του μετασχηματιστή. Επιπρόσθετα, πριν οι ενσωματώσεις των εισόδων και των υπολογισμένων εξόδων δοθούν στον κωδικοποιητή και στον αποκωδικοποιητή, αντίστοιχα, αθροίζονται με ένα διάνυσμα ίδιων διαστάσεων με των ενσωματώσεων. Η πρόσθεση των διανυσμάτων αυτών αποσκοπεί στο να επιτρέψει στον μετασχηματιστή να αξιοποιήσει την θέση των τμημάτων εισόδου και εξόδου στους υπολογισμούς του.

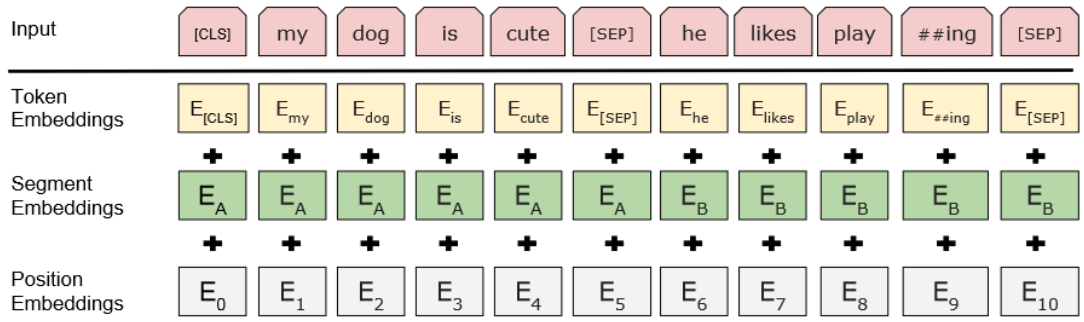
2.3.2 Ο Μετασχηματιστής BERT

Μετά την εμφάνιση των μετασχηματιστών, τα επόμενα χρόνια παρουσιάστηκαν πολλά μοντέλα τα οποία χρησιμοποίησαν σε μεγάλο βαθμό την αρχιτεκτονική τους. Ένας από τους πιο αξιοσημείωτους μετασχηματιστές, αποτελεί ο **BERT (Bidirectional Encoder Representations from Transformers)**, τον οποίο οι Devlin et al. [21] παρουσιάζουν στην εργασία τους το 2018.

Η αρχιτεκτονική του BERT

Ένας μετασχηματιστής της μορφής που εξηγήσαμε στην ενότητα 2.3.1 λαμβάνει υπόψιν στα τμήματα προσοχής του κατά την πρόβλεψη της τρέχουσας εξόδου, μόνο τα στοιχεία εισόδου και των προηγούμενων εξόδων και όχι τα στοιχεία των επόμενων εξόδων, κάτι που δεν είναι πάντα επιθυμητό ανάλογα την εργασία για την οποία προορίζεται ο μετασχηματιστής. Ο **BERT** σχεδιάστηκε και εκπαιδεύτηκε με τέτοιο τρόπο, ούτως ώστε να επιτρέπει στους μηχανισμούς προσοχής να αξιοποιήσουν χρήσιμες πληροφορίες κατά των υπολογισμό μιας εξόδου **τόσο από προηγούμενα όσο και από επόμενα** στοιχεία. Πιο συγκεκριμένα, ο BERT είναι δομημένος μόνο από μέρη κωδικοποιητών μετασχηματιστών, τους οποίους παρουσιάσαμε στην ενότητα 2.3.1, και δεν χρησιμοποιεί μέρη αποκωδικοποιητών στην αρχιτεκτονική του.

Όσον αφορά τις εισόδους του, Ο BERT εκτελεί τμηματοποίηση **WordPiece** [22] για το κείμενο εισόδου με λεξιλόγιο μεγέθους 30000 συμβόλων και στη συνέχεια κάθε σύμβολο αντιστοιχίζεται στην ενσωμάτωση του. Επιπλέον, εισάγονται δύο ειδικά σύμβολα στην τμηματοποιημένη ακολουθία εισόδου. Το πρώτο ειδικό σύμβολο, **[CLS]**, τοποθετείται στην **αρχή της ακολουθίας** και είναι χρήσιμο για την εξαγωγή πληροφορίας μιας γενικής αναπαράστασης της εισόδου. Το δεύτερο ειδικό σύμβολο, **[SEP]**, χρησιμοποιείται σε περιπτώσεις όπου επιθυμούμε η είσοδος του BERT να αποτελεί ένα ζεύγος προτάσεων. Το ειδικό σύμβολο [SEP] συνεπώς **τοποθετείται στο τέλος** κάθε πρότασης ώστε να σηματοδοτεί τη λήξη της. Επιπρόσθετα, σε κάθε ενσωμάτωση που αναφέραμε παραπάνω, προστίθενται **δύο** επιπλέον διανύσματα ίδιων διαστάσεων, ένα για την προσθήκη πληροφορίας της θέσης της ενσωμάτωσης στην ακολουθία όπως είδαμε και στους κλασσικούς μετασχηματιστές, και ένα για την προσθήκη πληροφορίας σχετικά με το αν η εκάστοτε ενσωμάτωση αποτελεί μέρος της πρώτης ή της δεύτερης πρότασης της εισόδου.



Εικόνα 8: Παράδειγμα Εισόδων του Μετασχηματιστή BERT [21]

Η διαδικασία προ-εκπαίδευσης και βελτίωσης

Η εκπαίδευση του μετασχηματιστή BERT μπορεί να διακριθεί σε δύο ξεχωριστά βήματα, αυτό της **προ-εκπαίδευσης** και αυτό της **βελτίωσης**.

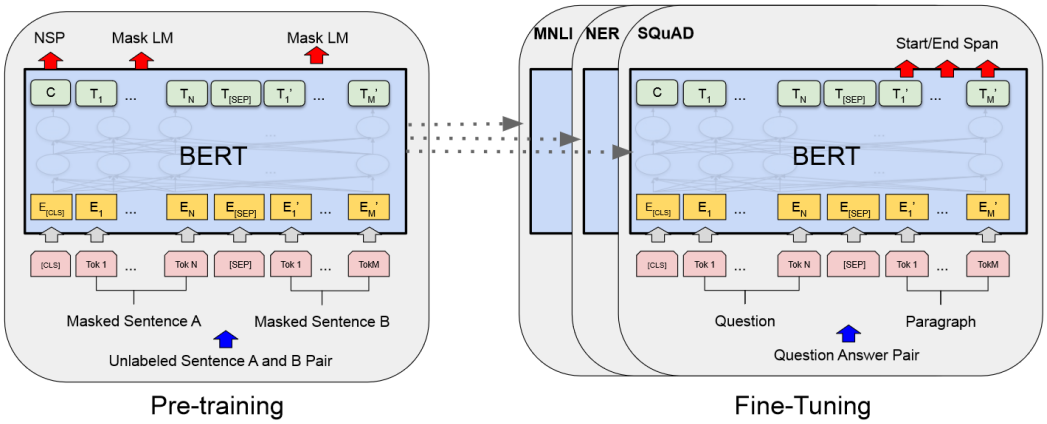
Στο βήμα προ-εκπαίδευσης, ο BERT εκπαιδεύεται σε **δύο** εργασίες μη επιβλεπόμενης μάθησης. Η πρώτη από αυτές αποτελεί την πρόβλεψη ενός ποσοστού των συμβόλων της εισόδου που έχουν αποκρυφθεί. Ειδικότερα, δίνεται ως είσοδος μια πρόταση όπου έπειτα από την εφαρμογή συμβολοποίησης γίνεται απόκρυψη ορισμένων συμβόλων με τυχαίο τρόπο επιλογής. Το ποσοστό αυτό των συμβόλων αντιστοιχεί στο 15% των συνολικών που προκύπτουν για κάθε πρόταση. Για κάθε τέτοιο σύμβολο που επιλέγεται, οι Devlin et al. [21] επιλέγουν να εφαρμόσουν τρία διαφορετικά ενδεχόμενα:

- Αντικατάσταση του συμβόλου που επιλέχθηκε με ένα τυχαίο σύμβολο, με πιθανότητα 10%.
- Διατήρηση του συμβόλου χωρίς κάποια αλλαγή και πάλι με πιθανότητα 10%.
- Αντικατάσταση του συμβόλου με ένα ειδικό σύμβολο (**[MASK]**) το οποίο σηματοδοτεί σύμβολο που έχει αποκρυφθεί, με πιθανότητα 80%.

Τα ενδεχόμενα αυτά εφαρμόστηκαν κυρίως ώστε να καταπολεμηθεί το γεγονός του ότι το ειδικό σύμβολο απόκρυψης δεν χρησιμοποιείται στο στάδιο της βελτίωσης. Η εργασία αυτή καλείται **μοντελοποίηση κρυμμένης γλώσσας** και οι Devlin et al. [21] αποδεικνύουν ότι μπορεί να επιτρέψει στους μηχανισμούς προσοχής του BERT να χρησιμοποιούν για την πρόβλεψη ενός συμβόλου που αντιστοιχεί σε μια θέση εισόδου-εξόδου, όπως αναφέραμε, όχι μόνο τις προηγούμενες αλλά και τις επόμενες από εκείνη θέσεις.

Η δεύτερη εργασία του βήματος προ-εκπαίδευσης αποτελεί η απάντηση στο ερώτημα του εάν σε ένα ζεύγος προτάσεων, η δεύτερη πρόταση έπεται λογικά της πρώτης. Στο σύνολο των ζευγών προτάσεων που χρησιμοποιήθηκαν στα μισά από αυτά η δεύτερη πρόταση ήταν πράγματι η επόμενη της πρώτης και στα άλλα μισά οι δύο προτάσεις δεν συσχετίζονταν μεταξύ τους. Για την απάντηση στο πρόβλημα αυτό χρησιμοποιείται η πρώτη έξοδος του τελευταίου επιπέδου του BERT που αντιστοιχεί στο σύμβολο [CLS] που όπως αναφέραμε προορίζεται για τέτοιου είδους εργασίες. Η εργασία αυτή ονομάζεται **πρόβλεψη επόμενης πρότασης** και οι Devlin et al. [21] αποδεικνύουν και πάλι ότι η χρήση αυτού του βήματος στη διαδικασία καταφέρνει να βελτιώσει τις επιδόσεις του BERT σε πληθώρα εργασιών.

Το στάδιο της βελτίωσης αποσκοπεί στο να μπορέσει ο BERT να εκτελεί μια συγκεκριμένη εργασία όσο το δυνατόν πιο βέλτιστα. Έχοντας ολοκληρώσει το στάδιο προ-εκπαίδευσης, το οποίο είναι αρκετά χρονοβόρο, οι παράμετροι του μετασχηματιστή έχουν προσαρμοστεί ως προς τις 2 εργασίες του βήματος προ-εκπαίδευσης που αναφέραμε παραπάνω. Επίσης, οι τεχνικές προσοχής που χρησιμοποιούνται από την αρχιτεκτονική των μετασχηματιστών καθιστούν τον BERT εύκολα προσαρμόσιμο έπειτα από την προ-εκπαίδευση του για την ανταπόκριση του σε πολλές εργασίες. Δηλαδή, για να προσαρμόσουμε τον BERT, τις περισσότερες φορές χρειάζεται να προσθέσουμε λίγα μόνο επιπλέον επίπεδα δικτύων απλής αρχιτεκτονικής σε ορισμένες από τις εξόδους του τελευταίου επιπέδου του BERT. Στην συνέχεια χρειάζεται να βελτιστοποιήσουμε κυρίως τις νέες παραμέτρους που εισήγαμε, κάτι που όπως εξηγούν και οι Devlin et al. [21] στην δημοσίευση τους απαιτεί σημαντικά λιγότερο χρόνο.



Εικόνα 9: Τα Στάδια Προ-εκπαίδευσης (Αριστερά) και Βελτίωσης (Δεξιά) του BERT [21]

2.3.3 Μετασχηματιστές για Χρήση σε Αναζήτηση Ομοιότητας Κειμένου

Έπειτα από την παρουσίαση του BERT, αναπτύχθηκε ένα μεγάλο πλήθος μοντέλων μηχανικής μάθησης τα οποία εμπνεύστηκαν από τον τρόπο με τον οποίο ο πρώτος είχε σχεδιαστεί και εκπαιδευτεί. Μια σημαντική κατηγορία μοντέλων που έχουν αναπτυχθεί έχουν ως έναν από τους βασικούς τους στόχους, λαμβάνοντας ως είσοδο συνήθως μικρά τμήματα κειμένου, την παραγωγή ενσωματώσεων που αντιπροσωπεύουν το εκάστοτε τμήμα εισόδου σε ένα διανυσματικό χώρο. Τα μοντέλα που έχουν ως στόχο την παραγωγή ενσωματώσεων συχνά αναφέρονται και απλώς ως **μοντέλα ενσωματώσεων**. Οι ενσωματώσεις που λαμβάνονται από τέτοια μοντέλα μπορούν στην συνέχεια να χρησιμοποιηθούν για εργασίες όπως αυτές της αναζήτησης ή ανάκτησης δεδομένων.

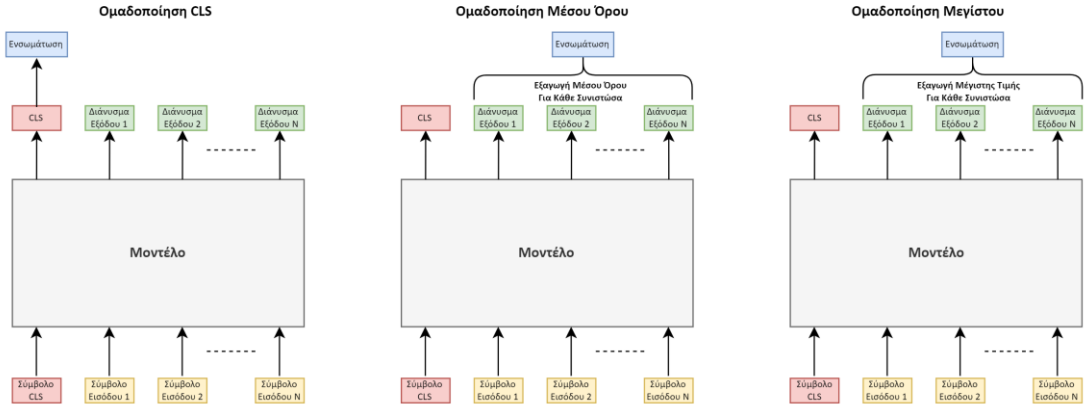
Τεχνικές Ομαδοποίησης

Ένας τρόπος που έχει δοκιμαστεί για την εξαγωγή ενσωματώσεων κειμένου από μοντέλα που ακολουθούν αρχιτεκτονική παρόμοια με αυτή του μετασχηματιστή BERT, είναι η αξιοποίηση των εξόδων των μοντέλων μέσω ορισμένων τεχνικών ομαδοποίησης των εξόδων αυτών. Τρεις τέτοιες τεχνικές που συχνά εφαρμόζονται είναι οι εξής [23]:

- **<u>Ομαδοποίηση CLS</u>**: Όπως είδαμε και στη προηγούμενη υπό-ενότητα, στην αρχιτεκτονική του μοντέλου BERT το σύμβολο CLS χρησιμοποιείται για την εξαγωγή πληροφορίας για όλη την ακολουθία εισόδου. Στην τεχνική αυτή, χρησιμοποιούμε το διάνυσμα εξόδου του μοντέλου που αντιστοιχεί στο σύμβολο CLS, ως την ενσωμάτωση που αντιπροσωπεύει το κείμενο που δίνεται ως είσοδος.

- **<u>Ομαδοποίηση Μέσου Όρου</u>**: Στην τεχνική αυτή, λαμβάνουμε την τελική ενσωμάτωση του κειμένου εισόδου, υπολογίζοντας ένα διάνυσμα όπου κάθε συνιστώσα αποτελεί τον μέσο όρο των αντίστοιχων συνιστωσών των διανυσμάτων εξόδου του μοντέλου μας.

- **<u>Ομαδοποίηση Μεγίστου</u>**: Στην τεχνική αυτή, η τελική ενσωμάτωση που επιλέγουμε για το κείμενο εισόδου αποτελεί ένα διάνυσμα όπου κάθε συνιστώσα του αποτελεί τη μέγιστη τιμή που εντοπίζουμε στις αντίστοιχες συνιστώσες των διανυσμάτων εξόδου του μοντέλου μας [24].



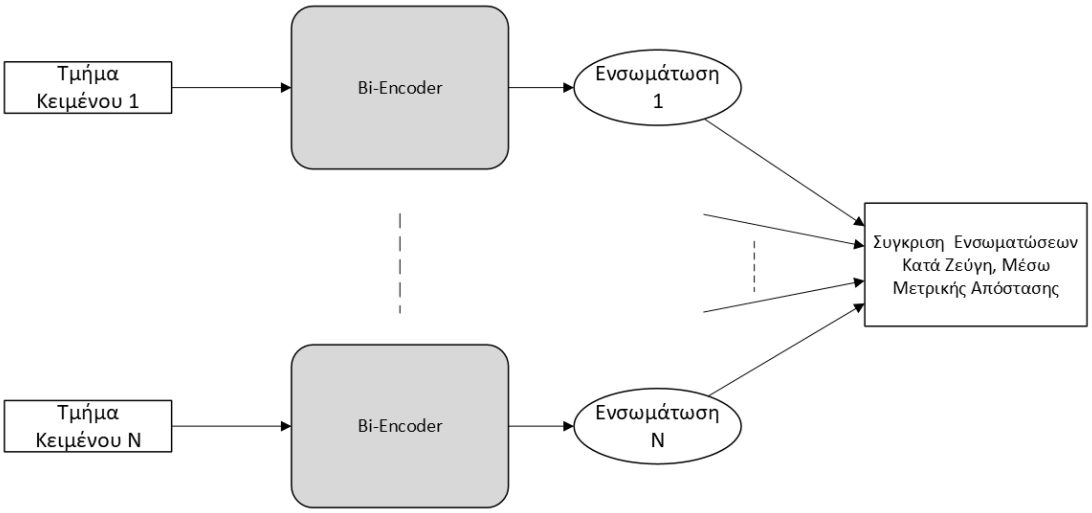
Εικόνα 10: Τεχνικές Ομαδοποίησης για Εξαγωγή Ενσωματώσεων

Καθεμιά από τις τεχνικές αυτές είναι δυνατόν να έχουν αποτελεσματικές επιδόσεις, ανάλογα με τον τρόπο που έχει εκπαιδευτεί το εκάστοτε μοντέλο ενσωματώσεων. Στην εργασία τους, οι Reimers και Gurevych [23] δοκιμάζουν να χρησιμοποιήσουν τον μετασχηματιστή BERT για να εξάγουν ενσωματώσεις κειμένου εφαρμόζοντας ορισμένες τεχνικές ομαδοποίησης από αυτές που αναφέραμε παραπάνω. Καταλήγουν στο συμπέρασμα του ότι οι ενσωματώσεις αυτές δεν είναι ιδιαίτερα χρήσιμες για εργασίες αναζήτησης όμοιων μεταξύ τους κειμένων μέσω της σύγκρισης των ενσωματώσεων, χρησιμοποιώντας κάποια μετρική απόστασης.

Προκειμένου να βελτιωθεί η διαδικασία της παραγωγής τέτοιων ενσωματώσεων κειμένου, μέχρι και σήμερα έχουν προταθεί διάφορα μοντέλα που χρησιμοποιούν συχνά την αρχιτεκτονική του BERT ή προ-εκπαιδευμένες εκδοχές του, ως δομικό στοιχείο. Τα μοντέλα αυτά έχουν βελτιώσει σημαντικά τις επιδόσεις για εργασίες όπου χρειάζεται να εφαρμοστεί αναζήτηση σχετικών κειμένων όπου οι παραγόμενες ενσωματώσεις συγκρίνονται μεταξύ τους, μέσω της χρήσης μιας μετρικής απόστασης όπως αυτή της απόστασης συνημιτόνου.

Bi-Encoders

Μια ιδιαίτερα δημοφιλής κατηγορία μοντέλων, οι **Bi-Encoders**, μπορούν να χρησιμοποιηθούν για εργασίες αναζήτησης ομοιότητας μεταξύ κειμένων. Συγκεκριμένα, οι Bi-Encoders, λαμβάνοντας ως είσοδο ένα στοιχείο, όπως μια πρόταση ή γενικότερα μια ακολουθία κειμένου, παράγουν ως έξοδο μια ενσωμάτωση που αντιστοιχεί στην ακολουθία αυτή. Στη συνέχεια, η ενσωμάτωση αυτή μπορεί να συγκριθεί με ενσωματώσεις άλλων ακολουθιών κειμένου που υπολογίζονται με όμοιο τρόπο [25].



Εικόνα 11: Διαδικασία Σύγκρισης Ομοιότητας για Ν Τμήματα Κειμένου, με Χρήση Bi-Encoder

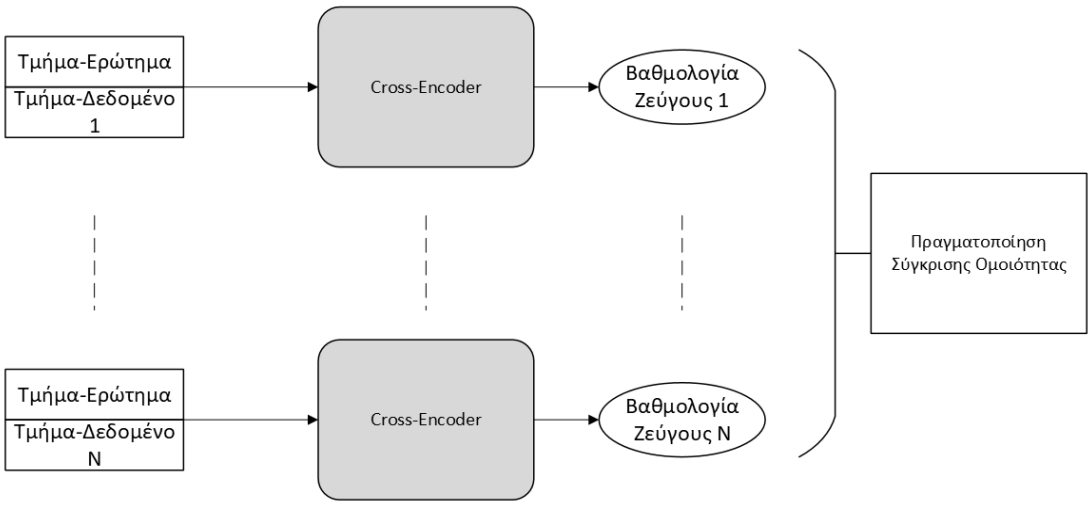
Ειδικότερα, Οι Bi-Encoders που βασίζονται σε μετασχηματιστές έχουν συχνά αρχιτεκτονική παρόμοια με αυτή του μετασχηματιστή BERT και χρησιμοποιώντας κάποια τεχνική ομαδοποίησης των εξόδων, όπως αυτές που αναφέραμε, επιτρέπουν την εξαγωγή ενσωματώσεων για την ακολουθία κειμένου που έχουν δεχτεί ως είσοδο. Έχοντας παράγει τις ενσωματώσεις για τις ακολουθίες κειμένων που έχουμε στη διάθεση μας, μπορούμε να τις αποθηκεύσουμε σε μια βάση δεδομένων και έπειτα, έχοντας σε μελλοντική χρονική στιγμή διαθέσιμες κάποιες ενσωματώσεις νέων ακολουθιών, είμαστε σε θέση να αναζητήσουμε τις πιο σχετικές υπάρχουσες ακολουθίες, υπολογίζοντας απλώς τις νέες ενσωματώσεις και συγκρίνοντας τες με αυτές των ήδη υπαρχόντων ακολουθιών [26]. Στην Εικόνα 11 φαίνεται η διαδικασία που ακολουθούμε για την διαδικασία παραγωγής και της σύγκρισης ενσωματώσεων για ακολουθίες κειμένων, χρησιμοποιώντας Bi-Encoders της προαναφερθείσας κατηγορίας.

Cross-Encoders

Μια άλλη σημαντική κατηγορία μοντέλων που έχει αξιοποιηθεί για την σύγκριση ομοιότητας μεταξύ τμημάτων κειμένου είναι οι **Cross-Encoders**. Όταν χρησιμοποιούνται για σύγκριση ομοιότητας τμημάτων κειμένων, οι Cross-Encoders δέχονται ως είσοδο, σε αντίθεση με τους Bi-Encoders, δύο τμήματα κειμένων και έχουν ως στόχο την βαθμολόγηση της ομοιότητας μεταξύ των δύο αυτών τμημάτων κειμένου [26].

Λόγω και του ότι δέχονται ως είσοδο και τα δύο τμήματα που επιθυμούμε να συγκρίνουμε, έχει παρατηρηθεί ότι οι Cross-Encoders που βασίζονται σε μετασχηματιστές είναι συχνά πιο αποτελεσματικοί συγκριτικά με τους αντίστοιχους Bi-Encoders,

καθώς οι πρώτοι μπορούν να αξιοποιήσουν τους μηχανισμούς προσοχής και στα δύο τμήματα κειμένου, ενώ οι τελευταίοι δεν έχουν αυτό το πλεονέκτημα [25]. Η χρήση όμως και των δύο τμημάτων κειμένου ως είσοδο, καθιστούν τους Cross-Encoders ιδιαίτερα ακριβούς σε υπολογιστικούς πόρους σε περιπτώσεις όπου θέλουμε να υπολογίσουμε την ομοιότητα μεταξύ ενός τμήματος κειμένου και ορισμένων άλλων τμημάτων, διότι χρειάζεται να τους χρησιμοποιήσουμε τόσες φορές όσο και το πλήθος των τμημάτων αυτών. Στην Εικόνα 12 φαίνεται η διαδικασία του υπολογισμού βαθμολογιών ομοιότητας, μεταξύ ενός τμήματος-ερωτήματος και ορισμένων τμημάτων που έχουμε ως δεδομένα, χρησιμοποιώντας έναν Cross-Encoder που έχει βασιστεί σε μετασχηματιστές.



Εικόνα 12: Σύγκριση Ομοιότητας Μεταξύ Ενός Τμήματος-Ερώτημα και Ν Τμημάτων-Δεδομένων, με Χρήση Cross-Encoder

Λόγω του ότι πολλές φορές ένα τέτοιο τμήμα-ερώτημα δεν είναι άμεσα διαθέσιμο, όπως για παράδειγμα σε περιπτώσεις όπου ένα σύστημα ανάκτησης πληροφορίας χρησιμοποιεί δεδομένα που δίνονται ως είσοδος από τον χρήστη τους, η διαδικασία της εύρεσης όμοιων τμημάτων κειμένου με αυτό του τμήματος-ερώτημα μπορεί πολλές φορές όπως αναφέραμε, ενώ είναι αποτελεσματική, να είναι ταυτόχρονα πολύ χρονοβόρα όταν χρησιμοποιούμε Cross-Encoders. Συνεπώς, σε περιπτώσεις όπου επιθυμούμε να δώσουμε έμφαση στην αρτιότερη σύγκριση ομοιότητας μεταξύ των τμημάτων είναι προτιμότερο να χρησιμοποιήσουμε Cross-Encoders. Όμως, σε περιπτώσεις όπου η ταχύτητα της σύγκρισης είναι πιο σημαντική ή το πλήθος των τμημάτων-δεδομένων μας είναι πολύ μεγάλο, συνηθίζεται να αξιοποιούμε τους Bi-Encoders, καθώς όπως είδαμε μπορούμε να αποθηκεύσουμε τις ενσωματώσεις των τμημάτων δεδομένων και να υπολογίζουμε κάθε φορά μόνο την ενσωμάτωση του εκάστοτε τμήματος-ερωτήματος και έπειτα την διαφορά των ενσωματώσεων κατά ζεύγη, ώστε να πραγματοποιήσουμε την σύγκριση ομοιότητας [26].

2.4 Γενετική Τεχνητή Νοημοσύνη

Στην ενότητα αυτή παρουσιάζουμε έννοιες και μοντέλα μηχανικής μάθησης που σχετίζονται με την Γενετική Τεχνητή Νοημοσύνη (Generative Artificial Intelligence). Στην υπό-ενότητα 2.4.1 κάνουμε μια σύντομη εισαγωγή σχετική με την Γενετική Τεχνητή Νοημοσύνη. Έπειτα, στην υπό-ενότητα 2.4.2 παρουσιάζουμε πληροφορίες σχετικά με τα Μεγάλα Γλωσσικά Μοντέλα και εξηγούμε τα βασικά χαρακτηριστικά τους, ενώ στην υπό-ενότητα 2.4.3 εξηγούμε ορισμένες τεχνικές που χρησιμοποιούνται κατά την παραγωγή κειμένου από τα μοντέλα αυτά. Στην υπό-ενότητα 2.4.4 παρουσιάζουμε μια οικογένεια Μεγάλων Γλωσσικών Μοντέλων, τα Llama. Τέλος, στην υπό-ενότητα 2.4.5 αναλύουμε ορισμένες τεχνικές μηχανικής προτροπών (prompt engineering).

2.4.1 Ορισμός / Εισαγωγή

Η **Γενετική Τεχνητή Νοημοσύνη** μπορεί να οριστεί ως μια υποκατηγορία μοντέλων Τεχνητής Νοημοσύνης. Καλείται γενετική καθώς χρησιμοποιείται για την παραγωγή πολλών ειδών εξόδων όπως κειμένου ή εικόνων. Τα τελευταία χρόνια έχει παρατηρηθεί ιδιαίτερη αύξηση της χρήσης Γενετικής Τεχνητής Νοημοσύνης μέσω εφαρμογών. Πιο συγκεκριμένα, στις εφαρμογές αυτές οι χρήστες συνήθως ορίζουν μια είσοδο σε μορφή κειμένου και αναμένουν την παραγωγή μιας εξόδου σε διάφορες μορφές, όπως κειμένου ή εικόνας, ανάλογα με την υπηρεσία που παρέχει η εκάστοτε εφαρμογή.

Η χρήση τέτοιων εφαρμογών από εκατομμύριες χρήστες του διαδικτύου σε παγκόσμιο επίπεδο αλλά και η ολοένα και αυξανόμενη εμφάνιση των πρώτων, καθιστά ξεκάθαρη την αποτελεσματικότητα των μοντέλων Γενετικής Τεχνητής Νοημοσύνης στην επίτευξη του στόχου τους. Αυτό έχει οδηγήσει στην διερεύνηση τρόπων με τους οποίους μπορούν να αναπτυχθούν νέα ικανότερα μοντέλα και εφαρμογές που χρησιμοποιούν τέτοιου είδους τεχνολογίες.

2.4.2 Μεγάλα Γλωσσικά Μοντέλα (LLMs)

Τα πλεονεκτήματα που προσφέρει η αρχιτεκτονική των Μετασχηματιστών, την οποία αναλύσαμε στην ενότητα 2.3, είχαν ως αποτέλεσμα την δημιουργία πολλών μοντέλων μηχανικής μάθησης που υιοθέτησαν την αρχιτεκτονική αυτή. Εκτός από μοντέλα που δομούνται μόνο από μέρη κωδικοποιητή ενός Μετασχηματιστή, όπως τον BERT τον οποίο παρουσιάσαμε στην ενότητα 2.3, υπάρχουν μοντέλα τα οποία χρησιμοποιούν μόνο μέρη αποκωδικοποιητή ενός Μετασχηματιστή ως το κύριο δομικό στοιχείο τους, όπως για παράδειγμα το GPT και το GPT-2 [27], [28].

Τα **Μεγάλα Γλωσσικά Μοντέλα** αποτελούν μια οικογένεια μοντέλων μηχανικής μάθησης που απαρτίζεται από γλωσσικά μοντέλα τα οποία εκπαιδεύονται σε ένα τεράστιο πλήθος δεδομένων, συνήθως κειμένων, κατά τη διάρκεια του σταδίου προεκπαίδευσης τους. Μετά την ολοκλήρωση του σταδίου αυτού, τα Μεγάλα Γλωσσικά μοντέλα έχουν αποκτήσει αξιοσημείωτη ποσότητα γνώσεων από τα δεδομένα προ-εκπαίδευσης και έχουν πλέον την ικανότητα να επεξεργάζονται και να εξάγουν χρήσιμες

πληροφορίες για την μελλοντική είσοδο που λαμβάνουν. Λόγω αυτής τους της ικανότητας, χρησιμοποιούνται με επιτυχία σε πλήθος εργασιών πετυχαίνοντας εξαιρετικά αποτελέσματα. Τα περισσότερα Μεγάλα Γλωσσικά μοντέλα εφαρμόζουν την αρχιτεκτονική των μετασχηματιστών, λόγω της υψηλής παραλληλοποίησης και των πλεονεκτημάτων της χρήσης των μηχανισμών προσοχής που παρουσιάζει η αρχιτεκτονική των τελευταίων. Όπως υποδηλώνει και η ονομασία τους, τα μοντέλα αυτά αποτελούνται συνήθως από πολλές εκατομμύριες ή ακόμα και δισεκατομμύριες παραμέτρους. Έχει παρατηρηθεί ότι η αύξηση των παραμέτρων των μοντέλων αυτών έχει τις περισσότερες φορές θετική επιρροή στις επιδόσεις τους, γεγονός που φαίνεται και από τις διαφορές στις επιδόσεις μεταξύ εκδοχών ίδιων μοντέλων που έχουν διαφορετικό πλήθος επιπέδων μετασχηματιστών και άρα και παραμέτρων [11], [21].

Τα προ-εκπαιδευμένα μοντέλα, ανάλογα και με τις τεχνικές που θα χρησιμοποιηθούν κατά τη διάρκεια της προ-εκπαίδευσης τους, μπορούν να χρησιμοποιηθούν χωρίς περαιτέρω εκπαίδευση σε πλήθος εργασιών ή να εκπαιδευτούν μέσω του σταδίου βελτίωσης ούτως ώστε να εξειδικευτούν στο να ανταποκρίνονται όσο το δυνατόν αρτιότερα σε συγκεκριμένες μόνο εργασίες. Σύμφωνα με τα παραπάνω, μπορούμε λοιπόν να διακρίνουμε δύο μεγάλες κατηγορίες μεγάλων γλωσσικών μοντέλων.

Η πρώτη αποτελεί μοντέλα που ακολουθούν μια αρχιτεκτονική παρόμοια με αυτή του BERT. Έχουν δηλαδή ως κύριο δομικό στοιχείο κωδικοποιητές ενός Μετασχηματιστή, ή όπως συχνά καλούνται στη βιβλιογραφία, **Κωδικοποιητές Μετασχηματιστές**. Όπως είδαμε, τα μοντέλα αυτά χρησιμοποιούνται συχνά σε προβλήματα ταξινόμησης κειμένου, για την αναγνώριση της ομοιότητας μεταξύ δύο προτάσεων και γενικότερα σε εργασίες όπου χρειάζεται να αντλήσουμε πληροφορία για το σύνολο της εισόδου και να την αξιοποιήσουμε για την επίλυση του εκάστοτε προβλήματος.

Η δεύτερη κατηγορία αποτελεί μοντέλα τα οποία έχουν ως κύριο δομικό στοιχείο τους αποκωδικοποιητές ενός Μετασχηματιστή που αναφέρονται, αντίστοιχα με τους κωδικοποιητές, συνήθως ως **Αποκωδικοποιητές Μετασχηματιστές.** Όπως αναφέραμε, λόγω της απόκρυψης των επόμενων θέσεων από αυτή για την οποία πραγματοποιείται πρόβλεψη της εξόδου στους μηχανισμούς προσοχής των επιπέδων αποκωδικοποιητών, τα μοντέλα αυτά είναι τις περισσότερες φορές κατάλληλα για την πρόβλεψη της επόμενης λέξης ή συμβόλου σε ένα κείμενο εισόδου και μπορούν να χρησιμοποιηθούν για την παραγωγή κειμένου.

Στο σημείο αυτό αξίζει να παραθέσουμε ορισμένες πληροφορίες για μια οικογένεια μοντέλων που ανήκουν στη δεύτερη κατηγορία που αναφέραμε, τους **Γενετικούς Προ-εκπαιδευμένους Μετασχηματιστές** (**Generative Pre-trained Transformers**). Τα μοντέλα αυτά χρησιμοποιούν όπως αναφέραμε και παραπάνω Αποκωδικοποιητές Μετασχηματιστές, όπου το κάθε επίπεδο αποκωδικοποιητή έχει ως κύρια τμήματα αυτά που παρουσιάζει ένα απλό επίπεδο Αποκωδικοποιητή, με εξαίρεση το τμήμα της προσοχής πολλαπλών κεφαλών που λαμβάνει ως είσοδο την έξοδο του μέρους του αποκωδικοποιητή στον αρχικό Μετασχηματιστή, καθώς το μέρος του κωδικοποιητή δεν χρησιμοποιείται στα μοντέλα αυτής της κατηγορίας. Τα μοντέλα αυτά χρησιμοποιούν ένα μεγάλο πλήθος δεδομένων για το βήμα προ-εκπαίδευσης τους ώστε να μπορούν μετά το πέρας του βήματος αυτού να είναι ικανά να υπολογίζουν μια πιθανότητα για κάθε σύμβολο που πλέον γνωρίζουν, η οποία εκφράζει το κατά πόσο πιθανό είναι το εκάστοτε σύμβολο να αποτελεί το επόμενο για ένα κείμενο εισόδου [27], [28].

Επίσης, αξίζει να αναφέρουμε ότι στην εργασία τους οι Radford et al. [28] διαπιστώνουν ότι τα μεγάλα γλωσσικά μοντέλα, εφόσον προ-εκπαιδευτούν σε αρκετά μεγάλο όγκο δεδομένων, είναι ικανά να ανταπεξέλθουν σε ικανοποιητικό βαθμό σε εύρος εργασιών χωρίς κάποια περαιτέρω καθοδήγηση. Επιπλέον, σημειώνουν ότι το μέγεθος των μοντέλων είναι μια ακόμα παράμετρος που μπορεί να επηρεάσει το κατά πόσο αυτά είναι ικανά να εμφανίσουν την προαναφερθείσα ιδιότητα.

2.4.3 Τεχνικές Παραγωγής Κειμένου

Όπως εξηγήσαμε και στην προηγούμενη υπό-ενότητα, τα μεγάλα γλωσσικά μοντέλα που προορίζονται για παραγωγή κειμένου εκπαιδεύονται στην εργασία της επιλογής του επόμενου συμβόλου σε μια ακολουθία εισόδου. Ό τρόπος με τον οποίο επιτυγχάνεται η επιλογή του πιθανότερου επόμενου συμβόλου γίνεται συνήθως με την μετατροπή της εξόδου του μοντέλου σε μια κατανομή πιθανοτήτων που εκφράζει την πιθανότητα κάθε διαθέσιμο σύμβολο να αποτελεί το επόμενο της ακολουθίας εισόδου και την εφαρμογή κάποιας τεχνικής ώστε να επιλέξουμε ένα από αυτά ως το πιο κατάλληλο.

Μια από τις πιο απλές τεχνικές αποτελεί η επιλογή του συμβόλου το οποίο εμφανίζει την μεγαλύτερη υπολογισμένη πιθανότητα. Η μέθοδος αυτή καλείται **άπληστη δειγματοληψία**. Ωστόσο, έχει παρατηρηθεί ότι η τεχνική αυτή έχει συχνά ως αποτέλεσμα το να παράγονται έξοδοι που μπορούν να χαρακτηριστούν ως επαναλαμβανόμενες, καθώς επιλέγουμε συνεχώς τα πιο πιθανά σύμβολα. Συνεπώς, σε περιπτώσεις όπου είναι επιθυμητή η παραγωγή κειμένου αξιοσημείωτου μεγέθους ή ενδεχομένως κειμένων που θα χαρακτηρίζαμε ως πιο δημιουργικά, η μέθοδος αυτή δεν είναι πολλές φορές κατάλληλη [11].

Προκειμένου να βελτιωθεί η διαδικασία επιλογής συμβόλων, έχουν προταθεί διάφορες πιο σύνθετες τεχνικές οι οποίες προσφέρουν πολλές φορές πλεονεκτήματα ανάλογα και με τις απαιτήσεις της εκάστοτε εργασίας παραγωγής κειμένου. Μία από τις τεχνικές που χρησιμοποιούνται ευρέως αποτελεί η εξαγωγή της κατανομής πιθανοτήτων που εξηγήσαμε στην αρχή αυτής της υπό-ενότητας και στη συνέχεια η επιλογή ενός συμβόλου με τυχαίο τρόπο, με βάση την πιθανότητα που αντιστοιχεί στο καθένα από αυτά. Για το σκοπό αυτό χρησιμοποιείται συνήθως η συνάρτηση **Softmax** [11], η οποία φαίνεται στην εξίσωση *(2. 5)*.

$$\text{softmax}(\vec{u})_i = \frac{e^{u_i}}{\sum_{m=1}^{k} u_m} \tag{2. 5}$$

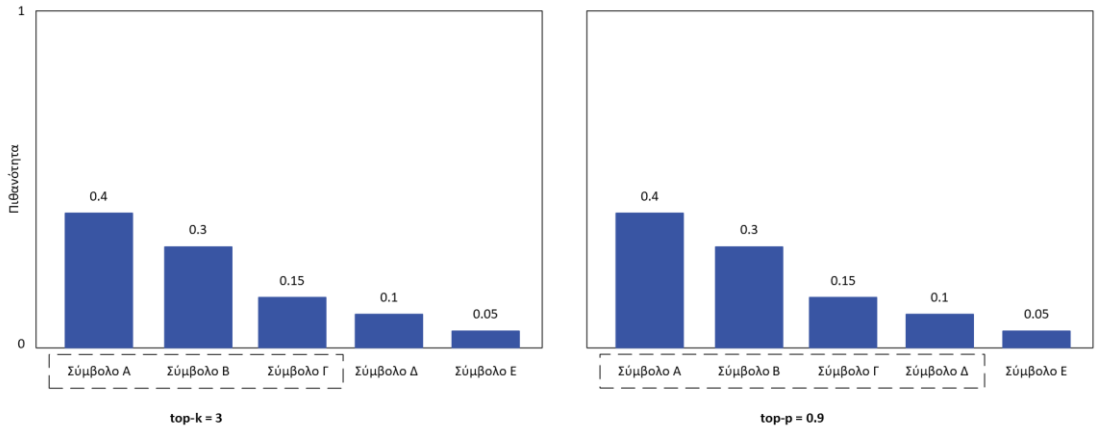
$$\text{softmax}\left(\left(\frac{\vec{u}}{T}\right)\right)_i = \frac{e^{\left(\frac{u}{T}\right)_i}}{\sum_{m=1}^{k}\left(\frac{u}{T}\right)_i} \tag{2. 6}$$

Μπορεί να χρησιμοποιηθεί επίσης μια ελαφρώς παραλλαγμένη εκδοχή της συνάρτησης Softmax, η οποία φαίνεται στην εξίσωση (2. 6), όπου εισάγουμε μια επιπλέον παράμετρο, που ονομάζεται θερμοκρασία. Η παραλλαγμένη αυτή εκδοχή δειγματοληψίας

ψίας καλείται **δειγματοληψία θερμοκρασίας** (temperature sampling) και μας προσφέρει την δυνατότητα να επηρεάσουμε την παραγόμενη κατανομή πιθανοτήτων. Πιο συγκεκριμένα, για τιμές της παραμέτρου της θερμοκρασίας στην περιοχή 0 έως 1, λαμβάνουμε κατανομές πιθανοτήτων όπου τα σύμβολα με υψηλή πιθανότητα έχουν πλέον ακόμα μεγαλύτερη πιθανότητα επιλογής, η οποία ολοένα και αυξάνεται όσο μειώνουμε την παράμετρο θερμοκρασίας. Θέτοντας την παράμετρο θερμοκρασίας με τιμές μεγαλύτερες του 1 λαμβάνουμε μια πιο ομοιόμορφη κατανομή πιθανοτήτων, κάτι που οδηγεί συνήθως στην δημιουργία ακολουθιών κειμένου που είναι πιο απρόβλεπτες και δημιουργικές [11].

Μια άλλη τεχνική που έχει εφαρμοστεί αποτελεί η **δειγματοληψία top-k**, στην οποία γίνεται επιλογή των κ συμβόλων που έχουν τις μεγαλύτερες πιθανότητες από την αρχική κατανομή που προκύπτει από την κατάλληλη επεξεργασία της εξόδου του μοντέλου [29]. Στη συνέχεια, αφού προσαρμόσουμε κατάλληλα την κατανομή πιθανοτήτων για τα σύμβολα αυτά [11], επιλέγεται βάση αυτής ένα σύμβολο με τυχαίο τρόπο.

Αργότερα, προτάθηκε επίσης μια ακόμη τεχνική, που καλείται **δειγματοληψία top-p ή δειγματοληψία πυρήνα**. Σύμφωνα με την τεχνική αυτή, ορίζεται αρχικά μια παράμετρος με εύρος τιμών 0 έως 1. Έπειτα επιλέγεται το μικρότερο δυνατό πλήθος συμβόλων για το οποίο το άθροισμα των πιθανοτήτων τους είναι μεγαλύτερο ή ίσο με τη τιμή της παραπάνω παραμέτρου και δημιουργείται μια νέα τροποποιημένη κατανομή πιθανοτήτων για τα σύμβολα αυτά. Η τεχνική αυτή παρουσιάζει το πλεονέκτημα του ότι επιλέγει με δυναμικό τρόπο το πλήθος των συμβόλων που θα χρησιμοποιηθούν στην τελική τυχαία επιλογή [30].



Εικόνα 13: Παράδειγμα Επιλογής Συμβόλων στην Top-k (Αριστερά) και Top-p (Δεξιά) Δειγματοληψία

2.4.4 Η Οικογένεια Μοντέλων Llama

Μια ιδιαίτερα δημοφιλής οικογένεια Μεγάλων Γλωσσικών Μοντέλων αποτελεί αυτή των μοντέλων **LLaMA**, τα οποία έχουν αναπτυχθεί από την Meta AI. Μέχρι και τη διάρκεια της συγγραφής αυτής της εργασίας, έχουν παρουσιαστεί τρείς εκδόσεις μοντέλων Llama, όπου κάθε έκδοση έχει εκδοχές μοντέλων με διαφορετικό πλήθος παραμέτρων. Για κάθε μια από τις εκδόσεις, οι εκδοχές των μοντέλων που έχουν τις

περισσότερες παραμέτρους πετυχαίνουν και καλύτερα αποτελέσματα από τις αντίστοιχες εκδοχές με λιγότερες παραμέτρους, κάτι που συμβαδίζει και με όσα αναφέραμε στην προηγούμενη υπό-ενότητα.

Τα Μοντέλα LLaMA

Στην εργασία τους το 2023, οι Touvron et al. [31], παρουσιάζουν τα μοντέλα **LLaMA**, τα οποία αποτελούν Μεγάλα Γλωσσικά Μοντέλα με εκδοχές που αποτελούνται από 7 δισεκατομμύριες παραμέτρους για την μικρότερη εκδοχή, έως και 65 δισεκατομμύριες για την μεγαλύτερη. Τα μοντέλα LLaMA, κατά το στάδιο της προ-εκπαίδευσης εφαρμόζουν συμβολοποίηση κωδικοποίησης Byte Pair και εκπαιδεύονται σε ένα σύνολο δεδομένων σημαντικού μεγέθους (αφού εφαρμοστεί ο αλγόριθμος συμβολοποίησης, προκύπτουν σχεδόν 1,4 τρισεκατομμύρια σύμβολα). Χρησιμοποιούν την κλασσική αρχιτεκτονική ενός Αποκωδικοποιητή Μετασχηματιστή, εφαρμόζοντας όμως ορισμένες αλλαγές, καθώς οι τελευταίες είχαν χρησιμοποιηθεί με επιτυχία σε διαφορετικά μοντέλα που είχαν ήδη παρουσιαστεί. Ειδικότερα, εφαρμόζουν κανονικοποίηση για κάθε υπό-τμήμα σε κάθε επίπεδο του Αποκωδικοποιητή Μετασχηματιστή όχι στην έξοδο κάθε υπό-τμήματος αλλά αντιθέτως στην είσοδο του. Επιπλέον, χρησιμοποιούν την συνάρτηση ενεργοποίησης SwiGLU και όχι την ReLU και αντικαθιστούν επίσης τις ενσωματώσεις που χρησιμοποιούνται για την προσφορά πληροφορίας για την θέση των εισόδων, με ορισμένες ενσωματώσεις που χρησιμοποιούνται σε κάθε επίπεδο του μετασχηματιστή, που καλούνται Περιστροφικές Θεσιακές Ενσωματώσεις. Επιπροσθέτως, οι Touvron et al. εξηγούν ότι εκπαιδεύοντας τα προ-εκπαιδευμένα μοντέλα LLaMA σε λίγα μόλις κομμάτια κειμένου τα οποία αποτελούνται από μια οδηγία και την αντίστοιχη απάντηση, οι βελτιώσεις που μπορούμε να παρατηρήσουμε όσον αφορά τις εργασίες με τις οποίες σχετίζονται τα παραπάνω κείμενα, είναι σημαντικές.

Τα Μοντέλα LLaMA 2

Το ίδιο έτος, έπειτα από μερικούς μήνες, παρουσιάζεται η δεύτερη έκδοση των μοντέλων LLaMA, τα **LLaMA 2**, στην εργασία των Touvron et al. [32]. Τα μοντέλα LLaMA 2 έχουν παρόμοια αρχιτεκτονική με αυτά της προηγούμενης έκδοσης, εφαρμόζοντας όμως ορισμένες αλλαγές η οποίες στοχεύουν στην περαιτέρω αύξηση των επιδόσεων των μοντέλων, όπως για παράδειγμα η αύξηση του μεγέθους των συμφραζομένων στο διπλάσιο συγκριτικά με την προηγούμενη έκδοση. Επιπλέον, και πάλι με στόχο την επίτευξη αρτιότερων επιδόσεων, πραγματοποιήθηκαν αλλαγές στη ποιότητα αλλά και στην ποσότητα των δεδομένων που χρησιμοποιήθηκαν στην προ-εκπαίδευση των μοντέλων της δεύτερης έκδοσης.

Τα μοντέλα LLaMA 2 που παρουσιάζονται στην εργασία των Touvron et al. [32] αποτελούνται από δύο εκδοχές. Η πρώτη αποτελεί τα μοντέλα που έχουν ολοκληρώσει μόνο το βήμα προ-εκπαίδευσης και έχουν συνεπώς αποκτήσει σημαντική ποσότητα γνώσεων από τα δεδομένα προ-εκπαίδευσης και είναι ικανά για την παραγωγή της συνέχειας ενός κειμένου που δέχονται ως είσοδο. Η δεύτερη κατηγορία αποτελείται από μοντέλα που έχουν, μέσω ενός επιπλέον βήματος βελτίωσης, προσαρμοστεί ούτως ώστε να παράγουν κείμενο σε ύφος μιας συζήτησης (όπου η είσοδος αποτελεί συνήθως μια ερώτηση και το μοντέλο αναμένεται να έχει ως έξοδο μια σχετική απάντηση). Ειδικότερα, το παραπάνω βήμα της βελτίωσης έχει δύο στάδια. Αρχικά πραγματοποιείται προσαρμογή του μοντέλου για την εργασία στόχο, με χρήση δεδομένων που είναι οργανωμένα σε μορφή ερώτησης και απάντησης, παρόμοια με το βήμα βελτίωσης των

μοντέλων LLaMA,. Στη συνέχεια εφαρμόζεται μια διαδικασία που καλείται *επιβλεπόμενη μάθηση με ανθρώπινη ανατροφοδότηση* [32], προκειμένου το μοντέλο να προσαρμοστεί ακόμα περισσότερο στην εργασία της συνομιλίας αλλά και για να επιτευχθεί ελάττωση επιλογής ορισμένων απαντήσεων από το βελτιωμένο μοντέλο που κρίνονται επικίνδυνες, όσον αφορά το περιεχόμενο τους, ή μη ωφέλιμες ως προς την αντίστοιχη ερώτηση.

Τα Μοντέλα LLaMA 3

Τον Απρίλιο του 2024, ανακοινώθηκαν ορισμένα μοντέλα της τρίτης έκδοσης των μοντέλων LLaMA, τα **LLaMA 3** [33] και τον Ιούλιο του ίδιου έτους ανακοινώθηκαν τα μοντέλα **LLaMA 3.1** [34] που αποτελούν βελτιωμένες εκδόσεις των μοντέλων LLaMA 3, οι οποίες αυξάνουν το μέγιστο επιτρεπτό μέγεθος εισόδου και πετυχαίνουν βελτιωμένες επιδόσεις. Κατά τη διάρκεια της συγγραφής της παρούσας εργασίας έχουν παρουσιαστεί μοντέλα της οικογένειας LLaMA 3 με πλήθος παραμέτρων 8 δισεκατομμυρίων, 70 δισεκατομμυρίων καθώς και 405 δισεκατομμυρίων, με τα μοντέλα του μεγαλύτερου μεγέθους να είναι όπως είδαμε και τα πιο αποτελεσματικά, όσον αφορά την ποιότητα των απαντήσεων. Όπως και τα μοντέλα LLaMA 2, τα μοντέλα LLaMA 3 αποτελούνται από δύο εκδοχές, αυτή των απλώς προ-εκπαιδευμένων μοντέλων και αυτή όπου έχει πραγματοποιηθεί ένα επιπλέον στάδιο βελτίωσης, για την προσαρμογή των μοντέλων στην εργασία της συνομιλίας.

Όσον αφορά την αρχιτεκτονική των μοντέλων LLaMA 3 και LLaMA 3.1, ακολουθούν και αυτά την βασική αρχιτεκτονική της προηγούμενης έκδοσης, εφαρμόζοντας αλλαγές που στοχεύουν στην περαιτέρω αύξηση των επιδόσεων, όπως την εφαρμογή διαφορετικού αλγορίθμου συμβολοποίησης, όπου χρησιμοποιείται μεγαλύτερο λεξικό συμβόλων, μέσω του οποίου επιτυγχάνεται ο αρτιότερος διαχωρισμός της εισόδου σε σύμβολα. Τα μοντέλα LLaMA 3 έχουν επίσης προ-εκπαιδευτεί σε σύνολο δεδομένων μεγέθους σημαντικά μεγαλύτερου από αυτό του αντίστοιχου συνόλου δεδομένων των μοντέλων LLaMA 2 [35].

2.4.5 Μηχανική Προτροπών (Prompt Engineering)

Η μηχανική προτροπών μπορεί να οριστεί ως η διαδικασία όπου μέσω της χρήσης κατάλληλων οδηγιών επιτυγχάνεται η προσαρμογή της λειτουργίας ενός μεγάλου γλωσσικού μοντέλου [36]. Πριν συνεχίσουμε, είναι σκόπιμο να παρουσιάσουμε την μορφή οργάνωσης που συνήθως παρουσιάζει η είσοδος ενός μεγάλου γλωσσικού μοντέλου σε περιπτώσεις χρήσης συζητήσεων. Ειδικότερα, η είσοδος διαχωρίζεται συνήθως σε ένα πλήθος μηνυμάτων όπου το καθένα από αυτά αντιστοιχεί σε κάποιο ρόλο [37], [38]. Παρακάτω εξηγούμε συνοπτικά τρία από τα πιο σημαντικά είδη μηνυμάτων:

- **Μήνυμα Συστήματος**: Το μήνυμα συστήματος περιέχει συνήθως ορισμένες εντολές και οδηγίες που το μοντέλο χρειάζεται να ακολουθεί σε όλη τη διάρκεια της συνομιλίας.

- **Μήνυμα Χρήστη**: Το μήνυμα χρήστη αντιστοιχεί σε μια είσοδο που έχει δώσει ο χρήστης με τον οποίο τον μοντέλο αλληλοεπιδρά.

- **<u>Μήνυμα Βοηθού</u>:** Το μήνυμα αυτό αντιστοιχεί σε μια έξοδο που έχει παραχθεί από το μοντέλο.

Οργανώνοντας τα μηνύματα με αυτόν τον τρόπο μας δίνεται η δυνατότητα να έχουμε ένα εύκολα διαχειρίσιμο ιστορικό της συνομιλίας του χρήστη και του μοντέλου.

<u>Προτροπή Με Χρήση Παραδειγμάτων</u>

Μια τεχνική μηχανικής προτροπών αποτελεί η **Προτροπή Πολλών Προσπαθειών (Few Shot Prompting)**. Σύμφωνα με την τεχνική αυτή, προκειμένου να προσαρμόσουμε ένα μεγάλο γλωσσικό μοντέλο στο να ακολουθεί κάποια συγκεκριμένη οδηγία που περιέχεται σε μια προτροπή προσφέρουμε ως είσοδο σε αυτό, μαζί με την προτροπή, μια σειρά από παραδείγματα του τρόπου με τον οποίο επιθυμούμε να ανταποκριθεί σε αυτή [31], [39]. Σε περίπτωση όπου το πλήθος των παραδειγμάτων είναι μόνο ένα η διαδικασία αυτή αναφέρεται συχνά ως Προτροπή μίας προσπάθειας , ενώ όταν παραθέτουμε την προτροπή χωρίς χρήση παραδειγμάτων ως προτροπή μηδενικών προσπαθειών. Όπως εξηγήσαμε, έχει παρατηρηθεί ότι τα μεγάλα γλωσσικά μοντέλα μεγέθους πολλών παραμέτρων τα οποία έχουν εκπαιδευτεί με κατάλληλο τρόπο, είναι σε θέση να αξιοποιήσουν τα παραδείγματα που τους δίνουμε ώστε πολλές φορές να βελτιώσουν αισθητά την ποιότητα της παραγόμενης εξόδου τους προσαρμόζοντας την στις ανάγκες της εκάστοτε εργασίας.

<u>Προτροπή με Σταδιακή Ανάπτυξη</u>

Μια άλλη τεχνική η οποία είναι ιδιαιτέρως δημοφιλής και έχει παρατηρηθεί ότι είναι ικανή να βελτιώσει πολλές φορές τις επιδόσεις ενός μεγάλου γλωσσικού μοντέλου, αποτελεί η **προτροπή με χρήση σταδιακής ανάπτυξης (Chain of Thought Prompting)**. Σύμφωνα με την τεχνική αυτή, δίνουμε στο μεγάλο γλωσσικό μοντέλο την οδηγία να παράγει τις απαντήσεις του σε στάδια, τροποποιώντας κατάλληλα την προτροπή που του δίνουμε ως είσοδο. Ακολουθώντας με αυτόν τον τρόπο μια σταδιακή διαδικασία ανάπτυξης της εξόδου του και όχι την άμεση παράθεση αυτής, δίνεται στο μοντέλο πολλές φορές η ευκαιρία να καταλήξει στην ακολουθία εξόδου που επιθυμούμε [40].

2.5. Διανυσματικές Βάσεις Δεδομένων

Στην ενότητα αυτή παρουσιάζουμε ένα είδος βάσεων δεδομένων, τις διανυσματικές βάσεις δεδομένων. Στην υπό ενότητα 2.5.1 γίνεται μια εισαγωγή στον τρόπο λειτουργίας των συστημάτων διαχείρισης διανυσματικών βάσεων δεδομένων. Στην υπόενότητα 2.5.2 παρουσιάζουμε τα είδη αλγορίθμων που αξιοποιούνται από τέτοια συστήματα και παρουσιάζουμε τον αλγόριθμο Ιεραρχικών Πλοηγήσιμων Τοπολογιών Μικρού Κόσμου.

2.5.1 Ορισμός

Οι **βάσεις δεδομένων** μπορούν να οριστούν ως ένα σημαντικό πλήθος πληροφοριών. Τη σημερινή εποχή, ο όγκος πληροφοριών που σχηματίζεται μέσω δραστηριοτήτων στο διαδίκτυο, όπως για παράδειγμα η αλληλεπίδραση σε κοινωνικά μέσα δικτύωσης ή ηλεκτρονικές συναλλαγές, αυξάνεται ολοένα και περισσότερο. Η πολυπλοκότητα αλλά και το μέγεθος μιας βάσης δεδομένων που περιέχει τέτοιου είδους πληροφορία είναι συνεπώς ιδιαίτερα περίπλοκη. Προκειμένου να διαχειριστούμε μια βάση δεδομένων με αποτελεσματικό τρόπο, χρησιμοποιούμε συνήθως ένα σύστημα διαχείρισης το οποίο είναι υπεύθυνο για την ορθή διαχείριση μιας βάσης δεδομένων. Ειδικότερα, το σύστημα αυτό είναι υπεύθυνο για την αποτελεσματική αποθήκευση της πληροφορίας που περιέχεται σε μια βάση δεδομένων, μέσω ορισμένων τεχνικών, καθώς όμως και για την προσφορά της δυνατότητας του να μπορεί κανείς να ανακτήσει και γενικότερα να επεξεργαστεί τις πληροφορίες που περιέχονται στη βάση αυτή, με όσο το δυνατόν πιο βέλτιστο τρόπο. Ένα τέτοιο σύστημα διαχείρισης μαζί με την βάση δεδομένων που διαχειρίζεται, καλείται **σύστημα διαχείρισης βάσεων δεδομένων** [41].

Έχουν αναπτυχθεί διάφορα τέτοια συστήματα που εξυπηρετούν διαφορετικές ανάγκες οι οποίες προφανώς καθορίζονται σε σημαντικό βαθμό από την δομή των δεδομένων που περιέχονται στην εκάστοτε βάση δεδομένων, καθώς η μορφή που λαμβάνουν τα δεδομένα που περιέχονται στη βάση συχνά απαιτεί εφαρμογή διαφορετικών τεχνικών αποθήκευσης και διαχείρισης των πρώτων, αλλά και από τις ανάγκες των χρηστών που εξυπηρετεί ένα τέτοιο σύστημα. Στην ενότητα αυτή θα εξηγήσουμε τον τρόπο με τον οποίο λειτουργεί μια κατηγορία των συστημάτων αυτών, που ονομάζονται **Συστήματα Διαχείρισης Διανυσματικών Βάσεων Δεδομένων**. Στα συστήματα αυτά, η βάση δεδομένων περιέχει εγγραφές που αποτελούνται από ένα διάνυσμα υψηλών συνήθως διαστάσεων και ενδεχομένως από δεδομένα που σχετίζονται με το διάνυσμα αυτό, γνωστά και ως μετά-δεδομένα. Μια τέτοια βάση δεδομένων καλείται συχνά απλώς Διανυσματική Βάση Δεδομένων [42], [43].

Ένα σύστημα διαχείρισης για μια τέτοια διανυσματική βάση έχει συνήθως τον στόχο, πέρα από την επεξεργασία των δεδομένων (είσοδος, ανανέωση, διαγραφή, ανάγνωση), να αποθηκεύσει τα διανύσματα που περιέχονται σε αυτή με τρόπο τέτοιο ώστε να μας επιτρέπεται, έχοντας στη διάθεση μας ένα διάνυσμα-ερώτηση, να ανακτήσουμε διανύσματα παρόμοια με αυτό του διανύσματος-ερώτησης. Προκειμένου να επιτευχθεί ο στόχος αυτός με ικανοποιητική ταχύτητα, έχοντας όσο το δυνατόν λιγότερη μείωση της ποιότητας της αναζήτησης, το σύστημα διαχείρισης εφαρμόζει συνήθως κάποιον προσεγγιστικό αλγόριθμο αναζήτησης πλησιέστερων γειτόνων [42], [43].

2.5.2 Αλγόριθμοι που Εφαρμόζονται από Συστήματα Διαχείρισης Διανυσματικών Βάσεων Δεδομένων

Όπως αναφέραμε, τα συστήματα διαχείρισης διανυσματικών βάσεων δεδομένων χρησιμοποιούν συχνά μια κατηγορία αλγορίθμων, τους αλγορίθμους προσεγγιστικής αναζήτησης πλησιέστερων γειτόνων, προκειμένου να μπορέσουν να αναζητήσουν αποδοτικά τα δεδομένα που περιέχονται στη διανυσματική βάση τα οποία είναι όσο το δυνατόν πιο όμοια με το διάνυσμα-ερώτηση που ορίζει ο χρήστης του συστήματος.

Αναζήτηση κ-Πλησιέστερων Γειτόνων

Το **πρόβλημα του πλησιέστερου γείτονα** αποτελεί ένα πρόβλημα που έχει απασχολήσει σημαντικά την ερευνητική κοινότητα ανά τα χρόνια. Ένας ορισμός που μπορεί να δοθεί για το πρόβλημα αυτό είναι ο εξής: *Σε έναν πολυδιάστατο χώρο όπου έχουμε ένα πλήθος δεδομένων αναζητούμε για ένα άλλο δεδομένο, που ονομάζεται δεδομένο-ερώτημα, το δεδομένο που βρίσκεται εγγύτερα στο δεδομένο-ερώτημα* [44]. Με αντίστοιχο τρόπο μπορούμε να ορίσουμε το **πρόβλημα των κ-πλησιέστερων γειτόνων**, όπου ψάχνουμε πλέον τα κ κοντινότερα δεδομένα του δεδομένου-ερωτήματος.

Στην υπό-ενότητα 2.1.2 παρουσιάσαμε τον αλγόριθμο των κ πλησιέστερων γειτόνων, που επιλύει το πρόβλημα των κ-πλησιέστερων γειτόνων χρησιμοποιώντας μια μετρική απόστασης για να προσδιορίσει τα εγγύτερα δεδομένα για κάθε νέο δεδομένο που λαμβάνει ως είσοδο. Εκτός από τον παραπάνω, έχουν υλοποιηθεί και άλλοι αλγόριθμοι που αντιμετωπίζουν το πρόβλημα των κ-πλησιέστερων γειτόνων οι οποίοι εφαρμόζονται για την επίλυση προβλημάτων που προκύπτουν σε πληθώρα πεδίων, όπως αυτό της ανάκτησης πληροφορίας και της μηχανικής μάθησης.

Προσεγγιστικοί Αλγόριθμοι Πλησιέστερων Γειτόνων

Έχει παρατηρηθεί ότι η αναζήτηση των κ-πλησιέστερων γειτόνων σε χώρους υψηλών διαστάσεων, με τον υπολογισμό της απόστασης του δεδομένου-ερωτήματος μεταξύ των υπόλοιπων δεδομένων με εξαντλητικό τρόπο προκειμένου να προσδιορίσουμε τα πλησιέστερα δεδομένα στο δεδομένο-ερώτημα, δεν είναι συνήθως αποτελεσματική την σημερινή εποχή, όπου το πλήθος των δεδομένων που έχουμε στη διάθεση μας είναι πολλές φορές σημαντικού μεγέθους. Οι **προσεγγιστικοί αλγόριθμοι πλησιέστερων γειτόνων** προσπαθούν να καταπολεμήσουν αυτό το πρόβλημα της υψηλής υπολογιστικής πολυπλοκότητας αναζητώντας, όπως υποδηλώνει και η ονομασία τους, τα κ κοντινότερα δεδομένα του δεδομένου-ερωτήματος με μια πιθανότητα να πραγματοποιηθεί εσφαλμένη πρόβλεψη, η οποία ωστόσο είναι όσο το δυνατόν μικρότερη [45].

Στο σημείο αυτό είναι σκόπιμο να αναλύσουμε έναν αλγόριθμο που ανήκει στην κατηγορία των προσεγγιστικών αλγορίθμων πλησιέστερων γειτόνων, τον αλγόριθμο **Ιεραρχικών Πλοηγήσιμων Τοπολογιών Μικρού Κόσμου (Hierarchical Navigable Small Worlds, HNSW)**. Ο αλγόριθμος αυτός παρουσιάζεται στην εργασία των Malkov et al. [45] και χρησιμοποιείται από διάφορες υλοποιήσεις συστημάτων διαχείρισης διανυσματικών βάσεων δεδομένων, για την δημιουργία ενός ευρετηρίου που επιτρέπει ιδιαίτερα αποτελεσματικές επιδόσεις όσον αφορά την αναζήτηση κ-πλησιέστερων γειτόνων για ένα διάνυσμα-ερώτηση.

Η υλοποίηση του αλγορίθμου HNSW που παρουσιάζουν στην εργασία τους οι Malkov et al. [45] μπορεί να διαχωριστεί σε δύο ξεχωριστά βήματα, αυτό της κατασκευής και αυτό της αναζήτησης.

Το Βήμα Κατασκευής

Κατά το **βήμα της κατασκευής** ο αλγόριθμος HNSW σχηματίζει ένα δίκτυο που αποτελείται από μια σειρά γραφημάτων τα οποία είναι οργανωμένα μεταξύ τους σε μορφή επιπέδων. Προκειμένου να επιτευχθεί αυτό, ο αλγόριθμος εισάγει στο δίκτυο τα διαθέσιμα δεδομένα που του δίνονται ως είσοδος ένα τη φορά, τροποποιώντας κατάλληλα το δίκτυο για κάθε εισαγωγή που πραγματοποιείται. Πιο συγκεκριμένα, η εισαγωγή του κάθε δεδομένου στο γράφημα μπορεί να διαχωριστεί σε δύο υπό-βήματα.

Πριν εξηγήσουμε τα υπό-βήματα αυτά αναλυτικότερα, χρειάζεται να αναφέρουμε τον τρόπο με τον οποίο πραγματοποιείται η αναζήτηση γειτόνων ενός κόμβου εισόδου για ένα επίπεδο του δικτύου στον αλγόριθμο HNSW. Πιο συγκεκριμένα, η διαδικασία που ακολουθείται περιγράφεται ως εξής: ξεκινώντας από μια λίστα τρεχόντων εγγύτερων γειτόνων που περιέχει αρχικά τους κόμβους από την οποία ξεκίνησε η αναζήτηση για το επίπεδο στο οποίο βρισκόμαστε, εξάγονται οι γειτονικοί κόμβοι των κόμβων που περιέχονται στη λίστα αυτή και προστίθενται σε ένα σύνολο εξέτασης. Έπειτα, σε κάθε βήμα της αναζήτησης λαμβάνουμε το πλησιέστερο στο κόμβο εισόδου κόμβο του συνόλου εξέτασης και στη συνέχεια, εξετάζουμε κάθε κόμβο που ανήκει στη γειτονιά του επιλεγμένου κόμβου. Σε περίπτωση που εντοπίσουμε κάποιον εγγύτερο κόμβο ως προς το κόμβο εισόδου, ανανεώνουμε κατάλληλα τη λίστα εγγύτερων γειτόνων. Το πλήθος των εγγύτερων γειτόνων που αναζητούμε καθορίζεται από μια παράμετρο που συμβολίζουμε ως ef. Μέσω του ελέγχου ορισμένων συνθηκών, επιτυγχάνεται επίσης η παράλειψη της εξέτασης κόμβων που η απόσταση τους από τον κόμβο εισόδου είναι μεγαλύτερη από αυτή που παρουσιάζουν οι τρέχοντες εγγύτεροι γείτονες, για το εκάστοτε βήμα της αναζήτησης. Η διαδικασία που περιγράψαμε φαίνεται αναλυτικότερα στην Εικόνα 14. Παρακάτω θα αναφερόμαστε στη διαδικασία αυτή ως **διαδικασία αναζήτησης γειτόνων σε ένα επίπεδο**.

SEARCH-LAYER(q, ep, ef, l_c)
Input: query element q, enter points ep, number of nearest to q elements to return ef, layer number l_c
Output: ef closest neighbors to q
1 $v \leftarrow ep$ // set of visited elements
2 $C \leftarrow ep$ // set of candidates
3 $W \leftarrow ep$ // dynamic list of found nearest neighbors
4 **while** $|C| > 0$
5 $c \leftarrow$ extract nearest element from C to q
6 $f \leftarrow$ get furthest element from W to q
7 **if** $distance(c, q) > distance(f, q)$
8 **break** // all elements in W are evaluated
9 **for** each $e \in neighbourhood(c)$ at layer l_c // update C and W
10 **if** $e \notin v$
11 $v \leftarrow v \cup e$
12 $f \leftarrow$ get furthest element from W to q
13 **if** $distance(e, q) < distance(f, q)$ or $|W| < ef$
14 $C \leftarrow C \cup e$
15 $W \leftarrow W \cup e$
16 **if** $|W| > ef$
17 remove furthest element from W to q
18 **return** W

Εικόνα 14: Διαδικασία Αναζήτησης Γειτόνων σε Επίπεδο [45]

Έχοντας εξηγήσει την διαδικασία αναζήτησης, μπορούμε να εξηγήσουμε το **πρώτο υπό-βήμα του βήματος κατασκευής**. Ξεκινώντας από το μέγιστο δυνατό επίπεδο υπολογίζουμε τον κόμβο που βρίσκεται πιο κοντά στο κόμβο που εισάγεται, μέσω της διαδικασίας αναζήτησης γειτόνων σε ένα επίπεδο. Στη συνέχεια, ο αλγόριθμος πραγματοποιεί παρόμοια αναζήτηση για το αμέσως κατώτερο επίπεδο, χρησιμοποιώντας το κοντινότερο κόμβο που βρέθηκε προηγουμένως ως κόμβο εκκίνησης. Έπειτα, έχουμε επανάληψη της διαδικασίας αυτής έως ότου ο αλγόριθμος βρεθεί στο λ-οστό επίπεδο. Ο αριθμός λ επιλέγεται με τυχαίο τρόπο, χρησιμοποιώντας μια κατανομή πιθανοτήτων σε συνδυασμό με μια σταθερά κανονικοποίησης τέτοιες ώστε οι τιμές του λ να είναι συνήθως μικρές, καθώς είναι επιθυμητό να δημιουργούνται συνδέσεις μεταξύ των κόμβων κυρίως στα κατώτερα επίπεδα του γραφήματος και όπως θα δούμε παρακάτω, οι συνδέσεις αυτές πραγματοποιούνται μόνο κατά το δεύτερο υπό-βήμα.

INSERT(*hnsw*, *q*, *M*, M_{max}, *efConstruction*, m_L)
Input: multilayer graph *hnsw*, new element *q*, number of established connections *M*, maximum number of connections for each element per layer M_{max}, size of the dynamic candidate list *efConstruction*, normalization factor for level generation m_L
Output: update *hnsw* inserting element *q*
1 $W \leftarrow \emptyset$ // list for the currently found nearest elements
2 *ep* ← get enter point for *hnsw*
3 $L \leftarrow$ level of *ep* // top layer for *hnsw*
4 $l \leftarrow \lfloor -\ln(unif(0..1)) \cdot m_L \rfloor$ // new element's level
5 **for** $l_c \leftarrow L \ldots l+1$
6 $W \leftarrow$ SEARCH-LAYER(q, *ep*, *ef*=1, l_c)
7 *ep* ← get the nearest element from *W* to *q*
8 **for** $l_c \leftarrow \min(L, l) \ldots 0$
9 $W \leftarrow$ SEARCH-LAYER(q, *ep*, *efConstruction*, l_c)
10 *neighbors* ← SELECT-NEIGHBORS(q, *W*, *M*, l_c) // alg. 3 or alg. 4
11 add bidirectionall connectionts from *neighbors* to *q* at layer l_c
12 **for** each $e \in$ *neighbors* // shrink connections if needed
13 *eConn* ← *neighbourhood*(*e*) at layer l_c
14 **if** $|eConn| > M_{max}$ // shrink connections of *e*
// if $l_c = 0$ then $M_{max} = M_{max0}$
15 *eNewConn* ← SELECT-NEIGHBORS(*e*, *eConn*, M_{max}, l_c)
// alg. 3 or alg. 4
16 set *neighbourhood*(*e*) at layer l_c to *eNewConn*
17 *ep* ← *W*
18 **if** $l > L$
19 set enter point for *hnsw* to *q*

Εικόνα 15: Το Βήμα Κατασκευής του Αλγορίθμου HNSW [45]

Από το λ-οστό επίπεδο και ύστερα, εφαρμόζεται το **δεύτερο υπό-βήμα του βήματος κατασκευής** του αλγορίθμου. Ειδικότερα, στο στάδιο αυτό το πλήθος των πλησιέστερων γειτόνων του κόμβου εισόδου που αναζητούνται καθορίζεται πλέον από μια ειδική παράμετρο (ef_c). Για κάθε επίπεδο εφαρμόζεται η εξής διαδικασία, κατά την οποία εισάγεται ο κόμβος εισόδου στο αντίστοιχο γράφημα: αναζητούνται οι ef_c πλησιέστεροι γείτονες του κόμβου εισόδου και στη συνέχεια εξετάζεται το ενδεχόμενο σύνδεσης των γειτόνων αυτών με τον κόμβο εισόδου, μέσω μιας ευριστικής συνάρτησης. Σημειώνουμε επίσης ότι σε κάθε επίπεδο δίνονται ως κόμβοι εκκίνησης οι πλησιέστεροι γείτονες που υπολογίστηκαν στο προηγούμενο επίπεδο. Όταν ο αλγόριθμος ολοκληρώσει τη διαδικασία αυτή για το κατώτερο επίπεδο, το δεύτερο υπό-βήμα έχει ολοκληρωθεί.

Το Βήμα Αναζήτησης

Έχοντας κατασκευάσει πλέον το δίκτυο των κόμβων μας κατά το βήμα κατασκευής, μπορούμε πλέον να χρησιμοποιήσουμε το **βήμα αναζήτησης** του αλγορίθμου HNSW. Πιο αναλυτικά, το βήμα αναζήτησης αξιοποιεί το δίκτυο αυτό ξεκινώντας και αυτό από το ανώτατο επίπεδο. Ακολουθείται διαδικασία όμοια με αυτή του πρώτου υπό-βήματος του βήματος κατασκευής. Πραγματοποιείται δηλαδή αναζήτηση του εγγύτερου κόμβου ως προς τον κόμβο εισόδου σε κάθε επίπεδο ώστε να γίνει χρησιμοποιηθεί ως σημείο εκκίνησης για το επόμενο επίπεδο. Όταν φτάσουμε στο τελευταίο επίπεδο, τότε αναζητούνται οι ef_s πλησιέστεροι γείτονες του δεδομένου εισόδου και τέλος επιστρέφονται οι x πιο κοντινοί από εκείνους, όπου x είναι μια παράμετρος που μπορούμε να ορίσουμε ανάλογα με το πόσους κόμβους / δεδομένα επιθυμούμε να ανακτήσουμε. Η παράμετρος ef_s έχει εισαχθεί προκειμένου να μας επιτρέψει να ελέγξουμε την ποιότητα του βήματος αυτού.

K-NN-SEARCH(*hnsw*, *q*, *K*, *ef*)
Input: multilayer graph *hnsw*, query element *q*, number of nearest neighbors to return *K*, size of the dynamic candidate list *ef*
Output: *K* nearest elements to *q*
1 $W \leftarrow \emptyset$ // set for the current nearest elements
2 *ep* ← get enter point for *hnsw*
3 $L \leftarrow$ level of *ep* // top layer for *hnsw*
4 **for** $l_c \leftarrow L \dots 1$
5 $W \leftarrow$ SEARCH-LAYER(*q*, *ep*, *ef*=1, l_c)
6 *ep* ← get nearest element from *W* to *q*
7 $W \leftarrow$ SEARCH-LAYER(*q*, *ep*, *ef*, l_c =0)
8 **return** *K* nearest elements from *W* to *q*

Εικόνα 16: Το Βήμα Αναζήτησης του Αλγορίθμου HNSW [45]

3. Chatbots

Στο κεφάλαιο αυτό παρουσιάζουμε πληροφορίες για τα συστήματα chatbot. Καταρχάς, κάνουμε μια σύντομη εισαγωγή σχετικά με τον στόχο των συστημάτων αυτών. Στη συνέχεια αναλύουμε ορισμένες κατηγορίες στις οποίες μπορούμε να διαχωρίσουμε τα συστήματα chatbot με βάση την ήδη υπάρχουσα βιβλιογραφία, ανάλογα με τον τρόπο που λειτουργούν και τις τεχνικές τις οποίες εφαρμόζουν. Στην συνέχεια, αναλύουμε την εξέλιξη των chatbot και των δυνατοτήτων τους και τέλος αναλύουμε ορισμένες τεχνικές που έχουν εφαρμοστεί από τέτοιου είδους συστήματα.

3.1 Εισαγωγή

3.1.1 Ορισμός

Το ενδιαφέρον γύρω από τα συστήματα chatbot, λόγω και των σημαντικών εξελίξεων που έχουν επιτευχθεί στο πεδίο της μηχανικής μάθησης τα τελευταία χρόνια, ολοένα και αυξάνεται. Μπορούμε να ορίσουμε ένα σύστημα chatbot ως ένα σύνολο λογισμικού που έχει ως στόχο την παροχή μιας εμπειρίας στον τελικό χρήστη του η οποία προσομοιώνει όσο το δυνατόν πιο βέλτιστα αυτή της συζήτησης μεταξύ δύο ανθρώπων [46]. Προκειμένου να επιτευχθεί ο στόχος τους, τα chatbots χρειάζεται να είναι ικανά να κατανοούν την είσοδο που δέχονται από τον χρήστη και να μπορούν στη συνέχεια να την επεξεργαστούν κατάλληλα ώστε να παράγουν μια χρήσιμη και ποιοτική απάντηση. Συνεπώς, τα chatbots είναι συχνά αναγκαίο να σχεδιαστούν με συγκεκριμένο τρόπο και να εφαρμόσουν ορισμένες τεχνικές ώστε να είναι σε θέση να πετύχουν τα παραπάνω.

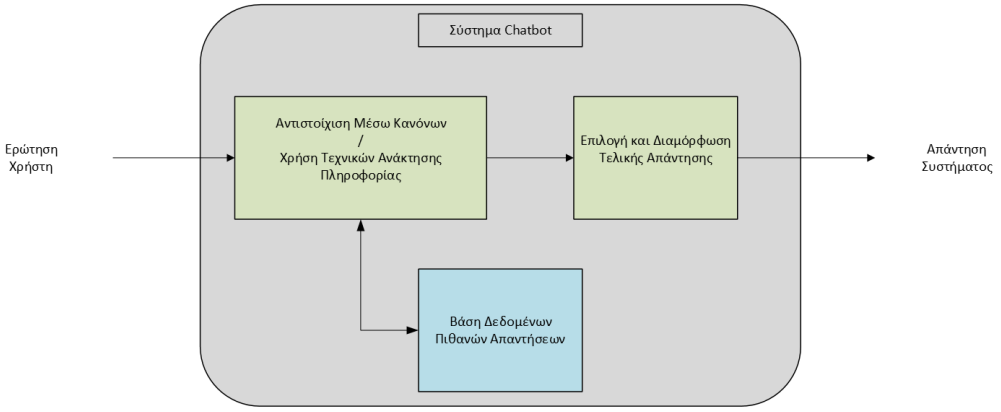
3.1.2 Είδη Συστημάτων Chatbot

Τα chatbots είναι συχνά σύνθετα συστήματα που μπορούν να χρησιμοποιηθούν σε ποικιλία εφαρμογών. Συνεπώς, η κατηγοριοποίηση τους είναι πολλές φορές δύσκολο να πραγματοποιηθεί με βάση ενός μόνο κριτηρίου. Ένα κριτήριο που έχει χρησιμοποιηθεί για το σκοπό αυτό είναι **ο τρόπος με τον οποίο είναι δομημένη η αρχιτεκτονική ενός chatbot**. Ειδικότερα, με βάση το κριτήριο αυτό, στη βιβλιογραφία διακρίνονται οι εξής τρεις κατηγορίες chatbot: αυτά που βασίζονται σε κανόνες, αυτά που βασίζονται σε ανάκτηση δεδομένων και αυτά που βασίζονται στην παραγωγή της απάντησης τους. Ωστόσο, λόγω του ότι ουσιαστικά οι δύο πρώτες κατηγορίες παρουσιάζουν όμοιο τρόπο λειτουργίας, μπορούμε να καταλήξουμε στη διάκριση δύο πιο γενικών κατηγοριών. Η πρώτη αποτελεί τα **συστήματα chatbot που αξιοποιούν τεχνικές ανάκτησης** και η δεύτερη **τεχνικές παραγωγής κειμένου**, προκειμένου να απαντήσουν στο ερώτημα που τίθεται από τον χρήστη [46], [47], [48].

Συστήματα Chatbot που Βασίζονται σε Τεχνικές Ανάκτησης

Στην κατηγορία αυτή μπορούμε να κατατάξουμε ένα chatbot, το οποίο έχει στη διάθεση του μια σειρά από απαντήσεις ή τμήματα απαντήσεων, οι οποίες έχουν προκαθοριστεί από τον κατασκευαστή του συστήματος. Προκειμένου να είναι ικανό να

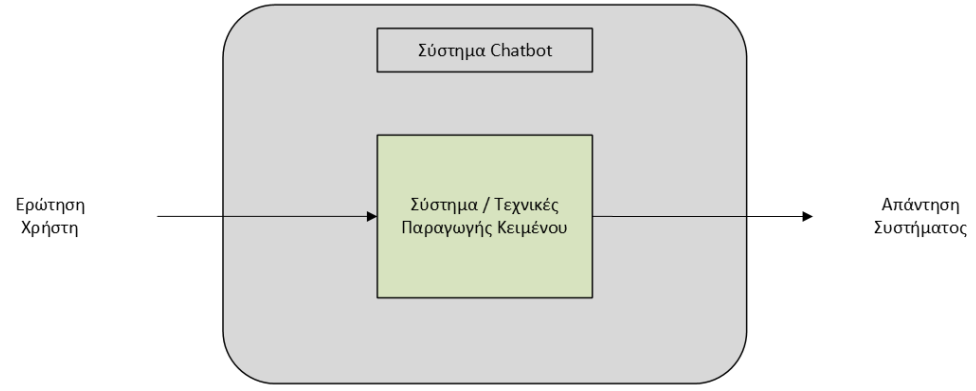
απαντήσει σε μια ερώτηση του χρήστη του, ένα τέτοιο σύστημα αξιοποιεί μία ή περισσότερες τεχνικές προκειμένου να αποφασίσει ποια από τις απαντήσεις που έχει στη διάθεση του είναι η καταλληλότερη. Για παράδειγμα, ένα σύστημα που ανήκει στη κατηγορία αυτή μπορεί να χρησιμοποιήσει ορισμένους κανόνες ώστε να καθορίσει την επιλεγόμενη απάντηση. Επίσης, πολλά συστήματα της κατηγορίας αυτής αξιοποιούν συστήματα διαχείρισης βάσεων δεδομένων για την αποτελεσματική αποθήκευση ενός συνόλου απαντήσεων, τεχνικές ανάκτησης πληροφορίας ώστε να ανακτήσουν σχετικές και χρήσιμες απαντήσεις που περιέχονται στην βάση δεδομένων τους καθώς και την εφαρμογή επιπλέον τεχνικών ώστε να καταλήξουν στην τελική απάντηση που προσφέρουν στον χρήστη τους.



Εικόνα 17:Σύστημα Chatbot που Βασίζεται σε Τεχνικές Ανάκτησης

Συστήματα Chatbot που Βασίζονται σε Τεχνικές Παραγωγής Κειμένου

Στην δεύτερη κατηγορία μπορούμε να εντάξουμε τα chatbot τα οποία, σε αντίθεση με αυτά που ανήκουν στην προηγούμενη κατηγορία, λαμβάνοντας μια είσοδο εφαρμόζουν τεχνικές και αλγορίθμους που στοχεύουν στην παραγωγή μιας νέας, μη προκαθορισμένης απάντησης. Τα συστήματα αυτής της κατηγορίας τα οποία έχουν αναπτυχθεί πρόσφατα αξιοποιούν τις περισσότερες φορές κατάλληλα εκπαιδευμένα μοντέλα μηχανικής μάθησης διαφόρων αρχιτεκτονικών, καθώς όπως είδαμε και στο Κεφάλαιο 2, πετυχαίνουν ιδιαιτέρως καλά αποτελέσματα σε εργασίες όπως η παραγωγή κειμένου. Χρησιμοποιώντας τέτοιου είδους τεχνικές, τα συστήματα αυτής της κατηγορίας είναι συχνά σε θέση να λάβουν υπόψιν τη γενικότερη θεματολογία της συνομιλίας μεταξύ αυτών και του χρήστη τους, προσφέροντας τους έτσι την δυνατότητα να απαντήσουν πιο ευέλικτα και αποτελεσματικά σε ένα πιο ευρύ σύνολο ερωτήσεων.

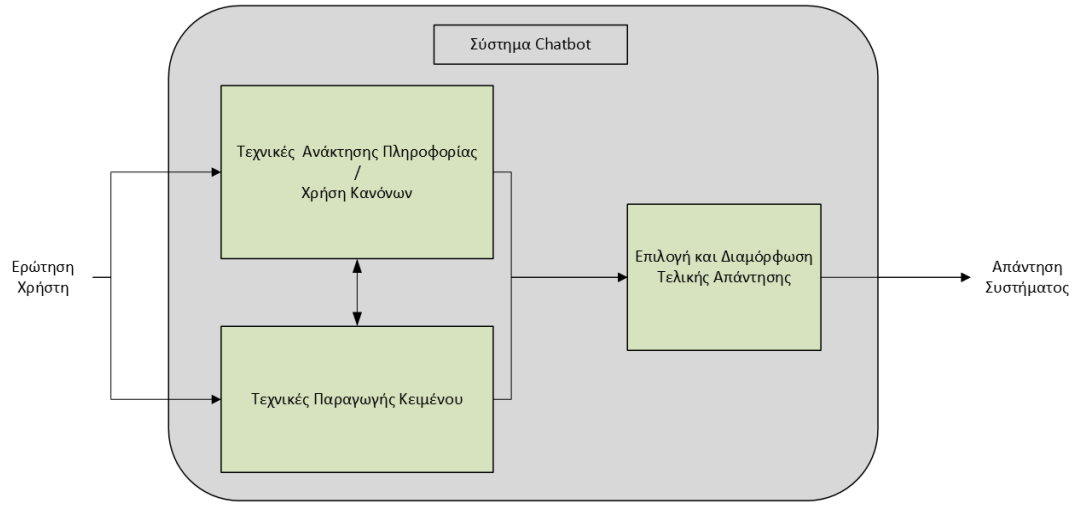


Εικόνα 18: Σύστημα Chatbot που Βασίζεται σε Τεχνικές Παραγωγής Κειμένου

Αξίζει να σημειωθεί ότι τα chatbot που ανήκουν στην πρώτη κατηγορία, έχουν πολλές φορές λιγότερες δυνατότητες συγκριτικά με αυτά της δεύτερης κατηγορίας. Αυτό οφείλεται στο ότι οι απαντήσεις των πρώτων περιορίζονται στο πλήθος των διαθέσιμων απαντήσεων που περιέχονται στη βάση δεδομένων τους, ενώ τα chatbot της δεύτερης κατηγορίας όπως αναφέραμε βασίζονται συνήθως σε μοντέλα μηχανικής μάθησης τα οποία προσφέρουν μεγαλύτερη ευελιξία στο σύστημα. Ωστόσο, τα συστήματα chatbot της πρώτης κατηγορίας έχουν το πλεονέκτημα του ότι ο σχεδιαστής είναι σε θέση να ελέγξει πιο εύκολα το περιεχόμενο των παραγόμενων απαντήσεων του συστήματος. Επίσης, τα συστήματα της πρώτης κατηγορίας απαιτούν τις περισσότερες φορές μικρό ή ακόμα και καθόλου χρόνο εκπαίδευσης, ενώ τα μοντέλα μηχανικής μάθησης που αξιοποιούνται συνήθως στα συστήματα της δεύτερης κατηγορίας χρειάζονται κατάλληλη εκπαίδευση προκειμένου να είναι σε θέση να παράγουν την απάντηση του συστήματος, μια διαδικασία η οποία ενδεχομένως να είναι ιδιαίτερα ακριβή σε χρόνο και πόρους ανάλογα με την αρχιτεκτονική και την πολυπλοκότητα του εκάστοτε συστήματος [46], [47], [48].

Υβριδικά Συστήματα Chatbot

Τα συστήματα chatbot που έχουν σχεδιαστεί τα τελευταία χρόνια συχνά εφαρμόζουν πολλές και ενίοτε σύνθετες τεχνικές, τόσο ανάκτησης πληροφορίας όσο και παραγωγής κειμένου. Με αυτό τον τρόπο γίνεται προσπάθεια να αξιοποιηθούν τα πλεονεκτήματα που προσφέρεται από το κάθε είδος των chatbot των δύο κατηγοριών που αναλύσαμε. Τα chatbot που επιχειρούν το παραπάνω καλούνται συχνά **υβριδικά chatbots**, καθώς μπορούν να ενταχθούν σε παραπάνω από μια από τις κατηγορίες που αναφέραμε. Το σύστημα chatbot που υλοποιήσαμε στην παρούσα εργασία θα μπορούσε να ενταχθεί στη κατηγορία αυτή καθώς όπως θα δούμε στο κεφάλαιο 5, η επιλεγμένη αρχιτεκτονική αξιοποιεί τόσο ανάκτηση δεδομένων όσο και μοντέλων παραγωγής κειμένου για την παραγωγή της τελικής απάντησης του chatbot.



Εικόνα 19: Υβριδικό Σύστημα Chatbot

3.2 Ανάπτυξη Συστημάτων Chatbot

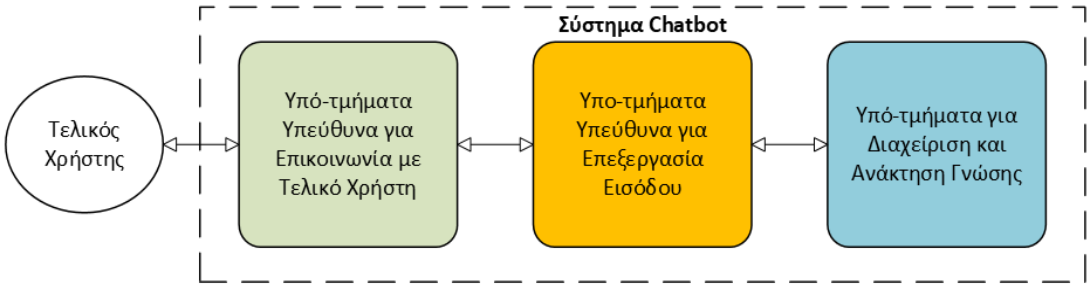
3.2.1 Εξέλιξη Συστημάτων Chatbot ανά τα Χρόνια

Ο σχεδιασμός συστημάτων chatbot αποτελεί μια εργασία της οποίας οι ρίζες εντοπίζονται αρκετά χρόνια στο παρελθόν. Ειδικότερα, το chatbot **ELIZA**, που αποτελεί ένα από τα παλαιότερα γνωστά chatbot, παρουσιάζεται το 1966. Το ELIZA εφαρμόζει ορισμένους κανόνες προκειμένου να επεξεργαστεί το κείμενο εισόδου με κατάλληλο τρόπο. Αξιοποιώντας τα αποτελέσματα της επεξεργασίας αυτής, το σύστημα εντοπίζει μέσα από ένα σύνολο απαντήσεων την καταλληλότερη από αυτές. Η μορφή των απαντήσεων αυτών έχει μια γενική μορφή την οποία το σύστημα, λαμβάνοντας υπόψιν της πληροφορίες που έλαβε από την είσοδο του, διαμορφώνει κατάλληλα και έπειτα τις παρουσιάζει στον χρήστη [47], [49].

Μετέπειτα, μέχρι και την σημερινή περίοδο έχει αναπτυχθεί ένα αξιοσημείωτο πλήθος συστημάτων chatbot, τα οποία έχουν αξιοποιήσει διάφορες τεχνικές. Ακολουθώντας παρόμοια λογική με το chatbot ELIZA, έχουν παρουσιαστεί διάφορα συστήματα τα οποία αξιοποιούν εργαλεία και γλώσσες που υποστηρίζουν τον ορισμό κανόνων προκειμένου να επιτευχθεί η παραγωγή των απαντήσεων τους. Ωστόσο, τα πιο πρόσφατα chatbots αξιοποιούν τις υπόλοιπες τεχνικές που αναφέραμε στην υπό-ενότητα 3.1.2. Δηλαδή, χρησιμοποιούν συνήθως τα υψηλών δυνατοτήτων μοντέλα που έχουν πλέον αναπτυχθεί στο πεδίο της μηχανικής μάθησης, καθώς και τεχνικές ανάκτησης δεδομένων [46], [48].

3.2.2 Η Γενική Αρχιτεκτονική Ενός Συστήματος Chatbot

Τα συστήματα chatbot χρειάζεται πολλές φορές να έχουν σύνθετη αρχιτεκτονική που εφαρμόζει μια πληθώρα τεχνικών προκειμένου να είναι αποτελεσματικά. Οι τεχνικές που χρησιμοποιεί ένα chatbot ποικίλλουν, ανάλογα και με το στόχο που έχει το εκάστοτε σύστημα. Μπορούμε όμως να διακρίνουμε μια **γενική αρχιτεκτονική** που αποτελείται από ορισμένα υπό-τμήματα, όπου το καθένα από αυτά είναι υπεύθυνο για ορισμένες από τις λειτουργίες που απαιτείται να εκτελεί ένα chatbot [48].



Εικόνα 20: Αρχιτεκτονική ενός Συστήματος Chatbot, Βασισμένη σε Υπό-τμήματα

Πιο συγκεκριμένα, ορισμένες αρχιτεκτονικές που έχουν τη μορφή υπό-τμημάτων που έχουν παρουσιαστεί στη βιβλιογραφία [47], [48], διαχωρίζουν την ευθύνη της επικοινωνίας με το τελικό χρήστη, την επεξεργασία της εισόδου με κατάλληλο τρόπο και την διαχείριση και ανάκτηση της γνώσης που έχει στη διάθεση του το σύστημα chatbot, αναθέτοντας τη καθεμία από αυτές σε μια ομάδα υπό-τμημάτων. Κάθε τέτοια

ομάδα μπορεί να αποτελείται από ένα ή περισσότερα τμήματα, ανάλογα και με τις απαιτήσεις της εκάστοτε εφαρμογής. Ένα γενικό παράδειγμα μιας αρχιτεκτονικής υπότμημάτων για συστήματα chatbot φαίνεται στην Εικόνα 20.

3.3. Προηγούμενες Τεχνικές Επεξεργασίας Εισόδου σε Συστήματα Chatbot

Όπως εξηγήσαμε, τα πιο σύγχρονα συστήματα chatbot έχουν βελτιώσει αξιοσημείωτα τις δυνατότητες τους, χρησιμοποιώντας πληθώρα μοντέλων μηχανικής μάθησης τα οποία είναι ιδιαίτερα βοηθητικά στην διαδικασία παραγωγής απαντήσεων. Παραδείγματα τέτοιων μοντέλων μηχανικής μάθησης αποτελούν τα αναδρομικά νευρωνικά δίκτυα, τα Αναδρομικά Νευρωνικά Δίκτυα Μακράς και Βραχείας Μνήμης αλλά και οι μετασχηματιστές, την λειτουργία τον οποίων αναλύσαμε στο κεφάλαιο 2. Τα τελευταία χρόνια, έχουν επίσης αναπτυχθεί εφαρμογές chatbots [50], [51] που χρησιμοποιούν μεγάλα γλωσσικά μοντέλα ως το κύριο εργαλείο για την παραγωγή απαντήσεων. Στην ενότητα αυτή θα εξηγήσουμε ορισμένες τεχνικές οι οποίες έχουν αξιοποιηθεί στο παρελθόν προκειμένου να επιτραπεί σε συστήματα chatbot να χρησιμοποιήσουν αποτελεσματικά την είσοδο την οποία δέχονται από τον χρήστη.

3.3.1 Ενσωματώσεις Λέξεων

Όπως αναφέραμε και στο Κεφάλαιο 2, οι ενσωματώσεις λέξεων αποτελούν μια ιδιαιτέρως χρήσιμη τεχνική με την οποία μπορούμε να αναπαραστήσουμε λέξεις ως διανύσματα. Στο σημείο αυτό κρίνεται σκόπιμο να αναφέρουμε και έναν άλλο ιδιαιτέρως δημοφιλή τρόπο με τον οποίο μας δίνεται η δυνατότητα να εξάγουμε τέτοιου είδους ενσωματώσεις, τις ενσωματώσεις **GloVe**.

Το **GloVe**, το οποίο παρουσιάζουν στην εργασία τους το 2014 οι Pennington et al. [52], αποτελεί ένα μοντέλο που εφαρμόζει στατιστικές μεθόδους και εκπαιδεύεται στο να παράγει ενσωματώσεις λέξεων σε ένα διανυσματικό χώρο. Πιο συγκεκριμένα, το GloVe αποτελεί ένα μοντέλο παλινδρόμησης σταθμισμένων ελαχίστων τετραγώνων το οποίο αξιοποιεί το πλήθος των ταυτόχρονων εμφανίσεων δύο λέξεων στο σύνολο δεδομένων, όπου για κάθε λέξη υπολογίζεται το πλήθος εμφάνισης κάθε άλλης λέξης που περιέχεται στο λεξιλόγιο, για ένα παράθυρο λέξεων προκαθορισμένου μεγέθους γύρω από την πρώτη. Επίσης, το μοντέλο χρησιμοποιεί μια συνάρτηση κόστους η οποία έχει επιλεχθεί με τρόπο τέτοιο ώστε να μην δίνεται υπερβολική σημασία σε ταυτόχρονες εμφανίσεις ζευγαριών λέξεων που παρατηρούνται πολύ συχνά ή σπάνια. Εκπαιδεύοντας το μοντέλο σε ένα ικανοποιητικό πλήθος δεδομένων με κατάλληλο τρόπο, οι Pennington et al. εξηγούν ότι το μοντέλο είναι σε θέση να παράγει ενσωματώσεις λέξεων που μπορούν να χρησιμοποιηθούν σε εργασίες όπως η σύγκριση ομοιότητας.

Τα συστήματα chatbot που βασίζονται σε τεχνητή νοημοσύνη αξιοποιούν συχνά ενσωματώσεις λέξεων σε ορισμένα τμήματα τους. Παρακάτω παρουσιάζουμε συνοπτικά ορισμένα συστήματα που αξιοποιούν ενσωματώσεις λέξεων.

Στην εργασία τους οι Xu et al. [53] παρουσιάζουν ένα σύστημα chatbot που προορίζεται για χρήση σε μέσα κοινωνικής δικτύωσης. Το chatbot αυτό αξιοποιεί νευρωνικά δίκτυα Μακράς και Βραχείας Μνήμης, τα οποία έχουν εκπαιδευτεί κατάλληλα,

προκειμένου να παράγει τις απαντήσεις που προσφέρει στον χρήστη. Για να μετατραπούν οι λέξεις που δίνονται ως είσοδος στα δίκτυα αυτά σε μορφή την οποία τα τελευταία μπορούν να αξιοποιήσουν, οι Xu et al. εκπαιδεύουν για το σκοπό αυτό μοντέλα ενσωματώσεων λέξεων Word2Vec, τη λειτουργία των οποίων εξηγήσαμε στο Κεφάλαιο 2. Ειδικότερα, για την εκπαίδευση των μοντέλων ενσωματώσεων λέξεων χρησιμοποιείται ένα κατάλληλα επεξεργασμένο σύνολο δεδομένων κειμένου, τα περιεχόμενα του οποίου σχετίζονται με το περιβάλλον χρήσης για το οποίο προορίζεται το chatbot αυτό. Με αυτό τον τρόπο, μετά το πέρας της εκπαίδευσης οι παραγόμενες ενσωματώσεις αντιπροσωπεύουν αρτιότερα τις λέξεις που δίνονται ως είσοδο στο σύστημα.

Ένα άλλο παράδειγμα αποτελεί το ιδιαίτερα γνωστό chatbot MILABOT [54]. Το MILABOT είναι ένα σύστημα chatbot που αξιοποιεί μια πληθώρα μοντέλων, καθένα από τα οποία παράγει μια απάντηση για την ερώτηση που λαμβάνεται από τον χρήστη του συστήματος. Στη συνέχεια μέσω κατάλληλων μηχανισμών επιλέγεται μία απάντηση από το ευρύτερο σύνολο απαντήσεων που παράχθηκαν, ως την τελική απάντηση του συστήματος. Σε αρκετά από τα παραπάνω μοντέλα παραγωγής απαντήσεων του συστήματος, γίνεται αξιοποίηση ενσωματώσεων λέξεων που παράγονται μέσω μοντέλων της οικογένειας Word2Vec αλλά και της οικογένειας GloVe, για την επίτευξη σύγκρισης ομοιότητας μεταξύ λέξεων με στόχο την βελτίωση της ικανότητας των μοντέλων αυτών στην ανάκτηση απαντήσεων σχετικών με το ερώτημα του χρήστη.

3.3.2 TF-IDF

Το **TF-IDF** αποτελεί έναν τρόπο με τον οποίο μπορούμε να εξάγουμε πληροφορίες για συγκεκριμένες λέξεις που περιέχονται σε ένα τμήμα κειμένου, που συχνά αποκαλούμε και απλά ως έγγραφο. Προκειμένου να υπολογίσουμε τη τιμή TF-IDF για μια λέξη, χρησιμοποιούνται οι δύο παρακάτω έννοιες:

- Συχνότητα Όρου (Term Frequency): Η Συχνότητα όρου για μια λέξη w σε ένα τμήμα κειμένου k , την οποία συμβολίζουμε ως $\boldsymbol{TF_{w,k}}$, ορίζεται ως το πλήθος των εμφανίσεων της λέξης w στο τμήμα αυτό [55].

- Αντίστροφη Συχνότητα Εγγράφου (Inverse Document Frequency): Η Αντίστροφη Συχνότητα Εγγράφου μιας λέξης w, έχοντας στη διάθεση μας ένα σύνολο K τμημάτων κειμένου, ορίζεται ως εξής:

$$\boldsymbol{IDF_w} = \log \frac{\boldsymbol{K}}{\boldsymbol{DF_w}} \tag{3. 1}$$

  όπου η ποσότητα $\boldsymbol{DF_w}$ αντιστοιχεί στο πλήθος των K τμημάτων στα οποία περιέχεται η λέξη w, και είναι γνωστή ως Συχνότητα Εγγράφου (Document Frequency) [55].

Χρησιμοποιώντας τους δύο παραπάνω όρους, το TF-IDF για μια λέξη w και ένα τμήμα κειμένου k που αποτελεί μέρος ενός συνόλου τμημάτων K, ορίζεται ως:

$$(\boldsymbol{TF-IDF})_{\boldsymbol{w,k}} = \boldsymbol{TF_{w,k} \cdot IDF_w} \quad (3.\ 2)$$

Το TF-IDF μιας λέξης w σε ένα τμήμα κειμένου k έχει το πλεονέκτημα του ότι λαμβάνει υπόψιν τόσο το πλήθος εμφάνισης της λέξης w στο έγγραφο k όσο όμως και το ποσοστό εμφάνισης στο γενικότερο σύνολο των διαθέσιμων τμημάτων. Συνεπώς, το TF-IDF μιας λέξης που εμφανίζεται σε μικρό ποσοστό των διαθέσιμων τμημάτων τείνει να παρουσιάζει μεγαλύτερη τιμή συγκριτικά με το TF-IDF μιας λέξης που εμφανίζεται σε μεγάλο πλήθος των διαθέσιμων τμημάτων.

Έχοντας ορίσει ένα σύνολο λέξεων και εξάγοντας το TF-IDF για καθεμία από αυτές για ένα τμήμα κειμένου, μπορούμε να ομαδοποιήσουμε τα αποτελέσματα σε ένα διάνυσμα-αναπαράσταση του τμήματος αυτού. Χρησιμοποιώντας τις αναπαραστάσεις αυτές, είμαστε σε θέση να πραγματοποιήσουμε σύγκριση ομοιότητας μέσω μιας μετρικής απόστασης. Η διαδικασία αυτή μπορεί να χρησιμοποιηθεί επομένως για εργασίες όπου επιθυμούμε να ανακτήσουμε σχετικά τμήματα κειμένων για ένα τμήμα-ερώτημα.

Ένα παράδειγμα εφαρμογής TF-IDF σε συστήματα chatbot αποτελεί το σύστημα chatbot που παρουσιάζουν στην εργασία τους οι Athota et al. [56]. Το σύστημα chatbot αυτό υπολογίζει τα TF-IDF για ορισμένες λέξεις των προτάσεων που το σύστημα έχει αποθηκεύσει στη βάση δεδομένων του και της ερώτησης που δίνεται από το χρήστη του. Στη συνέχεια, οι τιμές TF-IDF των λέξεων αυτών αξιοποιούνται για την πραγματοποίηση σύγκρισης ομοιότητας με χρήση της απόστασης συνημιτόνου, ώστε να πραγματοποιηθεί τελικά η ανάκτηση όσο το δυνατόν καταλληλότερων απαντήσεων για το ερώτημα του χρήστη.

Ένα ακόμα σύστημα chatbot που αξιοποιεί διάφορες τεχνικές, στις οποίες συμπεριλαμβάνεται και το TF-IDF, είναι το DBpedia Chatbot, που παρουσιάζουν στην εργασία τους οι Athreya et al. [57]. Συγκεκριμένα, προκειμένου το σύστημα αυτό να είναι σε θέση να απαντήσει σε ορισμένες ερωτήσεις που τίθενται από τον χρήστη, αξιοποιούνται διανύσματα που περιέχουν τα TF-IDF συγκεκριμένων λέξεων, ώστε να εντοπιστούν μέσω τεχνικών ομαδοποίησης των διανυσμάτων αυτών ορισμένες θεματικές ενότητες. Οι ενότητες αυτές στη συνέχεια χρησιμοποιούνται για την υλοποίηση ενός μηχανισμού κανόνων για την παραγωγή των απαντήσεων για τις σχετικές ερωτήσεις που αναφέραμε παραπάνω.

4. Σύνολο Δεδομένων

Στο κεφάλαιο αυτό παρουσιάζουμε το σύνολο δεδομένων που χρησιμοποιούμε για την υλοποίηση του συστήματος chatbot στα πλαίσια της παρούσας εργασίας. Στην ενότητα 4.1 παρουσιάζουμε μια περιγραφή της δομής των διαθέσιμων εγγράφων που θα χρησιμοποιήσουμε στα πλαίσια της παρούσας εργασίας. Στην ενότητα 4.2 περιγράφουμε τη διαδικασία που ακολουθήσαμε ώστε να επεξεργαστούμε κατάλληλα τα δεδομένα μας ώστε να είμαστε σε θέση να τα χρησιμοποιήσουμε. Τέλος, στην ενότητα 4.3 αναλύουμε το ζήτημα της εξαγωγής πινάκων από τα έγγραφα που έχουμε στη διάθεση μας.

4.1 Περιγραφή Συνόλου Δεδομένων

4.1.1 Εισαγωγή

Το **σύνολο δεδομένων** που έχουμε στη διάθεση μας αποτελείται από έγγραφα σχετικά με έργα στα οποία συμμετέχει το Εργαστήριο Συστημάτων Υποστήριξης Αποφάσεων και Διοίκησης του Εθνικού Μετσόβιου Πολυτεχνείου. Τα έγγραφα αυτά αποτελούνται από συμβατικά έγγραφα (Grant Agreements) καθώς και έγγραφα που σχετίζονται με τα παραδοτέα του εκάστοτε έργου. Στις παρακάτω υπό-ενότητες γίνεται αναλυτικότερη αναφορά στο κάθε είδος εγγράφου, παρουσιάζοντας την γενική μορφή που έχουν τα έγγραφα αυτά, ανεξαρτήτως του έργου στο οποίο αντιστοιχούν.

4.1.2 Grant Agreements

Τα **Grant Agreements** αποτελούν ένα έγγραφο αξιοσημείωτου μεγέθους, στα οποία αναφέρονται σημαντικές πληροφορίες σχετικές με το έργο που αντιστοιχεί το εκάστοτε Grant Agreement. Πιο αναλυτικά στα έγγραφα αυτά, πέραν από τους όρους, τις προϋποθέσεις και παραδείγματα εντύπων σχετικά με το αντίστοιχο έργο, περιέχεται επίσης μια εκτενής περιγραφή της εργασίας που αντιστοιχεί στο έργο. Η εκτενής περιγραφή αυτή καλείται **Περιγραφή της Εργασίας (Description of the Action, DoA)** και διαχωρίζεται σε δύο κύρια μέρη.

Πριν προχωρήσουμε στην ανάλυση των δύο μερών, χρειάζεται πρώτα να αναφέρουμε ότι η συνολική εργασία που αντιστοιχεί σε ένα έργο οργανώνεται σε μικρότερες ομάδες εργασιών, που καλούνται **πακέτα εργασίας**. Τα πακέτα εργασίας απαρτίζονται από μια σειρά εργασιών, για καθεμιά από τις οποίες ορίζεται ο υπεύθυνος και οι συμμετέχοντες εταίροι, που συσχετίζονται με ορισμένα από τα παραδοτέα του έργου. Κάθε τέτοιο πακέτο εργασίας έχει ορισμένους στόχους και δεν συμμετέχουν σε αυτά απαραίτητα όλοι οι εταίροι που συμμετέχουν στο έργο. Επίσης, ο χρόνος συμμετοχής του κάθε εταίρου που συμμετέχει σε ένα πακέτο εργασίας είναι διαφορετικός, ανάλογα με τον ρόλο που έχει αναλάβει.

Το πρώτο μέρος της Περιγραφής της Εργασίας αποτελείται από μια σειρά δομημένης πληροφορίας και έχει παρόμοια μορφή για κάθε έργο. Πιο συγκεκριμένα, το μέρος αυτό περιέχει μια περίληψη του έργου, βασικές πληροφορίες για τους εταίρους

που συμμετέχουν σε αυτό και μια περιγραφή για κάθε πακέτο εργασίας του έργου, στην οποία αναφέρονται για το καθένα όλες οι σχετικές πληροφορίες που αναφέραμε παραπάνω καθώς και δεδομένα όπως ο τίτλος, η περίοδος έναρξης και η περίοδος λήξης του πακέτου εργασίας. Περιέχονται επίσης βασικές πληροφορίες σχετικά με τα παραδοτέα του έργου και τέλος ένας πίνακας στον οποίο περιγράφεται συνολικά ο χρόνος συμμετοχής των εταίρων στα διάφορα πακέτα εργασίας. Ένα παράδειγμα των περιεχομένων του μέρους αυτού απεικονίζεται στην Εικόνα 21.

Table of Contents

Εικόνα 21: Παράδειγμα Περιεχομένων Πρώτου Μέρους της Περιγραφής της Εργασίας

Το δεύτερο μέρος της Περιγραφής της Εργασίας αποτελείται από αδόμητη πληροφορία της οποίας η δομή, σε αντίθεση με το πρώτο μέρος, διαφέρει ανάλογα με το εκάστοτε έργο. Ειδικότερα, το μέρος αυτό αποτελείται κυρίως από κείμενο και περιέχει πιο αναλυτικές και ειδικές πληροφορίες σχετικές με το έργο και τους εταίρους που συμμετέχουν σε αυτό. Ενδέχεται να περιέχονται επίσης ορισμένοι πίνακες και εικόνες στις οποίες παρουσιάζονται πληροφορίες σχετικές με το έργο και διαγράμματα Gantt που περιγράφουν τον χρονικό προγραμματισμό της εκτέλεσης των εργασιών του έργου. Ένα παράδειγμα των περιεχομένων του μέρους αυτού απεικονίζεται στην Εικόνα 22.

Table of Contents

Εικόνα 22: Παράδειγμα Περιεχομένων Δεύτερου Μέρους της Περιγραφής της Εργασίας

4.1.3 Παραδοτέα Έργου

Όπως εξηγήσαμε, κάθε έργο έχει ένα σημαντικό πλήθος παραδοτέων. Για τα περισσότερα από τα παραδοτέα, προκύπτουν συνήθως έγγραφες αναφορές σε μορφή pdf ή docx, που περιέχουν τα αποτελέσματα του αντίστοιχου παραδοτέου. Η δομή κάθε τέτοιου εγγράφου ενδέχεται να διαφέρει για κάθε παραδοτέο.

4.2 Επεξεργασία Συνόλου Δεδομένων

4.2.1 Εισαγωγή

Το σύνολο των εγγράφων που έχουμε στη διάθεση μας είναι δύσκολο να χρησιμοποιηθεί στην αρχική μορφή του από το σύστημα chatbot που υλοποιούμε στη παρούσα εργασία. Χρειάζεται συνεπώς να εξάγουμε τις χρήσιμες πληροφορίες από τα έγγραφα αυτά σε κατάλληλη μορφή και έπειτα να τις αποθηκεύσουμε κατάλληλα, ούτως ώστε να μπορούν να αξιοποιηθούν όσο το δυνατόν αρτιότερα από το σύστημα chatbot μας.

4.2.2 Μετατροπή είδους αρχείων

Όπως αναφέραμε, τα περισσότερα έγγραφα που έχουμε στη διάθεση μας είναι της μορφής pdf και docx. Επιθυμούμε να εξάγουμε τη πληροφορία που περιέχεται στα

έγγραφα αυτά, διατηρώντας ταυτόχρονα όμως όσο το δυνατόν περισσότερη πληροφορία και για την δομή που παρουσιάζει το κείμενο, όπως για παράδειγμα τον διαχωρισμό του σε παραγράφους, την οργάνωση κειμένου σε πίνακες κ.α.. Η εξαγωγή κειμένου, διατηρώντας τη δομή του, είναι σημαντικά πιο εύκολη όταν πραγματοποιείται από έγγραφα docx σε σχέση με έγγραφα pdf, καθώς τα πρώτα περιέχουν ενδείξεις που υποδεικνύουν την δομή του κειμένου, όπως για παράδειγμα την αλλαγή παραγράφου, την αλλαγή σελίδας, την έναρξη αλλά και τη λήξη στοιχείων πινάκων. Συνεπώς, λόγω του παραπάνω, η ορθή εξαγωγή κειμένου για έγγραφα pdf με σύνθετη δομή καθίσταται συχνά μια απαιτητική διαδικασία.

Για την εξαγωγή κειμένου με διατήρηση της δομής του κειμένου από έγγραφα τύπου docx χρησιμοποιήσαμε τη βιβλιοθήκη unstructured [58], μέσω της οποίας μπορούμε να εξάγουμε το κείμενο του εγγράφου σε ένα σύνολο μικρότερων τμημάτων, το καθένα από τα οποία συνοδεύεται από μια σειρά πληροφορίων σχετικές με εκείνο. Οι πληροφορίες αυτές σχετίζονται με το ρόλο που έχει το εκάστοτε τμήμα κειμένου στη δομή του κειμένου, όπως για παράδειγμα το αν αποτελεί επικεφαλίδα, τίτλο, ή μέρος κειμένου ενός πίνακα. Περιέχονται επίσης μετά-δεδομένα όπως ο τίτλος και ο αριθμός της σελίδας του εγγράφου στην οποία βρίσκεται το εκάστοτε τμήμα.

Όσον αφορά την εξαγωγή κειμένου από έγγραφα τύπου pdf, διατηρώντας όσο το δυνατόν αρτιότερα την δομή του, δοκιμάσαμε και πάλι να χρησιμοποιήσουμε την βιβλιοθήκη unstructured [58], η οποία εξάγει ικανοποιητικά τα περισσότερα τμήματα κειμένου. Ωστόσο, σε περιπτώσεις όπου έχουμε πίνακες που εμφανίζουν σύνθετη δομή ή έχουν μέγεθος μεγαλύτερο της μίας σελίδας, παρατηρούμε ότι λαμβάνουμε αρκετές φορές μη ικανοποιητικά αποτελέσματα. Αυτό οφείλεται σε σημαντικό βαθμό στο ότι στα έγγραφα pdf έχουμε, όπως αναφέραμε, απουσία ενδείξεων της δομής του κειμένου.

Στα πλαίσια της παρούσας εργασίας χρησιμοποιήσαμε τη βιβλιοθήκη pdfplumber [59] προκειμένου να εξάγουμε ορθά το κείμενο και τους πίνακες που περιέχονται σε ένα από τα συμβατικά έγγραφα τα οποία έχουμε στη διάθεση μας σε μορφή pdf, προκειμένου να μπορέσουμε να το αξιοποιήσουμε για την αξιολόγηση του συστήματος chatbot μας. Πιο συγκεκριμένα, χρησιμοποιήσαμε ορισμένες συναρτήσεις της βιβλιοθήκης αυτής, που χρησιμεύουν στην εξαγωγή πινάκων και κειμένων. Σε περιπτώσεις όπου παρατηρήσαμε λανθασμένες επεξεργασίες πινάκων, αξιοποιήσαμε το γεγονός του ότι οι συναρτήσεις αυτές μπορούν να δεχτούν όρια πινάκων που ορίζονται χειροκίνητα, ούτως ώστε να εξάγουμε τους πίνακες αυτούς με την επιθυμητή τους δομή.

4.2.3 Καθαρισμός Συνόλου Δεδομένων

Έπειτα από την μετατροπή των δεδομένων μας, είναι ιδιαίτερα σημαντικό να μεριμνήσουμε για τον καθαρισμό τους. Χρειάζεται δηλαδή να ελέγξουμε ότι δεν υπάρχουν χαρακτήρες που αποτελούν θόρυβο στο κείμενο μας, όπως διπλά σημεία στίξης ή χαρακτήρες που δεν έχουν επεξεργαστεί σωστά. Τα παραπάνω κρίνονται σκόπιμα καθώς όπως θα δούμε στο κεφάλαιο 5, τα δεδομένα μας προορίζονται ως είσοδος σε ένα μεγάλο γλωσσικό μοντέλο και άρα η ποιότητα του κειμένου αυτού είναι επιθυμητό να είναι όσο το δυνατόν αρτιότερη, καθώς επηρεάζει άμεσα τις απαντήσεις που παράγει το μοντέλο που θα χρησιμοποιήσουμε. Για το σκοπό αυτό αξιοποιήσαμε τη βιβλιοθήκη

re της python, ώστε να ορίσουμε κανόνες καθαρισμού περιττών ή λανθασμένων χαρακτήρων που περιέχονται στο σύνολο των κειμένων που λαμβάνουμε κατά τη διαδικασία εξαγωγής από τα έγγραφα μας.

4.3 Εξαγωγή Πινάκων

Προκειμένου το μεγάλο γλωσσικό μοντέλο που θα χρησιμοποιήσουμε στο σύστημα chatbot μας να είναι ικανό να επεξεργαστεί τους πίνακες που έχουμε εξάγει, είναι χρήσιμο οι πίνακες αυτοί να δίνονται ως είσοδος σε μορφή που να δηλώνεται η δομή τους. Μια τέτοια μορφή αποτελεί η html, η οποία διαχωρίζει το κείμενο του πίνακα με ετικέτες που καθορίζουν την δομή του πίνακα.

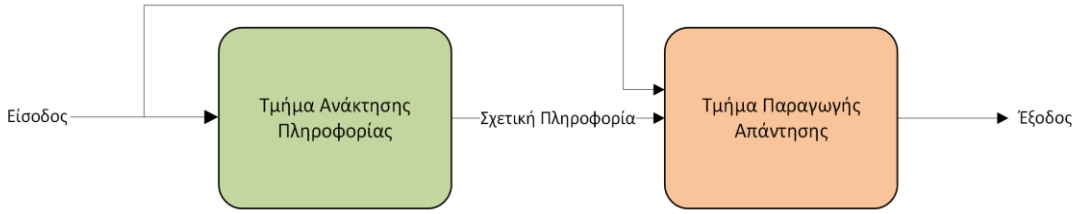
Για τα έγγραφα τύπου docx αξιοποιούμε το μετά-δεδομένο **text_as_html** των στοιχείων τύπου πίνακα που εξαγάγαμε μέσω της βιβλιοθήκης unstructured, ώστε να λάβουμε το κείμενο του πίνακα δομημένο σε μορφή html. Όσον αφορά τα έγγραφα τύπου pdf, όπως εξηγήσαμε και στις προηγούμενες ενότητες η εξαγωγή πινάκων δεν είναι πάντα το ίδιο αποτελεσματική, με αποτέλεσμα το αντίστοιχο μετά-δεδομένο να περιέχει πολλές φορές το κείμενο του πίνακα δομημένο με λανθασμένο τρόπο. Πειραματιστήκαμε επίσης με χρήση και άλλων βιβλιοθηκών που χρησιμεύουν στην εξαγωγή πινάκων όπως την pdfplumber [59] και την PyMuPDF [60], αλλά παρατηρούμε ότι δεν μπορούμε να εξάγουμε με αυτόματο τρόπο όλα τα είδη πινάκων στα έγγραφα μας και συνήθως απαιτείται χρήση ημιαυτόματων τρόπων εξαγωγής όπως αυτός που εξηγήσαμε στην υπό-ενότητα 4.2.2.

5. Μεθοδολογία

Στο κεφάλαιο αυτό παρουσιάζουμε την μεθοδολογία που ακολουθήσαμε για την δημιουργία του συστήματος chatbot στα πλαίσια της παρούσας διπλωματικής εργασίας.

5.1 Το Μοτίβο Σχεδίασης RAG

Το μοτίβο **Retrieval Augmented Generation (RAG)** προτάθηκε το 2020 στην εργασία των Lewis et al. [61]. Το μοτίβο σχεδίασης RAG αποτελεί ένα τρόπο με τον οποίο μπορούμε να βελτιώσουμε ένα γλωσσικό μοντέλο το οποίο είναι ικανό να παράγει κείμενο μέσω της γνώσης που έχει αποκτήσει στο στάδιο προ-εκπαίδευσης του. Ειδικότερα, για την επίτευξη του στόχου αυτού χρησιμοποιείται ένα άλλο σύστημα που είναι υπεύθυνο για την παροχή περαιτέρω γνώσης στο γλωσσικό μοντέλο, η οποία σχετίζεται με την είσοδο που λαμβάνει το τελευταίο. Με αυτόν τον τρόπο, το γλωσσικό μοντέλο εμπλουτίζει την ήδη υπάρχουσα γνώση που κατέχει με τις πληροφορίες που έλαβε και ενδεχομένως δεν γνώριζε εκ των προτέρων, κάτι που του επιτρέπει να ανταποκριθεί πολλές φορές πιο άρτια στην παραγωγή της εξόδου του.



Εικόνα 23: Περιληπτική Λειτουργία του Μοτίβου Ανάπτυξης RAG

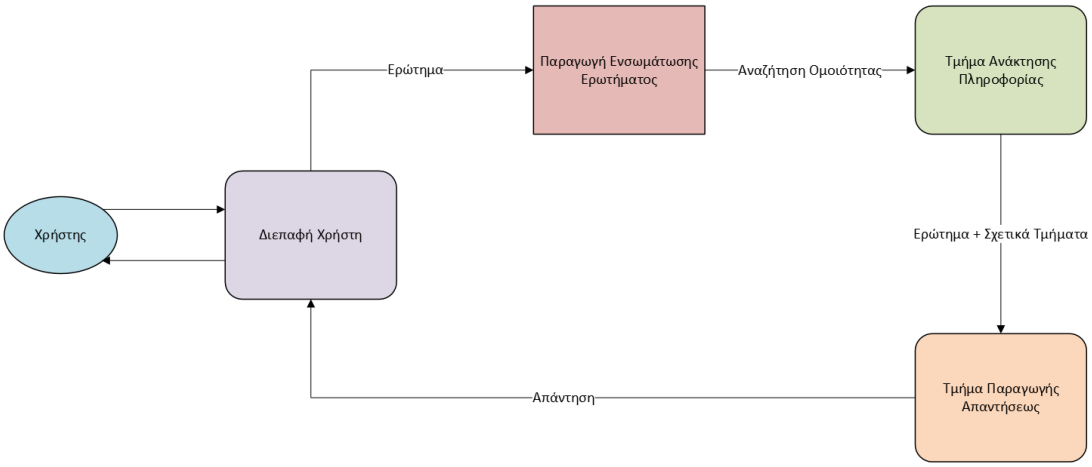
Το τμήμα που είναι υπεύθυνο για την ανάκτηση πληροφορίας αναφέρεται συχνά ως *τμήμα ανάκτησης πληροφορίας*, ενώ το γλωσσικό μοντέλο το οποίο αξιοποιεί την περαιτέρω γνώση που του προσφέρεται είναι γνωστό ως *τμήμα παραγωγής απάντησης*. Μια γενική απεικόνιση της διαδικασίας που ακολουθείται στο μοτίβο RAG φαίνεται και στην Εικόνα 23. Αρχικά, το τμήμα ανάκτησης λαμβάνει την είσοδο που προορίζεται για το τμήμα παραγωγής απάντησης και ανακτά ένα πλήθος δεδομένων που σχετίζονται με την είσοδο αυτή. Έπειτα, τα δεδομένα που ανακτήθηκαν, μαζί με την αρχική είσοδο, δίνονται ως είσοδος στο τμήμα παραγωγής απάντησης το οποίο με τη σειρά του τα αξιοποιεί για την παραγωγή της τελικής εξόδου.

Τα μεγάλα γλωσσικά μοντέλα που προορίζονται για παραγωγή φυσικού κειμένου, παρόλο του ότι όπως αναφέραμε κατέχουν μεγάλη ποσότητα γνώσεων, είναι δυνατόν όταν δέχονται ως είσοδο ερωτήσεις, για την απάντηση των οποίων απαιτούνται πληροφορίες που δεν γνωρίζουν, να παράγουν ως έξοδο μια απάντηση η οποία είναι λανθασμένη. Όταν κάτι τέτοιο συμβαίνει, συνήθως λέμε ότι το μοντέλο έχει παραισθήσεις, παράγοντας απαντήσεις που θεωρεί σωστές ενώ είναι στην πραγματικότητα εσφαλμένες. Στην εργασία τους οι Lewis et al. [61] εξηγούν ότι το μοτίβο RAG είναι ικανό να καταπολεμήσει σημαντικά το φαινόμενο αυτό, καθώς η γνώση που προσφέρεται στο τμήμα παραγωγής απάντησης μπορεί να βοηθήσει στην αποφυγή καταστάσεων παραισθήσεων από το μοντέλο παραγωγής κειμένου που χρησιμοποιείται στο τμήμα αυτό.

Τα μεγάλα γλωσσικά μοντέλα που χρησιμοποιούνται σήμερα για παραγωγή κειμένου, όπως για παράδειγμα τα LLaMA, αποτελούνται συνήθως από πολλές δισεκατομμύριες παραμέτρους. Το γεγονός αυτό καθιστά την περαιτέρω εκπαίδευση ενός τέτοιου προ-εκπαιδευμένου μοντέλου μέσω μιας διαδικασίας βελτίωσης εξαιρετικά ακριβή σε χρόνο και υπολογιστικούς πόρους. Συνεπώς, αν έχουμε στη διάθεση μας μια συλλογή δεδομένων όπου τα δεδομένα ανανεώνονται συχνά, η συνεχής επανεκπαίδευση του μοντέλου πολλές φορές δεν είναι επιθυμητή. Το μοτίβο RAG, επιτρέπει μέσω της ανανέωσης των δεδομένων που έχει στη διάθεση του το τμήμα ανάκτησης πληροφορίας, όποτε αυτό κρίνεται αναγκαίο, την παροχή σχετικής και ταυτόχρονα πρόσφατης πληροφορίας για την εκάστοτε είσοδο του τμήματος παραγωγής απάντησης [61]. Η διαδικασία αυτή απαιτεί σημαντικά λιγότερο χρόνο και πόρους σε σχέση με τη διαδικασία βελτίωσης του μεγάλου γλωσσικού μοντέλου, καθώς στο μοτίβο RAG η επιπλέον πληροφορία δίνεται μόνο ως ένα μέρος της εισόδου του μεγάλου γλωσσικού μοντέλου και δεν απαιτεί την περαιτέρω εκπαίδευση του.

5.2 Μεθοδολογία Υλοποίησης Εφαρμογής Chatbot

Για την εφαρμογή chatbot που αναπτύξαμε στα πλαίσια της διπλωματικής αυτής εργασίας εφαρμόζουμε, ως κύρια αρχιτεκτονική, το μοτίβο RAG που αναλύσαμε συνοπτικά παραπάνω. Επιλέγουμε το μοτίβο αυτό καθώς το σύνολο δεδομένων μας είναι σημαντικού μεγέθους και συχνά ανανεώνεται με νέα δεδομένα. Επιθυμούμε επιπλέον να αξιοποιήσουμε το πλεονέκτημα του ότι μπορούμε να ανακτήσουμε για κάθε ερώτημα που θα δίνεται ως είσοδο την εκάστοτε πιο σχετική πληροφορία, ώστε να μπορέσουμε να προσφέρουμε στο τμήμα παραγωγής απάντησης αρκετά συμφραζόμενα σχετικά με το ερώτημα και να μπορέσει έτσι να σχηματίσει, με όσο το δυνατόν μεγαλύτερη ακρίβεια και ορθότητα, την απάντηση του.



Εικόνα 24: Γενική Λειτουργία της Εφαρμογής Chatbot μας

Η διαδικασία που ακολουθείται προκειμένου να παραχθεί η απάντηση στο ερώτημα του χρήστη περιγράφεται ως εξής: Αρχικά ο χρήστης θέτει ένα ερώτημα μέσω της διεπαφής χρήστη. Στη συνέχεια, το ερώτημα δίνεται ως είσοδος στο μοντέλο παραγωγής ενσωματώσεων. Έπειτα, η ενσωμάτωση που παράχθηκε χρησιμοποιείται από το τμήμα ανάκτησης πληροφορίας προκειμένου να πραγματοποιηθεί η διαδικασία αναζήτησης όμοιων σημασιολογικά τμημάτων κειμένου με το ερώτημα του χρήστη. Τα αποτελέσματα της αναζήτησης αποστέλλονται μαζί με το ερώτημα στο τμήμα παραγωγής απαντήσεως το οποίο τα αξιοποιεί για την παραγωγή της τελικής απάντησης του

συστήματος. Τέλος, η απάντηση παρουσιάζεται στον χρήστη μέσω της διεπαφής χρήστη. Η διαδικασία που περιγράψαμε φαίνεται στην Εικόνα 24. Στις επόμενες ενότητες αυτού του κεφαλαίου, θα εξηγήσουμε πιο αναλυτικά τον τρόπο λειτουργίας του εκάστοτε τμήματος της εφαρμογής μας.

5.3 Τμήμα Ανάκτησης Πληροφορίας

Για το τμήμα ανάκτησης πληροφορίας της εφαρμογής μας, χρησιμοποιούμε ένα μοντέλο μηχανικής μάθησης προκειμένου να εξάγουμε ενσωματώσεις για καθένα από τα δεδομένα που επιθυμούμε να έχουμε διαθέσιμα για ανάκτηση. Έπειτα χρησιμοποιούμε ένα σύστημα διαχείρισης διανυσματικής βάσης δεδομένων, ούτως ώστε να δημιουργήσουμε ένα ευρετήριο για τα δεδομένα μας, που θα επιτρέπει την γρήγορη και ταυτόχρονα αποτελεσματική ανάκτηση των τελευταίων. Στις ακόλουθες υπό-ενότητες εξηγούμε πιο αναλυτικά τις τεχνικές και τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση του τμήματος ανάκτησης πληροφορίας.

5.3.1 Διαδικασία Διαχωρισμού σε τμήματα

Το μέγεθος της εισόδου ενός μεγάλου γλωσσικού μοντέλου καλείται συχνά ως μέγεθος συμφραζομένων και είναι συνήθως της τάξης των μερικών έως αρκετών χιλιάδων συμβόλων. Όπως εξηγήσαμε, για ένα σύστημα που ακολουθεί το μοτίβο RAG, το τμήμα παραγωγής απάντησης λαμβάνει ως είσοδο την αρχική είσοδο του συστήματος και επιπλέον τα δεδομένα που ανακτήθηκαν από το τμήμα ανάκτησης πληροφορίας. Για την περίπτωση όπου ένα τέτοιο σύστημα χρησιμοποιεί ένα μεγάλο γλωσσικό μοντέλο ως το κύριο συστατικό του τμήματος παραγωγής απάντησης, η είσοδος του τμήματος αυτού είναι σημαντικό να μην αποτελείται από περισσότερα σύμβολα από το μέγεθος συμφραζομένων που υποστηρίζει το μεγάλο γλωσσικό μοντέλο. Ταυτόχρονα όμως, δεν είναι επιθυμητό να δώσουμε ένα τεράστιο πλήθος δεδομένων στο τμήμα παραγωγής απάντησης, καθώς το τελευταίο ενδεχομένως να μην μπορέσει να το αξιοποιήσει αποτελεσματικά. Χρειάζεται συνεπώς, λαμβάνοντας υπόψιν τα παραπάνω, να επιλέξουμε ένα κατάλληλο μέγεθος της εισόδου.

Τα δεδομένα που έχουμε στη διάθεση μας προέρχονται από έγγραφα σημαντικού μεγέθους, που αποτελούνται από πολλές σελίδες. Χρειάζεται να διαχωρίσουμε τα δεδομένα αυτά σε μικρότερα τμήματα καθώς, πέρα από το ότι με αυτό τον τρόπο θα μπορεί να τα επεξεργαστεί το μεγάλο γλωσσικό μοντέλο για τους λόγους που εξηγήσαμε παραπάνω, το μοντέλο παραγωγής ενσωματώσεων παρουσιάζει και αυτό ένα μέγιστο μέγεθος εισόδου για την οποία είναι ικανό να παράγει μια ενσωμάτωση που να αναπαριστά την είσοδο αυτή με ακρίβεια.

Παρακάτω αναλύουμε ορισμένες τεχνικές για την επίτευξη του διαχωρισμού του συνόλου δεδομένων μας σε τμήματα.

<u>Διαχωρισμός Ανά Πλήθος Χαρακτήρων</u>

Η τεχνική αυτή αποτελεί μια από τις πιο απλές τεχνικές διαχωρισμού, όπου χωρίζουμε το κείμενο μας σε τμήματα ενός καθορισμένου πλήθους χαρακτήρων. Ωστόσο, ο χωρισμός σε τμήματα με έναν τέτοιο τρόπο δεν λαμβάνει υπόψιν τα σημεία στίξης

στο κείμενο. Αυτό έχει ως απότοκο τη δημιουργία τμημάτων που δεν είναι ικανά από μόνα τους να προσφέρουν πολλές φορές το πλήρες νόημα μιας πρότασης ή παραγράφου. Προκειμένου να αντιμετωπιστεί αυτό το φαινόμενο, μια ελαφρά παραλλαγμένη εκδοχή αυτής της τεχνικής αποτελεί η διάσπαση σε τμήματα πάλι καθορισμένου μεγέθους, ορίζοντας όμως κάποιους επιπλέον κανόνες διάσπασης, όπως για παράδειγμα η διάσπαση σε σημεία στίξης που σηματοδοτούν το τέλος μιας πρότασης. Με αυτό τον τρόπο, μειώνουμε σημαντικά τη πιθανότητα μια πρόταση ή παράγραφος να διασπαστεί σε διαφορετικά τμήματα, κάτι που αναμένουμε να αυξήσει την ποιότητα του διαχωρισμού της πληροφορίας μας.

Διαχωρισμός Ανά Πλήθος Συμβόλων

Μια άλλη τεχνική που μπορούμε να χρησιμοποιήσουμε είναι ο διαχωρισμός της πληροφορίας μας σε τμήματα που αποτελούνται από καθορισμένο πλήθος συμβόλων. Ειδικότερα, μπορούμε να χρησιμοποιήσουμε τον συμβολοποιητή (tokenizer) του μοντέλου ενσωματώσεων ή του μεγάλου γλωσσικού μοντέλου που χρησιμοποιείται για την παραγωγή απάντησης. Έτσι μπορούμε να είμαστε σίγουροι ότι τα τμήματα θα έχουν ένα μέγιστο πλήθος συμβόλων, ελέγχοντας συνεπώς με αυτό τον τρόπο το μέγιστο μήκος των εισόδων των μεγάλων γλωσσικών μοντέλων.

Για την εφαρμογή των τεχνικών αυτών χρησιμοποιήσαμε τους διαχωριστές κειμένου που προσφέρονται από το εργαλείο λογισμικού Llamaindex [62]. Στο σημείο αυτό αξίζει να σημειωθεί ότι για τα πειράματα αξιολόγησης που θα αναλύσουμε στο Κεφάλαιο 6, χρησιμοποιήσαμε διαχωρισμό ανά πλήθος συμβόλων, καθώς έτσι μπορούμε να είμαστε σίγουροι ότι το κείμενο μας θα διαχωρίζεται σε τμήματα ίδιου περίπου μεγέθους.

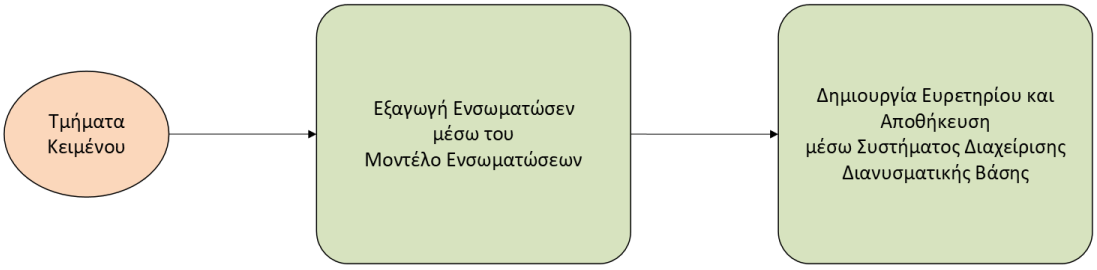
5.3.2 Εξαγωγή και Αποθήκευση Ενσωματώσεων

Αφού διαχωρίσουμε κατάλληλα το κείμενο σε τμήματα, μέσω μίας από της στρατηγικές που εξηγήσαμε παραπάνω, χρειάζεται στη συνέχεια να χρησιμοποιήσουμε κάποιο μοντέλο μηχανικής μάθησης προκειμένου να εξάγουμε μια ενσωμάτωση για καθένα από τα τμήματα αυτά. Για το σκοπό αυτό χρησιμοποιήσαμε ένα προ-εκπαιδευμένο μοντέλο μηχανικής μάθησης.

Τα τελευταία χρόνια έχουν αναπτυχθεί διάφορα προ-εκπαιδευμένα μοντέλα τα οποία έχουν εκπαιδευτεί για την παραγωγή ενσωματώσεων που προορίζονται για ανάκτηση σχετικών τμημάτων κειμένου. Προκειμένου να επιλέξουμε ένα μοντέλο που αντιστοιχεί στην περίπτωση χρήσης της εφαρμογής που αναπτύσσουμε, συμβουλευτήκαμε το **MTEB** [63] (Massive Text Embedding Leaderboard), το οποίο αποτελεί ένα τρόπο αξιολόγησης μοντέλων παραγωγής ενσωματώσεων κειμένου, σε πληθώρα εργασιών. Καθώς επιθυμούμε να χρησιμοποιήσουμε το προ-εκπαιδευμένο μοντέλο σε εργασίες ανάκτησης σχετικών τμημάτων για το εκάστοτε ερώτημα, αναζητούμε μοντέλα τα οποία έχουν καλές επιδόσεις στην αντίστοιχη κατηγορία του MTEB (εργασία retrieval).

Επιλέγουμε το μοντέλο **bge-en-large-en-v1.5** [64], καθώς πετυχαίνει καλά αποτελέσματα σύμφωνα με το MTEB και ταυτόχρονα δεν είναι ιδιαίτερα ακριβό σε πό-

ρους. Ένας άλλος λόγος για τον οποίο επιλέξαμε μοντέλο ενσωματώσεων μικρού σχετικά μεγέθους είναι το ότι επιθυμούμε οι ενσωματώσεις των ερωτημάτων του χρήστη του chatbot να υπολογίζονται γρήγορα, εφόσον δεν ελαττώνεται ιδιαίτερα η ποιότητα της διαδικασίας ανάκτησης, για να επιτευχθεί έτσι μια αρτιότερη εμπειρία χρήσης του chatbot.

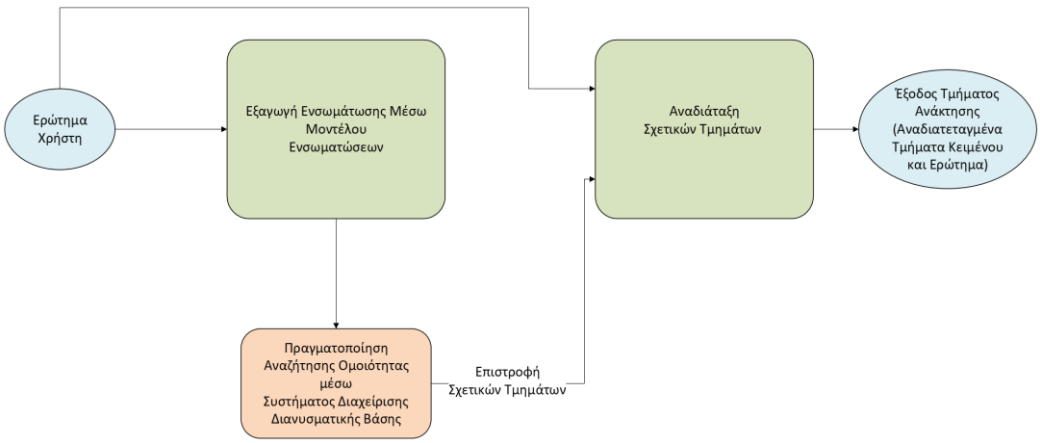


Εικόνα 25: Διαδικασία Παραγωγής και Αποθήκευσης Ενσωματώσεων

Μετά την εξαγωγή των ενσωματώσεων για καθένα από τα τμήματα δεδομένων μας, χρειάζεται, όπως εξηγήσαμε, να αποθηκεύσουμε με κατάλληλο τρόπο το σύνολο των τμημάτων κειμένου ομαδοποιημένα με την ενσωμάτωση που αντιστοιχεί στο καθένα, δημιουργώντας ένα ευρετήριο ώστε να είμαστε σε θέση να το αξιοποιήσουμε στη διαδικασία της ανάκτησης. Για την επίτευξη του παραπάνω στόχου επιλέξαμε το σύστημα διαχείρισης διανυσματικής βάσης δεδομένων qdrant [65]. Το qdrant υποστηρίζει, εκτός από την αναζήτηση όμοιων διανυσμάτων που υπάρχουν στη διανυσματική βάση δεδομένων για ένα διάνυσμα-ερώτηση, την συμμετοχή ορισμένων μόνο διανυσμάτων της βάσης στη διαδικασία της αναζήτησης μέσω ορισμένων φίλτρων τα οποία μπορούμε κάθε φορά να καθορίσουμε. Για παράδειγμα, εισάγοντας επιπλέον μετά-δεδομένα που θα συνοδεύουν τις εγγραφές της διανυσματικής βάσης δεδομένων, μπορούμε μέσω αυτών να περιορίσουμε τις αναζητήσεις στις εγγραφές που κάθε φορά είναι επιθυμητό, ανάλογα το ερώτημα που έχει τεθεί από τον χρήστη.

5.3.3 Ανάκτηση Πληροφορίας

Η ανάκτηση πληροφορίας επιτυγχάνεται μέσω της εφαρμογής αναζήτησης ομοιότητας για την ενσωμάτωση του ερωτήματος που το τμήμα ανάκτησης δέχεται ως είσοδο. Ειδικότερα αναζητούμε μέσω του συστήματος qdrant, το οποίο εφαρμόζει προσεγγιστικούς αλγορίθμους αναζήτησης πλησιέστερων γειτόνων, k τμήματα κειμένου των οποίων οι ενσωματώσεις είναι ομοιότερες με αυτές του ερωτήματος και άρα σχετίζονται σημασιολογικά μεταξύ τους. Δοκιμάσαμε επίσης να χρησιμοποιήσουμε ένα επιπλέον μοντέλο μηχανικής μάθησης ώστε να αναδιατάξουμε τα k τμήματα κειμένου. Στο Κεφάλαιο 6 θα εξηγήσουμε αναλυτικότερα την διαδικασία της αναδιάταξης και θα αναφέρουμε τα πλεονεκτήματα και τα αποτελέσματα της εφαρμογής της.

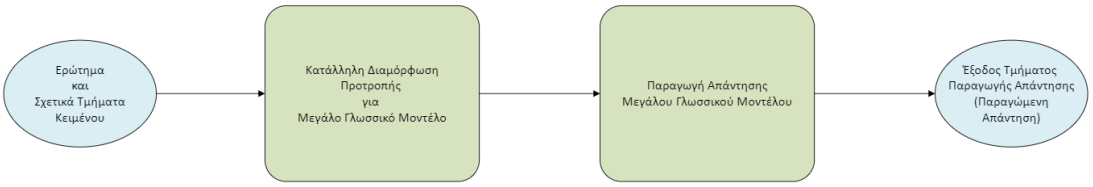


Εικόνα 26: Διαδικασία Ανάκτησης Πληροφορίας

Για την υλοποίηση του τμήματος ανάκτησης αξιοποιήσαμε το εργαλείο Llamaindex [62] και για την εφαρμογή της αναδιάταξης την βιβλιοθήκη SentenceTransformers [23], [66].

5.4 Τμήμα Παραγωγής Απάντησης

Το σημαντικότερο στοιχείο του τμήματος παραγωγής απάντησης της εφαρμογής μας αποτελείται από ένα μεγάλο γλωσσικό μοντέλο. Πιο συγκεκριμένα, χρησιμοποιήσαμε το μοντέλο **Llama 3.1 8B Instruct** που συνιστά την εκδοχή των μοντέλων Llama 3.1 8B η οποία έχει εκπαιδευτεί για παραγωγή εξόδων που αντιστοιχούν σε περιπτώσεις χρήσης συζήτησης. Για την χρήση του μοντέλου χρησιμοποιήσαμε την βιβλιοθήκη **Transformers**, μέσω της οποίας ορίσαμε τις επιθυμητές παραμέτρους δειγματοληψίας των εξόδων του μοντέλου καθώς και τις υπόλοιπες παραμέτρους, όπως για παράδειγμα το μέγιστο επιτρεπτό μήκος της απάντησης.



Εικόνα 27:Διαδικασία Παραγωγής Απάντησης

Στο κεφάλαιο 6 θα παρουσιάσουμε τα αποτελέσματα ορισμένων δοκιμών που πραγματοποιήσαμε προκειμένου να αξιολογήσουμε διάφορες παραμέτρους που επηρεάζουν την παραγόμενη έξοδο του μεγάλου γλωσσικού μοντέλου που χρησιμοποιήσαμε, όπως τη προτροπή που δίνουμε ως είσοδο και το μέγεθος των σχετικών στο ερώτημα τμημάτων κειμένου που λαμβάνουμε κατά το στάδιο της ανάκτησης.

5.5 Εξυπηρετητής Εφαρμογής Chatbot

Ο εξυπηρετητής που υλοποιήσαμε στα πλαίσια της εργασίας υλοποιεί τα βήματα ανάκτησης σχετικών τμημάτων για το ερώτημα του χρήστη και της παραγωγής

απάντησης, που εξηγήσαμε παραπάνω. Για την ανάπτυξη του εξυπηρετητή χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python και η βιβλιοθήκη FastAPI [67].

5.6 Διεπαφή Τελικού Χρήστη

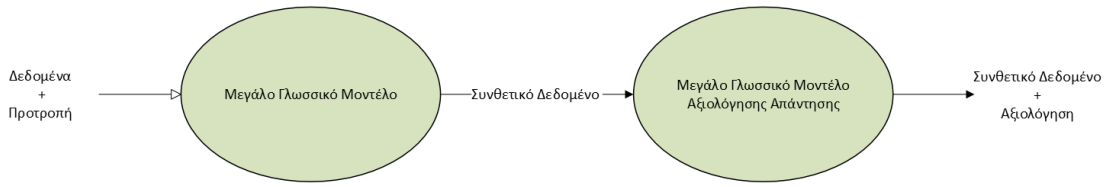
Η διεπαφή του τελικού χρήστη αποτελεί μια απλή διεπαφή που αναπτύχθηκε στα πλαίσια της εργασίας, χρησιμοποιώντας το εργαλείο Vue.js [68], με σκοπό να δοθεί η δυνατότητα στον χρήστη του συστήματος να αξιοποιήσει μέσω αυτής της δυνατότητες του συστήματος chatbot. Η διεπαφή αυτή επικοινωνεί με τον εξυπηρετητή, ο οποίος δέχεται ως είσοδο την ερώτηση του χρήστη μέσω της διεπαφής, εκτελεί την διαδικασία ανάκτησης και παραγωγής απάντησης και έπειτα επιστρέφει την απάντηση του συστήματος προς την διεπαφή ώστε να παρουσιαστεί στη συνέχεια στον τελικό χρήστη.

6. Αξιολόγηση

Στο κεφάλαιο αυτό παρουσιάζουμε τον τρόπο με τον οποίο αξιολογήσαμε την επίδοση του συστήματος μας. Αρχικά, εξηγούμε τη διαδικασία που ακολουθήσαμε για την παραγωγή ορισμένων συνόλων δεδομένων αξιολόγησης. Επιπλέον, παραθέτουμε τα αποτελέσματα πειραμάτων που πραγματοποιήσαμε, όπου δοκιμάζουμε την τροποποίηση παραμέτρων που αξιοποιούν οι τεχνικές που εφαρμόζονται στα βήματα που εξηγήσαμε στο κεφάλαιο 5, προκειμένου να διακρίνουμε το πως η κάθε παράμετρος επηρεάζει την αποτελεσματικότητα του συστήματος chatbot μας.

6.1 Δημιουργία Συνόλων Δεδομένων Αξιολόγησης

Η δημιουργία συνθετικών συνόλων δεδομένων με χρήση μεγάλων γλωσσικών μοντέλων αποτελεί μια ολοένα και πιο δημοφιλή τεχνική που έχει εφαρμοστεί σε σημαντικό βαθμό προκειμένου να αξιολογηθούν συστήματα τεχνητής νοημοσύνης [69], καθώς οι δυνατότητες των μοντέλων αυτών ολοένα και αυξάνονται. Ειδικότερα μπορούμε για παράδειγμα, δίνοντας κατάλληλες προτροπές μαζί με ένα ικανό πλήθος δεδομένων ως είσοδο σε ένα τέτοιο μοντέλο, να εξάγουμε χρήσιμα συνθετικά δεδομένα όπως ερωτήσεις που βασίζονται πάνω στα δεδομένα μας. Σε περίπτωση που τα δεδομένα που δίνουμε ως είσοδο αποτελούνται από ζεύγη δεδομένων και ερωτήσεων, μπορούμε να εξάγουμε επίσης υποθετικές απαντήσεις. Η εξαγωγή τέτοιου είδους δεδομένων μπορούν στη συνέχεια να αξιοποιηθούν ώστε να σχηματίσουμε ένα σύνολο δεδομένων με το οποίο μπορούμε να αξιολογήσουμε το σύστημα chatbot μας ως σύνολο, ή τα επιμέρους τμήματα του ξεχωριστά.



Εικόνα 28: Διαδικασία Παραγωγής και Αξιολόγησης Συνθετικών Δεδομένων

Η ποιότητα τέτοιων τεχνητών δεδομένων είναι φυσικά πολλές φορές μη αποδεκτή και τα δεδομένα περιέχουν συχνά λανθασμένες πληροφορίες που προκύπτουν για παράδειγμα από φαινόμενα παραισθήσεων του μεγάλου γλωσσικού μοντέλου. Ένας ακόμα παράγοντας της ποιότητας των συνθετικών δεδομένων αποτελούν και οι δυνατότητες του εκάστοτε μεγάλου γλωσσικού μοντέλου που χρησιμοποιείται για την παραγωγή των δεδομένων, οι οποίες εξαρτώνται για παράδειγμα από την ποιότητα των δεδομένων εκπαίδευσης του ή το πλήθος των παραμέτρων του. Μια λύση που έχει προταθεί για την αξιολόγηση των εξόδων ενός μεγάλου γλωσσικού μοντέλου είναι η χρήση ενός άλλου μεγάλου γλωσσικού μοντέλου το οποίο καλείται να αξιολογήσει τις παραγόμενες εξόδους του πρώτου [70].

Όπως αναφέραμε και στην υπό-ενότητα 4.2.2, για την δημιουργία των συνθετικών δεδομένων θα αξιοποιήσουμε τα περιεχόμενα που εξήγαμε από ένα από τα συμβατικά έγγραφα που έχουμε στη διάθεση μας.

6.1.1. Σύνολο Δεδομένων για Αξιολόγηση Τμήματος Ανάκτησης

Προκειμένου να αξιολογήσουμε τις επιδόσεις του τμήματος ανάκτησης της εφαρμογής chatbot μας, χρησιμοποιήσαμε το μεγάλο γλωσσικό μοντέλο Llama 3.1 Instruct 8B, προκειμένου να παράγουμε ένα συνθετικό σύνολο δεδομένων. Καταρχάς, διαχωρίσαμε τα δεδομένα που χρησιμοποιούμε για την δημιουργία των συνθετικών δεδομένων σε μικρότερα τμήματα, που αντιστοιχούν σε τμήματα μεγέθους της τάξης των περίπου 500 συμβόλων του μοντέλου παραγωγής ενσωματώσεων που χρησιμοποιήσαμε. Στη συνέχεια διαχωρίζουμε τα τμήματα αυτά σε ζεύγη όπου το δεύτερο τμήμα σε κάθε ζεύγος ακολουθεί νοηματικά το πρώτο. Μέσω κατάλληλα διαμορφωμένων προτροπών, τις οποίες δίνουμε ως είσοδο στο μεγάλο γλωσσικό μοντέλο, λαμβάνουμε ως έξοδο για το εκάστοτε ζεύγος τμημάτων μια ερώτηση που βασίζεται στα περιεχόμενα των εκάστοτε τμημάτων.

Για να αξιολογήσουμε την ποιότητα των ερωτήσεων που παράχθηκαν, χρησιμοποιήσαμε και πάλι το Llama 3.1 Instruct 8B με μια προτροπή αξιολόγησης. Η γενική μορφή των προτροπών που χρησιμοποιήσαμε για την παραγωγή και την αξιολόγηση των ερωτήσεων φαίνεται στον Πίνακας 1.

Στόχος Προτροπής	Προτροπή
Παραγωγή Ερώτησης	**“System message”**: Act as a question generation system. You will be given 2 pieces of context. Using information found in the context and not any other information, provide only a factual question. **“User message”**: ---Context 1: {context2_content} ---Context 2: {context2_content} ---Question:
Αξιολόγηση Ερώτησης	**“System message”**: Act as a judge system. You will be given 2 pieces of context and a question. You must provide the grade and a short explanation on whether the question can be answered through the context. Use a scale of 1 to 5. **“User message”**: ---Context 1:{context1_content} ---Context 2: {context2_content} ---Question: {question_content} ---Grade:

Πίνακας 1: Προτροπές για Παραγωγή και Αξιολόγηση Συνθετικού Συνόλου Δεδομένων για Αξιολόγηση Τμήματος Ανάκτησης

Το τελικό σύνολο δεδομένων για την αξιολόγηση του τμήματος ανάκτησης αποτελείται από τα ζεύγη των τμημάτων κειμένου ομαδοποιημένα με την αντίστοιχη ερώτηση και αξιολόγηση που παράχθηκαν.

6.1.2. Σύνολο Δεδομένων για Αξιολόγηση Συνολικού Συστήματος Chatbot

Ακολουθώντας παρόμοια διαδικασία με αυτή της υπό-ενότητας 6.1.1, δημιουργήσαμε ένα συνθετικό σύνολο δεδομένων για την γενική αξιολόγηση του συστήματος chatbot μας. Ειδικότερα, διαχωρίσαμε τα δεδομένα που χρησιμοποιούμε για τη παραγωγή συνθετικών δεδομένων σε τμήματα που αντιστοιχούν σε περίπου μία σελίδα του εγγράφου. Έπειτα, χρησιμοποιώντας το μεγάλο γλωσσικό μοντέλο που αναφέραμε, για ορισμένες από τις σελίδες αυτές εξάγουμε μια ερώτηση βασισμένη στα περιεχόμενα της και στη συνέχεια εξάγουμε μια υποθετική απάντηση για κάθε τέτοια ερώτηση. Για την παραγωγή των ερωτήσεων και των υποθετικών απαντήσεων καθώς και για την τελική αξιολόγηση του συνόλου δεδομένων μας, χρησιμοποιήσαμε και πάλι κατάλληλα διαμορφωμένες προτροπές τις οποίες δίνουμε ως είσοδο στο μοντέλο, οι οποίες παρουσιάζονται στον Πίνακας 2.

Στόχος Προτροπής	Προτροπή
Παραγωγή Ερώτησης	**“System Message”**: Act as a question generation system. You will be given 1 piece of context. Using information found in the context and not any other information, provide only a factual question. **“User Message”**: ---Context 1: {context_content} ---Question:
Παραγωγή Υποθετικής Απάντησης	“**System Message”**: Act as an answer generation system. You will be given 1 piece of context and 1 question. Using information found in the context and not any other information, provide an answer to the question. **“User Message”**: ---Context 1: {context_content} ---Question:{question_content} ---Answer:"'
Αξιολόγηση Δεδομένων	**“System Message”**: Act as a judge system. You will be given a piece of context and a question. You must provide the grade and a short explanation on whether the question can be answered through the context. Use a scale of 1 to 5. **“User message”**:---Context: {context_content} ---Question: {question_content} ---Grade:

Πίνακας 2: Προτροπές για Παραγωγή και Αξιολόγηση Συνθετικού Συνόλου Δεδομένων

Εκτός του παραπάνω συνθετικού συνόλου δεδομένων, δημιουργήσαμε με χειροκίνητο τρόπο ένα επιπλέον σύνολο δεδομένων μικρού σχετικά μεγέθους. Το σύνολο δεδομένων αυτό περιέχει ερωτήσεις γενικού χαρακτήρα και συνόψεως συγκεκριμένης πληροφορίας, των οποίων η απάντηση μπορεί να εντοπιστεί στο έγγραφο που χρησιμοποιούμε για την αξιολόγηση του συστήματος chatbot μας.

6.2 Αξιολόγηση Ανάκτησης Πληροφορίας

6.2.1 Μετρικές Αξιολόγησης Ανάκτησης Πληροφορίας

Για την αξιολόγηση συστημάτων που έχουν ως στόχο την ανάκτηση πληροφορίας έχουν χρησιμοποιηθεί διάφορες μετρικές αξιολόγησης. Καθεμία από αυτές τις μετρικές αποσκοπεί στο να αξιολογήσει συγκεκριμένες ιδιότητες που κρίνονται σημαντικές σε μία εργασία ανάκτησης πληροφορίας. Παρακάτω παρουσιάζουμε ορισμένες από τις μετρικές αυτές, τις οποίες θα αξιοποιήσουμε προκειμένου να αξιολογήσουμε το τμήμα ανάκτησης πληροφορίας του συστήματος μας.

Recall

Η μετρική **Recall** αποτελεί μια ιδιαίτερα δημοφιλή μετρική η οποία έχει χρησιμοποιηθεί για την αξιολόγηση σε ένα ευρύ πλήθος εργασιών. Συγκεκριμένα για την εργασία της ανάκτησης πληροφορίας, η μετρική **Recall** ορίζεται ως ο λόγος των τμημάτων που ανακτά το σύστημα ανάκτησης πληροφορίας μας τα οποία είναι σχετικά ως προς το ερώτημα-στόχο που το σύστημα έλαβε ως είσοδο, προς το συνολικό πλήθος τμημάτων που είναι σχετικά με το ερώτημα-στόχο [55].

$$Recall = \frac{Πλήθος\ Σχετικών\ Τμημάτων\ που\ Ανακτήθηκαν}{Συνολικό\ Πλήθος\ Σχετικών\ Τμημάτων} \quad (6.\ 1)$$

Hit Rate

Μια ακόμη χρήσιμη μετρική αποτελεί η μετρική **Hit Rate**. Η μετρική αυτή είναι αρκετά όμοια με τη μετρική recall. Διαφέρει στο ότι αξιολογεί απλώς το αν το σύστημα ανάκτησης πληροφορίας μας ανακτά οποιοδήποτε πλήθος σχετικών τμημάτων. Συνεπώς η μετρική Hit Rate λαμβάνει τιμή 1 στη περίπτωση όπου στα τμήματα που ανακτήθηκαν περιέχεται έστω και ένα σχετικό τμήμα με το ερώτημα-στόχο, ενώ σε αντίθετη περίπτωση λαμβάνει τιμή 0.

Mean Reciprocal Rank (MRR)

Οι δύο προηγούμενες μετρικές μας βοηθούν να αξιολογήσουμε το σύστημα μας ως το κατά πόσο είναι ικανό να ανακτήσει τα σχετικά με ένα ερώτημα τμήματα πληροφορίας. Ωστόσο, δεν μας προσφέρουν κάποια πληροφορία σχετικά με τη θέση των σχετικών τμημάτων στο συνολικό πλήθος των τμημάτων που ανακτήθηκαν. Μια μετρική που έχει χρησιμοποιηθεί και μας επιτρέπει να εξάγουμε πληροφορία για τον παραπάνω σκοπό αποτελεί η μετρική MRR (Mean Reciprocal Rank). Η μετρική MRR μπορεί να λάβει διάφορες τιμές μεταξύ των αριθμών 1 και 0. Πιο αναλυτικά, για ένα πλήθος ανακτημένων τμημάτων του συστήματος μας για ένα ερώτημα-στόχο, η τιμή της μετρικής MRR αντιστοιχεί κάθε φορά στον αντίστροφο αριθμό της θέσης στην οποία μπορούμε να εντοπίσουμε για πρώτη φορά ένα σχετικό τμήμα στο πλήθος των ανακτημένων δεδομένων. Σε περίπτωση που κανένα σχετικό τμήμα δεν περιέχεται στο σύνολο των ανακτημένων δεδομένων, η μετρική λαμβάνει την τιμή 0 [71].

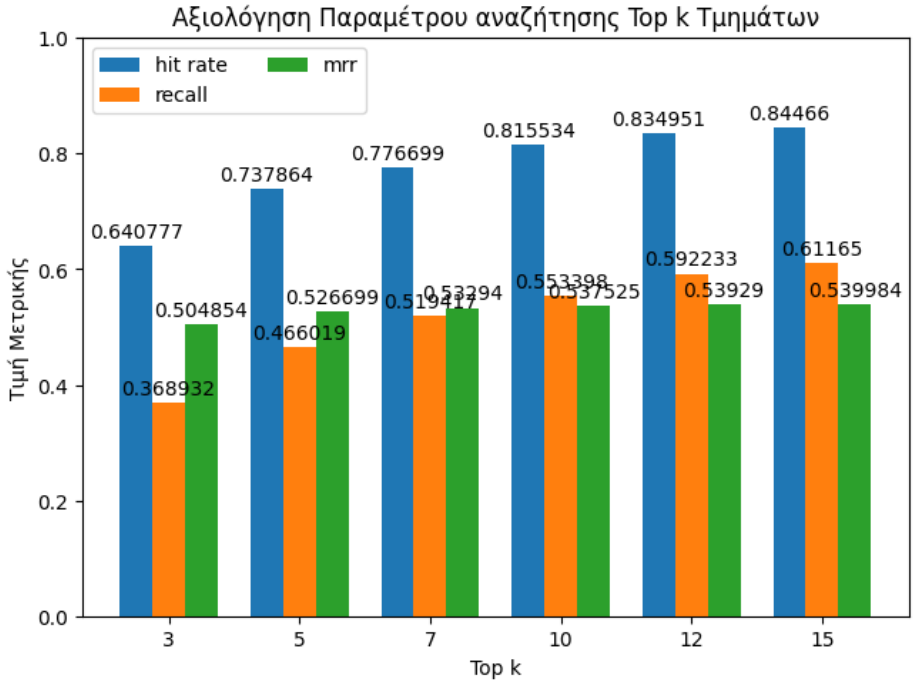
$$MRR = \begin{cases} \frac{1}{n}, (n: \Theta\acute{\epsilon}\sigma\eta\ \epsilon\mu\phi\acute{\alpha}\nu\iota\sigma\eta\varsigma\ \pi\rho\acute{\omega}\tau\sigma\upsilon\ \sigma\chi\epsilon\tau\iota\kappa\sigma\acute{\upsilon}\ \tau\mu\acute{\eta}\mu\alpha\tau\sigma\varsigma) \\ 0, (\Delta\epsilon\nu\ \alpha\nu\alpha\kappa\tau\acute{\eta}\theta\eta\kappa\epsilon\ \sigma\chi\epsilon\tau\iota\kappa\acute{\sigma}\ \tau\mu\acute{\eta}\mu\alpha) \end{cases} \quad (6.\ 2)$$

6.2.2 Αξιολόγηση Ανάκτησης Πληροφορίας

Για την αξιολόγηση του τμήματος ανάκτησης πληροφορίας αξιοποιήσαμε, όπως αναφέραμε, το σύνολο δεδομένων που δημιουργήσαμε για το σκοπό αυτό. Αρχικά υπολογίσαμε με χρήση του μοντέλου ενσωματώσεων τις ενσωματώσεις για τα τμήματα δεδομένων από τα οποία εξαγάγαμε τις ερωτήσεις που περιέχονται στο σύνολο δεδομένων μας και έπειτα τα αποθηκεύσαμε στη διανυσματική βάση η οποία δημιουργεί ένα ευρετήριο για τα τμήματα αυτά. Στη συνέχεια χρησιμοποιούμε τα ερωτήματα που περιέχονται στο αντίστοιχο σύνολο δεδομένων, υπολογίζοντας για καθένα από αυτά την ενσωμάτωση του. Ακολούθως εκτελούμε αναζήτηση ομοιότητας στη διανυσματική βάση προκειμένου να ανακτήσουμε ένα πλήθος τμήματα δεδομένων των οποίων τα περιεχόμενα είναι σχετικά με το εκάστοτε ερώτημα. Παρακάτω παρουσιάζουμε τα αποτελέσματα που λάβαμε κατά την εκτέλεση ορισμένων πειραμάτων.

Αποτελέσματα για Παράμετρο Αναζήτησης k Σχετικών Τμημάτων

Έχοντας αποθηκεύσει τα τμήματα δεδομένων του συνόλου αξιολόγησης στη διανυσματική βάση, ελέγχουμε τις επιδόσεις του τμήματος ανάκτησης, ανακτώντας κάθε φορά διαφορετικό πλήθος σχετικών τμημάτων για το κάθε ερώτημα θέτοντας διαφορετικές τιμές στην αντίστοιχη παράμετρο αναζήτησης (top k) και υπολογίζοντας τις τιμές των μετρικών recall, hit rate και mrr για κάθε τιμή της παραμέτρου. Τα αποτελέσματα για τις μετρικές αξιολόγησης φαίνονται στην Εικόνα 29.



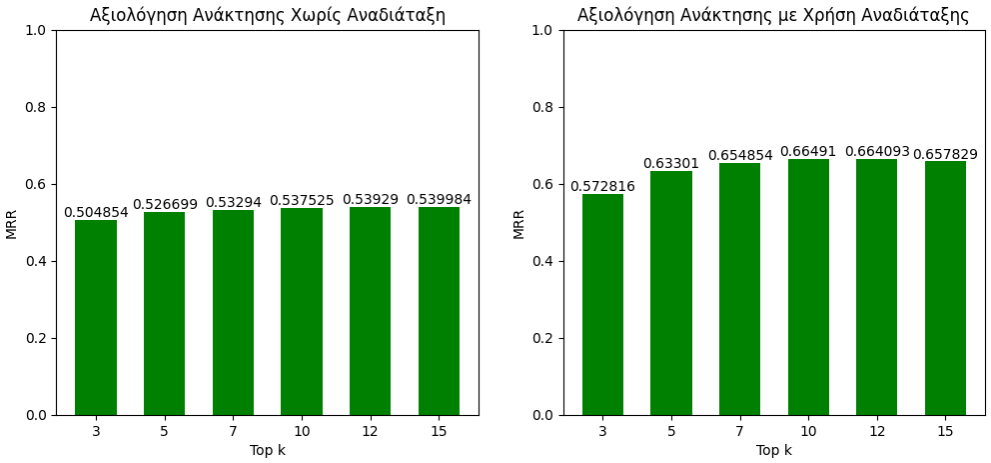
Εικόνα 29: Επιδόσεις Τμήματος Ανάκτησης για Διάφορες Τιμές της Παραμέτρου Top k

Παρατηρούμε ότι με αύξηση της παραμέτρου Top k οι μετρικές Hit Rate και Recall αυξάνονται σημαντικά, κάτι που είναι αναμενόμενο καθώς ανακτώντας περισσότερα τμήματα έχουμε περισσότερες πιθανότητες να ανακτήσουμε τα σχετικά με το ερώτημα τμήματα. Η μετρική mrr, παρουσιάζει και αυτή αύξηση αλλά σε πολύ μικρότερο βαθμό σε σχέση με τις υπόλοιπες μετρικές, κάτι που πιθανότητα οφείλεται στο ότι η εύρεση παραπάνω του ενός σχετικού τμήματος δεν την επηρεάζει. Λαμβάνοντας υπόψιν ότι η αύξηση των μετρικών για τις τιμές 10 και έπειτα είναι αρκετά μικρότερη σε σχέση με αυτές των προηγούμενων τιμών, καταλήγουμε στο ότι το εύρος τιμών 5-10 για την παράμετρο Top k αποτελεί μια ικανοποιητική επιλογή.

Αποτελέσματα Χρήσεως Αναδιάταξης Τμημάτων

Το σύστημα chatbot που σχεδιάζουμε χρειάζεται να είναι σε θέση να αξιοποιήσει όσο το δυνατόν πιο βέλτιστα τα τμήματα που ανακτώνται από το τμήμα ανάκτησης. Στην εργασία τους, οι Liu et al. [72] διαπιστώνουν ότι τα μεγάλα γλωσσικά μοντέλα αξιοποιούν με διαφορετικό τρόπο τις πληροφορίες που δέχονται ως είσοδο, ανάλογα με την θέση στην οποία βρίσκεται η καθεμία από αυτές. Πιο συγκεκριμένα, εξηγούν ότι αν πραγματοποιηθεί τοποθέτηση μιας χρήσιμης πληροφορίας, όσον αφορά την απάντηση ενός ερωτήματος, σε ενδιάμεσα τμήματα της εισόδου ενός τέτοιου μοντέλου, η ποιότητα της εξόδου του μοντέλου τείνει να είναι χειρότερη σε σχέση με το αν η πληροφορία είχε τοποθετηθεί, αντιθέτως, στην αρχή ή το τέλος της εισόδου του.

Μια τεχνική που έχει προταθεί για την αντιμετώπιση του παραπάνω ζητήματος αποτελεί το να αναδιατάξουμε τα τμήματα που έχουν ανακτηθεί από το τμήμα ανάκτησης μας, με στόχο την παρουσία των σχετικών τμημάτων σε όσο πιο αρχικό σημείο της εισόδου του μεγάλου γλωσσικού μοντέλου [72]. Ένας τρόπος για να επιτευχθεί η αναδιάταξη αυτή αποτελεί η χρήση ενός Cross Encoder, που όπως είδαμε αποτελεί ένα μοντέλο μηχανικής μάθησης που έχοντας ως είσοδο το σχετικό ερώτημα ακολουθούμενο κάθε φορά από ένα ανακτημένο τμήμα δεδομένων, μας δίνει ως έξοδο μια τιμή που αντιστοιχεί στη βαθμολογία της σχετικότητας μεταξύ των δύο κειμένων που δόθηκαν ως είσοδος. Όπως εξηγήσαμε, η βαθμολογία αυτή είναι πολλές φορές πιο ορθή, καθώς πετυχαίνεται εφαρμογή των μηχανισμών προσοχής όχι μόνο για μια πρόταση, όπως γίνεται κατά τη διαδικασία της παραγωγής ενσωματώσεων για χρήση στο τμήμα ανάκτησης, αλλά για το ζεύγος του ερωτήματος και του εκάστοτε ανακτημένου τμήματος [73]. Υπολογίζοντας άρα τις βαθμολογίες αυτές για κάθε ανακτημένο τμήμα, είμαστε σε θέση να τα αναδιατάξουμε αρτιότερα.



Εικόνα 30: Αποτελέσματα Απλής Ανάκτησης (Αριστερά) και Ανάκτησης με Χρήση Αναδιάταξης (Δεξιά)

Για την εφαρμογή της διαδικασίας ανάκτησης που περιγράψαμε, δοκιμάσαμε να χρησιμοποιήσουμε το προ-εκπαιδευμένο μοντέλο **ms-marco-MiniLM-L6-v2** [74]. Το προ-εκπαιδευμένο μοντέλο αυτό έχει εκπαιδευτεί στο σύνολο δεδομένων MS MARCO, το οποίο αποτελεί ένα σύνολο δεδομένων στο οποίο περιέχονται ζεύγη ερωτήσεων και σχετικών παραγράφων, πετυχαίνοντας ικανοποιητικές επιδόσεις. Στην Εικόνα 30 παρουσιάζουμε τα αποτελέσματα που είχε η διαδικασία αναδιάταξης στο σύνολο αξιολόγησης μας. Αξιολογήσαμε τη τεχνική αυτή ως προς τη μετρική mrr, καθώς στοχεύει στην εμφάνιση των σχετικών τμημάτων όσο το δυνατόν πιο νωρίς. Παρατηρούμε ότι πράγματι η χρήση αναδιάταξης ενός κατάλληλου προ-εκπαιδευμένου μοντέλου, ακόμα και χωρίς περαιτέρω εκπαίδευση, αυξάνει σημαντικά την ποιότητα της ανάκτησης ως προς το ζητούμενο κριτήριο.

6.3 Συνολική Αξιολόγηση Συστήματος Chatbot

6.3.1 Μετρικές

Ένα σύστημα chatbot που αξιοποιεί το μοτίβο RAG είναι δύσκολο να αξιολογηθεί ως σύνολο, καθώς αποτελείται από σύνθετα επιμέρους τμήματα που αξιοποιούν πληθώρα τεχνικών, οι οποίες με τη σειρά τους αυξάνουν την πολυπλοκότητα του συστήματος. Για την επίτευξη της αξιολόγησης τέτοιων συστημάτων ως σύνολο, έχουν προταθεί διάφορες μετρικές, καθεμιά από τις οποίες αποσκοπεί στο να αξιολογήσει ένα διαφορετικό στόχο του συστήματος [75]. Πολλές από αυτές παρουσιάζουν επίσης το πλεονέκτημα του ότι δεν απαιτούν την ύπαρξη ιδανικών απαντήσεων για την πραγματοποίηση σύγκρισης μεταξύ αυτής και της παραγόμενης από το σύστημα απάντησης.

Ο υπολογισμός των μετρικών αυτών είναι συνήθως δυνατόν να πραγματοποιηθεί με παραπάνω από έναν τρόπο. Μια πιθανή εφαρμογή που έχει δοκιμαστεί είναι η χρήση ενός μεγάλου γλωσσικού μοντέλου που πραγματοποιεί τις αξιολογήσεις, έχοντας ως είσοδο κατάλληλες προτροπές, στις οποίες περιέχεται και ο τρόπος με τον οποίο υπολογίζεται η εκάστοτε μετρική [75]. Ανάλογα με το πλήθος και την ποιότητα των προτροπών, αλλά και τις δυνατότητες του μεγάλου γλωσσικού μοντέλου που θα χρησιμοποιηθεί, μπορούμε να πραγματοποιήσουμε μια ικανοποιητική αξιολόγηση του συστήματος μας ως προς τους στόχους που κρίνουμε σημαντικούς. Για την αξιολόγηση του συστήματος chatbot μας επιλέγουμε τις παρακάτω μετρικές.

Ακριβολογία (Faithfulness)

Η μετρική **Ακριβολογίας** έχει ως στόχο να αναδείξει το κατά πόσο το περιεχόμενο μιας απάντησης που παράγεται από το σύστημα chatbot μας είναι βασισμένη στα τμήματα δεδομένων που έχουν ανακτηθεί από το τμήμα ανάκτησης πληροφορίας του συστήματος μας. Συγκεκριμένα, στην προτροπή που χρησιμοποιήσαμε, αξιολογούμε την ποιότητα της παραγόμενης απάντησης ως προς το παραπάνω κριτήριο χρησιμοποιώντας ως βαθμολογική κλίμακα τις τιμές 1 έως 5, όπου μια απάντηση που δε βασίζεται καθόλου στα ανακτημένα τμήματα δεδομένων βαθμολογείται με την τιμή 1, ενώ μια απάντηση της οποίας τα περιεχόμενα βασίζονται πλήρως στα ανακτημένα τμήματα βαθμολογείται με την τιμή 5.

Σχετικότητα (Relevance)

Μια δεύτερη χρήσιμη μετρική αποτελεί αυτή της **Σχετικότητας.** Η μετρική αυτή στοχεύει στο να αξιολογήσει το κατά πόσο τα περιεχόμενα μιας απάντησης που παράγεται από το σύστημα chatbot μας είναι χρήσιμα και σχετικά ως προς το ερώτημα που το chatbot μας έλαβε ως είσοδο. Ακολουθώντας αντίστοιχη διαδικασία με αυτή που περιγράψαμε για την προηγούμενη μετρική, χρησιμοποιήσαμε και σε αυτή τη περίπτωση μια κατάλληλα διαμορφωμένη προτροπή ώστε να βαθμολογήσουμε τις παραγόμενες απαντήσεις ως προς τον ορισμό της μετρικής της Σχετικότητας, σε βαθμολογική κλίμακα από 1 έως 5.

6.3.2 Διαδικασία Αξιολόγησης

Στην υπό-ενότητα αυτή εξηγούμε την γενική διαδικασία που ακολουθούμε προκειμένου να αξιολογήσουμε την επίδοση του chatbot μας.

Για την αξιολόγηση βάση των δύο μετρικών που αναφέραμε στη προηγούμενη υπό-ενότητα, αξιοποιήσαμε το μεγάλο γλωσσικό μοντέλο Llama 3.1 8B Instruct. Ειδικότερα, δίνοντας στο μοντέλο ως είσοδο μια κατάλληλα διαμορφωμένη προτροπή για καθεμία από τις μετρικές Ακριβολογίας και Σχετικότητας, εξάγουμε τις δύο αντίστοιχες βαθμολογίες για κάθε μία από τις παραγόμενες απαντήσεις του chatbot μας.

Μετρική	**Προτροπή**
Ακριβολογίας	**System**: You will be given a question and an answer. Your task is to grade whether the claims made in the answer are relevant and useful for the question being asked. Provide a very short explanation (1-2 sentences) followed by your grade in a scale of 1-5. Grade Explanation: {1. No Relevancy, 2. Low Relevancy, 3. Medium Relevancy, 4. Relevant, 5. Very Relevant} --Grading Rules: You should lower your grade if the answer contains : -Claims that are not helpful for answering the question. -Claims that are not relevant to the given question. **User**: ---Question: [{question_placeholder}] ---Answer: [{answer_placeholder}] ---Grade:

Σχετικότητας	**System**: You will be given a question and an answer. Your task is to grade whether the claims made in the answer are relevant and useful for the question being asked. Provide a very short explanation (1-2 sentences) followed by your grade in a scale of 1-5. Grade Explanation: {1. No Relevancy, 2. Low Relevancy, 3. Medium Relevancy, 4. Relevant, 5. Very Relevant} --Grading Rules: You should lower your grade if the answer contains : -Claims that are not helpful for answering the question. -Claims that are not relevant to the given question. **User**: ---Question: [{question_placeholder}] ---Answer: [{answer_placeholder}] ---Grade:

Πίνακας 3: Προτροπές για Εξαγωγή Βαθμολογίας των Μετρικών Αξιολόγησης

Στον Πίνακας 3 φαίνονται οι σκελετοί των προτροπών που χρησιμοποιήσαμε προκειμένου να εξάγουμε τις βαθμολογίες για τις δύο μετρικές αξιολόγησης μας. Αξίζει να σημειώσουμε ότι μεταξύ του μηνύματος συστήματος και του τελικού μηνύματος χρήστη προσθέσαμε ορισμένα παραδείγματα σε μορφή ζευγών μηνυμάτων χρήστη-βοηθού, προκειμένου να βοηθήσουμε το μεγάλο γλωσσικό μοντέλο να κατανοήσει ορθότερα τις απαιτήσεις της εκάστοτε εργασίας αξιολόγησης.

Για την διαδικασία αξιολόγησης του συστήματος chatbot μας, έχοντας καθορίσει τις παραμέτρους που επιθυμούμε να αξιολογήσουμε, χρησιμοποιούμε ένα σύνολο 50 ερωτήσεων. Πιο αναλυτικά, το σύνολο αυτό αποτελείται από 30 ερωτήσεις του συνθετικού συνόλου δεδομένων και 20 ερωτήσεις του συνόλου δεδομένων που δημιουργήσαμε με χειροκίνητο τρόπο.

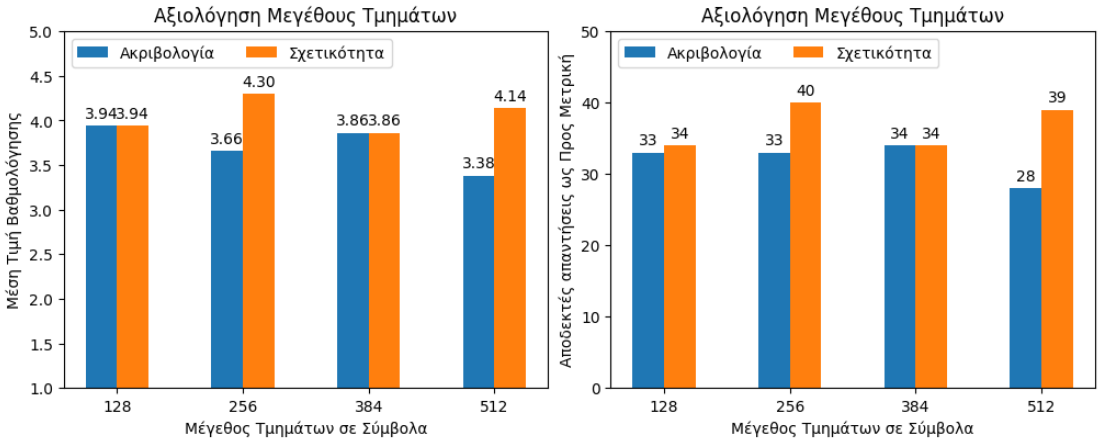
6.3.3 Αποτελέσματα Πειραμάτων Αξιολόγησης του Συνολικού Συστήματος

Στην υπό-ενότητα αυτή παρουσιάζουμε τα αποτελέσματα ορισμένων πειραμάτων που εκτελέσαμε, προκειμένου να ελέγξουμε τον τρόπο με τον οποίο επηρεάζονται οι επιδόσεις του chatbot μας, αν τροποιήσουμε ορισμένες από τις παραμέτρους του.

Μέγεθος Τμημάτων Σχετικών Δεδομένων

Το μέγεθος των τμημάτων σχετικών δεδομένων είναι μια σημαντική παράμετρος του συστήματος μας, καθώς επηρεάζει την ποιότητα της διαδικασίας παραγωγής των ενσωματώσεων που τα αντιπροσωπεύουν αλλά και την ποσότητα της σχετικής με το ερώτημα πληροφορίας που θα δοθεί ως είσοδος στο τμήμα παραγωγής απάντησης του chatbot μας. Τα αποτελέσματα της αξιολόγησης για τέσσερα διαφορετικά μεγέθη τμημάτων (128, 256, 384 και 512 σύμβολα του μοντέλου ενσωματώσεων) φαίνονται στην Εικόνα 31. Συγκεκριμένα, παρουσιάζονται οι μέσες τιμές των βαθμολογιών και

το πλήθος των αποδεκτών απαντήσεων για τις δύο μετρικές αξιολόγησης, για το σύνολο των 50 ερωτήσεων αξιολόγησης. Ως αποδεκτές απαντήσεις για μια μετρική, θεωρούμε αυτές που έλαβαν βαθμολογία μεγαλύτερη ή ίση με 4.

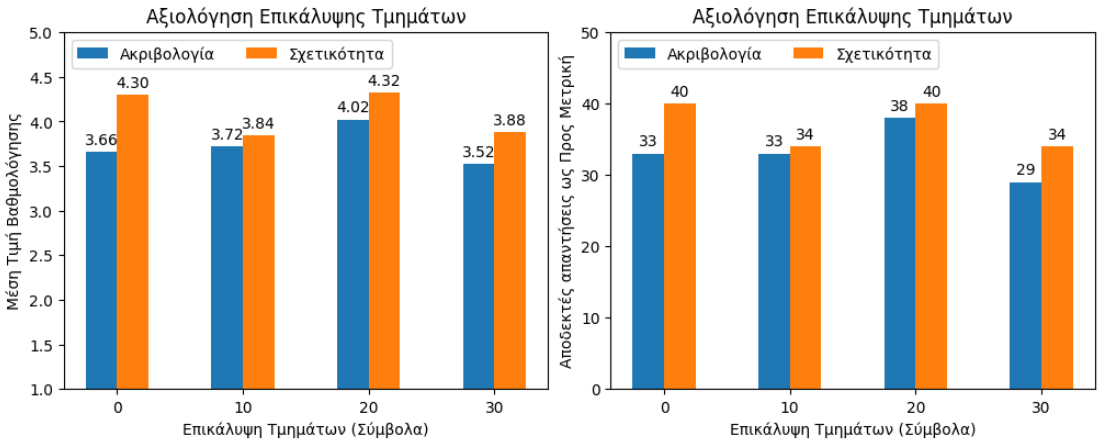


Εικόνα 31: Αποτελέσματα μετρικών (Αριστερά) και Αποδεκτές Ερωτήσεις (Δεξιά) για Διαφορετικά Μεγέθη Τμημάτων

Παρατηρούμε ότι η μέση τιμή αλλά και το πλήθος αποδεκτών απαντήσεων της μετρικής Ακριβολογίας είναι μεγαλύτερη για τα μικρότερα μεγέθη τμημάτων, κάτι που πιθανόν οφείλεται στο ότι το μεγάλο γλωσσικό μοντέλο λαμβάνει λιγότερη πληροφορία-θόρυβο όσο μικρότερο είναι το μέγεθος των τμημάτων, μειώνοντας έτσι τα φαινόμενα παραισθήσεων στις παραγόμενες απαντήσεις. Το μέγεθος τμήματος 256 συμβόλων θα μπορούσε να αποτελέσει την προτιμότερη επιλογή καθώς πετυχαίνει αρκετά καλά αποτελέσματα για την μετρική της Ακριβολογίας αλλά ταυτόχρονα το αρτιότερο αποτέλεσμα για την μετρική Σχετικότητας.

Επικάλυψη Τμημάτων Σχετικών Δεδομένων

Μια επιπλέον παράμετρος που μπορούμε να αξιοποιήσουμε κατά τη διάσπαση του κειμένου μας σε μικρότερα τμήματα, αποτελεί το να επιτρέψουμε την επικάλυψη των τμημάτων για ορισμένο πλήθος συμβόλων. Επιτρέποντας την επικάλυψη μεταξύ των τμημάτων του κειμένου προκύπτουν περισσότερα τέτοια τμήματα κατά τον διαχωρισμό και άρα αναμένουμε να έχουμε υψηλότερη ακρίβεια στη διαδικασία ανάκτησης τμημάτων κειμένου που κρίνονται ως σχετικά από το τμήμα ανάκτησης.



Εικόνα 32: Αποτελέσματα μετρικών (Αριστερά) και Αποδεκτές Ερωτήσεις (Δεξιά) για Διαφορετικό Μέγεθος Παραθύρου Επικάλυψης

Στην Εικόνα 32 παρουσιάζονται τα αποτελέσματα των δοκιμών αξιολόγησης μας, για μέγεθος του παραθύρου επικάλυψης 0 (μη εφαρμογή επικάλυψης), 10, 20 και 30 συμβόλων του μοντέλου ενσωματώσεων, για το σύνολο των 50 ερωτήσεων αξιολόγησης. Για τις δοκιμές αυτές θέσαμε το μέγεθος των παραγόμενων τμημάτων στα 256 σύμβολα. Παρατηρούμε ότι η μετρική της ακριβολογίας αυξάνεται για τα μεγέθη παραθύρων 10 και 20 συμβόλων και μειώνεται για μέγεθος παραθύρου 30 συμβόλων, σε σχέση με την περίπτωση μη εφαρμογής επικάλυψης. Όσον αφορά την σχετικότητα, παρατηρούμε ελάχιστη αύξηση για μέγεθος παραθύρου 20 συμβόλων και μείωση για τα υπόλοιπα μεγέθη. Με βάση τις παραπάνω δοκιμές, μπορούμε να καταλήξουμε στο ότι η χρήση ενός παραθύρου επικάλυψης καταλλήλου μεγέθους, όπου στην περίπτωση των δεδομένων μας αυτό αντιστοιχεί σε περίπου 20 σύμβολα, είναι σε θέση να αυξήσει τις επιδόσεις του συστήματος μας.

Προτροπή Μεγάλου Γλωσσικού Μοντέλου

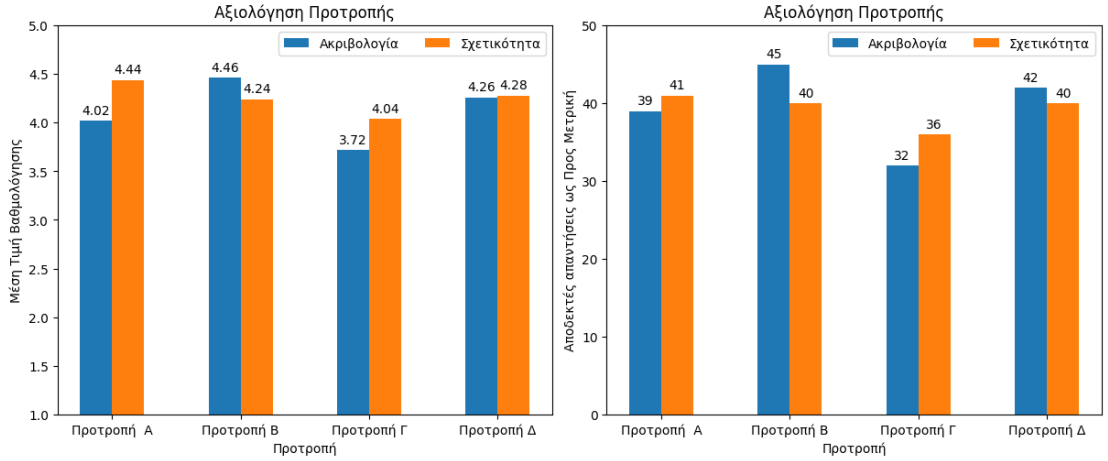
Η μορφή της προτροπής που δίνουμε ως είσοδο στο μεγάλο γλωσσικό μοντέλο μας είναι μια ιδιαίτερα σημαντική παράμετρος στη διαδικασία της παραγωγής της απάντησης του συστήματος chatbot μας. Προκειμένου να αξιολογήσουμε διάφορες τεχνικές με τις οποίες μπορούμε να σχηματίσουμε την προαναφερθείσα προτροπή, εξήγαμε και αξιολογήσαμε τις απαντήσεις που παράγεται από το chatbot μας χρησιμοποιώντας κάθε φορά μια διαφορετική προτροπή συστήματος, το σύνολο των οποίων φαίνεται στον Πίνακας 4, μαζί με την προτροπή χρήστη την οποία διατηρούμε ίδια για όλες τις εκδοχές. Η Προτροπή Α αποτελεί την απλή εκδοχή της προτροπής, όπου δίνουμε στο μεγάλο γλωσσικό μοντέλο την οδηγία να παράγει την απάντηση στο ερώτημα του χρήστη, χρησιμοποιώντας μόνο τα ανακτημένα τμήματα δεδομένων και όχι κάποια άλλη μη σχετική γνώση. Η προτροπή Β δίνει επιπλέον την οδηγία στο μοντέλο να παράγει μια απάντηση ενός καθορισμένου μέγιστου μεγέθους. Τέλος, οι προτροπές Γ και Δ αποτελούν παραλλαγές της Προτροπής Α, όπου στην πρώτη δίνονται επίσης δύο παραδείγματα της απαιτούμενης εργασίας στο μοντέλο, ενώ στη δεύτερη δίνεται η οδηγία να παράγει την απάντηση του σταδιακά.

Όνομα Προτροπής	**Πρότυπο Προτροπής**
Προτροπή Α	You are a helpful assistant. You are given some pieces of context and a question. Using information from the context and not any other prior knowledge, answer the question.
Προτροπή Β	You are a helpful assistant. You are given some pieces of context and a question. Using information from the context and not any other prior knowledge, answer the question. Your answer should be around three to five sentences long.
Προτροπή Γ	You are a helpful assistant. You are given some pieces of context and a question. Using information from the context and not any other prior knowledge, answer the question. Reference examples: -Example 1: (example_content) -Example 2: (example_content)

Προτροπή Δ	You are a helpful assistant. You are given some pieces of context and a question. Using information from the context and not any other prior knowledge, answer the question. Please think of your answer step by step.
Προτροπή Χρήστη	---Context: {context_placeholder} ---Question: {question_placeholder} ---Answer:

Πίνακας 4: Προτροπές Συστήματος και Πρότυπο Προτροπής Χρήστη

Τα αποτελέσματα των μετρικών αξιολόγησης για τις 4 αυτές προτροπές συστήματος καθώς και οι αποδεκτές ερωτήσεις για κάθε μετρική, για το σύνολο των ερωτήσεων αξιολόγησης μας, παρουσιάζονται στην Εικόνα 33.



Εικόνα 33:Αποτελέσματα Αξιολόγησης Προτροπής Μεγάλου Γλωσσικού Μοντέλου

Παρατηρούμε ότι η προτροπές Β και Δ πετυχαίνουν σημαντικά βελτιωμένες επιδόσεις όσον αφορά τη μετρική της Ακριβολογίας έχοντας ταυτόχρονα μικρές επιπτώσεις ως προς την μετρική Σχετικότητας, σε σχέση με τις επιδόσεις στη περίπτωση όπου το chatbot χρησιμοποιεί την προτροπή Α που αποτελεί όπως αναφέραμε την απλούστερη εκδοχή. Συνεπώς, μπορούμε να οδηγηθούμε στο συμπέρασμα του ότι ο περιορισμός του μεγέθους της απάντησης του chatbot μας σε κατάλληλο μέγεθος αλλά και η παροχή της εντολής για σταδιακή ανάπτυξη των απαντήσεων του είναι σε θέση να αυξήσουν την ορθότητα των τελευταίων, έχοντας ως αντίτιμο σχετικά αποδεκτές μειώσεις στην σχετικότητα τους.

Όσον αφορά την προτροπή Γ, παρατηρούμε ότι και οι δύο μετρικές αξιολόγησης μειώνονται αισθητά συγκριτικά με την προτροπή Α. Αυτό ενδεχομένως οφείλεται στο ότι τα παραδείγματα που εισήγαμε στην προτροπή προσαρμόζουν τις εξόδους που παράγονται από το μεγάλο γλωσσικό μοντέλο πάνω σε αυτά και στο ότι πιθανότατα τα παραδείγματα δεν σχετίζονται άμεσα με όλα τα ερωτήματα αξιολόγησης μας, μειώνοντας έτσι την ποιότητα των παραγόμενων απαντήσεων του chatbot.

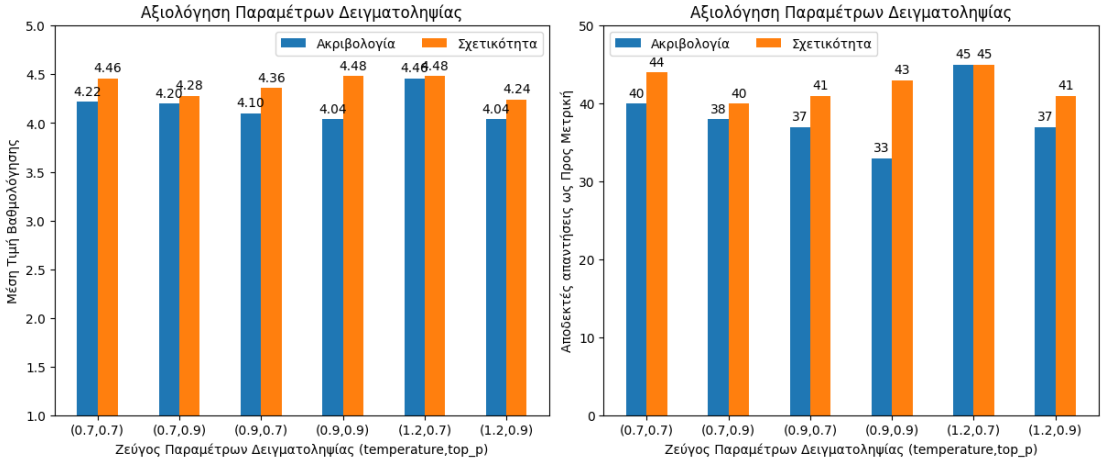
Παράμετροι Δειγματοληψίας Μεγάλου Γλωσσικού Μοντέλου

Στο Κεφάλαιο 2 παρουσιάσαμε τις μεθόδους δειγματοληψίας θερμοκρασίας (temperature sampling) και δειγματοληψίας πυρήνα (top_p sampling), οι οποίες μπορούν να χρησιμοποιηθούν κατά την παραγωγή της εξόδου ενός μεγάλου γλωσσικού μοντέλου. Δοκιμάσαμε να παράγουμε απαντήσεις για το σύνολο των ερωτήσεων αξιολόγησης μας χρησιμοποιώντας το chatbot μας, θέτοντας κάθε φορά διαφορετικές τιμές των παραμέτρων temperature και top_p, που χρησιμοποιούνται από τις δύο τεχνικές δειγματοληψίας που αναφέραμε παραπάνω. Τα ζεύγη που χρησιμοποιήσαμε φαίνονται στον Πίνακας 5.

# Ζεύγους Παραμέτρων	temperature	top_p
1	0.7	0.7
2	0.7	0.9
3	0.9	0.7
4	0.9	0.9
5	1.2	0.7
6	1.2	0.9

Πίνακας 5: Ζεύγη παραμέτρων δειγματοληψίας συμβόλων εξόδου temperature και top_p που αξιολογήθηκαν

Τα αποτελέσματα για το σύνολο των ερωτήσεων αξιολόγησης μας, για όλα τα παραπάνω ζεύγη παραμέτρων temperature και top_p, φαίνονται στην Εικόνα 34.



Εικόνα 34: Αποτελέσματα Αξιολόγησης Παραμέτρων Δειγματοληψίας Μεγάλου Γλωσσικού Μοντέλου

Παρατηρούμε ότι τα ζεύγη με τιμή της παραμέτρου top_p ίση με 0.7 πετυχαίνουν καλύτερες επιδόσεις ως προς την μετρική ακριβολογίας συγκριτικά με τα αντίστοιχα ζεύγη τους, όπου η παράμετρος top_p λαμβάνει τιμή 0.9. Αυτό πιθανότατα οφείλεται στο ότι τα παραγόμενα σύμβολα εξόδου περιορίζονται σε αυτά που το μεγάλο γλωσσικό μοντέλο πιστεύει ότι είναι πιο πιθανά, όσο μειώνουμε την παράμετρο top_p. Παρατηρούμε επίσης ότι ο συνδυασμός της τιμής 1.2 για την παράμετρο temperature και της τιμής 0.7 για την παράμετρο top_p, πετυχαίνει τις ισχυρότερες επιδόσεις για τις δύο μετρικές για το σύνολο αξιολόγησης μας, σε σχέση με τα υπόλοιπα ζεύγη τιμών. Συνεπώς, καταλήγουμε στο ότι η επιλογή των τιμών των παραμέτρων δειγματοληψίας στη περιοχή των παραπάνω τιμών, είναι προτιμότερη για το chatbot μας.

7. Μελλοντική Εργασία / Συμπεράσματα

Το σύστημα chatbot που υλοποιήσαμε στα πλαίσια της παρούσας διπλωματικής εργασίας, όπως είδαμε και στο Κεφάλαιο 6, είναι σε θέση να απαντήσει ικανοποιητικά σε διάφορες ερωτήσεις σχετικές με το σύνολο δεδομένων μας. Συνεπώς, το chatbot αυτό είναι ικανό να συμβάλλει σημαντικά στην εύρεση της εκάστοτε επιθυμητής πληροφορίας που αναζητά ο χρήστης του, διευκολύνοντας τον. Ωστόσο, αξίζει να αναφέρουμε ότι παρόλο που οι τεχνικές που εφαρμόσαμε για τη σχεδίαση του chatbot μας αποσκοπούν στο να ελαττώσουν τη πιθανότητα εσφαλμένων απαντήσεων, δεν μπορούμε να αποκλείσουμε απόλυτα το ενδεχόμενο της παραγωγής μιας ανακριβής ή λανθασμένης απάντησης.

Προκειμένου να επιτευχθεί η περαιτέρω αύξηση των επιδόσεων του συστήματος chatbot, κρίνεται σκόπιμη η διερεύνηση του αν η εφαρμογή επιπλέον τεχνικών πέρα από αυτές που ήδη εφαρμόσαμε στα τμήματα ανάκτησης δεδομένων και παραγωγής απάντησης του chatbot μας είναι ικανές να αυξήσουν την ποιότητα των απαντήσεων που παράγονται από αυτό. Επίσης, μια άλλη παράμετρος της οποίας η επιρροή στις επιδόσεις του συστήματος chatbot θα μπορούσε να διερευνηθεί, αποτελεί η περαιτέρω εκπαίδευση των μοντέλων παραγωγής ενσωματώσεων κειμένου αλλά και του μεγάλου γλωσσικού μοντέλου που είναι υπεύθυνο για την παραγωγή της απάντησης του chatbot, μέσω μιας διαδικασίας βελτίωσης με χρήση κατάλληλων δεδομένων.

Το σύστημα chatbot μας θα μπορούσε στο μέλλον να επεκταθεί προκειμένου να μπορεί να επεξεργαστεί όχι μόνο απλό κείμενο, αλλά και εικόνες που περιέχονται στα έγγραφα που αποτελούν το σύνολο δεδομένων μας. Για το σκοπό αυτό θα μπορούσαν να αξιοποιηθούν μοντέλα παραγωγής ενσωματώσεων και μεγάλα γλωσσικά μοντέλα που δέχονται ως είσοδο δεδομένα και σε μορφή εικόνας, προκειμένου το σύστημα chatbot να είναι σε θέση να αξιοποιήσει και τέτοιου είδους πληροφορία κατά την παραγωγή των απαντήσεων του.

Τέλος, όπως εξηγήσαμε και στα προηγούμενα κεφάλαια, η ποιότητα και ο τρόπος αποθήκευσης των δεδομένων μας είναι ιδιαίτερα σημαντικοί παράγοντες όσον αφορά την ποιότητα των απαντήσεων του συστήματος chatbot μας. Για τον λόγο αυτό, θα προτείναμε την περαιτέρω διερεύνηση και άλλων τρόπων, πέρα από αυτούς που αξιοποιήσαμε, με τους οποίους θα μπορούσε να πραγματοποιηθεί η εξαγωγή των δεδομένων μας από τα αρχικά έγγραφα και η μετατροπή τους σε μορφή χρήσιμη ως προς το μοντέλο παραγωγής της απάντησης του συστήματος chatbot. Ένας τέτοιος πιθανός τρόπος θα μπορούσε να αποτελεί η χρήση ενός κατάλληλου μηχανισμού ο οποίος να είναι σε θέση να βοηθήσει στον εντοπισμό της δομής ορισμένων στοιχείων δεδομένων, όπως για παράδειγμα οι πίνακες, σε περιπτώσεις όπου η δομή των στοιχείων αυτών είναι περίπλοκη ούτως ώστε να αυξηθεί το ποσοστό της ορθής εξαγωγής τους.

Βιβλιογραφία

[1] "What Is a Chatbot? | IBM." Accessed: Apr. 27, 2024. [Online]. Available: https://www.ibm.com/topics/chatbots

[2] "What Is Machine Learning (ML)? | IBM." Accessed: Apr. 27, 2024. [Online]. Available: https://www.ibm.com/topics/machine-learning

[3] T. M. Mitchell, *Machine Learning*. in McGraw-Hill series in computer science. New York: McGraw-Hill, 1997.

[4] S. Haykin, *Νευρωνικά Δίκτυα και Μηχανική Μάθηση*, 3rd ed. Αθήνα: Εκδόσεις Παπασωτηρίου, 2010.

[5] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. in Springer Series in Statistics. New York, NY: Springer, 2009. doi: 10.1007/978-0-387-84858-7.

[7] S. Russell and P. Norvig, *Τεχνητή Νοημοσύνη, Μια σύγχρονη προσέγγιση*, 4th ed. Αθήνα: Εκδόσεις Κλειδάριθμος, 2021.

[8] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," *Neural Comput.*, vol. 9, pp. 1735–80, Dec. 1997, doi: 10.1162/neco.1997.9.8.1735.

[9] R. Baeza-Yates *et al.*, "Modern Information Retrieval," Jul. 1999.

[10] "What is Sentence Similarity? - Hugging Face." Accessed: May 21, 2024. [Online]. Available: https://huggingface.co/tasks/sentence-similarity

[11] D. Jurafsky and J. Martin, *Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Third Edition draft. Accessed: Jun. 20, 2024. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/

[12] R. Sennrich, B. Haddow, and A. Birch, "Neural Machine Translation of Rare Words with Subword Units," Jun. 10, 2016, *arXiv*: arXiv:1508.07909. doi: 10.48550/arXiv.1508.07909.

[13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," Sep. 06, 2013, *arXiv*: arXiv:1301.3781. doi: 10.48550/arXiv.1301.3781.

[14] R. Schwaber-Cohen, "Vector Similarity Explained | Pinecone." Accessed: Jun. 22, 2024. [Online]. Available: https://www.pinecone.io/learn/vector-similarity/

[15] "Measuring Similarity from Embeddings | Machine Learning," Google for Developers. Accessed: Jun. 22, 2024. [Online]. Available: https://developers.google.com/machine-learning/clustering/similarity/measuring-similarity

[16] A. Vaswani *et al.*, "Attention Is All You Need," Aug. 01, 2023, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.

[17] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," May 19, 2016, *arXiv*: arXiv:1409.0473. doi: 10.48550/arXiv.1409.0473.

[18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Dec. 10, 2015, *arXiv*: arXiv:1512.03385. doi: 10.48550/arXiv.1512.03385.

[20] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," Jul. 21, 2016, *arXiv*: arXiv:1607.06450. doi: 10.48550/arXiv.1607.06450.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” May 24, 2019, *arXiv*: arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805.
[22] Y. Wu *et al.*, “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation,” Oct. 08, 2016, *arXiv*: arXiv:1609.08144. doi: 10.48550/arXiv.1609.08144.
[23] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” Aug. 27, 2019, *arXiv*: arXiv:1908.10084. doi: 10.48550/arXiv.1908.10084.
[24] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, “Supervised Learning of Universal Sentence Representations from Natural Language Inference Data,” Jul. 08, 2018, *arXiv*: arXiv:1705.02364. doi: 10.48550/arXiv.1705.02364.
[25] S. Humeau, K. Shuster, M.-A. Lachaux, and J. Weston, “Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring,” *ArXiv Prepr. ArXiv190501969*, 2019.
[26] N. Thakur, N. Reimers, J. Daxenberger, and I. Gurevych, “Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks,” *ArXiv Prepr. ArXiv201008240*, 2020.
[27] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training”.
[28] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners”.
[29] A. Fan, M. Lewis, and Y. Dauphin, “Hierarchical neural story generation,” *ArXiv Prepr. ArXiv180504833*, 2018.
[30] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, “The Curious Case of Neural Text Degeneration,” Feb. 14, 2020, *arXiv*: arXiv:1904.09751. doi: 10.48550/arXiv.1904.09751.
[31] H. Touvron *et al.*, “LLaMA: Open and Efficient Foundation Language Models,” Feb. 27, 2023, *arXiv*: arXiv:2302.13971. doi: 10.48550/arXiv.2302.13971.
[32] H. Touvron *et al.*, “Llama 2: Open Foundation and Fine-Tuned Chat Models,” Jul. 19, 2023, *arXiv*: arXiv:2307.09288. doi: 10.48550/arXiv.2307.09288.
[33] “Introducing Meta Llama 3: The most capable openly available LLM to date,” Meta AI. Accessed: Jun. 26, 2024. [Online]. Available: https://ai.meta.com/blog/meta-llama-3/
[34] “Introducing Llama 3.1: Our most capable models to date,” Meta AI. Accessed: Aug. 26, 2024. [Online]. Available: https://ai.meta.com/blog/meta-llama-3-1/
[35] A. Dubey *et al.*, “The llama 3 herd of models,” *ArXiv Prepr. ArXiv240721783*, 2024.
[36] J. White *et al.*, “A prompt pattern catalog to enhance prompt engineering with chatgpt,” *ArXiv Prepr. ArXiv230211382*, 2023.
[37] “Llama 3.1 | Model Cards and Prompt formats.” Accessed: Aug. 07, 2024. [Online]. Available: https://llama.meta.com/docs/model-cards-and-prompt-formats/llama3_1/#supported-roles
[38] “OpenAI Platform.” Accessed: Aug. 07, 2024. [Online]. Available: https://platform.openai.com
[39] T. Brown *et al.*, “Language models are few-shot learners,” *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 1877–1901, 2020.
[40] J. Wei *et al.*, “Chain-of-thought prompting elicits reasoning in large language models,” *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 24824–24837, 2022.

[41] A. Silberschatz, H. Korth, and S. Sudarshan, *Συστήματα Βάσεων Δεδομένων*, 7th ed. Αθήνα: Εκδόσεις: Μόσχος Γκιούρδας, 2021.
[42] R. Schwaber-Cohen, “What is a Vector Database & How Does it Work? Use Cases + Examples | Pinecone.” Accessed: Jul. 09, 2024. [Online]. Available: https://www.pinecone.io/learn/vector-database/
[43] “What Is a Vector Database? | IBM.” Accessed: Jul. 09, 2024. [Online]. Available: https://www.ibm.com/topics/vector-database
[44] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, “When Is ‘Nearest Neighbor’ Meaningful?,” in *Database Theory — ICDT’99*, C. Beeri and P. Buneman, Eds., Berlin, Heidelberg: Springer, 1999, pp. 217–235. doi: 10.1007/3-540-49257-7_15.
[45] Y. A. Malkov and D. A. Yashunin, “Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs,” Aug. 14, 2018, *arXiv*: arXiv:1603.09320. doi: 10.48550/arXiv.1603.09320.
[46] S. Hussain, O. Ameri Sianaki, and N. Ababneh, “A Survey on Conversational Agents/Chatbots Classification and Design Techniques,” in *Web, Artificial Intelligence and Network Applications*, L. Barolli, M. Takizawa, F. Xhafa, and T. Enokido, Eds., Cham: Springer International Publishing, 2019, pp. 946–956. doi: 10.1007/978-3-030-15035-8_93.
[47] E. Adamopoulou and L. Moussiades, “An Overview of Chatbot Technology,” in *Artificial Intelligence Applications and Innovations*, I. Maglogiannis, L. Iliadis, and E. Pimenidis, Eds., Cham: Springer International Publishing, 2020, pp. 373–383. doi: 10.1007/978-3-030-49186-4_31.
[48] K. Ramesh, S. Ravishankaran, A. Joshi, and K. Chandrasekaran, “A Survey of Design Techniques for Conversational Agents,” in *Information, Communication and Computing Technology*, S. Kaushik, D. Gupta, L. Kharb, and D. Chahal, Eds., Singapore: Springer, 2017, pp. 336–350. doi: 10.1007/978-981-10-6544-6_31.
[49] J. Weizenbaum, “ELIZA—a computer program for the study of natural language communication between man and machine,” *Commun. ACM*, vol. 9, no. 1, pp. 36–45, Jan. 1966, doi: 10.1145/365153.365168.
[50] “ChatGPT.” Accessed: Jul. 23, 2024. [Online]. Available: https://chatgpt.com
[51] “Gemini,” Gemini. Accessed: Jul. 23, 2024. [Online]. Available: https://gemini.google.com
[52] J. Pennington, R. Socher, and C. Manning, “Glove: Global Vectors for Word Representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162.
[53] A. Xu, Z. Liu, Y. Guo, V. Sinha, and R. Akkiraju, “A new chatbot for customer service on social media,” in *Proceedings of the 2017 CHI conference on human factors in computing systems*, 2017, pp. 3506–3510.
[54] I. V. Serban *et al.*, “A deep reinforcement learning chatbot,” *ArXiv Prepr. ArXiv170902349*, 2017.
[55] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008. [Online]. Available: https://nlp.stanford.edu/IR-book/
[56] L. Athota, V. K. Shukla, N. Pandey, and A. Rana, “Chatbot for healthcare system using artificial intelligence,” in *2020 8th International conference on reliability, infocom technologies and optimization (trends and future directions)(ICRITO)*, IEEE, 2020, pp. 619–622.

[57] R. G. Athreya, A.-C. Ngonga Ngomo, and R. Usbeck, “Enhancing Community Interactions with Data-Driven Chatbots–The DBpedia Chatbot,” in *Companion proceedings of the the web conference 2018*, 2018, pp. 143–146.
[58] *Unstructured-IO/unstructured*. (Aug. 02, 2024). HTML. Unstructured. Accessed: Aug. 02, 2024. [Online]. Available: https://github.com/Unstructured-IO/unstructured
[59] J. Singer-Vine and The pdfplumber contributors, *pdfplumber*. (Apr. 2024). Python. Accessed: Aug. 02, 2024. [Online]. Available: https://github.com/jsvine/pdfplumber
[60] *pymupdf/PyMuPDF*. (Aug. 02, 2024). Python. PyMuPDF. Accessed: Aug. 02, 2024. [Online]. Available: https://github.com/pymupdf/PyMuPDF
[61] P. Lewis *et al.*, “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 9459–9474. Accessed: Jul. 11, 2024. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html
[62] J. Liu, *LlamaIndex*. (Aug. 2022). Python. doi: 10.5281/zenodo.1234.
[63] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, “MTEB: Massive Text Embedding Benchmark,” *ArXiv Prepr. ArXiv221007316*, 2022, doi: 10.48550/ARXIV.2210.07316.
[64] “BAAI/bge-large-en-v1.5 · Hugging Face.” Accessed: Aug. 26, 2024. [Online]. Available: https://huggingface.co/BAAI/bge-large-en-v1.5
[65] *qdrant/qdrant*. (Jul. 15, 2024). Rust. Qdrant. Accessed: Jul. 15, 2024. [Online]. Available: https://github.com/qdrant/qdrant
[66] *UKPLab/sentence-transformers*. (Aug. 26, 2024). Python. Ubiquitous Knowledge Processing Lab. Accessed: Aug. 26, 2024. [Online]. Available: https://github.com/UKPLab/sentence-transformers
[67] S. Ramírez, *FastAPI*. [Online]. Available: https://github.com/fastapi/fastapi
[68] “Vue.js.” Accessed: Aug. 26, 2024. [Online]. Available: https://vuejs.org/
[69] R. Liu *et al.*, “Best practices and lessons learned on synthetic data for language models,” *ArXiv Prepr. ArXiv240407503*, 2024.
[70] L. Zheng *et al.*, “Judging llm-as-a-judge with mt-bench and chatbot arena,” *Adv. Neural Inf. Process. Syst.*, vol. 36, 2024.
[71] E. M. Voorhees and others, “The trec-8 question answering track report.,” in *Trec*, 1999, pp. 77–82.
[72] N. F. Liu *et al.*, “Lost in the Middle: How Language Models Use Long Contexts,” *Trans. Assoc. Comput. Linguist.*, vol. 12, pp. 157–173, Feb. 2024, doi: 10.1162/tacl_a_00638.
[73] “Retrieve & Re-Rank — Sentence Transformers documentation.” Accessed: Aug. 17, 2024. [Online]. Available: https://www.sbert.net/examples/applications/retrieve_rerank/README.html
[74] “cross-encoder/ms-marco-MiniLM-L-6-v2 · Hugging Face.” Accessed: Aug. 17, 2024. [Online]. Available: https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2
[75] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, “Ragas: Automated evaluation of retrieval augmented generation,” *ArXiv Prepr. ArXiv230915217*, 2023.