

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
«Ρομποτικός Σύμβουλος Σπουδών»



Του φοιτητή
Φραγκίσκου Πελαλίδη
Αρ. Μητρώου: 2021127

Επιβλέπων
Ευκλείδης Κεραμόπουλος
Καθηγητής

Ημερομηνία 26/08/2026

Τίτλος Δ.Ε. Ρομποτικός Σύμβουλος Σπουδών
Κωδικός Δ.Ε. 26201
Ονοματεπώνυμο φοιτητή/τών Φραγκίσκος Πελαλίδης
Ονοματεπώνυμο εισηγητή Κεραμόπουλος Ευκλείδης
Ημερομηνία ανάληψης Δ.Ε. 20/03/2026
Ημερομηνία περάτωσης Δ.Ε. ...

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Φραγκίσκου Πελαλίδη που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Στους αγαπημένους μου»

Πρόλογος

Η συγκεκριμένη διπλωματική εργασία έχει ως βασικό αντικείμενο την επικοινωνία ανθρώπου και μηχανής με την αξιοποίηση του ανθρωποειδούς ρομπότ NAO V6 όσον αφορά τις πληροφορίες οι οποίες βρίσκονται στον οδηγό σπουδών τόσο για τα προπτυχιακά πρόγραμμα όσο και για τις μετέπειτα σπουδές, οι οποίες αφορούν τα μεταπτυχιακά και τα διδακτορικά προγράμματα. Αυτή η επικοινωνία ρομπότ και ανθρώπου επιτυγχάνεται με την κατάλληλη διασύνδεση μεταξύ του ανθρωποειδούς NAO V6 και ενός ευφυούς RAG συστήματος σχεδιασμένο κατάλληλα με τις απαραίτητες πληροφορίες για το πανεπιστήμιο. Σε αυτή την διασύνδεση υπάρχει και ένα ενδιάμεσο κομμάτι το οποίο χρησιμοποιείται για να μετατρέπει την ανθρώπινη φωνή σε κείμενο.

Η επιλογή του συγκεκριμένου θέματος έγινε διότι συνδυάζει τομείς όπως η ρομποτική, η τεχνητή νοημοσύνη και η ανάκτηση πληροφορίας. Οι οποίοι είναι κάποιοι τομείς στους οποίους ήθελα να αποκτήσω περισσότερη τριβή και να εμβαθύνω περισσότερο στην μελλοντική μου καριέρα. Παράλληλα, επέλεξα το συγκεκριμένο θέμα, καθώς επιθυμούσα να αξιοποιήσω τον εξοπλισμό του πανεπιστημίου και ειδικότερα το ρομπότ NAO V6, με σκοπό την ενημέρωση και την προβολή του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων προς υποψήφιους φοιτητές.

Σκοπός αυτής είναι η κατάλληλη αναπτυξη ενός εξειδικευμένου RAG συστήματος με τα έγγραφα του πανεπιστημίου, το οποίο θα ανακτά τις σχετικές πληροφορίες ανάλογα με την ερώτηση του χρήστη, θα παράγει σε συνδυασμο με την τεχνητή νοημοσύνη εξανθρωπισμένες απαντήσεις με το ερώτημα. Τέλος θα υπάρχει η κατάλληλη διασύνδεση των συστημάτων με το ρομπότ NAO V6 ώστε ο κάθε χρήστης να μπορεί να αλληλεπιδρά με ένα μοντέλο το οποίο θυμίζει άνθρωπο. Ωστε τελικά η αλληλεπίδραση να είναι όσο φιλική απαιτείται

Περίληψη

Η ανάγκη για αλληλεπίδραση και επικοινωνία ανθρώπου-ρομπότ σχετικά με πληροφορίες της σχολής οδήγησε στην δημιουργία ενός Chatbot και στην διασύνδεση αυτού στο ρομπότ NAO V6. Η διπλωματική εργασία συνδυάζει πλήθος τεχνολογιών και μεθοδολογιών με σκοπό την επίτευξη μίας φυσικής, σχεδόν ανθρώπινης, αλληλεπίδρασης. Τα τελευταία χρόνια η χρήση των κοινωνικών ρομποτ (social robotics) έχει αυξηθεί ραγδαία και σε συνδυασμό με την τεχνητή νοημοσύνη παρουσιάζουν εντυπωσιακά αποτελέσματα όσον αφορά τις ικανότητες τους για επικοινωνία. Η συγκεκριμένη διπλωματική εργασία έχει ως σκοπό να παρουσιάσει το τρόπο με τον οποίο υλοποιήθηκε το τελικό αποτέλεσμα του ρομποτικού συμβούλου σπουδών.

Αρχικά εξετάζεται το θεωρητικό υπόβαθρο όσον αφορά τον συνδυασμό των μεθόδων που χρησιμοποιήθηκαν για την δημιουργία ενός Ρομποτικού Συμβούλου Σπουδών. Όπως τα χαρακτηριστικά του ανθρωποειδούς ρομπότ NAO V6, τα συστήματα RAG (Retrieval-Augmented Generation), τα LLMs, κάποιες γενικές αρχές για τις τεχνολογίες και μεθοδολογίες ανάκτησης πληροφορίας, την αλληλεπίδραση ανθρώπου-ρομπότ, τις μεθόδους που χρησιμοποιούνται για τη μετατροπή της ομιλίας σε κείμενο, καθώς και τις επιπλέον μεθόδους που κρίθηκαν απαραίτητες για τη λειτουργία τέτοιων συστημάτων.

Η Διπλωματική εργασία αναλύει την πρακτική υλοποίηση του Ρομποτικού Συμβούλου Σπουδών. Τον τρόπο με τον οποίο υλοποιήθηκε η συλλογή και η κανονικοποίηση των εγγράφων, τον τρόπο με τον οποίον τα έγγραφα χωρίστηκαν σε μικρότερα κομμάτια. Την μέθοδο η οποία χρησιμοποιήθηκε για να ανακτούνται τα κατάλληλα κομμάτια με βάση την ερώτηση χρησιμοποιώντας embeddings σε συνδυασμό με την αποθήκευση αυτών. Την επέμβαση των LLMs για την καλύτερη εμπειρία του χρήστη. Την σύνδεση όλου αυτού του συστήματος με το NAO V6, αλλά και τον τρόπο αναγνώρισης και σύνθεσης της ομιλίας.

Η διπλωματική εργασία παρουσιάζει ένα σύστημα το οποίο αναγνωρίζει την ομιλία του χρήστη, απαντάει στις ερωτήσεις του χρησιμοποιώντας τις σωστές πηγές και επιτρέπει την φυσική αλληλεπίδραση μεταξύ ανθρώπου-ρομποτ.

Η συνολική συνεισφορά της εργασίας είναι η ανάδειξη των συνδυασμών του RAG με τα χαρακτηριστικά του ρομπότ NAO V6 ώστε να δημιουργηθεί ένα σύστημα το οποίο θα επικοινωνεί και θα αλληλεπιδρά με τον άνθρωπο σχετικά με τις ανάγκες πληροφόρησης για το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων.

«Robotic Academic Advisor»

«Fragkiskos Pelalidis»

Abstract

The need for human-robot interaction and communication regarding information about the school led to the creation of a chatbot and its integration with the NAO V6 robot. This thesis combines various technologies and methodologies with the aim of achieving natural interaction. In recent years, the use of social robots has grown rapidly, and when combined with artificial intelligence, they have demonstrated impressive results in terms of their communication capabilities. The purpose of this thesis is to present how the final version of the robotic academic advisor was implemented.

First, the theoretical background regarding the combination of methods used to create a Robotic Academic Advisor is examined. These include the characteristics of the NAO V6 humanoid robot, RAG (Retrieval-Augmented Generation) systems, LLMs, and some general principles regarding information retrieval technologies and methodologies, as well as human-robot interaction.

This thesis analyzes the practical implementation of the Robotic Academic Advisor. It examines how the collection and normalization of documents were carried out, and how the documents were divided into smaller segments. It describes the method used to retrieve the appropriate segments based on the query using embeddings, as well as how these segments were stored. It also covers the integration of Large Language Models (LLMs) to enhance the user experience. The integration of this entire system with the NAO V6, as well as the method for speech recognition and synthesis.

This thesis presents a system that recognizes the user's speech, answers their questions using the appropriate sources, and enables natural human-robot interaction.

The overall contribution of this thesis is to highlight the integration of RAG with the capabilities of the NAO V6 robot to create a system that will communicate and interact with humans regarding information needs for the Department of Computer Engineering and Systems.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την οικογένεια μου και τους κοντινούς μου ανθρώπους

Περιεχόμενα

Πρόλογος	V
Περίληψη	VI
Abstract	VII
Ευχαριστίες	VIII
Περιεχόμενα	IX
Κατάλογος Σχημάτων	XI
Κατάλογος Πινάκων	XI
Συντομογραφίες	XII
Κεφάλαιο 1ο: Εισαγωγή	1
1.1 Εισαγωγή	1
1.2 Ιστορική Αναδρομή	1
1.2.1 Η γέννηση της ρομποτικής	1
1.2.2 Η εξέλιξη των Social Robotics	2
1.3 Βασικές αρχές Human Robot Interaction	2
1.4 Συνεισφορά	4
1.5 Οργάνωση της εργασίας	4
Κεφάλαιο 2ο: Θεωρητικό Υπόβαθρο	7
2.1 Εισαγωγή	7
2.2 Τεχνητή Νοημοσύνη	7
2.3 Μεγάλα Γλωσσικά Μοντέλα	8
2.4 Retrieval-Augmented Generation	8
2.5 Chunking	10
2.6 Embeddings	10
2.7 Vector DB- Semantic Search	11
2.8 Speech Recognition	11
2.9 Χαρακτηριστικά NAO V6.	12
2.10 Επίλογος	13
Κεφάλαιο 3ο: Υλοποίηση Retrieval Augmented Generation (RAG) Συστήματος	15
3.1 Εισαγωγή	15
3.2 RAG VS Fine Tuning : Επιλογή αρχιτεκτονικής RAG	16
3.3 Επιλογή Αρχιτεκτονικής Συστήματος Retrieval-Augmented Generation	19
3.3.1 Εισαγωγή	19
3.3.2 Χαρακτηριστικά Υλοποίησης Αρχιτεκτονικών RAG	19
3.3.3 Χαρακτηριστικά Υλοποίησης Αρχιτεκτονικών RAG	21
3.4 Επίλογος	24
Κεφάλαιο 4ο: Υλοποίηση Επιμέρους συστημάτων του Retrieval Augmented Generation (RAG) Συστήματος	27
4.1 Δημιουργία Βάση Γνώσης	27
4.1.1 Μέρος Πρώτο: Μη δομημένα δεδομένα	27
4.1.2 Μέρος Δεύτερο: Δομημένα δεδομένα	29
4.1.3 Μέρος Τρίτο : Συχνές ερωτήσεις	30

4.2 Στρατηγική Chunking	31
4.3 Embedding Model	35
4.4 Hybrid Retrieval	38
4.4.1 Retrieval Pipeline	38
4.4.2 Αρχικοποίηση Retrieval Engine	38
4.5 Context Construction και LLM Generation	40
4.6 Επίλογος	42
Κεφάλαιο 5ο: Σύστημα αναγνώρισης Φωνής	43
5.1 Αρχιτεκτονική και επιλογή μοντέλων αναγνώρισης φωνής	43
5.2 Διαχείριση εισόδου ήχου και μηχανισμός Voice Activity Detection	44
5.3 Μηχανισμός Voice Activity Detection (VAD)	46
5.4 Προεπεξεργασία και βελτίωση αποτελέσματος αναγνώρισης φωνής 49ερωτήματος για το RAG παρουσιάζεται στο Σχήμα 5.3.	49
5.5 Επίλογος	50
Κεφάλαιο 6ο: Τρόποι διασύνδεσης του NAO με το υπόλοιπο σύστημα	53
6.1 Εισαγωγή	53
6.2 Διασύνδεση NAO V6 με Speech to Text και RAG	54
6.2.1 Καταγραφή και μεταφορά Ήχου	54
6.2.2 One Shot Επικοινωνία NAO V6 προς Υπολογιστή.	56
6.3 Συνολική Αρχιτεκτονική Διασύνδεσης των Συστημάτων	57
6.4 Επίλογος	58
Κεφάλαιο 7ο: Συμπεράσματα και αποτίμηση της Υλοποίησης	61
7.1 Στόχοι και υλοποίηση	61
7.2 Συνολική Λειτουργία Συστήματος	62
7.3 Πλεονεκτήματα και περιορισμοί της υλοποίησης	63
7.3.1 Πλεονεκτήματα	63
7.3.2 Περιορισμοί της υλοποίησης	63
7.4 Έλεγχος γραπτής και φωνητικών αιτημάτων στο σύστημα	64
7.4.1 Σκοπός Αξιολόγησης	64
7.4.2 Μεθοδολογία Αξιολόγησης	64
7.4.3 Συνολικά αποτελέσματα	65
7.4.4 Διαγράμματα	66
7.4.5 Ερμηνεία αποτελεσμάτων και περιορισμοί	67
7.5 Συμπεράσματα	68
7.6 Επίλογος	69
Κεφάλαιο 8ο: Μελλοντικές επεκτάσεις	71
8.1 Εισαγωγή	71
8.2 Δυναμικός Εμπλουτισμός Βάσης Γνώσης	71
8.3 Μετάβαση σε υποδομή Server	71
8.4 Επέκταση κινητικής συμπεριφοράς	71
8.5 Εκτενότερη Πειραματική Αξιολόγηση	71
8.6 Επέκταση κινητικής συμπεριφοράς	71
ΒΙΒΛΙΟΓΡΑΦΙΑ	75

Κατάλογος Σχημάτων

Σχήμα 3.1: Συνολική αρχιτεκτονική του υλοποιημένου RAG συστήματος	16
Σχήμα 4.1: Διαδικασία Query Routing και ανάκτησης δεδομένων	29
Σχήμα 5.1: Ροή επεξεργασίας φωνητικού ερωτήματος	35
Σχήμα 5.2: Διαδικασία ανίχνευσης ομιλίας μέσω VAD	37
Σχήμα 5.3: Ροή επεξεργασίας ομιλίας και δημιουργίας ερωτήματος RAG	38
Σχήμα 5.4: Αρχιτεκτονική επεξεργασίας φωνητικής εισόδου του Ρομποτικού Συμβούλου Σπουδών	39
Σχήμα 6.1: Ροή αλληλεπίδρασης χρήστη με τον Ρομποτικό Σύμβουλο Σπουδών	41
Σχήμα 6.2: Συνολική Αρχιτεκτονική Διασύνδεσης των Συστημάτων	44
Σχήμα 7.1: Κατανομή αξιολόγησης απαντήσεων ανά τρόπο εισόδου	52
Σχήμα 7.2: Μέση επίδοση του RAG συστήματος	53

Κατάλογος Πινάκων

Πίνακας 3.1: Πίνακας Σύγκρισης Fine Tuning και RAG.	13
Πίνακας 3.2: Πίνακας Σύγκριση προσεγγίσεων διαχείρισης γνώσης σε μοντέλα τεχνητής νοημοσύνης	14
Πίνακας 4.1: Δομή του πίνακα Courses που χρησιμοποιείται για την αποθήκευση πληροφοριών μαθημάτων.	21
Πίνακας 4.2: Δομή του πίνακα Staff που χρησιμοποιείται για την αποθήκευση πληροφοριών μελών ΔΕΠ.	22
Πίνακας 4.3: Περιγραφή των μεταδεδομένων που αποθηκεύονται για κάθε chunk	24
Πίνακας 4.4: Δομή αποθήκευσης embeddings και μεταδεδομένων	27
Πίνακας 4.5: Αρχεία που χρησιμοποιούνται για την αποθήκευση και ανάκτηση της γνώσης	29
Πίνακας 5.1: Σύγκριση των μοντέλων αναγνώρισης φωνής που χρησιμοποιούνται στην υλοποίηση	33
Πίνακας 5.2: Παράμετροι εισόδου και τμηματοποίησης του ηχητικού σήματος.	36
Πίνακας 5.3: Ενδεικτικά επίπεδα ηχητικού σήματος σε κλίμακα dBFS	36
Πίνακας 5.4: Παράμετροι του μηχανισμού ανίχνευσης και τμηματοποίησης ομιλίας	38
Πίνακας 5.5: Παραδείγματα διόρθωσης οντοτήτων μετά το STT	38
Πίνακας 6.1: Παράμετροι της εντολής arecord για την καταγραφή και μεταφορά ήχου από τον NAO V6	42
Πίνακας 6.2: Σύγκριση των τρόπων επικοινωνίας NAO V6–υπολογιστή	43
Πίνακας 7.1: Συνοπτική παρουσίαση στόχων και αντίστοιχων υλοποιήσεων	47
Πίνακας 7.2: Κλίμακα αξιολόγησης απαντήσεων	51
Πίνακας 7.3: Αποτελέσματα αξιολόγησης γραπτής και φωνητικής εισόδου	51
Πίνακας 7.4: Πιθανές αιτίες μερικώς ή μη ικανοποιητικών απαντήσεων	54
Πίνακας 8.1: Προτεινόμενο πλαίσιο μελλοντικής αξιολόγησης	58
Πίνακας 8.2: Προτεραιοποίηση προτεινόμενων μελλοντικών επεκτάσεων	59

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΙΠΙΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
RAG	Retrieval Augmented Generation
LLM / LLMs	Large Language Model(s)
NLP	Natural Language Processing
HRI	Human-Robot Interaction
VAD	Voice Activity Detection
STT	Speech-to-Text
TTS	Text-to-Speech
FAQ	Frequently Asked Questions
RRF	Reciprocal Rank Fusion
FT	Fine-Tuning

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Ο άνθρωπος είναι ένα κοινωνικό ον το οποίο προσπαθεί συνεχώς να βελτιώσει την ποιότητα ζωής του, να ανακαλύψει νέους τρόπους μάθησης και επικοινωνίας. Ο τομέας των Social Robotics είναι ένας τομέας ο οποίος ασχολείται και με την συγκεκριμένη θεματολογία μέσω όχι της επικοινωνίας μεταξύ ανθρώπων αλλά μέσω της επικοινωνίας ανθρώπου και ρομπότ. Τα σύγχρονα κοινωνικά ρομπότ λειτουργούν ως πράκτορες με δυνατότητα αντίληψης, αξιοποιώντας τους ποικίλους αισθητήρες τους, την υπολογιστική τους ισχύ, μοντέλα τεχνητής νοημοσύνης και αλγορίθμους επεξεργασίας. Με αυτά τα εφόδια καταφέρνουν να επεξεργάζονται δεδομένα, να παράγουν πληροφορία προς τον χρήστη και εν τέλη να επιτυγχάνουν επικοινωνία μεταξύ ανθρώπου και μηχανής, ακολουθώντας τον κύκλο Αντίληψης, Επεξεργασίας και Δράσης (perception–reasoning–action loop) [1].

Τα Social Robots ανήκουν στην κατηγορία των ρομπότ τα οποία δημιουργήθηκαν με κύριο σκοπό την αλληλεπίδραση με τους ανθρώπους χρησιμοποιώντας σαν μέσο επικοινωνίας απευθείας την φυσική γλώσσα [2]. Ταυτόχρονα μπορεί να περιέχουν και ένα πλήθος κινητικών εκφράσεων οι οποίες με την σειρά τους βοηθούν στην επίτευξη μίας πιο αποδοτικής και εξανθρωπισμένης επικοινωνίας. Είναι τα πλέον κατάλληλα ρομπότ για να χρησιμοποιηθούν σε πληθος δραστηριοτήτων διαδραστικής εκπαίδευσης, καθώς είναι ιδιαίτερα ικανά στην διαδραστική μάθηση. Όλα τα παραπάνω χαρακτηριστικά έχουν οδηγήσει στην ραγδαία αύξηση χρήσης τους [2] τόσο στον χώρο της εκπαίδευσης, όσο και σε χώρους υγείας και γενικότερα δημόσιων υπηρεσιών που αφορούν αλληλεπίδραση με ανθρώπους.

Στον χώρο του πανεπιστημίου υπάρχει μεγάλη ανάγκη για αλληλεπίδραση, επικοινωνία και για καλύτερη υποστήριξη και εξυπηρέτηση φοιτητών. Ένα παράδειγμα είναι η γραμματειακή υποστήριξη που αυξάνεται κατά την διάρκεια των περιόδων εγγραφής των φοιτητών [3] αλλά και κατά την έλευση νέων φοιτητών στο πανεπιστήμιο [3]. Ένας τρόπος κάλυψης αυτών των αναγκών είναι μέσω των Social Robotics και πιο συγκεκριμένα ενός ρομπότ το οποίο θα εξειδικεύεται πάνω στις βάσεις γνώσης του πανεπιστημίου με τελικό σκοπό να παράγει κατάλληλες απαντήσεις με το εκάστοτε αίτημα του χρήστη. Καθώς τα κοινωνικά ρομπότ λειτουργούν σαν δίαυλος επικοινωνίας μεταξύ των φοιτητών και των πληροφοριών του πανεπιστημίου.

1.2 Ιστορική Αναδρομή

1.2.1 Η γέννηση της ρομποτικής

Τα ρομπότ αποτελούσαν ανέκαθεν μεγάλο μέρος της κουλτούρας των τεχνών συγκεκριμένα το 1920 ο Κάρελ Τσάπεκ εισήγαγε τον όρο ρομπότ στο θεατρικό του έργο R.U.R. Όπου τότε μηχανές λειτουργούν σαν υπηρέτες. Η έννοια ακόμα ρομπότ δεν έχει περάσει στην υλοποίηση της τεχνολογίας και παρέμεινε στον κόσμο των τεχνών του θεάτρου. Λίγο πιο μετά ο Asimov το 1942 στο Runaround [4] αναφέρει τους τρεις νόμους της ρομποτικής χωρίς όμως ακόμα να γίνεται κάποια ξεκάθαρη τεχνολογική αναφορά. Στη συνέχεια υπάρχει το 1973 το πρώτο ανδροειδές ρομπότ το ρομπότ WABOT-1 [5] το οποίο βέβαια είναι ένα ρομπότ το οποίο δεν συγκαταλέγεται στα κοινωνικά ρομπότ. Επειτα εξαιτίας αυτής της επιρροής συναντάμε διάφορα ρομπότ και στην κουλτούρα του κινηματογράφου, χωρίς κάποιο τεχνολογικό υπόβαθρο, ένα από τα πρώτα ανδροειδή εμφανίστηκε το 1977 και συγκεκριμένα στην ταινία Star Wars : Episode IV – A New Hope . Μέχρι εκείνο το σημείο ο όρος των Social Robots υπήρχε αλλά χρησιμοποιούνταν σπάνια καθώς δεν είχε χτιστεί ακόμα το

κατάλληλο τεχνολογικό περιβάλλον χρήσης, η σπάνια χρήση του όρου συνεχίστηκε για κάποια ακόμα χρόνια μέχρι τα τέλη του 1990 όπου η Cynthia Breazeal να αναπτύξει ουσιαστικά ένα από τα πρώτα Social Robot το ρομπότ Kismet [6]. Το ρομπότ Kismet αποτελεί ουσιαστικά την γέννηση των Social Robotics. Θεωρείται ένα από τα πρώτα ρομπότ το οποίο αντιλαμβάνεται πρόσωπα, εκφράσεις, παράγει το ίδιο εκφράσεις και είναι σχεδιασμένο για αλληλεπίδραση μεταξύ ανθρώπων και ρομπότ. Η πρωτοπορία και η επιτυχία του ρομπότ Kismet οδήγησε και συνέβαλε σημαντικά στην εξέλιξη αυτού του τομέα και στην ανάπτυξη νέων κοινωνικών ρομπότ, όπως για παράδειγμα τα ASIMO, PEPPER και NAO.

Πλέον τα σύγχρονα ρομπότ με την τεράστια βοήθεια της τεχνολογικής εξέλιξης σε τομείς όπως η τεχνητή νοημοσύνη, στην φωνητική αναγνώριση και στα μεγάλα γλωσσικά μοντέλα έχουν επιτύχει εξαιρετικές δυνατότητες επικοινωνίας και αλληλεπίδρασης συγκριτικά με τις πρώτες υλοποιήσεις πριν από περίπου 30 χρόνια.

1.2.2 Η εξέλιξη των Social Robotics

Το στοιχείο που ξεχωρίζει τα ανθρωποειδή ρομπότ από τα κοινωνικά είναι το γεγονός ότι τα δεύτερα έχουν σχεδιαστεί με κατάλληλο τρόπο με κύριο σκοπό την αλληλεπίδραση ανθρώπων και ρομπότ. Χωρίς να σημαίνει ότι ένα ρομπότ δεν μπορεί να συνυπάρχει και στις δύο κατηγορίες ταυτόχρονα. Ένα κοινωνικό ρομπότ πρέπει να είναι σε θέση να αντιλαμβάνεται και να επικοινωνεί συναισθήματα, να μπορεί να χρησιμοποιήσει διάλογο, να αναγνωρίζει άλλους χρήστες, να μαθαίνει νέα πράγματα [7].

- Kismet (1997-2000) :

Θεωρείται από πολλούς το ρομπότ ορόσημο για τα Social Robotics καθώς σχεδιάστηκε με γνώμονα την κοινωνική αλληλεπίδραση. Περιείχε εκφράσεις προσώπου, αναγνώριση προσώπου, δείχνει συναισθήματα και δημιουργήθηκε με απώτερο σκοπό την επικοινωνία ρομπότ – ανθρώπου [6].

- ASIMO (2000) :

Με μικρή χρονική διαφορά ακολούθησε το ρομπότ ASIMO, το συγκεκριμένο ήταν ένα ρομπότ το οποίο είναι κυρίως ανθρωποειδές ρομπότ το οποίο όμως παρέχει και κάποιες λειτουργίες που αφορούν αλληλεπίδραση ρομπότ με άνθρωπο . Καθώς το συγκεκριμένο συνδύαζε την ανθρωπόμορφη εμφάνιση, την αναγνώριση φωνής και αντικειμένων[8].

- NAO(2006):

Οι προθέσεις αυτού του ρομπότ πλέον είναι ακόμα πιο ξεκάθαρες είναι ένα κοινωνικό ρομπότ το οποίο φτιάχτηκε από το μηδέν για να είναι κατάλληλο για εκπαίδευση, αλληλεπίδραση ανθρώπου – ρομπότ και γενικότερα για κοινωνική επικοινωνία [9]. Πέρα από το πρώτο του 2006 ακολούθησαν μετέπειτα και άλλες εκδόσεις που βελτιώνουν κάθε φορά την προηγούμενη παραμένοντας στην ίδια φιλοσοφία και αρχιτεκτονική. Στην συγκεκριμένη Διπλωματική Εργασία έχει χρησιμοποιηθεί η έκδοση NAO V6 του 2018.

1.3 Βασικές αρχές Human Robot Interaction

Η αλληλεπίδραση μεταξύ ανθρώπων και ρομπότ αποτελεί έναν ιδιαίτερο επιστημονικό τομέα, ο οποίος ασχολείται με την οργάνωση, την αξιολόγηση και, γενικότερα, την κατανόηση των ρομποτικών συσκευών που έχουν δημιουργηθεί για να χρησιμοποιούνται από ανθρώπους ή να συνυπάρχουν μαζί τους [10]. Η καταλληλότητα ενός ρομπότ για αλληλεπίδραση με ανθρώπους εξαρτάται, μεταξύ άλλων, από τον βαθμό αυτονομίας των λειτουργιών του. Ένα ρομπότ θεωρείται αυτόνομο όταν μπορεί να

δέχεται εξωτερικά ερεθίσματα, να τα επεξεργάζεται και, με βάση αυτά, να σχεδιάζει και να εκτελεί τις επόμενες ενέργειές του [10], χωρίς άμεση ανθρώπινη παρέμβαση στο ρομποτικό σύστημα. Η συγκεκριμένη διπλωματική εργασία αξιοποιεί την αρχή της αυτονομίας, με στόχο την επίτευξη αποτελεσματικής αλληλεπίδρασης μεταξύ ανθρώπου και ρομπότ. Συγκεκριμένα, το σύστημα ανιχνεύει, μέσω των αισθητήρων του, την παρουσία ανθρώπου στον χώρο και αντιλαμβάνεται τη φωνή του χρήστη. Στη συνέχεια, επεξεργάζεται την είσοδο, προσπαθεί να κατανοήσει το αίτημα και, ανάλογα με το περιεχόμενό του, αναλαμβάνει την κατάλληλη δράση, χρησιμοποιώντας τη φωνή του NAO V6 για την παρουσίαση του αποτελέσματος. Ένας τρόπος αξιολόγησης της αυτονομίας ενός ρομπότ είναι το χρονικό διάστημα κατά το οποίο μπορεί να λειτουργεί αποτελεσματικά για την επίτευξη του εκάστοτε στόχου του, χωρίς να μειώνεται η απόδοσή του και χωρίς να δυσχεραίνεται η αλληλεπίδραση με τον άνθρωπο. Το χρονικό αυτό διάστημα συνδέεται με την έννοια του Neglect Time (NT) [11]. Η παρούσα διπλωματική εργασία έχει σχεδιαστεί έτσι ώστε οι λειτουργίες του ρομπότ να παραμένουν σταθερές κατά τη διάρκεια της λειτουργίας του και να διατηρείται η συνολική εμπειρία του χρήστη. Ένας ακόμη δείκτης που σχετίζεται με την απόδοση ενός ρομπότ είναι ο χρόνος που απαιτείται για να φτάσει στη μέγιστη δυνατή απόδοσή του, ο οποίος αναφέρεται ως Interaction Time (IT) [12]. Η παρούσα υλοποίηση έχει σχεδιαστεί ώστε να παρέχει σταθερή απόδοση από την αρχή έως το τέλος της λειτουργίας της, χωρίς η απόδοσή της να επηρεάζεται από παράγοντες που σχετίζονται με τον χρόνο. Παράλληλα, θεμελιώδης παράγοντας της αλληλεπίδρασης ανθρώπου-ρομπότ είναι η αντίληψη (Perception), η οποία σχετίζεται με τον τρόπο με τον οποίο ο άνθρωπος αντιλαμβάνεται και αξιολογεί το ρομπότ. Η ανθρωπόμορφη εμφάνιση του NAO V6, σε συνδυασμό με τον σχεδιασμό των επιμέρους λειτουργιών της παρούσας διπλωματικής εργασίας, μπορεί να συμβάλλει στην αύξηση της κοινωνικής αποδοχής του. Παράλληλα, η διατήρηση ενός φιλικού και υποστηρικτικού ύφους μπορεί να βελτιώσει συνολικά την αλληλεπίδραση ανθρώπου-ρομπότ. Οι παράγοντες που επηρεάζουν αυτή την αρχή της HRI δεν περιορίζονται στα τεχνικά χαρακτηριστικά του NAO V6 [13], αλλά περιλαμβάνουν και τις συμπεριφορικές λειτουργίες που προστέθηκαν στο πλαίσιο της παρούσας υλοποίησης. Ένα επιπλέον στοιχείο που συμβάλλει στη βελτίωση της εμπειρίας και της επικοινωνίας μεταξύ ανθρώπων και ρομπότ είναι η φυσική επικοινωνία (Natural Communication). Η προσέγγιση αυτή αφορά μια πιο φυσική μορφή επικοινωνίας ανθρώπου-ρομπότ, η οποία προσεγγίζει, όσο είναι δυνατόν, την επικοινωνία μεταξύ ανθρώπων. Επομένως, τα ρομπότ δεν θα πρέπει να αντιδρούν μόνο σε συγκεκριμένες, προκαθορισμένες εντολές, αλλά να μπορούν να προσαρμόζουν τη συμπεριφορά τους ανάλογα με την αλληλεπίδραση που πραγματοποιείται με τον άνθρωπο [14]. Στην παρούσα διπλωματική εργασία, αυτό επιτυγχάνεται, για παράδειγμα, μέσω της δυνατότητας του NAO V6 να προσανατολίζει το βλέμμα του προς την κατεύθυνση από την οποία αντιλαμβάνεται φωνή ή παρουσία προσώπου, αλλά και μέσω της προσαρμογής των απαντήσεων που παράγονται με τη χρήση τεχνητής νοημοσύνης. Ιδιαίτερο χαρακτηριστικό της συγκεκριμένης υλοποίησης είναι η προσαρμοστικότητα, η οποία αφορά τόσο τον σχεδιασμό του ρομπότ όσο και τις λειτουργίες που προστέθηκαν στο πλαίσιο της διπλωματικής εργασίας. Χαρακτηριστικό παράδειγμα αποτελεί η κίνηση του κεφαλιού του ρομπότ, το οποίο μπορεί να στρέφεται προς τον ομιλητή, καθώς και οι κινήσεις των χεριών, όπως ο χαιρετισμός, ανάλογα με τις ανάγκες της επικοινωνίας. Επιπρόσθετα, η διπλωματική εργασία έχει σχεδιαστεί έτσι ώστε και η ομιλία του ρομπότ να προσαρμόζεται στο περιεχόμενο και στο ύφος της ερώτησης. Το σύστημα RAG αναζητά και ανακτά τις κατάλληλες πληροφορίες, ώστε να εντοπιστεί η απάντηση που σχετίζεται με το ερώτημα του χρήστη. Στη συνέχεια, η τεχνητή νοημοσύνη συμβάλλει στην προσαρμογή του ύφους της απάντησης. Η προσαρμοστικότητα (Adaptability) αποτελεί βασικό στοιχείο της Human-Robot Interaction, καθώς καθιστά την αλληλεπίδραση πιο φυσική και εξατομικευμένη [15]. Όπως οι άνθρωποι προσαρμόζουν τις κινήσεις του σώματός τους ανάλογα με τη θέση του συνομιλητή και τον

τρόπο διατύπωσης της απάντησής τους, έτσι και το ρομπότ της παρούσας διπλωματικής εργασίας προσαρμόζει τη συμπεριφορά και την επικοινωνία του ανάλογα με την εκάστοτε περίπτωση. Οι άνθρωποι και οι μηχανές μπορούν να αποκομίσουν σημαντικά οφέλη από την αλληλεπίδραση και την ανταλλαγή μηνυμάτων μεταξύ τους. Η επίτευξη μιας καλής εμπειρίας αλληλεπίδρασης δεν εξαρτάται μόνο από τις τεχνικές ικανότητες του ρομπότ. Υπάρχουν πολλοί παράγοντες που επηρεάζουν τη συνολική αλληλεπίδραση, όπως το φύλο που αναπαριστά το ρομπότ και το αν χρησιμοποιεί ανδρική ή γυναικεία φωνή. Οι παράγοντες αυτοί επηρεάζουν την ποιότητα της εμπειρίας, αλλά και τον τρόπο με τον οποίο αντιλαμβανόμαστε τα κοινωνικά φύλα [16]. Το γεγονός αυτό αποτελεί χαρακτηριστικό παράδειγμα της αμφίδρομης επικοινωνίας μεταξύ ανθρώπου και ρομπότ, καθώς η φωνή του ρομπότ επηρεάζει τον τρόπο με τον οποίο βιώνεται η αλληλεπίδραση, ενώ παράλληλα μπορεί να ενισχύσει ή να αμφισβητήσει ήδη υπάρχοντα στερεότυπα και αντιλήψεις [16].

1.4 Συνεισφορά

Η διπλωματική εργασία σαν γενικό σκοπό έχει την ανάλυση της γνώσης, της βιβλιογραφίας, την δημιουργία και εκτέλεση νέων ιδεών από τα υπάρχοντα δεδομένα με τρόπο ολοκληρωμένο. Ταυτόχρονα στοχεύει και στην πρακτική εφαρμογή αυτών των μεθόδων. Η συγκεκριμένη διπλωματική προσφέρει έναν σημαντικό συνδυασμό των παραπάνω στόχων. Αρχικά πρωταρχικός στόχος της διπλωματικής είναι μέσω της συνένωσης σύγχρονων τεχνολογιών οι οποίες συνδυάζουν τομείς της ρομποτικής, ανάκτηση πληροφορίας και τεχνητής νοημοσύνης με σκοπό να δημιουργηθεί ένα οργανωμένο σύστημα το οποίο θα φέρει εις πέρας την αποστολή για αλληλεπίδραση με ήδη εγγεγραμμένους φοιτητές του τμήματος ή άλλων τμημάτων ή με εν δυνάμει νέους φοιτητές του τμήματος, ή τέλος και με ακόμα έμμεσα ενδιαφερόμενους για το τμήμα όπως οι γονείς των παραπάνω κατηγοριών. Η αλληλεπίδραση επιπλέον συνεισφέρει και στην απόκτηση γνώσης για θέματα όπου αφορούν το πανεπιστήμιο και συγκεκριμένα το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Για την καλύτερη απόδοση στην επικοινωνία, στην σύνθεση και δημιουργία κατάλληλων απαντήσεων μέσω της ανακτημένης πληροφορίας έγινε αξιολόγηση σε βάθος τεχνολογιών και μεθόδων οι οποίες προυπάρχουν, συνδυαστικά με κατάλληλη εκτενή μελέτη αυτών. Ουσιαστικά η συνεισφορά χωρίζεται σε δύο κομμάτια. Στην θεωρητική μελέτη και αξιολόγηση χρησιμών εργασιών, δημοσιεύσεων και στην κατάλληλη σχεδίαση του συστήματος για την επίτευξη της καλύτερης αλληλεπίδρασης μεταξύ ανθρώπου και ρομπότ. Τα δύο παραπάνω στοιχεία τελικά συνδυάστηκαν ώστε να οδηγηθεί η διπλωματική εργασία στην κατάλληλη πρακτική εφαρμογή όπου τελικά υπάρχει η πραγματική αλληλεπίδραση ανθρώπου και ρομπότ, μέσω φωνητικής επικοινωνίας πάνω σε πληροφορίες σχετικές με το πανεπιστήμιο και το τμήμα ενδιαφέροντός.

1.5 Οργάνωση της εργασίας

Η παρούσα διπλωματική εργασία έχει σχεδιαστεί και οργανωθεί σε οκτώ κεφάλαια με σκοπό να αναλυθούν ενδελεχώς το θεωρητικό υπόβαθρο της εργασίας, των ρομπότ, του RAG συστήματος και των επιπλέον τεχνολογιών και μεθόδων που συνέβαλαν στην συγκεκριμένη διπλωματική. Να αναλυθεί επιπλέον ο τρόπος υλοποίησης της διπλωματικής εργασίας αλλά και τα τελικά αποτελέσματα και συμπεράσματα αυτής και η τελική αξιολόγηση της.

Στο 1ο κεφάλαιο παρουσιάζεται ο επιστημονικός τομέας των κοινωνικών ρομπότ, μία ιστορική αναδρομή σε αυτόν και διαφορά χαρακτηριστικά του. Στην συνέχεια του κεφαλαίου γίνεται αναφορά στην αλληλεπίδραση μεταξύ ρομπότ και ανθρώπων και στον τρόπο με τον οποίο η διπλωματική εργασία καταφέρνει να διατηρήσει τις βασικές αρχές αυτού του αντικείμενου. Κλείνοντας σε αυτό το κεφάλαιο αναλύεται και η συνεισφορά της παρούσας διπλωματικής εργασίας.

Στο 2ο κεφάλαιο γίνεται μια περισσότερο σε βάθος και μία πιο στοχευμένη περιγραφή στις τεχνολογίες που συμβάλλουν και συμμετέχουν ενεργά στην υλοποίηση της διπλωματικής εργασίας, όπως είναι για παράδειγμα, η συμβολή της τεχνητής νοημοσύνης, των μεγάλων γλωσσικών μοντέλων, των RAG συστημάτων, των embeddings αλλά και του chunking που χρησιμοποιήθηκε. Αναλύθηκε επίσης με ποιόν τρόπο αποθηκεύεται αυτή η πληροφορία, πως ανακτάται. Πως μέσω της ανάπτυξης και της σύνδεσης με μοντέλα LLM και τεχνητής νοημοσύνης συντίθεται η τελική απάντηση. Και τελικά αναπτύσσονται όλα τα χαρακτηριστικά του ρομπότ τα οποία φαίνονται χρήσιμα για την υλοποίηση ενός Ρομποτικού Συμβούλου Σπουδών. Ουσιαστικά αυτό το κεφάλαιο χρησιμοποιείται σαν γέφυρα μεταξύ του θεωρητικού υπόβαθρου και στην μεθοδολογία υλοποίησης.

Στο 3ο κεφάλαιο γίνεται αναφορά στους λόγους επιλογής του RAG έναντι τεχνικών όπως το Fine Tuning ενώ παράλληλα γίνεται αναφορά και σε μία παρουσίαση της συνολικής δομής και αρχιτεκτονικής του συστήματος RAG το οποίο τελικά υλοποιήθηκε στην διπλωματική εργασία.

Στο 4ο κεφάλαιο επεκτεινόμαστε στον τρόπο υλοποίησης του RAG συστήματος, στην γλώσσα προγραμματισμού που χρησιμοποιήθηκε. Αναλύουμε επίσης σε βάθος κάθε τμήμα τέτοιων συστημάτων. Καταλήγουμε τελικά στον τρόπο με τον οποίο δημιουργήθηκε η τελική μορφή του RAG συστήματος η οποία επιλέχθηκε για την παρούσα πρακτική εφαρμογή της διπλωματικής εργασίας.

Στο 5ο κεφάλαιο αναλύεται σε βάθος η τελική μορφή της φωνητικής αναγνώρισης η οποία απαιτείται για την παρούσα διπλωματική εργασία. Μοντέλα φωνητικής αναγνώρισης τα οποία προτιμήθηκαν έναντι άλλων και για ποιους λόγους συμβαίνει αυτό. Ενώ παράλληλα αναφέρονται οι διάφορες τεχνικές οι οποίες χρησιμοποιήθηκαν ώστε η μετατροπή αυτή να είναι κατάλληλη για τις ανάγκες αλληλεπίδρασης ανθρώπου και ρομπότ.

Στο 6ο κεφάλαιο γίνεται αναφορά στον τρόπο με τον οποίο διασυνδέονται τα παραπάνω συστήματα του RAG και της φωνητικής αναγνώρισης μεταξύ τους αλλά κυρίως επάνω στο ρομπότ NAO V6. Παράλληλα αναλύονται και διάφορες τεχνικές και διάφορα χαρακτηριστικά του ρομπότ τα οποία αξιοποιούνται για την επίτευξη καλύτερης αλληλεπίδρασης μεταξύ ρομπότ και ανθρώπου, τέτοιες όπως η δυνατότητα κινήσεων, η χρήση των αισθητήρων του ρομπότ αλλά και η χρήση της φυσικής του παρουσίας και σωματικής σύνθεσης για την παραγωγή κινήσεων.

Στο 7ο κεφάλαιο γίνεται μία συνολική αναφορά στα αποτελέσματα της εργασίας, στο τελικό προϊόν το οποίο προκύπτει από την συνεργασία όλων των προηγούμενων ιδεών, ενώ σχολιάζεται και το τι έχει καταφέρει να πετύχει και στον τρόπο αξιοποίησης κάθε επιπέδου. Τέλος ακολουθεί και μία πειραματική αξιολόγηση αυτής.

Στο 8ο και τελευταίο κεφάλαιο της διπλωματικής εργασίας περιγράφονται διάφορες μελλοντικές επεκτάσεις και βελτιώσεις του συστήματος. Ενώ καταλήγουμε και στον τρόπο με τον οποίο ολοκληρώνεται συνολικά η διπλωματική εργασία σαν ένας επίλογος αυτής.

Κεφάλαιο 2ο: Θεωρητικό Υπόβαθρο

2.1 Εισαγωγή

Στο προηγούμενο κεφάλαιο έγινε μία εισαγωγή στα κοινωνικά ρομπότ και στον τρόπο με τον οποίο αυτά αλληλεπιδρούν με τους ανθρώπους. Στο συγκεκριμένο κεφάλαιο θα γίνει σε βάθος ανάλυση στις τεχνολογίες που χρησιμοποιήθηκαν στην υλοποίηση του Ρομποτικού Συμβούλου Σπουδών οι οποίες αποτελούν τα θεωρητικά θεμέλια της διπλωματικής εργασίας πριν από την υλοποίηση της. Το συγκεκριμένο κεφάλαιο είναι αυτό το οποίο θα μας τροφοδοτήσει με τις θεωρητικές γνώσεις όσον αφορά τις θεματικές ενότητες τις οποίες θα υλοποιήσουμε στα παρακάτω κεφάλαια της παρούσας διπλωματικής εργασίας.

2.2 Τεχνητή Νοημοσύνη

Ο τομέας της Τεχνητής Νοημοσύνης είναι δύσκολο να περιοριστεί εντός κάποιου μοναδικού ολοκληρωτικού ορισμού. Ανάλογα με την προσέγγιση που υιοθετείται, διατυπώνονται και διαφορετικοί ορισμοί, με αποτέλεσμα να υπάρχει μεγάλη ποικιλία σχετικών ορισμών. Ο ορισμός Τεχνητή Νοημοσύνη μπορεί να ταξινομηθεί σε τέσσερις διαφορετικές κατηγορίες ανάλογα με το πόσο εστιάζουν στην επίτευξη ανθρώπινης σκέψης ή στην συμπεριφορά των αποτελεσμάτων της τεχνητής νοημοσύνης [17]. Ενώ παράλληλα χωρίζονται και από τον τρόπο με τον οποίο αξιολογούνται, αν έχουν ως σημείο αναφοράς την ανθρώπινη συμπεριφορά ή την ορθολογική λειτουργία [17].

Η παρούσα διπλωματική εστιάζει μόνο σε δύο είδη κατηγοριών των ορισμών διότι βρίσκονται πιο νοηματικά κοντά για χρήση από έναν Ρομποτικό Σύμβουλο Σπουδών. Στην επίτευξη δηλαδή ενός μοντέλου κατάλληλου για να αντιδρά, να αλληλεπιδρά με γνώμονα και επίκεντρο την ανθρώπινη σκέψη και συμπεριφορά. Στην πρώτη κατηγορία ανήκουν οι ορισμοί οι οποίοι προσπαθούν να κάνουν τις μηχανές να σκεφτούν σαν άνθρωποι και να μιμηθούν ουσιαστικά την ανθρώπινη συμπεριφορά [17]. Στην δεύτερη κατηγορία ανήκουν οι ορισμοί οι οποίοι παρουσιάζουν μία τεχνητή νοημοσύνη η οποία αντιλαμβάνεται, σκέφτεται και αντιδρά με σκοπό να επιτύχει το καλύτερο δυνατό αποτέλεσμα[17].

Οι συγκεκριμένες προσεγγίσεις αποτέλεσαν τα θεμέλια για την ανάπτυξη, εξέλιξη και ενδελεχούς μελέτης των σύγχρονων conversational agents και Chatbots. Η πρόοδος του τομέα της τεχνητής νοημοσύνης και του αριθμού των δημοσιεύσεων σχετικά με αυτές είναι τεράστια και φυσικά έχει επηρεάσει πολλούς τομείς. Ο τομέας των Chatbot όπως είναι για παράδειγμα ένας ρομποτικός σύμβουλος σπουδών είναι ένας τομέας ο οποίος επηρεάστηκε σε πολύ μεγάλο βαθμό. Ακόμα και αν οι πρώτες αναφορές για chatbot είναι αρκετά παλαιότερες η εξέλιξη της τεχνητής νοημοσύνης βοήθησε σημαντικά αυτόν τον κλάδο της επιστήμης [18]. Αυτό συνέβη καθώς ένας ρομποτικός σύμβουλος σπουδών και οποιαδήποτε εφαρμογή η οποία στηρίζεται σε αυτά τα θεμέλια έχει ανάγκη για καλύτερη αντίληψη για το πλαίσιο για το οποίο απαντάει ανάλογα με την εκάστοτε ερώτηση. Το γεγονός της αντίληψης του σωστού πλαισίου είναι ένα σημαντικό προσόν της Τεχνητής Νοημοσύνης. Επιπρόσθετα η εξέλιξη του επιστημονικού αντικειμένου της Τεχνητής Νοημοσύνης έχει τα απαραίτητα χαρακτηριστικά έτσι ώστε τέτοιες εφαρμογές να μπορούν να δημιουργούν προσωποποιημένες απαντήσεις αναλαμβάνοντας και επεξεργάζοντας καταλλήλως το ύψος απεύθυνσης προς τον χρήστη.

Τέλος, η προαναφερόμενη πρόοδος των τελευταίων δέκα ετών δημιούργησε πρόσφορο έδαφος για την ανάπτυξη των μεγάλων γλωσσικών μοντέλων (Large Language Models - LLMs). Χάρη στις

δυνατότητές τους, οι οποίες θα αναλυθούν στα επόμενα κεφάλαια, τα LLMs συνέβαλαν σημαντικά στην υποστήριξη και την εξέλιξη των chatbots.

2.3 Μεγάλα Γλωσσικά Μοντέλα

Σε αυτό το υποκεφάλαιο περιγράφονται τα μεγάλα γλωσσικά μοντέλα (Large Language Models - LLMs), ο τρόπος χρήσης τους, η συμβολή τους σε έναν Ρομποτικό Σύμβουλο Σπουδών, καθώς και ο ρόλος της Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing - NLP) στις συγκεκριμένες εφαρμογές..

Ως μεγάλα γλωσσικά μοντέλα ορίζουμε τα γλωσσικά μοντέλα τα οποία έχουν εκπαιδευτεί με χρήση μεγάλου μεγέθους όγκου δεδομένων, κυρίως κειμένων αλλά γίνεται εκπαίδευση με χρήση άλλων μορφών δεδομένων, όπως για παράδειγμα εικόνες και βίντεο, αξιοποιώντας αρχιτεκτονικές βαθιάς μάθησης (Deep Learning). Τα μοντέλα αυτά χρησιμοποιούνται για την επεξεργασία, τη μετάφραση και τη δημιουργία κειμένου, καθώς και για την εκτέλεση διαφόρων διεργασιών Επεξεργασίας Φυσικής Γλώσσας (NLP) [19]. Ουσιαστικά, τα μεγάλα γλωσσικά μοντέλα αποτελούν εξελιγμένες εκδοχές των γλωσσικών μοντέλων (LMs). Έχουν εκπαιδευτεί σε μεγάλους όγκους δεδομένων και, μέσω της εκπαίδευσης, εντοπίζουν μοτίβα και συσχετισμούς στα δεδομένα αυτά [20]. Οι ικανότητες των LLMs όπως είναι για παράδειγμα τα LLM μοντέλα GPT της OpenAI επεκτείνουν τα παραπάνω χαρακτηριστικά και είναι εκπαιδευμένα για να μπορούν να έχουν ικανότητες σε κώδικα, σε δημιουργία κειμένου, εικόνων και βίντεο [20]. Οι οργανισμοί οι οποίοι παρέχουν τέτοια γλωσσικά μοντέλα είναι αρκετοί όπως για παράδειγμα το Gemini της Google, το Deepseek της DeepSeek AI, το Llama της Meta και άλλοι. Στην συγκεκριμένη διπλωματική γίνεται εστίαση στα μοντέλα μόνο της OpenAI καθώς αυτό το μοντέλο χρησιμοποιήθηκε στην υλοποίηση.

Η μεγάλη και γρήγορη ανάπτυξη των μεγάλων γλωσσικών μοντέλων (LLMs), όπως για παράδειγμα το GPT-4 και LLaMa, έχει επηρεάσει σημαντικά τον τομέα εφαρμογών τύπου Chatbot [18]. Το ρομπότ NAO V6 δεν έχει κάποια σχετική λειτουργία φυσικής επεξεργασίας γλώσσας και χρησιμοποιεί μια στατική δομή δεδομένων, με συγκεκριμένες και προκαθορισμένες ερωτήσεις και απαντήσεις [21] πολλές φορές διασυνδέεται το ίδιο το ρομπότ με ένα μεγάλο γλωσσικό μοντέλο με σκοπό να απαντάει με βάση το κατάλληλο πλαίσιο, ύφος και με φυσικότητα.

Μία μέθοδος για να εξελιχθεί περαιτέρω μία τέτοιου είδους υλοποίηση ενός Ρομποτικού συμβούλου σπουδών, είναι μέσω της συνεργασίας των μεγάλων γλωσσικών μοντέλων (LLM) με συστήματα RAG (Retrieval-Augmented Generation), καθώς μία τέτοια συνεργασία βοηθάει στην καλύτερη αντίληψη του πλαισίου, παρέχει μεγαλύτερη ακρίβεια και μετατρέπει αυτά τα συστήματα σε πιο έμπιστα και κατάλληλα για ανθρώπινη επικοινωνία μεταξύ ανθρώπων και ρομπότ [3].

2.4 Retrieval-Augmented Generation

Τα LLMs είναι ιδιαίτερα βοηθητικά στον τομέα των Social Robotics όμως δεν αρκούν για να προσφέρουν την απόλυτη λύση. Ένα από τα σημαντικά μειονεκτήματα που παρουσιάζουν είναι το φαινόμενο των “hallucination”, το συγκεκριμένο φαινόμενο μεταφράζεται στα ελληνικά ως οφθαλμαπάτη ή ψευδαίσθηση και αναφέρεται στο γεγονός στο οποίο ένα LLM μπορεί και παρουσιάζει μία απάντηση η οποία βασίζεται σε μη επικαιροποιημένα και αληθή στοιχεία. Σε τομείς υγείας ή ακόμα και σε εφαρμογές όπως ο Ρομποτικός Σύμβουλος Σπουδών είναι ιδιαίτερα αρνητικό φαινόμενο και μειώνει κατά πολύ την αξιοπιστία της εφαρμογής. Για την επίλυση τέτοιων προβλημάτων δημιουργήθηκε η ανάγκη για τα RAG συστήματα [22].

Η έννοια του συστήματος RAG προτάθηκε από τον Lewis et al. ως μία υβριδική αρχιτεκτονική η οποία χρησιμοποιούσε γενικούς κανόνες και παραμέτρους από τα μεγάλα γλωσσικά μοντέλα, όμως τα συνδύαζε με κάποια εσωτερική συγκεκριμένη πηγή δεδομένων [22]. Αυτός ο τρόπος λειτουργίας είναι αρκετά αποδοτικός ιδιαίτερα στους τομείς που προαναφέρθηκαν, καθώς συνδυάζει δύο πολύ σημαντικά πλεονεκτήματα των μεθόδων. Τα θετικά στοιχεία της δημιουργίας νέας πληροφορίας όπως είναι για παράδειγμα η δημιουργία νέου κειμένου τα οποία προσφέρουν τα μεγάλα γλωσσικά μοντέλα. Και τα θετικά στοιχεία της κατάλληλης παροχής πλαισίου που απαιτείται για να αποφευχθούν φαινόμενα όπως το “hallucination”. Επιπλέον το πρόσθετο πλαίσιο το οποίο προσφέρει, χρησιμοποιείται και αυξάνει την συνολική ακρίβεια στην διαδικασία παραγωγής κειμένου.

Αυτά τα στοιχεία τα οποία προσδίδει ένα σύστημα RAG είναι παραπάνω από χρήσιμα για έναν Ρομποτικό Σύμβουλο Σπουδών καθώς κατά την δημιουργία και παραγωγή μιας απάντησης σε κάποια ενδεχόμενη ερώτηση του χρήστη χωρίς την προσθήκη των RAG συστημάτων και αποκλειστικά χρησιμοποιώντας μόνο τα LLMs η απάντηση, λόγω των εξαιρετικά μεγάλου όγκου δεδομένων με τα οποία είναι εκπαιδευμένα αυτά τα γλωσσικά μοντέλα, θα μπορούσε να οδηγηθεί σε κάποιο λανθασμένο συμπέρασμα το οποίο θα επέστρεφε και μία λάθος απάντηση. Η συμβολή των RAG λοιπόν προσανατολίζει κατάλληλα τον όγκο πληροφορίας και δεδομένων που καλείται να ψάξει το εκάστοτε LLM και με αυτόν τον τρόπο η γνώση που επιστρέφεται είναι συγκεκριμένη από ένα εξωτερικό σύστημα βάσης γνώσης το οποίο έχει προσθέσει για αυτόν τον σκοπό ο χρήστης. Παράλληλα το LLM χρησιμοποιεί τις ικανότητες του για παραγωγή κειμένου και μπορεί ουσιαστικά την ανακτώμενη πληροφορία που επιστρέφει το RAG σύστημα να την εξανθρωπίζει και να την παρουσιάζει με τρόπο φιλικό για τον χρήστη της εφαρμογής.

Η λειτουργία ενός RAG συστήματος χωρίζεται πρακτικά σε δύο μέρη[23]. Το πρώτο μέρος αφορά την ανάκτηση της σχετικής πληροφορίας από τα κειμενικά δεδομένα μιας εξωτερικής βάσης, με βάση το πλαίσιο το οποίο έχει τεθεί από την ερώτηση του χρήστη. Το δεύτερο μέρος είναι υπεύθυνο για την παραγωγή του κειμενικού αντικειμένου με βάση τα ανακτημένα δεδομένα τα οποία τα δέχεται ως είσοδο από το πρώτο μέρος και κατά αυτόν τον τρόπο δημιουργείται η τελική απάντηση του RAG συστήματος[22].

Ο τρόπος με τον οποίο λειτουργεί το πρώτο μέρος χωρίζεται και αυτό σε κάποιες επιπλέον επιμέρους διαδικασίες οι οποίες θα περιγραφούν σε εισαγωγικό επίπεδο σε αυτό το κομμάτι της διπλωματικής εργασίας και μετέπειτα θα αναλυθούν εκτενότερα στα επιμέρους υποκεφάλαια. Αρχικά το RAG σύστημα περιέχει μία βάση γνώσης, η οποία αποτελείται από τα δεδομένα με τα οποία την έχει τροφοδοτήσει ο διαχειριστής του συστήματος. Αυτά τα αρχεία χωρίζονται σε επιμέρους μικρότερα τμήματα με βάση κάποια αρχιτεκτονική Chunking και δημιουργούνται τα chunks. Μετέπειτα αυτά με βάση μίας μεθόδου Embeddings κωδικοποιούνται σε διανύσματα, σε πιο κατανοητή μορφή για τους ηλεκτρονικούς υπολογιστές. Υστέρα αφού αυτά τα τμήματα έχουν χωριστεί σε διανύσματα αποθηκεύονται σε μία διανυσματική βάση δεδομένων (Vector Database). Κατά την στιγμή της ανάκτησης επιστρέφουν τα κομμάτια τα οποία θεωρούνταν πιο σχετικά με το αίτημα του εκάστοτε χρήστη και τέλος αυτά τα κομμάτια στέλνονται σαν σχετικό πλαίσιο στο LLM, όπου αυτό παράγει την απάντηση η οποία επιστρέφεται στον τελικό χρήστη.

2.5 Chunking

Ο όγκος των δεδομένων με τα οποία εκπαιδεύονται τα μεγάλα γλωσσικά μοντέλα είναι τεράστιος. Στα συστήματα RAG, όμως, ο όγκος των δεδομένων που χρησιμοποιούνται είναι συνήθως αρκετά μικρότερος και καθορίζεται από τον διαχειριστή του εκάστοτε συστήματος RAG. Μπορεί να κυμαίνεται από μερικά έως και δεκάδες χιλιάδες αρχεία, ανάλογα με το είδος και τον σκοπό του

συστήματος. Σε κάθε περίπτωση, ο όγκος αυτός παραμένει σημαντικά μικρότερος από τον όγκο των δεδομένων εκπαίδευσης ενός LLM. Παρότι ο αριθμός των αρχείων είναι μικρότερος, τα αρχεία αυτά χρειάζονται επιπλέον διάσπαση, με στόχο την καλύτερη λειτουργία του συστήματος ανάκτησης. Η διαδικασία διάσπασης του κειμένου σε μικρότερα τμήματα, με τη χρήση διαφόρων τεχνικών επεξεργασίας, ονομάζεται chunking. Η διαδικασία αυτή μπορεί να διαφέρει ανάλογα με το είδος του κειμένου και άλλα χαρακτηριστικά του, όπως η ύπαρξη εικόνων ή πινάκων. Αξίζει να σημειωθεί ότι η στρατηγική που θα ακολουθηθεί είναι ιδιαίτερα σημαντική, καθώς τα chunks μπορεί να είναι πολύ μικρά και να παρέχουν μη ολοκληρωμένη πληροφορία. Αντίστοιχα, κάθε chunk μπορεί να περιλαμβάνει μεγάλο όγκο δεδομένων, με αποτέλεσμα να συνδυάζονται διαφορετικές πληροφορίες και να δημιουργείται θόρυβος κατά τη διαδικασία της ανάκτησης [22]. Η χρησιμότητα του chunking έγκειται στο γεγονός ότι ένα κείμενο που περιέχει μεγάλο όγκο πληροφοριών διασπάται σε μικρότερα τμήματα. Για παράδειγμα, στην παρούσα διπλωματική εργασία, ένα κείμενο σχετικά με τα μεταπτυχιακά προγράμματα σπουδών χωρίζεται σε πολλά μικρότερα τμήματα. Ανάλογα με τη στρατηγική που ακολουθείται, το μέγεθος κάθε chunk διαφέρει. Ο βασικός στόχος, όμως, παραμένει ο ίδιος: κάθε chunk να περιέχει ολοκληρωμένη και αυτοτελή πληροφορία. Στη συνέχεια, σε κάθε τμήμα αντιστοιχίζεται ένα embedding. Κατά τη διαδικασία της ανάκτησης, η ερώτηση του χρήστη μετατρέπεται επίσης σε διάνυσμα και αναζητούνται τα chunks των οποίων τα embeddings βρίσκονται κοντά στο διάνυσμα της ερώτησης.

2.6 Embeddings

Ο τεμαχισμός του κειμένου σε επιμέρους τμήματα, από μόνος του, δεν επαρκεί, καθώς το σύστημα RAG δεν μπορεί να κατανοήσει τη σημασιολογική πληροφορία κάθε chunk που δημιουργήθηκε. Επομένως, είναι απαραίτητη μια μέθοδος που να αποτυπώνει με κάποιον τρόπο το θεματικό υπόβαθρο κάθε chunk, καθώς και κάθε νέας ερώτησης που υποβάλλει ο χρήστης στο σύστημα. Αυτό ισχύει είτε πρόκειται για έναν Ρομποτικό Σύμβουλο Σπουδών είτε για οποιαδήποτε άλλη εφαρμογή τύπου chatbot, στην οποία ο χρήστης υποβάλλει ερωτήσεις και λαμβάνει απαντήσεις μέσω μιας βάσης γνώσης. Η μέθοδος που καλύπτει αυτή την ανάγκη είναι τα text embeddings ή, απλώς, embeddings. Η μέθοδος αυτή ανήκει στον τομέα της Επεξεργασίας Φυσικής Γλώσσας ή, αλλιώς, Natural Language Processing (NLP). Μέσω αυτής, οι λέξεις ή τα τμήματα κειμένου αναπαρίστανται ως διανύσματα πραγματικών αριθμών. Τα διανύσματα που βρίσκονται κοντά μεταξύ τους στον διανυσματικό χώρο αντιστοιχούν συνήθως σε νοηματικά κοντινά δεδομένα [24]. Το χαρακτηριστικό αυτό είναι ιδιαίτερα χρήσιμο σε εφαρμογές ανάκτησης πληροφορίας, αλλά χρησιμοποιείται ευρέως και σε εφαρμογές μηχανικής μάθησης. Η εφαρμογή των embeddings στον Ρομποτικό Σύμβουλο Σπουδών παρουσιάζει ιδιαίτερο ενδιαφέρον, καθώς το μοντέλο embeddings που θα χρησιμοποιηθεί επηρεάζει το αποτέλεσμα της ανάκτησης. Το γεγονός αυτό έχει ιδιαίτερη σημασία για τη συγκεκριμένη διπλωματική εργασία, επειδή τα έγγραφα και τα chunks τους είναι γραμμένα στην ελληνική γλώσσα. Επομένως, τα embeddings πρέπει να είναι κατάλληλα για την αναπαράσταση ελληνικών κειμένων. Αν δεν είναι, θεματικές ενότητες που είναι νοηματικά κοντινές ενδέχεται να μην αποτυπώνονται ως κοντινά διανύσματα στον διανυσματικό χώρο. Στο πλαίσιο του Ρομποτικού Συμβούλου Σπουδών, κάθε chunk που δημιουργείται μέσω της στρατηγικής chunking αντιστοιχίζεται σε ένα διάνυσμα πραγματικών αριθμών. Όσο πιο κοντά είναι νοηματικά δύο chunks, τόσο πιο κοντά αναμένεται να βρίσκονται και τα αντίστοιχα embeddings τους. Η αντιστοίχιση ενός chunk με ένα embedding, με βάση τη θεματική του ενότητα, δεν αρκεί από μόνη της. Τα διανύσματα αυτά πρέπει να αποθηκεύονται σε μια κατάλληλη βάση δεδομένων, όπως μια διανυσματική βάση δεδομένων. Όταν ο

χρήστης υποβάλλει μια ερώτηση, αυτή μετατρέπεται επίσης σε διάνυσμα μέσω του μοντέλου embeddings. Στη συνέχεια, πραγματοποιείται αναζήτηση στη διανυσματική βάση δεδομένων και ανακτώνται τα chunks των οποίων τα embeddings είναι περισσότερο κοντά στο διάνυσμα της ερώτησης.

2.7 Vector DB- Semantic Search

Το embedding model είναι αυτό που μετατρέπει λέξεις και κειμενικά τμήματα σε διανύσματα πραγματικών αριθμών. Τα διανύσματα αυτά συνήθως αποθηκεύονται όχι σε βάσεις δεδομένων SQL, αλλά σε ειδικές διανυσματικές βάσεις δεδομένων (Vector Databases), οι οποίες παρέχουν εξειδικευμένες μεθόδους αναζήτησης. Μία τέτοια μέθοδος είναι η προσέγγιση πλησιέστερου γείτονα (Approximate Nearest Neighbor - ANN) [22]. Μέσω της συγκεκριμένης μεθόδου, όταν ένας χρήστης υποβάλλει μια ερώτηση, αυτή μετατρέπεται σε διάνυσμα πραγματικών αριθμών και εντοπίζονται τα embeddings των chunks που έχουν αποθηκευτεί στη διανυσματική βάση δεδομένων και βρίσκονται πλησιέστερα στο διάνυσμα της ερώτησης. Η πρακτική αυτή χρησιμοποιείται ευρέως σε συστήματα RAG και είναι ιδιαίτερα χρήσιμη σε εφαρμογές τύπου chatbot, καθώς οι ερωτήσεις των χρηστών σπάνια ταυτίζονται αυτούσιες με τα δεδομένα της βάσης γνώσης. Για τον λόγο αυτό, η αναζήτηση δεν περιορίζεται στην ακριβή αντιστοίχιση λέξεων ή φράσεων, αλλά μπορεί να βασίζεται και στη σημασιολογική ομοιότητα. Αυτό είναι ιδιαίτερα σημαντικό στην επικοινωνία μεταξύ ανθρώπων και ρομπότ, καθώς η ίδια πληροφορία μπορεί να διατυπωθεί με πολλούς διαφορετικούς τρόπους. Οι διανυσματικές βάσεις δεδομένων υποστηρίζουν αυτή τη διαδικασία, καθώς μέσω της ANN επιτρέπουν την αποδοτική αναζήτηση των πλησιέστερων διανυσμάτων, τα οποία αντιστοιχούν συνήθως σε νοηματικά συναφή δεδομένα. Η διαδικασία που περιγράφηκε αποτελεί τη βάση της σημασιολογικής αναζήτησης (Semantic Search). Η σημασιολογική αναζήτηση επιστρέφει αποτελέσματα με βάση το νόημα και τη θεματική ενότητα ενός ερωτήματος, αντί να περιορίζεται αποκλειστικά στην αναζήτηση κοινών λέξεων. Με άλλα λόγια, η αναζήτηση πραγματοποιείται με βάση τη σημασιολογική ομοιότητα μεταξύ του ερωτήματος και των δεδομένων που είναι αποθηκευμένα στη διανυσματική βάση δεδομένων. Χαρακτηριστικό παράδειγμα της μεθόδου στην εφαρμογή του Ρομποτικού Συμβούλου Σπουδών είναι το εξής: έστω ότι ο χρήστης θέλει να υποβάλει την ερώτηση «Πού βρίσκεται το ΔΠΙΑΕ;». Η συγκεκριμένη ερώτηση μπορεί να μην υπάρχει αυτούσια στη βάση γνώσης. Το σύστημα αναζητά, επομένως, chunks που είναι σημασιολογικά συναφή με την ερώτηση. Πρόκειται για σημασιολογική αναζήτηση (Semantic Search), κατά την οποία επιστρέφονται τα καταλληλότερα chunks με βάση την ομοιότητα των embeddings τους με το διάνυσμα που προκύπτει από την ερώτηση.

2.8 Speech Recognition

Η φωνητική αναγνώριση είναι ένας επιστημονικός τομέας που χρησιμοποιείται ευρέως σε εφαρμογές όπως τα κοινωνικά ρομπότ. Οι ρομποτικοί σύμβουλοι σπουδών, τα πιο εξελιγμένα chatbots και γενικά παρόμοιες εφαρμογές, οι οποίες απαιτούν αποτελεσματική αλληλεπίδραση και επικοινωνία μεταξύ ανθρώπων και ρομπότ, αξιοποιούν σε μεγάλο βαθμό τη φωνητική αναγνώριση. Αυτό συμβαίνει επειδή υπάρχει ανάγκη για την αναπαράσταση ενός πραγματικού διαλόγου μεταξύ ρομπότ και ανθρώπου. Το συγκεκριμένο χαρακτηριστικό επιτυγχάνεται όταν και οι δύο πλευρές μπορούν να χρησιμοποιούν την ομιλία για να διατυπώνουν ερωτήματα ή να παρέχουν απαντήσεις. Ως φωνητική

αναγνώριση ή αναγνώριση ομιλίας (Speech Recognition) ορίζουμε τη διαδικασία αναγνώρισης της ακολουθίας των λέξεων που εκφωνεί ένας χρήστης, με βάση το ακουστικό σήμα που λαμβάνεται και καταγράφεται από μια συσκευή [17]. Στη συγκεκριμένη διπλωματική εργασία, η οποία δίνει ιδιαίτερη έμφαση στην επικοινωνία μεταξύ του ρομπότ NAO V6 και των φοιτητών που αλληλεπιδρούν με αυτό, η φωνητική αναγνώριση αποτελεί ιδιαίτερα σημαντικό χαρακτηριστικό. Για να απαντήσει σωστά το ρομπότ σε μια ερώτηση του φοιτητή, απαιτείται η κατάλληλη αναγνώριση της ερώτησης. Η διαδικασία αυτή παρουσιάζει αρκετές προκλήσεις, καθώς ένα σφάλμα στη φωνητική αναγνώριση μπορεί να οδηγήσει στην ανάκτηση λανθασμένων chunks. Με αυτόν τον τρόπο, το LLM, το οποίο είναι υπεύθυνο για την παραγωγή της απάντησης, ενδέχεται να μην έχει τις κατάλληλες πληροφορίες για να απαντήσει σωστά στο αίτημα του χρήστη. Επομένως, υπάρχει κίνδυνος η ερώτηση να μην απαντηθεί σωστά ή να μην απαντηθεί καθόλου. Στα κοινωνικά ρομπότ, η φωνητική αναγνώριση χρησιμοποιείται από τα πρώτα στάδια ανάπτυξής τους. Το ρομπότ NAO V6, το οποίο χρησιμοποιείται στην παρούσα διπλωματική εργασία, διαθέτει επίσης ενσωματωμένη λειτουργία φωνητικής αναγνώρισης, η οποία όμως δεν επαρκεί για την πλήρη κατανόηση της ανθρώπινης γλώσσας [21]. Για την κάλυψη αυτής της ανάγκης χρησιμοποιήθηκαν εξωτερικά μοντέλα φωνητικής αναγνώρισης, όπως το μοντέλο Whisper της OpenAI και, ως δεύτερη επιλογή, το μοντέλο Vosk της Alpha Cephei, το οποίο είναι μοντέλο ανοιχτού κώδικα.

2.9 Χαρακτηριστικά NAO V6.

Τα παραπάνω στοιχεία συνθέτουν το κατάλληλο περιβάλλον για την υποστήριξη της αλληλεπίδρασης, κατά την οποία ο χρήστης υποβάλλει μια ερώτηση μέσω ομιλίας. Αρχικά, ανακτώνται τα κατάλληλα τμήματα κειμένου και η πληροφορία αυτή μεταβιβάζεται στο LLM, το οποίο συνθέτει την απάντηση που θα παρουσιαστεί στον χρήστη. Στη συνέχεια, η απάντηση εκφωνείται από τα ηχεία του ρομπότ NAO V6. Παράλληλα, εκτελούνται διάφορες κινήσεις του σώματος, ώστε να προσδίδεται ένα πιο φιλικό ύφος στη συνολική εφαρμογή. Το ρομπότ χρησιμοποιείται, ουσιαστικά, ως η φωνητική και κινητική διεπαφή αλληλεπίδρασης (interaction interface) του Ρομποτικού Συμβούλου Σπουδών με τον τελικό χρήστη. Σε μια τέτοια εφαρμογή αλληλεπίδρασης ανθρώπου-ρομπότ, πέρα από το σύστημα που λειτουργεί στο παρασκήνιο, αξίζει να αναφερθούμε και στο ίδιο το ρομπότ. Το NAO V6 είναι ένα αυτόνομο ανθρωποειδές ρομπότ, το οποίο κυκλοφόρησε το 2018 [21]. Έχει συνολικό ύψος 57 έως 58 εκατοστά και ζυγίζει περίπου 4,5 κιλά [9]. Τα παραπάνω χαρακτηριστικά, σε συνδυασμό με τη σχεδίαση του ρομπότ, προσδίδουν μια φιλική εμφάνιση, κατάλληλη για αλληλεπίδραση με ανθρώπους. Επιπλέον, μαζί με το ρομπότ παρέχεται το λογισμικό Choregraphe για γραφικό προγραμματισμό. Το χαρακτηριστικό αυτό διαμορφώνει ένα αρκετά φιλικό περιβάλλον για νέους χρήστες του ρομπότ, αλλά και για νέους προγραμματιστές. Το συγκεκριμένο ρομπότ επιλέχθηκε με βάση τα προαναφερθέντα χαρακτηριστικά. Επιπρόσθετα, το NAO V6 παρέχει εργαλεία και υποστήριξη για γλώσσες όπως η Python και η C++, καθώς διαθέτει API για την ανάπτυξη εφαρμογών με τη χρήση των συγκεκριμένων γλωσσών. Χάρη στην αρχιτεκτονική NAOqi, οι προγραμματιστές μπορούν να αξιοποιήσουν ένα πλήθος αισθητήρων, διαθέσιμων κινήσεων, καθώς και λειτουργιών ομιλίας και ακοής. Αυτά τα στοιχεία αποτελούν βασικούς λόγους για τους οποίους το NAO χρησιμοποιείται συχνά σε ρομποτικές εφαρμογές, καθώς καθιστούν τη χρήση του φιλική τόσο για τον προγραμματιστή όσο και για τον τελικό χρήστη.

2.10 Επίλογος

Στο συγκεκριμένο κεφάλαιο έγινε μία παρουσίαση των τεχνολογιών αλλά και των τεχνικών οι οποίες χρησιμοποιούνται στην παρούσα εργασία για να δοθεί το κατάλληλο θεωρητικό υπόβαθρο στον αναγνώστη. Σκοπός αυτού του κεφαλαίου είναι η εν μέρει κατανόηση των τεχνοτροπιών οι οποίες

χρησιμοποιήθηκαν στον Ρομποτικό Σύμβουλο Σπουδών της παρούσας διπλωματικής εργασίας. Συγκεκριμένα στο παρόν κεφάλαιο παρουσιάστηκαν οι τεχνολογίες οι οποίες συνθέτουν ένα RAG σύστημα, επιμέρους τεχνικές οι οποίες χρησιμοποιούνται σε αυτού του είδους τα συστήματα, όπως είναι για παράδειγμα το Chunking και τα Embeddings. Παράλληλα παρουσιάστηκε σε θεωρητικό υπόβαθρο το Speech Recognition, ένας τομέας ο οποίος χρησιμοποιείται σε απόλυτο βαθμό στην παρούσα διπλωματική εργασία και επηρεάζει την συνολική της λειτουργία. Έστερα στο παρόν κεφάλαιο παρουσιάστηκε το ρομπότ NAO V6, τα διάφορα χαρακτηριστικά του και οι μέθοδοι οι οποίοι θα συμβάλλουν στην υλοποίηση της παρούσας Διπλωματικής Εργασίας. Ουσιαστικά το συγκεκριμένο κεφάλαιο περιέχει κάποιες εισαγωγικές γνώσεις οι οποίες θα βοηθήσουν τον χρήστη να κατανοήσει πιο ολοκληρωμένα τα επόμενα κεφάλαια τα οποία αφορούν την συνολική υλοποίηση του συστήματος.

Κεφάλαιο 3ο: Υλοποίηση Retrieval Augmented Generation (RAG) Συστήματος

3.1 Εισαγωγή

Στο κεφάλαιο 2 περιγράφηκαν τα θεωρητικά θεμέλια της παρούσας διπλωματικής εργασίας. Δηλαδή αναλύθηκαν θεωρητικά οι έννοιες της Τεχνητής Νοημοσύνης, τα μεγάλα γλωσσικά μοντέλα (LLMs), τα συστήματα Retrieval-Augmented Generation (RAG), τεχνικές Chunking, τεχνικές Embedding. Αναλύθηκαν επίσης τεχνικές αποθήκευσης των Embeddings με χρήση Διανυσματικών Βάσεων Δεδομένων, το Semantic Search, το θεωρητικό υπόβαθρο της φωνητικής αναγνώρισης και τέλος παρουσιάστηκε η Διεπαφή Χρήστη μεταξύ ανθρώπου και ρομπότ, δηλαδή το Ρομπότ ΝΑΟ V6. Όλες οι παραπάνω τεχνολογίες συνδυάστηκαν και δημιούργησαν την παρούσα διπλωματική του Ρομποτικού Συμβούλου Σπουδών.

Στο τρίτο κεφάλαιο της Διπλωματικής Εργασίας θα παρουσιαστεί ο τρόπος υλοποίησης του συστήματος Retrieval-Augmented Generation (RAG). Για τους λόγους για τον οποίο επιλέχθηκε να χρησιμοποιηθεί το σύστημα του RAG έναντι μίας υλοποίησης Fine Tuning ενός ήδη υπάρχοντος Μεγάλου Γλωσσικού Μοντέλου (LLM) από διάφορες οπτικές. Θα περιγραφεί επίσης η συνολική αρχιτεκτονική του RAG συστήματος αναδεικνύοντας τα πλεονεκτήματα κάθε επιλογής, με τελικό στόχο την ολική συναρμογή του συστήματος.

Παράλληλα θα γίνει και μία ενδότερη ανάλυση των τεχνικών στοιχείων, των τεχνολογιών και των μεθόδων που επιλέχθηκαν σε κάθε μεμονωμένο κομμάτι του RAG συστήματος. Δηλαδή τον τρόπο με τον οποίο δημιουργήθηκε η συνολική βάση γνώσης για το συγκεκριμένο σύστημα. Πως χωρίστηκαν τα αρχεία, τον τρόπο με τον οποίο αυτά επεξεργάστηκαν. Μετέπειτα θα εξηγηθεί ο τρόπος με τον οποίο τα επεξεργασμένα κατάλληλα αρχεία τεμαχίστηκαν σε πολλά επιμέρους τμήματα. Θα αναλυθεί ο τρόπος και η στρατηγική των Embeddings η οποία χρησιμοποιήθηκε για να αποδώσει κάθε τμήμα κειμένου που δημιουργήθηκε το κατάλληλο διάνυσμα πραγματικών αριθμών. Έπειτα θα γίνει αναφορά στον τρόπο με τον οποίο οι ερωτήσεις ανακτούν ως απαντήσεις τα κατάλληλα επιμέρους κομμάτια του κειμένου. Όστε μελλοντικά να δοθούν όλα αυτά σαν πλαίσιο, μαζί με την ερώτηση του χρήστη και να δημιουργηθεί μία απάντηση από το εκάστοτε LLM το οποίο χρησιμοποιήθηκε. Αφού αυτή η απάντηση δημιουργηθεί, τότε δίνεται σε συνδυασμό με ένα πλήθος κινητικών εντολών ώστε το ΝΑΟ V6, η διεπαφή μεταξύ ανθρώπου και ρομπότ να παρουσιάσει στον χρήστη το τελικό αποτέλεσμα.

Η επιλογή της εκάστοτε στρατηγικής σε οποιοδήποτε σημείο βασίστηκε σε διάφορους παράγοντες ώστε να υλοποιηθεί αρμονικά και λειτουργικά ο Ρομποτικός Σύμβουλος Σπουδών. Στα διαφορικά στάδια της διπλωματικής εργασίας έπρεπε να ληφθούν υπόψη διάφοροι παράγοντες. Παραδείγματα αυτών είναι το υπολογιστικό κόστος, το χρηματικό κόστος, η αποτελεσματικότητα, η ταχύτητα, η αξιοπιστία, η ευκολία συντήρησης και άλλα. Διότι όλοι αυτοί οι παράγοντες συνεργατικά επηρεάζουν σε μεγαλύτερο ή μικρότερο βαθμό την συνολική αλληλεπίδραση μεταξύ ενός ανθρώπου και του ρομποτικού συμβούλου σπουδών της παρούσας διπλωματικής εργασίας. Για παράδειγμα μία μέθοδος φωνητικής αναγνώρισης μπορεί να ήταν ιδιαίτερα αξιόπιστη και να καταλάβαινε το μεγαλύτερο ποσοστό της ομιλίας του χρήστη σωστά αλλά να καθυστερούσε σημαντικά στον χρόνο παρουσίασης του αποτελέσματος της καταγραφής. Μία τέτοια μέθοδος θα απορριπτόταν και θα κυριαρχούσε μία άλλη μέθοδος με λιγότερη αξιοπιστία αλλά με μεγαλύτερη ταχύτητα.

3.2 RAG VS Fine Tuning : Επιλογή αρχιτεκτονικής RAG

Κατά την υλοποίηση το πρώτο δίλημμα το οποίο αντιμετωπίστηκε είχε να κάνει με τον τρόπο με τον οποίο θα χτιζόταν ολόκληρη λογική για τον τρόπο αναζήτησης και ανάκτησης της πληροφορίας. Τα δύο συστήματα που εξετάστηκαν ήταν αυτού του Fine Tuning (FT) και αυτό του RAG. Και τα δύο συστήματα χρησιμοποιούνται για εφαρμογές ρομποτικών συμβούλων, διότι τα αποτελέσματα τους είναι ιδιαίτερα ικανοποιητικά. Για τους παραπάνω λόγους υπήρξε η ανάγκη σύγκρισης των παραπάνω μεθόδων ώστε να αναλυθούν τα πλεονεκτήματα και τα μειονεκτήματα αυτών. Όστε τελικά να αναλύσουμε τους λόγους για τους οποίους επιλέχθηκε η μέθοδος του RAG συστήματος και τους λόγους για τους οποίους υπήρχε αυτό το δίλημμα.

Τα RAG συστήματα και αυτά που χρησιμοποιούν Fine Tuning (FT) είναι δύο μέθοδοι οι οποίες χρησιμοποιούνται σε γλωσσικά μοντέλα με σκοπό να γίνουν οι κατάλληλες προσαρμογές σε αυτά με στόχο να επικεντρωθούν σε κάποιο συγκεκριμένο αντικείμενο γνώσης. Η μέθοδος του RAG ουσιαστικά χτίζει το κατάλληλο πλαίσιο έτσι ώστε να επέμβουν σε αυτό οι ικανότητές του εκάστοτε γλωσσικού μοντέλου και να παραχθεί μία απάντηση η οποία ακολουθεί τους κανόνες του γλωσσικού μοντέλου. Ενώ παράλληλα η χρήση αυτού αντιλαμβάνεται το κατάλληλο πλαίσιο από την εξωτερική δομή δεδομένων του RAG συστήματος και το χρησιμοποιεί αντίστοιχα. Από την άλλη πλευρά ο τρόπος με τον οποίο το Fine Tuning προσαρμόζει τις τελικές απαντήσεις ως προς τα εκάστοτε δεδομένα διαφέρει σημαντικά συγκριτικά με αυτή του RAG. Η στρατηγική Fine Tuning αλλάζει τα βάρη του εκάστοτε γλωσσικού μοντέλου το οποίο χρησιμοποιείται με τον κατάλληλο τρόπο ώστε οι συγκεκριμένες σχετικές πληροφορίες να ευνοούνται έναντι πιο γενικών πληροφοριών. Το συγκεκριμένο επιτυγχάνεται μέσω μίας διαδικασίας εκ νέου εκπαίδευσης με χρήση νέων εξωτερικών δεδομένων πληροφορίας, με τελικό στόχο την προσαρμογή του μοντέλου σε ένα συγκεκριμένο γνωστικό πεδίο και τη βελτίωση της απόδοσής του σε σχετικές ερωτήσεις [25].

Αν και το Fine Tuning καταφέρνει να βελτιώσει αρκετά την απόδοση του γλωσσικού μοντέλου το οποίο χρησιμοποιείται καθώς εκπαιδεύεται επιπλέον σε συγκεκριμένα δεδομένα. Η μέθοδος του Retrieval-Augmented Generation καταγράφει καλύτερη επίδοση σχετικά με τις γνώσεις πάνω σε αυτή την εξωτερική δομή δεδομένων[25]. Από την άλλη πλευρά όμως η τεχνοτροπία του Fine Tuning μπορεί να χρησιμοποιηθεί με κατάλληλο τρόπο με τον οποίο θα προσδώσει ένα καταλληλότερο και πιο εξειδικευμένο τρόπο απάντησης [26]. Ουσιαστικά μέσω της επιπλέον εκπαίδευσης και της προσαρμογής των βαρών του γλωσσικού μοντέλου κατάλληλα για την αναγκαία ανακατεύθυνση του προς τα εξωτερικά δεδομένα γνώσης υπάρχει μεγαλύτερη εξειδίκευση σε ορολογία ανάλογη του εκάστοτε περιβάλλοντος [26]. Το παραπάνω γεγονός οδηγεί σε μία καταλληλότερη ίσως περσόνα για έναν ρομποτικό σύμβουλο ο οποίος είναι κατάλληλα εκπαιδευμένος για την συγκεκριμένη αρμοδιότητα.

Αφού αναφερθήκαμε στις δύο ειδοποιές διαφορές των δύο διαφορετικών αρχιτεκτονικών, του RAG και του Fine Tuning, αξίζει να αναφερθούμε στα στοιχεία στα οποία υπερτερεί η μεθοδολογία των RAG συστημάτων για μία υλοποίηση όπως ο ρομποτικός σύμβουλος σπουδών. Ένα από τα πιο σημαντικά πλεονεκτήματα του RAG έναντι της μεθόδου Fine Tuning στα γλωσσικά μοντέλα είναι το γεγονός της έλλειψης ανάγκης για επανεκπαίδευση [3] και παράλληλα της εύκολης ενημέρωσης των εξωτερικών αρχείων [22] τα οποία χρησιμοποιούνται ως η δομή άντλησης δεδομένων πληροφορίας. Συνδυαστικά αυτά τα δύο χαρακτηριστικά μας οδηγούνε στο συμπέρασμα ότι η μέθοδος RAG πλεονεκτεί στο γεγονός ότι προσφέρει εύκολη επικαιροποίηση στα αρχεία τα οποία χρησιμοποιεί. Δηλαδή στο παράδειγμα του ρομποτικού συμβούλου σπουδών αν ενημερωθεί ή αλλάξει κάποιος κανονισμός της σχολής, για παράδειγμα ο αριθμός μαθημάτων ο οποίος απαιτείται για την επίτευξη

και την απονομή πτυχίου, τότε η επικαιροποίηση των κανόνων μπορεί να γίνει ιδιαίτερα γρήγορα και εύκολα. Από την άλλη πλευρά αν ενημερωθούν αντίστοιχα οι κανονισμοί ή γενικότερα τα εξωτερικά δεδομένα μίας εφαρμογής ενός ρομποτικού συμβούλου η οποία χρησιμοποιεί Fine Tuning υπάρχει σημαντικό μειονέκτημα. Το μειονέκτημα αυτό συναντάται διότι αυτή η μέθοδος αν ενημερωθούν εκ νέου τα δεδομένα απαιτεί και εκ νέου εκπαίδευση στα επικαιροποιημένα νέα δεδομένα [26]. Αυτό είναι ιδιαίτερα αρνητικό στοιχείο καθώς αυτή η διαδικασία είναι ιδιαίτερα κοστοβόρα σε υπολογιστικούς πόρους.

Τα μεγάλα γλωσσικά μοντέλα είναι ιδιαίτερα επικίνδυνα να παρουσιάσουν ελλιπή ή λάθος πληροφορία ως γεγονός. Αυτό το φαινόμενο ονομάζεται φαινόμενο ψευδαισθήσεων αλλιώς στα αγγλικά αναφέρεται ως Hallucination, το παρόν φαινόμενο έχει αναφερθεί και στο κεφάλαιο 2. Όπως αναφέρθηκε και τότε τα RAG συστήματα είναι ιδιαίτερα ικανά στο να μην οδηγούνται σε τέτοια φαινόμενα καθώς το πλαίσιο το οποίο φτάνει στο εκάστοτε γλωσσικό μοντέλο έχει δομηθεί από τα συγκεκριμένα αρχεία που υπάρχουνε στην βάση γνώσης του RAG. Από την άλλη η επιλογή μίας στρατηγικής Fine Tuning είναι πιο επιρρεπής σε έναν τέτοιο κίνδυνο. Ακόμα και εάν το γλωσσικό μοντέλο επανεκπαιδεύεται με τα νέα δεδομένα και εξειδικεύεται ουσιαστικά σε αυτά υπάρχει ακόμα μεγάλο πλήθος πληροφοριών στο οποίο είχε εκπαιδευτεί πρώτα το γλωσσικό μοντέλο. Αυτό το χαρακτηριστικό της Fine Tuning στρατηγικής αποτελεί σημαντικό μειονέκτημα για μία υλοποίηση ενός ρομποτικού συμβούλου σπουδών καθώς μπορεί να απαντήσει μία ερώτηση πλήρως και με αυτοπεποίθηση ότι το μοντέλο έχει βρει την σωστή απάντηση χωρίς όμως κάτι τέτοιο να ισχύει και δίνοντας τελικά μία εσφαλμένη απάντηση προς τον χρήστη.

Ένας ακόμα παράγοντας ο οποίος εξετάστηκε για να ληφθεί η τελική απόφαση σχετικά με το εάν θα χρησιμοποιηθεί μία αρχιτεκτονική RAG συστήματος ή μία αρχιτεκτονική Fine Tuning είναι ο παράγοντας του κόστους. Για μία εφαρμογή όπως είναι ένας ρομποτικός σύμβουλος για το τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων έπρεπε να φτιαχτεί ένα σύστημα το οποίο θα ήταν οικονομικό για την συνεχή συντήρηση του. Αυτό το οποίο ισχύει για το RAG σύστημα είναι ότι απαιτεί ένα πολύ μικρό αρχικό κόστος υλοποίησης [26]. Το κόστος ενός RAG συστήματος ουσιαστικά περιλαμβάνει την δημιουργία των embeddings για τα chunks τα οποία έχουν δημιουργηθεί, την αποθήκευση αυτών σε μία vector Database, τον τρόπο ανάκτησης των δεδομένων ώστε να δοθούν σαν πλαίσιο στο LLM και τέλος το κόστος κλήσης για το ίδιο το LLM. Το συνολικό κόστος των παραπάνω διεργασιών είναι πολύ μικρό και σχεδόν αμελητέο. Από την άλλη το Fine Tuning απαιτεί συγκεκριμένου είδους δεδομένα τα οποία είναι επισημασμένα (labeled) και απαιτεί και Datasets τα οποία είναι κατάλληλα διαμορφωμένα για εκπαίδευση ενός γλωσσικού μοντέλου [26]. Αυτές οι ιδιαιτερότητες καθιστούν το Fine Tuning μία πιο οικονομικά κοστοβόρα στρατηγική. Ενώ ταυτόχρονα αυτή η μέθοδος απαιτεί και χρόνο και αρκετούς υπολογιστικούς πόρους για την διαδικασία εκπαίδευσης του γλωσσικού μοντέλου.

Συνολικά και οι δύο στρατηγικές με τις κατάλληλες τροποποιήσεις μπορούν να αξιοποιηθούν σωστά για να χρησιμοποιηθούν σε κάποιον Ρομποτικό Σύμβουλο Σπουδών ή σε παρόμοιες εφαρμογές. Όμως η υλοποίηση μίας τέτοιου είδους διπλωματικής με χρήση της στρατηγικής του RAG συστήματος παρέχει συνολικά μεγαλύτερη ευελιξία χάρις στην πιο εύκολη ενημέρωση της βάσης γνώσης. Παρέχει μικρότερο κόστος, τόσο υπολογιστικά όσο αντίστοιχα και οικονομικά. Υπερτερεί παράλληλα στο γεγονός ότι η δομή των δεδομένων του είναι λιγότερο περιορισμένη και μπορεί να βασιστεί σε δεδομένα ελεύθερου κειμένου. Ενώ από την άλλη το Fine Tuning χρειάζεται κατάλληλη επεξεργασία των δεδομένων ώστε να δοθούν στο γλωσσικό μοντέλο με την απαιτούμενη μορφή. Επιπρόσθετα το Fine Tuning είναι πολύ πιο χρονοβόρο στην εκπαίδευση του. Αυτά τα παραπάνω χαρακτηριστικά διετέλεσαν τους σημαντικότερους λόγους ώστε να προτιμηθεί μία υλοποίηση με χρήση της

στρατηγικής RAG έναντι του Fine Tuning. Η διπλωματική εργασία με θέμα έναν ρομποτικό σύμβολο σπουδών είναι μία εργασία η οποία λαμβάνει σε ύψιστη σημασία τέτοια στοιχεία καθώς πέρα από την αλληλεπίδραση μεταξύ ανθρώπου και ρομπότ. Είναι μία εργασία η οποία πρέπει να είναι ανεπηρέαστη ως προς τον χρόνο όσο το δυνατόν πιο εφικτό είναι αυτό το στοιχείο. Τα βασικά πλεονεκτήματα και μειονεκτήματα των δύο προσεγγίσεων συνοψίζονται στον Πίνακα 3.1.

Πίνακας 3.1: Πίνακας Σύγκρισης Fine Tuning και RAG.

Κριτήριο	Fine-Tuning	RAG
Ενημέρωση γνώσης	Χρειάζεται retraining	Απλή ενημέρωση εγγράφων
Κόστος	Υψηλό	Μικρό
Hallucinations	Περισσότερες πιθανότητες	Μειωμένες
Πηγή πληροφορίας	Δύσκολη	Υπάρχει
Ελληνικά δεδομένα	Απαιτεί dataset	Αρκούν τα έγγραφα
Χρόνος υλοποίησης	Μεγάλος	Μικρός

Οι προσεγγίσεις διαχείρισης γνώσης συγκρίνονται στον Πίνακα 3.2 ως προς την ενημέρωση της γνώσης, τον έλεγχο των ψευδαισθήσεων, το υπολογιστικό κόστος, τις απαιτήσεις δεδομένων και την ιδανική χρήση τους.

Πίνακας 3.2: Πίνακας Σύγκριση προσεγγίσεων διαχείρισης γνώσης σε μοντέλα τεχνητής νοημοσύνης

Τύπος Γνώσης	Μη παραμετρική (Εξωτερικά έγγραφα)	Παραμετρική (Βάση μοντέλου)	Συνδυασμός και των δύο
Ενημέρωση Γνώσης	Δυναμική & άμεση χωρίς επανεκπαίδευση	Στατική & δαπανηρή (χρειάζεται retraining)	Δυναμική με περιοδικό συντονισμό
Έλεγχος Ψευδαισθήσεων	Υψηλός (θεμελίωση σε πηγές)	Χαμηλότερος (κίνδυνος fabrication)	Βέλτιστος έλεγχος και ακρίβεια
Υπολογιστικό Κόστος	Χαμηλό αρχικό κόστος (embeddings)	Υψηλό κόστος εκπαίδευσης & δεδομένων	Πολύ υψηλό (συνδυαστικό κόστος)
Απαιτήσεις Δεδομένων	Απλά κείμενα/έγγραφα	Επισημασμένα QA ζεύγη (High-cost)	Έγγραφα & επισημασμένα δεδομένα
Ιδανική Χρήση	Πληροφορίες εισαγωγής, κανονισμοί	Εξειδικευμένο ύφος συμβούλου, ορολογία	Σύνθετη ακαδημαϊκή υποστήριξη

3.3 Επιλογή Αρχιτεκτονικής Συστήματος Retrieval-Augmented Generation

3.3.1 Εισαγωγή

Στο προηγούμενο υποκεφάλαιο αναλύθηκαν οι λόγοι για τους οποίους προτιμήθηκε η στρατηγική των RAG συστημάτων έναντι της στρατηγικής του Fine Tuning. Τα πλεονεκτήματα και τα μειονεκτήματα της κάθε μεθόδου.

Στο παρόν υποκεφάλαιο θα αναλυθούν οι διάφορες στρατηγικές των συστημάτων RAG. Θα γίνει η κατάλληλη περιγραφή αυτών όχι απλά θεωρητικά αλλά έχοντας ως πρίσμα την χρήση των διαφόρων στρατηγικών για μία διπλωματική εργασία με κεντρικό θέμα την υλοποίηση ενός ρομποτικού συμβούλου σπουδών.

Ο στόχος της συγκεκριμένης ενότητας είναι να προβληθούν κατάλληλα οι διάφορες υλοποιήσεις και στρατηγικές του RAG. Οι οποίες μετέπειτα θα χρησιμοποιηθούν έμμεσα, δηλαδή κάποια συγκεκριμένα χαρακτηριστικά αυτών για κάποια συγκεκριμένη λειτουργία του RAG. Ουσιαστικά, δηλαδή θα εξεταστούν 3 είδη στρατηγικών των RAG συστημάτων. Η στρατηγική του Naive RAG, του Advanced RAG, του Modular RAG. Στα οποία μετέπειτα θα αναλυθεί συγκεκριμένα ποιο χαρακτηριστικό αυτών χρησιμοποιήθηκε για ποιο συγκεκριμένα στάδιο υλοποίησης.

Για την καλύτερη επίδοση της διπλωματικής χρησιμοποιήθηκαν διάφορα είδη RAG ανάλογα με τις απαραίτητες ανάγκες του συστήματος. Για παράδειγμα στην διαδικασία του Chunking χρησιμοποιείται ο τρόπος για διαίρεση του κείμενο σε μικρότερα τμήματα της στρατηγικής του RAG Naive. Θα αναλύονται οι λόγοι για τους οποίους προτιμήθηκε αυτή η στρατηγική έναντι κάποιας άλλης.

3.3.2 Χαρακτηριστικά Υλοποίησης Αρχιτεκτονικών RAG

Στο μεγαλύτερο μέρος του κεφαλαίου 2 αναλύθηκαν τα χαρακτηριστικά της πιο απλής και συνηθισμένης μεθόδου RAG η οποία ονομάζεται Naive RAG ουσιαστικά αυτή η μέθοδος αποτελεί την βάση για τα υπόλοιπα είδη συστημάτων RAG. Οι υπόλοιπες μέθοδοι που χρησιμοποιούνται απλά προσθέτουν διάφορα στοιχεία στις τεχνικές όπου ορίζονται και προυπάρχουν στην στρατηγική του Naive RAG. Η τελική αρχιτεκτονική του συστήματος του ρομποτικού συμβούλου σπουδών ανήκει στα είδη του RAG τα οποία θα παρουσιάσουμε και στον συνδυασμό αυτών. Δηλαδή αποτελείται από δύο βασικά τμήματα. Από αυτό της Ανάκτησης της πληροφορίας. Το οποίο περιέχει την δημιουργία βάσης γνώσης, το chunking, το embedding, το retrieval. Και από το στάδιο της παραγωγής αποτελέσματος, το οποίο συγκεντρώνει το συνολικό πλαίσιο το οποίο δημιουργήθηκε προηγουμένως και παράγει μία απάντηση.

Ως πιο εξελιγμένο στάδιο αρχιτεκτονικής του RAG συστήματος μετά το Naive RAG ακολουθεί η τεχνοτροπία του Advanced RAG η οποία δημιουργήθηκε για την αντιμετώπιση ορισμένων περιορισμών που παρουσιάζουν τα Naive RAG ως στρατηγική υλοποίησης. Ένα στάδιο στο οποίο τα Advanced RAG παρουσιάζουν βελτιώσεις είναι η πιο εξεζητημένη και λεπτομερής ανάκτηση. Συγκεκριμένα παρέχουν επιπλέον δυνατότητες και για αραιή ανάκτηση (sparse retrieval). Η αραιή ανάκτηση ουσιαστικά βασίζεται στην αντιστοίχιση λέξεων κλειδιών με όρους που χρησιμοποιούνται στο εκάστοτε query της απάντησης[22]. Ουσιαστικά στα Advanced RAG χρησιμοποιείται ένα υβριδικό σύστημα ανάκτησης. Σε αυτό το σύστημα χρησιμοποιούνται παράλληλα και η μέθοδος αραιής ανάκτησης αλλά και η μέθοδος της πυκνής ανάκτησης. Η υλοποίηση της αραιής ανάκτησης υλοποιείται με την συνάρτηση βαθμολόγησης BM25. Αυτή η συνάρτηση βαθμολόγησης δέχεται ένα αίτημα σαν είσοδο. Ύστερα ψάχνει όλα τα chunks και βαθμολογεί υψηλότερα αυτά που περιέχουν

όρους του αιτήματος. Ως πυκνή ανάκτηση ονομάζουμε τον τρόπο με τον οποίο ανακτούνται τα Chunks με την χρήση των Embeddings.

Για ένα Advanced RAG η παρουσία της υβριδικής αναζήτησης (Hybrid Search) δηλαδή ο συνδυασμός αραιής και πυκνής αναζήτησης απαιτεί και ένα είδος συγχώνευσης. Όπου η πυκνή και η αραιή ανάκτηση συγχωνεύονται μέσω μίας μεθόδου η οποία λέγεται Reciprocal Rank Fusion (RRF). Έπειτα στο συγκεκριμένο σύστημα εφαρμόζεται και ένα στάδιο επανακατάταξης (rerank) των ανακτημένων τμημάτων κειμένου. Το reranking βασίζεται στην συγχώνευση του RRF, στην προτεραιότητα των πηγών και στο ταίριασμα κριτηρίων με βάση την θεματολογία και την σημασία του εγγράφου. Τέλος εφαρμόζεται και αποκοπή όλων των chunks των οποίων δεν συγκεντρώνουν ικανή βαθμολογία και ενώ ανακτούνται μέσα στο σύνολο chunks δεν μεταφέρονται σε μετέπειτα στάδια του RAG. Η υλοποίηση της συγκεκριμένης διπλωματικής δεν χρησιμοποιεί κάποιο rerank το οποίο να είναι βασισμένο σε νευρωνικό δίκτυο, λεγόμενο και ως cross-Encoder παρά μόνο μία έξτρα αναβαθμολόγηση τηρώντας κανόνες.

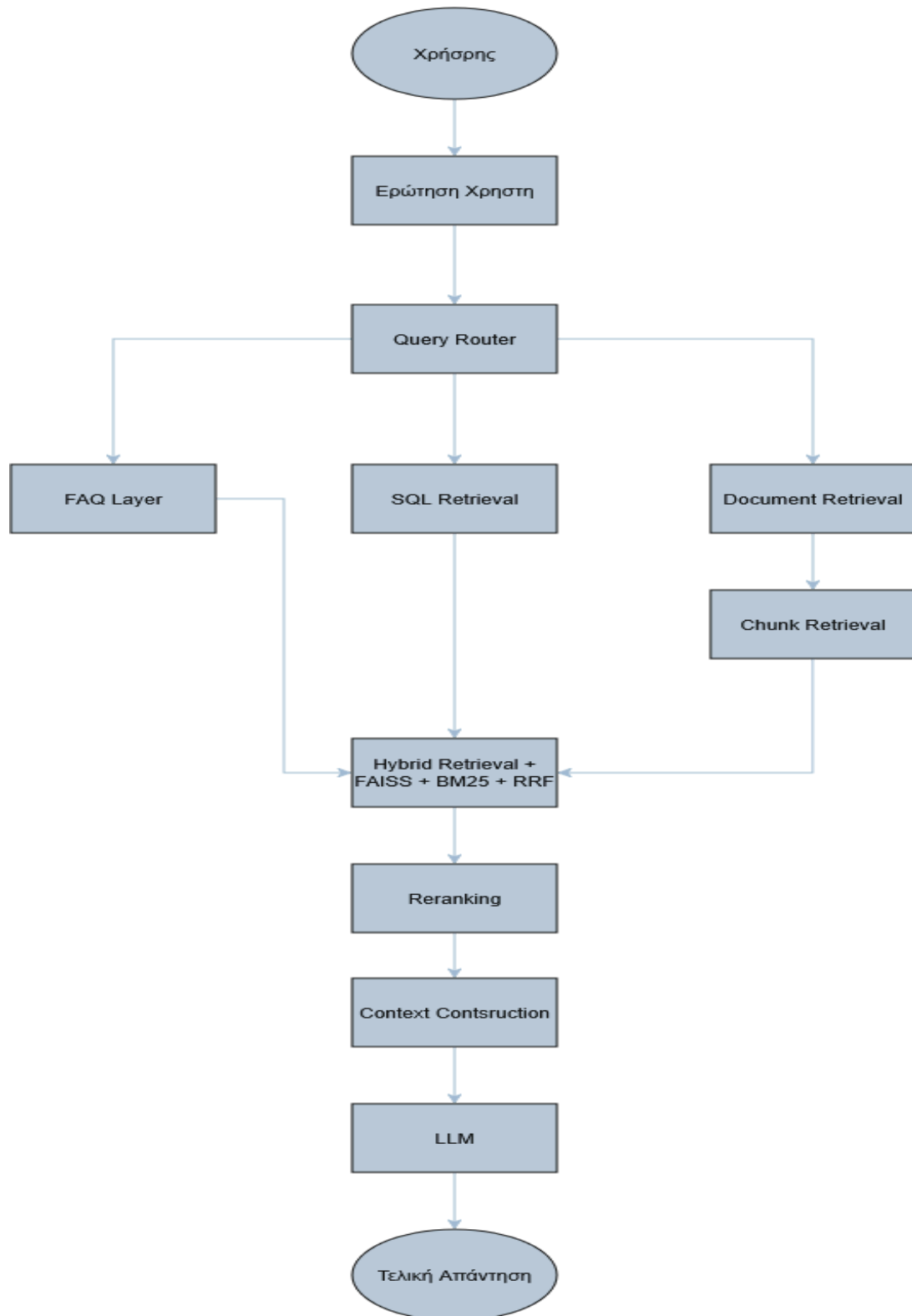
Η παραπάνω επιλογή υλοποίησης έγινε με σκοπό μίας ελαφρύτερης εκδοχής επανακατάταξης των chunks η οποία προσφέρει υψηλότερες ταχύτητες ανάκτησης. Στην παρούσα διπλωματική εργασία αυτό το χαρακτηριστικό είναι ιδιαίτερα σημαντικό καθώς επιτρέπει γρήγορες ανακτήσεις, διατηρώντας την αξιοπιστία ενώ παράλληλα επιτρέπει και την καλύτερη δυνατή αλληλεπίδραση μεταξύ ανθρώπου και ρομπότ. Αυτό συμβαίνει διότι ο χρήστης λαμβάνει απάντηση σε σύντομο χρονικό διάστημα, χαρακτηριστικό το οποίο προσδίδει ρεαλιστικότητα στην επικοινωνία.

Επιπρόσθετα εξετάστηκαν πολλά επιπλέον χαρακτηριστικά με σκοπό την επίτευξη μεγαλύτερης ευελιξίας και εξειδίκευσης του συστήματος. Η συνεχής προσθήκη επιπρόσθετων ιδιοτήτων συνολικά οδήγησε στην νέα αρχιτεκτονική των Modular RAG. Ουσιαστικά το Modular RAG είναι μία εξέλιξη των Advanced RAG συστημάτων στο οποίο ανήκουν κάποια επιπρόσθετα στοιχεία τα οποία προσαρμόζουν κατάλληλα τα συστήματα για τις εκάστοτε ανάγκες [22].

Στην υλοποίηση της συγκεκριμένης διπλωματικής εργασίας χρησιμοποιήθηκε το χαρακτηριστικό στοιχείο της κατάλληλης δρομολόγησης των αιτημάτων χρήστη, το οποίο είναι γνωστό και ως Query Routing. Το Query Routing είναι μια διαδικασία κατά την οποία ένα σύστημα, συχνά ένας ξεχωριστός πράκτορας, αναλύει το ερώτημα του χρήστη και το κατευθύνει στην κατάλληλη πηγή δεδομένων, όπου είναι πιθανό να βρίσκεται η σχετική απάντηση [27]. Αυτό το οποίο επιτυγχάνει αυτή η συγκεκριμένη ιδιότητα είναι η δρομολόγηση των ερωτημάτων ανάλογα με την κατηγορία στην οποία ανήκουν. Επιλέγεται δηλαδή η καταλληλότερη πηγή απάντησης, ώστε μετά η αναζήτηση να είναι πιο στοχευμένη. Η υλοποίηση αυτού του Module, αυτού του επιπλέον χαρακτηριστικού, στην συγκεκριμένη Διπλωματική Εργασία χρησιμοποιείται με σκοπό ανάλογα με κάποιους ορισμένους κανόνες να γίνει αναζήτηση στην κατάλληλη βάση γνώσης του συστήματος. Συγκεκριμένα οι κανόνες καθορίζουν εάν η απάντηση του ερωτήματος θα αναζητηθεί στις Βάσεις Δεδομένων SQL για τους καθηγητές ή στις βάσεις δεδομένων SQL για τα μαθήματα ή στο αρχείο με τις FAQ ερωτήσεις ή αν θα πρέπει να γίνει αναζήτηση στα chunks των αρχείων γνώσης αυτού.

Επιπλέον το σύστημα RAG το οποίο δημιουργήθηκε για την διπλωματική εργασία έχει την δυνατότητα να αντλεί δεδομένα και απαντήσεις παράλληλα και από αρχεία με FAQ (Frequently Asked Questions) αλλά και από βάσεις δεδομένων SQL. Το συγκεκριμένο χαρακτηριστικό είναι ένα στοιχείο των Modular Rag. Ουσιαστικά ενώ τα πιο απλά είδη των RAG έχουν την δυνατότητα να ανακτούν πληροφορίες μόνο από τα επεξεργασμένα αρχεία του RAG το συγκεκριμένο Module προσφέρει την δυνατότητα να χρησιμοποιούνται και επιπλέον πηγές δεδομένων, όπως είναι τα παραδείγματα τα οποία αναφέρθηκαν στην συγκεκριμένη παράγραφο.

3.3.3 Χαρακτηριστικά Υλοποίησης Αρχιτεκτονικών RAG



Σχήμα 3.1: Συνολική αρχιτεκτονική του υλοποιημένου RAG συστήματος

Το σχήμα 3.1 παρουσιάζει την συνολική αρχιτεκτονική η οποία υλοποιήθηκε για το RAG σύστημα του Ρομποτικού συμβούλου σπουδών. Συνολικά περιέχει τρία είδη αρχιτεκτονικών RAG. Περιέχει χαρακτηριστικά του είδους RAG Naïve, του είδους RAG Advanced και του Modular RAG. Από κάθε

τέτοια στρατηγική χρησιμοποιεί στοιχεία τα οποία βοηθάνε στην συνολική υλοποίηση ενός RAG κατάλληλου για την εφαρμογή ενός ρομποτικού συμβούλου. Η συνολική στρατηγική δεν ταιριάζει με κάποιο συγκεκριμένο είδος αλλά είναι μία υβριδική υλοποίηση. Κατά αυτόν τον τρόπο ανάλογα με τις απαιτήσεις χρησιμοποιούνται διαφορετικά στοιχεία από την εκάστοτε μέθοδο.

Ουσιαστικά η υλοποίηση σπάει σε τρία διαφορετικά επίπεδα. Αυτά τα μέρη αποτελούνται με την σειρά τους από επιπλέον επιμέρους τμήματα.

- **Επίπεδο 1 Query Processing και Routing:** Σε αυτό το επίπεδο το κάθε αίτημα του χρήστη κατευθύνεται σε μία επιπλέον μέθοδο η οποία αποφασίζει την πηγή στην οποία θα κάνει αναζήτηση των σχετικών πληροφοριών. Το συγκεκριμένο χαρακτηριστικό ανήκει στην στρατηγική των Modular RAG. Για παράδειγμα αν κάποια ερώτηση αφορά Καθηγητές τότε το σύστημα δεν ψάχνει την πληροφορία ούτε στα chunks και στα embeddings τους μέσα σε μία διανυσματική βάση δεδομένων, ούτε ψαχνει το αρχείο FAQ αλλά ψάχνει τις SQL βάσεις δεδομένων. Στο συγκεκριμένο επίπεδο πρώτα θα γίνει έλεγχος για το αν το αίτημα του χρήστη κατατάσσεται στις ερωτήσεις που βρίσκονται στην βάση δεδομένων για τα μαθήματα, μετά θα γίνει αναζήτηση στη βάση δεδομένων για τους καθηγητές. Έπειτα ελέγχεται εάν η ερώτηση ανήκει και ταιριάζει στο FAQ. Ενώ αν δεν ταιριάζει πουθενά από τις παραπάνω κατηγορίες τότε ακολουθείται η κανονική ροή του RAG συστήματος. Η διαδικασία Query Processing και Routing, καθώς και η σειρά ελέγχου των επιμέρους πηγών, παρουσιάζονται στο Snippet 1.

Snippet 1: Query Processing και Routing: Μέθοδος Δρομολόγησης των ερωτημάτων του χρήστη ανάλογα με την κατηγορία πηγής στην οποία βρίσκεται η απάντηση.

```
#Έλεγχος εάν το ερώτημα αφορά πληροφορίες μαθημάτων
structured_course_result= self.structured_courses.answer_course_query_structured(q,self.db_path,)

if structured_course_result is not None:
    ...
# Έλεγχος εάν το ερώτημα αφορά πληροφορίες καθηγητών
if self.structured.should_route_to_staff_db(q):
    ...
# Έλεγχος αντιστοίχισης με τη βάση συχνών ερωτήσεων (FAQ)
faq_match = self.faq_searcher.search(q)

if faq_match is not None:
    ...
# Εάν δεν υπάρχει αντιστοίχιση, ακολουθείται η διαδικασία ανάκτησης RAG
res = self.retrieval.retrieve(...)
```

- **Επίπεδο 2 Στάδιο ανάκτησης πληροφορίας:** Αυτό είναι το επίπεδο στο οποίο αν μία ερώτηση δεν αντιστοιχηθεί σε FAQ ή σε ερωτήσεις που βρίσκεται η πληροφορία στις βάσεις δεδομένων καθηγητών ή μαθημάτων. Σε αυτό το στάδιο χρησιμοποιούνται τα στοιχεία του Advanced RAG και του Modular RAG. Συγκεκριμένα στο επίπεδο αυτό συναντάμε την υβριδική μέθοδο ανάκτησης. Μία μέθοδος ανάκτησης η οποία συνδυάζει ανάκτηση δεδομένων μέσω των embeddings, τα οποία αποθηκεύονται σε μία διανυσματική βάση δεδομένων, με την ανάκτηση με βάση λέξεις ή φράσεις κλειδιά μεταξύ του αιτήματος του χρήστη και των chunks που έχουν δημιουργηθεί. Η ανάκτηση

αυτή αναφέρθηκε εκτενότερα στο προηγούμενο υποκεφάλαιο. Τέλος για την δημιουργία καλύτερου κοινού πλαισίου, το οποίο θα πάει στο LLM, χρησιμοποιείται η μέθοδος RRF όπου ενώνει αραιό και πυκνό retrieval. Η κλήση της διαδικασίας ανάκτησης και οι βασικές παράμετροι που χρησιμοποιούνται παρουσιάζονται στο Snippet 2.

Snippet 2: Στάδιο ανάκτησης πληροφορίας: Στο συγκεκριμένο τμήμα κώδικα προβάλλεται ο τρόπος κλήσης της διαδικασίας ανάκτησης.

```
res = self.retrieval.retrieve(
    q, # το κείμενο της ερώτησης (query)
    self.model, # το φορτωμένο embedding model (για embed_query)
    self.faiss_idx, # FAISS index ή None (για γρήγορη dense αναζήτηση)
    self.embeddings, # numpy embeddings fallback αν δεν υπάρχει FAISS
    self.meta, # λίστα metadata (χαρτογράφηση index -> έγγραφο)
    bm25_index=self.bm25_idx, # προαιρετικός BM25 index για sparse search
    dense_top_n=20, # πόσα dense candidates να κρατηθούν πριν fusion
    sparse_top_n=20, # πόσα sparse (BM25) candidates να ληφθούν
    rrf_k=60, # παράμετρος k για reciprocal rank fusion
```

- Επίπεδο 3 Στάδιο παραγωγής πληροφορίας ως απάντηση: Στο παρόν επίπεδο συμβαίνει η δημιουργία της ολοκληρωμένης απάντησης του χρήστη. Σε αυτό το επίπεδο λαμβάνονται από το Επίπεδο 2 τα κατάλληλα chunks αφότου έχουν γίνει όλες οι διαδικασίες του επιπέδου αυτού. Αυτά συμπλέκονται μεταξύ τους και δημιουργούν τελικά, ένα ενιαίο πλαίσιο πληροφορίας. Μετέπειτα ακολουθεί η συμμετοχή του LLM το οποίο με βάση κάποιες οδηγίες δημιουργεί την απάντηση η οποία θα μεταφερθεί στο επόμενο στάδιο ώστε να παρουσιαστεί κατάλληλα από το NAO V6. Σε περίπτωση στην οποία τα chunks δεν παρέχουν αρκετή πληροφορία ή κατάλληλη με το ερώτημα για να απαντηθεί υπάρχει μία διαδικασία fallback στην οποία παραδίδεται μία απάντηση φιλική προς το χρήστη όπου τον ενημερώνει για αυτό το γεγονός.

Η δημιουργία του prompt, στο οποίο συνδυάζονται το πλαίσιο που προέκυψε από τη διαδικασία ανάκτησης και το ερώτημα του χρήστη, παρουσιάζεται στο Snippet 3.

Snippet 3: Στάδιο παραγωγής πληροφορίας: Στο συγκεκριμένο τμήμα κώδικα εμφανίζεται ο τρόπος δημιουργίας του Prompt, με ολόκληρο το πλαίσιο που χρησιμοποιείται για αυτόν τον σκοπό. Δηλαδή φαίνεται η συγχώνευση του πλαισίου πληροφορίας το οποίο δημιούργησε η ανάκτηση στο προηγούμενο στάδιο, με το ερώτημα το οποίο έκανε ο χρήστης. Τέλος το Prompt με βάση τα συγκεκριμένα στοιχεία καλείται να παράγει μία απάντηση

```
def build_prompt_from_context(context: str, query: str) -> str:
    return ( "### CONTEXT\n"
    f"{context}\n\n"
    "### QUESTION\n"
    f"{query}\n\n"
    "### ANSWER\n" )
```

Η κλήση του LLM και οι ρυθμίσεις που χρησιμοποιούνται για την παραγωγή της τελικής απάντησης παρουσιάζονται στο Snippet 4.

Snippet 4: Στάδιο παραγωγής πληροφορίας: Το συγκεκριμένο κομμάτι κώδικα ακολουθεί αυτό του Snippet 3 και περιέχει τις κατάλληλες ρυθμίσεις για να σταλεί στο LLM. Το συγκεκριμένο στάδιο είναι το τελευταίο βήμα του σταδίου παραγωγής πληροφορίας.

```
answer = openai_chat(  
    self.api_base_url, #To URL του API endpoint  
    api_key, #To token για το endpoint  
    self.model_name, #To μοντέλο LLM το οποίο θα χρησιμοποιηθεί  
    prompt, #To prompt το οποίο φτιάχτηκε στο Snippet 3  
    temperature=0.0, #Καθορίζει την δημιουργικότητα της απαντησης με 0.0 να σημαίνει λιγότερη  
    δημιουργικότητα  
    system_prompt=STRICT_RAG_SYSTEM_PROMPT, #Καθορίζει τους κανόνες με τους  
    οποίους παράγει το LLM την απάντηση  
).strip() #Αφαιρεί κενούς χαρακτήρες από την αρχή και το τέλος της απάντησης
```

3.4 Επίλογος

Στο Τρίτο κεφάλαιο της διπλωματικής εργασίας εξηγήθηκε ο τρόπος με τον οποίο επιλέχθηκε το RAG έναντι άλλων εκδοχών στην συγκεκριμένη υλοποίηση αλλά και ο τρόπος με τον οποίο υλοποιήθηκε το RAG σύστημα συνολικά. Ουσιαστικά σε αυτό το κεφάλαιο αναλύονται τα βασικά θεμέλια της Διπλωματικής εργασίας του Ρομποτικού Συμβούλου Σπουδών στα οποία συγκαταλέγεται το RAG σύστημα. Στο πρώτο μέρος αυτού του κεφαλαίου γίνεται τεκμηρίωση των λόγων που οδήγησαν ουσιαστικά στην επιλογή της RAG αρχιτεκτονικής ενάντια της αρχιτεκτονικής Fine Tuning. Έγινε δηλαδή μία επιμέρους ανάλυση των πλεονεκτημάτων και των μειονεκτημάτων του κάθε συστήματος, εξηγήθηκαν και αναλύθηκαν με βάση τις ανάγκες της διπλωματικής εργασίας του Ρομποτικού Συμβούλου σπουδών για το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων.

Ουσιαστικά η τελική επιλογή έγινε διότι τα πλεονεκτήματα της αρχιτεκτονικής RAG ταιριάζουν καλύτερα σε μία υλοποίηση ακαδημαϊκού περιβάλλοντος καθώς διάφορα στοιχεία , διάφορες πληροφορίες ενδέχεται να διαφοροποιηθούν. Αυτή η αλλαγή δεν είναι ένα σπάνιο φαινόμενο οπότε αποτέλεσε ένα σημείο κλειδί για την επιλογή του RAG έναντι του Fine Tuning καθώς είναι μία ειδοποιός διαφορά των δύο αρχιτεκτονικών. Επιπρόσθετα το δεύτερο αλλά εξίσου σημαντικό στοιχείο το οποίο οδήγησε στην προτίμηση της μίας αρχιτεκτονικής έναντι της άλλης είναι το σημαντικό μειονέκτημα το οποίο υπάρχει στο Fine Tuning. Αυτό το μειονέκτημα είναι η ενδεχόμενη ύπαρξη ψευδαισθήσεων κατά την παραγωγή της απάντησης. Ένα στοιχείο όπως και αυτό είναι ιδιαίτερης σημασίας καθώς επηρεάζει συνολικά την αξιοπιστία ολόκληρης της εφαρμογής.

Στο συγκεκριμένο κεφάλαιο επίσης αναλύθηκε ο τρόπος με τον οποίο έγινε η συνολική σύνθεση του RAG συστήματος. Ουσιαστικά η παρούσα υλοποίηση της αρχιτεκτονικής του RAG αποτελείται από τρία είδη διαφορετικών RAG. Συνδυάζονται δηλαδή χαρακτηριστικά της προσέγγισης Naive, Advanced και Modular RAG. Από τα Naive RAG, χρησιμοποιούνται οι τρόποι δημιουργίας της συνολικής βάσης γνώσης, κατακερματισμού των εγγράφων και δημιουργία Embeddings. Ενώ από το Advanced RAG αυτό το χαρακτηριστικό το οποίο ενσωματώθηκε στην συγκεκριμένη συνδυαστική υλοποίηση του συστήματος και είναι παράλληλα ένα από τα πιο σημαντικά για να αναφέρουμε είναι αυτό της υβριδικής αναζήτησης. Στην οποία συνδυάζεται η σημασιολογική ή αλλιώς αναφερόμενη και ως πυκνή ανάκτηση με την αραιή. Επιπλέον από το είδος Modular RAG αξιοποιήθηκε στο τελικό

σύστημα ο μηχανισμός της δρομολόγησης των ερωτημάτων. Κατά αυτήν την διαδικασία το σύστημα διαλέγει ουσιαστικά την καταλληλότερη πηγή γνώσης για να απαντηθεί το εκάστοτε ερώτημα του χρήστη της εφαρμογής.

Τέλος στο παρόν κεφάλαιο έγινε και μία αναφορά στον τρόπο και στην μέθοδο με την οποία παράγεται τελικά από το μεγάλο γλωσσικό μοντέλο η απάντηση του χρήστη. Επηρεαζόμενη άμεσα από την διαδικασία ανάκτησης του RAG και από διάφορες οδηγίες στο prompt. Με στόχο την παραγωγή μιας απάντησης η οποία συμπυκνώνει όλα τα χαρακτηριστικά αυτά και τα μεταφέρει προς παρουσίαση στο ρομπότ NAO V6. Κατά αυτόν τον τρόπο δημιουργείται ένας πλήρης κύκλος αλληλεπίδρασης μεταξύ ανθρώπων και ρομπότ αξιοποιώντας χαρακτηριστικά όπως η αξιοπιστία, η ταχύτητα, η δυνατότητα επεκτασιμότητας και συντήρησης.

Κεφάλαιο 4ο: Υλοποίηση Επιμέρους συστημάτων του Retrieval Augmented Generation (RAG) Συστήματος

4.1 Δημιουργία Βάση Γνώσης

Στο συγκεκριμένο υποκεφάλαιο γίνεται αναλυτική αναφορά και περιγραφή του τρόπου με τον οποίο υλοποιήθηκε η Βάση Γνώσης για την Διπλωματική Εργασία με θέμα το Ρομποτικό Σύμβουλο σπουδών. Ο στόχος της δημιουργίας της βάσης γνώσης είναι να παρέχει τα κατάλληλα εφόδια ώστε να απαντιέται το μεγαλύτερο πλήθος ερωτήσεων του χρήστη σχετικά με το θέμα της βάσης γνώσης.

Ουσιαστικά η Βάση Γνώσης χωρίζεται σε τρία μέρη. Το πρώτο μέρος αυτής αφορά τα PDF αρχεία που χρησιμοποιήθηκαν για αυτήν. Επίσης αφορά τον τρόπο προεπεξεργασίας αυτών των αρχείων πριν ακόμα καταλήξουν στην διαδικασία του Chunking. Στο δεύτερο μέρος συναντάμε τα δομημένα δεδομένα τα οποία περιέχονται στην βάση γνώσης του ρομποτικού συμβούλου σπουδών. Τα δομημένα δεδομένα της συγκεκριμένης βάσης αφορούν πληροφορίες σχετικές με καθηγητές και μαθήματα του τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Τέλος η συγκεκριμένη βάση γνώσης περιέχει και πληροφορίες σε μορφή αρχείου ερωτοαπαντήσεων. Δηλαδή υπάρχει ένα συγκεκριμένο αρχείο στο οποίο έχουν προστεθεί συγκεκριμένες ερωτήσεις οι οποίες έχουν αντιστοιχηθεί κατάλληλα με τις απαντήσεις τους. Αυτές οι ερωτήσεις είναι κάποιες όπου δεν μπορούσαν να απαντηθούν με βάση τα προηγούμενα μέρη της βάσης γνώσης.

4.1.1 Μέρος Πρώτο: Μη δομημένα δεδομένα

Σε αυτό το τμήμα της βάσης γνώσης συλλέχθηκαν όλα τα μη δομημένα κείμενα, όλα τα PDF αρχεία τα οποία αφορούν πληροφορίες οι οποίες βρίσκονται στον οδηγό σπουδών της σχολής. Όπως και στο επόμενο μέρος της βάσης γνώσης αντίστοιχα και στο πρώτο μέρος αυτής τα αρχεία τα οποία χρησιμοποιούνται χωρίζονται σε δύο τμήματα. Το ένα τμήμα αφορά πληροφορίες σχετικά με τον οδηγό σπουδών για το προπτυχιακό πρόγραμμα σπουδών. Το δεύτερο τμήμα αφορά τα κατάλληλα αρχεία για τα μεταπτυχιακά προγράμματα του τμήματος και του διδακτορικού.

Το πρώτο βήμα αυτής της διαδικασίας είναι η εξαγωγή των κειμένων από τα αρχεία PDF τα οποία χρησιμοποιούνται στην βάση γνώσης. Η υλοποίηση αυτής της διαδικασίας εξαγωγής γίνεται μέσω της βιβλιοθήκης PyMuPDF. Αφού συλλέξει όλα τα τμήματα κειμένων τα οποία υπάρχουν στα PDF ακολουθεί το στάδιο της προεπεξεργασίας.

Τα αρχεία αυτά πριν από το στάδιο του chunking δέχονται ενός είδους φιλτράρισμα, αυτό το φιλτράρισμα είναι ενός είδους προεπεξεργασίας η οποία γίνεται με σκοπό να καθαριστεί το εκάστοτε κείμενο των αρχείων για να περιέχονται σε αυτό μόνο χρήσιμες πληροφορίες. Η αρχή του φιλτραρίσματος γίνεται με την κανονικοποίηση Unicode NFKC, στην συγκεκριμένη κανονικοποίηση οι διαφορετικές αναπαραστάσεις χαρακτήρων μετατρέπονται σε μία ενιαία μορφή. Παράλληλα εφαρμόζονται και πιο απλού είδους κανονικοποιήσεων. Μία τέτοιου είδους κανονικοποίηση είναι αυτή η οποία είναι υπεύθυνη για την μετατροπή των κεφαλαίων χαρακτήρων σε πεζούς. Επιπλέον τα κείμενα δέχονται και ένα Trim, δηλαδή τα κένα τα οποία είναι περιττά από την αρχή και το τέλος μιας πρότασης αφαιρούνται με σκοπό την μείωση του όγκου της πληροφορίας από τα κειμενικά αντικείμενα, τα οποία θα προκύψουν μετέπειτα της προεπεξεργασίας.

Παράλληλα μία επιπλέον διαδικασία η οποία χρησιμοποιείται με σκοπό την καλύτερη και πιο ουσιαστική δημιουργία κειμένων από τα αρχεία PDF είναι το Hyphenation correction. Σε αυτή την διαδικασία υλοποιείται η ένωση δύο λέξεων οι οποίες δεν χωρίζονται πραγματικά και νοηματικά αλλά χωρίζονται κειμενικά. Δηλαδή αν μία λέξη χωρίζεται ενδιάμεσα από μία παύλα (-), διότι για παράδειγμα τα μέρη αυτής διαχωρίζονται από μία σειρά στην επόμενη. Τότε αυτή η μέθοδος χρησιμοποιείται ώστε τα δύο αυτά μέρη να ενωθούν. Επιπρόσθετα ένα από τα τελικά αλλά εξίσου σημαντικά στάδια προεπεξεργασίας είναι αυτό της αφαίρεσης θορύβου. Σε αυτό το στάδιο αφαιρούνται τα κομμάτια κειμένου, όπως είναι για παράδειγμα η αρίθμηση μίας σελίδας, τα οποία δεν έχουν πραγματικό νόημα στην εξαγωγή κάποιου κειμένου, ειδικά εφόσον αυτό τεμαχίζεται μετέπειτα σε επιπλέον κομμάτια. Η διαδικασία εξαγωγής κειμένου από τα PDF μέσω της βιβλιοθήκης PyMuPDF παρουσιάζεται στο Snippet 5.

Snippet 5 Εξαγωγή κειμένων με PyMuPDF: Στο συγκεκριμένο κομμάτι κώδικα φαίνεται ο τρόπος με τον οποίο γίνεται η εξαγωγή της κειμενικής πληροφορίας από τα PDF.

```
def extract_pdf_text(pdf_path: Path) -> str:
    doc = fitz.open(str(pdf_path))
    parts = []
    for page in doc:
        parts.append(page.get_text("text"))
    doc.close()
    raw_text = "\n".join(parts)
    return normalize_text(raw_text)
```

Στο Snippet 6 παρουσιάζεται η εφαρμογή των βασικών βημάτων προεπεξεργασίας. Αρχικά, το κείμενο μετατρέπεται σε πεζούς χαρακτήρες. Στη συνέχεια, ενώνονται οι λέξεις που χωρίστηκαν λόγω αλλαγής γραμμής, αφαιρούνται οι κενές γραμμές και απομακρύνεται η αρίθμηση των σελίδων.

Snippet 6 Εξαγωγή κειμένων με PyMuPDF: Στο συγκεκριμένο κομμάτι κώδικα φαίνεται ο τρόπος με τον οποίο γίνεται η κανονικοποίηση Unicode, το Hyphenation και το Noise Removal.

```
def normalize_text(text: str) -> str:
    text = text.lower() # Μετατρέπει όλους του χαρακτήρες σε πεζούς

    # Ενώνει τις λέξεις που χωρίστηκαν στα PDF
    text = re.sub(r"(\w+)-\s+(\w+)", r"\1\2", text)

    lines = text.splitlines() # Χωρίζει το κείμενο σε γραμμές
    norm_lines = []

    for line in lines:
        line = line.strip() # Αφαιρεί τα κενά από την αρχή και το τέλος μίας γραμμής

        if not line: # Αφαιρεί κενές γραμμές
            continue

        # Αφαιρεί την αρίθμηση των σελίδων
        if re.fullmatch(r"\d+", line):
            continue

        norm_lines.append(line) # Κρατάει τις καθαρές γραμμές
```

```
text = "\n".join(norm_lines) # Ενώνει ξανά τις καθαρές γραμμές

# Επιστρέφει το τελικό κείμενο και αφαιρεί τα κενά από την αρχή και το τέλος
return text.strip()
```

4.1.2 Μέρος Δεύτερο: Δομημένα δεδομένα

Το δεύτερο μέρος της βάσης γνώσης του RAG συστήματος αποτελείται από τα δομημένα δεδομένα. Δηλαδή από τις Βάσεις Δεδομένων SQL οι οποίες περιέχουν πίνακες με σχετικές πληροφορίες. Η ύπαρξη των δομημένων δομών δεδομένων αποσκοπεί να καλύψει τις ερωτήσεις οι οποίες είναι πιο συγκεκριμένες και αφορούν μαθήματα ή καθηγητές. Ουσιαστικά όταν ο χρήστης αναφέρει κάποιες ορισμένες λέξεις κλειδιά τότε η αναζήτηση γίνεται ταχύτατα στις βάσεις δεδομένων SQL. Με τον τρόπο αυτό δεν υπάρχει η ανάγκη των Chunking, Embeddings και η αναζήτηση σε κάποια διανυσματική βάση δεδομένων, διαδικασίες οι οποίες μπορούν να είναι πιο χρονοβόρες.

Τα συγκεκριμένα δεδομένα των πινάκων τα οποία περιγράφονται παρακάτω έχουν ληφθεί από την ιστοσελίδα της σχολής. Η δομή του εκάστοτε πίνακα ακολουθεί μία συγκεκριμένη μορφή. Αυτή η συγκεκριμένη μορφή επιλέχθηκε με στόχο αυτές οι πληροφορίες να μπορούν να δοθούν μετέπειτα στο LLM με τέτοιο τρόπο ώστε να απαντηθεί το εκάστοτε ερώτημα με τον ανάλογο τρόπο. Υπάρχει βέβαια και η επιλογή να μην χρησιμοποιηθεί καθόλου το LLM και η απάντηση να προέλθει μέσα από ένα συγκεκριμένο και προκαθορισμένο τρόπο απάντησης. Αυτό το είδος επιλογής προστέθηκε για δύο λόγους. Ο πρώτος λόγος είναι η ανάγκη για να μην χρειάζεται να επεμβαίνει το LLM κάθε φορά υπό το πρίσμα της μείωσης του κόστους. Ο δεύτερος λόγος είναι και για αύξηση της ταχύτητας ανάκτησης τέτοιων ερωτημάτων. Ενώ ταυτόχρονα μπορεί να χρησιμοποιηθεί και ως fallback σε περιπτώσεις όπου το LLM δεν είναι διαθέσιμο. Η δομή του πίνακα Courses και τα επιμέρους πεδία που χρησιμοποιούνται για την αποθήκευση των πληροφοριών παρουσιάζονται στον Πίνακα 4.1. Αντίστοιχα, η δομή του πίνακα Staff και τα πεδία που χρησιμοποιούνται για την αποθήκευση των πληροφοριών των μελών ΔΕΠ παρουσιάζονται στον Πίνακα 4.2.

Πίνακας 4.1: Δομή του πίνακα Courses που χρησιμοποιείται για την αποθήκευση πληροφοριών μαθημάτων.

Πεδίο	Τύπος δεδομένων	Περιγραφή
course_id	INTEGER	Μοναδικό αναγνωριστικό μαθήματος
course_code	TEXT	Κωδικός μαθήματος
title	TEXT	Τίτλος μαθήματος
semester	TEXT	Εξάμηνο διδασκαλίας
ects	INTEGER	Πιστωτικές μονάδες ECTS
course_type	TEXT	Κατηγορία μαθήματος
course_kind	TEXT	Υποχρεωτικό ή επιλογής
knowledge_area	TEXT	Γνωστικό πεδίο
weekly_hours	TEXT	Ωρες διδασκαλίας
course_url	TEXT	Σύνδεσμος περιγραφής

Πίνακας 4.2: Δομή του πίνακα Staff που χρησιμοποιείται για την αποθήκευση πληροφοριών μελών ΔΕΠ.

Πεδίο	Τύπος δεδομένων	Περιγραφή
staff_id	INTEGER	Μοναδικό αναγνωριστικό καθηγητή
first_name	TEXT	Όνομα
last_name	TEXT	Επώνυμο
full_name	TEXT	Πλήρες ονοματεπώνυμο
academic_rank	TEXT	Ακαδημαϊκή βαθμίδα
administrative_roles	TEXT	Διοικητικές αρμοδιότητες
courses	TEXT	Μαθήματα διδασκαλίας
office_location	TEXT	Τοποθεσία γραφείου

4.1.3 Μέρος Τρίτο : Συχνές ερωτήσεις

Στο τρίτο μέρος της βάσης γνώσης χρησιμοποιείται ένα αρχείο με συγκεκριμένες ερωτήσεις και απαντήσεις. Στο συγκεκριμένο αρχείο βρίσκονται κάποιες συχνές ερωτήσεις των χρηστών οι οποίες δεν απαντώνται ούτε από τα αρχεία κειμένου ούτε και από την βάση δεδομένων η οποία έχει δημιουργηθεί. Το συγκεκριμένο αρχείο έχει μία δομή και ακολουθώντας αυτή την δομή μπορεί κάποια

επιπλέον νέα ερώτηση να προστεθεί. Η δομή του αρχείου παρουσιάζεται στο Snippet 7. Αυτό είναι ένα ιδιαίτερα χρήσιμο χαρακτηριστικό καθώς οποιαδήποτε στιγμή μπορεί μία νέα ερώτηση και να ενημερωθεί και να προστεθεί στο σύνολο των ερωτήσεων.

Snippet 7 Δομή αρχείου FAQ: Στο παρακάτω Snippet κώδικα γίνεται παρουσίαση της δομής που ακολουθεί το αρχείο το οποίο περιέχει τις συχνές ερωτήσεις μέσα από ένα παράδειγμα

```
{
  "id": 25,
  "question": "Ποιος είναι ο προεδρος της σχολής;",
  "answer": "Ο πρόεδρος της σχολής είναι ο Καθηγητής Αντωνίου Ευστάθιος.",
  "category": "Γενικές Πληροφορίες"
},
```

Η δομή του αρχείου το οποίο περιέχει τις συχνές ερωτήσεις αποτελείται από ένα μοναδικό αναγνωριστικό αριθμό το id. Από το πεδίο question στο οποίο αναγράφεται η ερώτηση την οποία μπορεί να κάνει ο χρήστης και δεν μπορεί να απαντηθεί μέσα από τις βάσεις δεδομένων SQL ή τα μη δομημένα δεδομένα των chunks. Αποτελείται επίσης από το πεδίο answer. Το πεδίο αυτό περιέχει την απάντηση της εκάστοτε ερώτησης. Ενώ τέλος περιέχει και το πεδίο Category στο οποίο αναφέρεται η κατηγορία του ερωτήματος για βοηθητικούς σκοπούς και για να μπορεί να γίνεται πιο εύκολα η κατηγοριοποίηση των ερωτήσεων.

4.2 Στρατηγική Chunking

Σε αυτό το στάδιο της Διπλωματικής εργασίας θα μπορούμε στην διαδικασία να εξηγήσουμε τον τρόπο με τον οποίο γίνεται το Chunking. Η διαδικασία του Chunking εξηγήθηκε αναλυτικά στο Δεύτερο κεφάλαιο σχετικά με τον τρόπο που λειτουργεί, το τι προσπαθεί να πετύχει αλλά και ποιες τεχνικές χρησιμοποιούνται για την επίτευξη των στόχων της διαδικασίας.

Στο συγκεκριμένο υποκεφάλαιο η εργασία θα ασχοληθεί αποκλειστικά με τον τρόπο με τον οποίο υλοποιείται η συγκεκριμένη διαδικασία στο RAG σύστημα του ρομποτικού συμβούλου σπουδών. Ουσιαστικά θα περιγραφεί η επόμενη διαδικασία μετά την προεπεξεργασία των εγγράφων αλλά και μετά από την συνολική δημιουργία της βάσης γνώσης για τον ρομποτικό σύμβουλο σπουδών.

Η διαδικασία του κατακερματισμού των αρχείων, δημιουργώντας έτσι τα Chunks, δηλαδή μικρότερων κειμενικών τμημάτων, ξεκινάει χρησιμοποιώντας την βιβλιοθήκη SpaCy. Η συγκεκριμένη βιβλιοθήκη χρησιμοποιείται με σκοπό να χωρίζονται τα κείμενα σε μικρότερα κομμάτια όχι τυχαία αλλά υλοποιώντας τεμαχισμό του κειμένου έχοντας επίγνωση των προτάσεων [28]. Επειδή το σύνολο των πλήθος των αρχείων είναι στην ελληνική γλώσσα για την καλύτερη κατανόηση του νοήματος των προτάσεων χρησιμοποιείται ειδικό μοντέλο του SpaCy κατάλληλο για την ελληνική γλώσσα. Το μοντέλο του SpaCy είναι το el_core_news_sm.

Snippet 8 Χρήση SpaCy στο Chunking: Στο παρακάτω τμήμα κώδικα φαίνεται ο τρόπος με τον οποίο γίνεται η διαδικασία του chunking με βάση το SpaCy.

```
chunks = chunk_text_with_spacy(
    text, nlp,
    max_chars=args.max_chars,
    max_words=args.max_words,
    overlap_sents=1, # <<<< set overlap to 1 sentence
    verbose=args.verbose
)
```

Στο Snippet 8 παρουσιάζεται η κλήση της συνάρτησης `chunk_text_with_spacy`, η οποία πραγματοποιεί τον κατακερματισμό του κειμένου με χρήση του μοντέλου SpaCy. Παίρνει σαν είσοδο το `text`, μετά το σπάει σε κομμάτια με την λογική του μοντέλου SpaCy. Ύστερα δημιουργεί τα `chunks` με ένα συγκεκριμένο μήκος χαρακτήρων και λέξεων. Μετέπειτα κρατάει σαν χώρο επικάλυψη μεταξύ των `chunk` μία πρόταση και τελικά επιστρέφει όλα τα `chunks`.

Η συγκεκριμένη στρατηγική `chunking` η οποία ακολουθείται στο συγκεκριμένο κομμάτι κώδικα έχει και ένα επιπλέον χαρακτηριστικό. Το χαρακτηριστικό είναι το όριο μεγέθους του κάθε `chunk`. Στην συγκεκριμένη υλοποίηση το όριο των `chunks` είναι συνολικά 300 χαρακτήρες. Το συγκεκριμένο όριο επιλέχθηκε ώστε σε κάθε `chunk` να υπάρχει ένα συγκεκριμένο μέγεθος πληροφορίας για να διατηρείται το κατάλληλο πλαίσιο και να μην συγκεντρώνεται πολύ μεγάλο μέγεθος πληροφορίας σε ένα μόνο `chunk`. Εάν ξεπεραστεί το όριο των μεγίστων χαρακτήρων σε ένα `chunk` τότε αυτό δεν κόβει το κείμενο στον χαρακτήρα νούμερο 300 του `chunk` αλλά κόβει ολόκληρη την τελευταία πρόταση η οποία προστέθηκε στο συγκεκριμένο `chunk`.

Ένα επιπλέον χαρακτηριστικό της στρατηγικής με την οποία δημιουργούνται τα `chunks` είναι το `overlap` το οποίο υπάρχει και αφορά τον χώρο επικάλυψης μεταξύ των διαδοχικών `chunks`. Η συγκεκριμένη υλοποίηση χρησιμοποιεί σαν `overlap` μία πρόταση. Δηλαδή η τελευταία πρόταση ενός `chunk` είναι η πρώτη του επόμενου. Αυτή η διαδικασία υπάρχει διότι με αυτόν τον τρόπο υπάρχουν δύο σημαντικά πλεονεκτήματα. Αρχικά διατηρείται καλύτερα το πλαίσιο πληροφορίας μεταξύ των διαδοχικών `chunks`. Ενώ παράλληλα συμβάλλει στην μείωση της πιθανότητας να υπάρξει απώλεια πληροφορίας. Διότι αν μία σημαντική πληροφορία βρίσκεται σε ενδιάμεσο σημείο μεταξύ των `chunks` τότε υπάρχει ο κίνδυνος να υπάρξει απώλεια σημαντικής πληροφορίας η οποία βρίσκεται στα όρια δύο τμημάτων.

Επιπρόσθετα στην υλοποίηση του `Chunking` δόθηκε ιδιαίτερη έμφαση στον τρόπο αλλά και στο είδος των μεταδεδομένων ή αλλιώς `metadata`. Για κάθε `chunk` αποθηκεύονται και επιπλέον στοιχεία πέρα από αυτά τα κομμάτια κειμένου που είναι το κεντρικό σημείο ενδιαφέροντος της διαδικασίας του `chunking`. Αποθηκεύονται επιπλέον το αρχείο από το οποίο είναι τμήμα του το `chunk`. Η συγκεκριμένη ενότητα στην οποία ανήκει. Ο αύξων αριθμός του `chunk`. Ο αριθμός του πρώτου και του τελευταίου χαρακτήρα του `chunk`. Ενώ επίσης αναφέρεται και η γλώσσα που χρησιμοποιείται στο συγκεκριμένο `chunk`. Η χρήση όλων των συγκεκριμένων μεταδεδομένων αφορά την βελτίωση της διαδικασίας ανάκτηση. Ουσιαστικά κάθε `chunk` πέρα από το τμήμα κειμένου το οποίο περιέχει απαραίτητα για να θεωρηθεί `chunk`, περιέχει και έξτρα πληροφορίες μέσω των επιπρόσθετων `metadata` του. Οι επιπλέον πληροφορίες οι οποίες περιέχονται δημιουργούν ένα καλύτερο πλαίσιο πληροφορίας. Αυτό το πλαίσιο πληροφορίας βοηθάει ιδιαίτερα στην βελτίωση της απόδοσης της διαδικασίας ανάκτησης. Τα μεταδεδομένα που αποθηκεύονται για κάθε `chunk` παρουσιάζονται στον Πίνακα 4.3.

Πίνακας 4.3: Περιγραφή των μεταδεδομένων που αποθηκεύονται για κάθε chunk

Πεδίο	Περιγραφή
id	Μοναδικό αναγνωριστικό του chunk μέσα στη βάση γνώσης.
text	Το πραγματικό κείμενο του chunk που προέκυψε μετά τον κατακερματισμό του εγγράφου.
retrieval_text	Το κείμενο που χρησιμοποιείται κατά την ανάκτηση. Περιλαμβάνει το περιεχόμενο του chunk και, όπου υπάρχει, την ενότητα (section) ώστε να ενισχύεται η σημασιολογική αναζήτηση.
source	Το όνομα του αρχικού αρχείου από το οποίο προήλθε το chunk.
chunk_index	Ο σειριακός αριθμός του chunk μέσα στο συγκεκριμένο αρχείο.
start_char	Η θέση (offset) του πρώτου χαρακτήρα του chunk στο αρχικό κείμενο.
end_char	Η θέση (offset) του τελευταίου χαρακτήρα του chunk στο αρχικό κείμενο.
section	Η ενότητα (επικεφαλίδα) του εγγράφου στην οποία ανήκει το chunk, εφόσον αυτή έχει εντοπιστεί κατά την προεπεξεργασία.
language	Η γλώσσα του κειμένου (στην παρούσα εργασία χρησιμοποιείται η τιμή el για τα ελληνικά).

Snippet 9 Χρήση Δημιουργία Start-End Char Metadata: Παρουσίαση κώδικα υπεύθυνο για την δημιουργία των metadata Star και End Char.

```
for i in range(0, len(text), batch_size): #Ανά batch size δημιουργούνται διαδοχικές θέσεις
    part = text[i:i+batch_size] #Παίρνει το εκάστοτε κομμάτι του κειμένου, ανάλογα με το i
    doc = nlp(part) #Χρήση SpaCy στο κομμάτι και το αποθηκεύει στο doc
    for sent in doc.sents: # Η λούπα επαναλαμβάνεται για κάθε πρόταση που αναγνωρίστηκε
        raw_text = sent.text # Εδώ βρίσκεται το κείμενο της πρότασης όπως βγήκε από το SpaCy
        ...
    # Το sent.start_char είναι η θέση μέσα στο κομμάτι κειμένου,
    # i η θέση από όπου ξεκινάει το κομμάτι part στο αρχικό έγγραφο
    # leading_ws ο αριθμός κενών χαρακτήρων στην αρχή της πρότασης.
    start = i + sent.start_char + leading_ws
    end = i + sent.end_char - trailing_ws
    # Αποθηκεύει την πρόταση και τις θέσεις σε μία λίστα
    sentences.append({"text": stripped_text, "start": start, "end": end})
```

Στο Snippet 9 παρουσιάζεται ο υπολογισμός των θέσεων αρχής και τέλους κάθε πρότασης στο αρχικό κείμενο.

Snippet 10 Χρήση Δημιουργία Ενότητας Metadata: Ο υπεύθυνος κώδικας για την δημιουργία του metadata για την ενότητα στην οποία ανήκει κάθε Chunk.

```
for sentence in raw_sentences:
    section_header, remainder, consumed_chars = extract_section_prefix(sentence["text"])
    if section_header:
        current_section = section_header
    ...
    start = sentence["start"] + consumed_chars
    ...
    content_units.append({
        "text": remainder.strip(),
        "section": current_section,
        "start": start,
        "end": sentence["end"],
        "is_bullet": is_bullet_sentence(remainder),
    })
```

Στο Snippet 10 παρουσιάζεται η διαδικασία εντοπισμού και αποθήκευσης της ενότητας στην οποία ανήκει κάθε τμήμα κειμένου.

Snippet 11 Συνολική δημιουργία όλων των Metadata Ο συγκεκριμένος κώδικας είναι αυτός ο οποίος είναι υπεύθυνος για την συνένωση και την τελική δημιουργία όλων των metadata τα οποία χρειάζεται ένα chunk.

```
for i, chunk_data in enumerate(chunks):
    ...
    chunk_preview.append({
        "id": chunk_id,
        "text": chunk,
        "retrieval_text": chunk_data["retrieval_text"],
        "source": txt_file.name,
        "chunk_index": i + 1,
        "start_char": chunk_data["start"],
        "end_char": chunk_data["end"],
        "section": section_title,
        "language": "el"
    })
```

Στο Snippet 11 παρουσιάζεται η τελική δημιουργία των μεταδεδομένων που αποθηκεύονται για κάθε chunk.

Ο λόγος για τον οποίο επιλέχθηκε η συγκεκριμένη στρατηγική για το Chunking είναι για να καλύψει τις ανάγκες του ρομποτικού συμβούλου σπουδών. Επειδή μία τέτοια εφαρμογή περιέχει νοηματικά κοντινές έννοιες δημιουργήθηκε η απαίτηση για την δημιουργία ενός κατάλληλου πλαισίου πληροφορίας. Με την αναγνώριση προτάσεων, ενοτήτων και γενικότερα του συνολικού πλήθους μεταδεδομένων επιτυγχάνεται μεγαλύτερη λεπτομέρεια. Αυτή η λεπτομέρεια θα οδηγηθεί μετέπειτα στο στάδιο της ανάκτησης. Η ανάκτηση θα κατέχει σε εκείνο το επίπεδο κατάλληλη οργάνωση της γνώσης η οποία λαμβάνεται από τα chunks. Κατά αυτόν τον τρόπο τα κοντινά νοηματικά πλαίσια πληροφορίας θα πλαισιώνονται και από επιπλέον λεπτομέρειες οι οποίες βοηθούν στην καλύτερη διαχώριση του νοήματος.

4.3 Embedding Model

Ως Embedding ορίσαμε την διαδικασία στην οποία ένα chunk, όπως αυτά που δημιουργήθηκαν με την στρατηγική η οποία αναλύθηκε στο κεφάλαιο 4.5, μετατρέπεται σε κάποιο διάνυσμα πραγματικών αριθμών. Στο συγκεκριμένο κεφάλαιο θα ορίσουμε τον τρόπο υλοποίησης αυτού. Την διαδικασία και την στρατηγική η οποία θα ακολουθηθεί με σκοπό τα chunks να μετατραπούν σε διανύσματα πραγματικών αριθμών. Αυτού του είδους η διαδικασία είναι αυτή η οποία μετέπειτα θα χρησιμοποιηθεί για την ανάκτηση των δεδομένων. Αρχικά θα αναλυθεί ο τρόπος με τον οποίο δέχονται σαν είσοδο τα chunks, πώς τα επεξεργάζονται αυτά. Θα αναλυθούν οι λόγοι για τους οποίους επιλέχθηκε το συγκεκριμένο μοντέλο Embeddings και πώς γίνεται η υλοποίηση αυτού. Πώς γίνεται δηλαδή μετέπειτα η δημιουργία διανυσμάτων.

Αφότου δημιουργηθούν τα Chunks τότε αυτά χρησιμοποιούνται ως η είσοδος για τα Embeddings. Για να βεβαιωθούμε ότι κάποιο chunk δεν υπάρχει δύο φορές μετά από την διαδικασία Chunking κάνουμε τις απαραίτητες διαδικασίες. Οι απαραίτητες διαδικασίες είναι πρακτικά η εφαρμογή της συνάρτησης dedupe_items(). Μέσω αυτής της συνάρτησης γίνεται αφαίρεση των διπλών chunks, ή γενικότερα των Chunks τα οποία μοιράζονται ακριβώς την ίδια πληροφορία. Ο λόγος για τον οποίο γίνεται η αφαίρεση τέτοιων κειμενικών τμημάτων είναι διότι το σύστημα των Embeddings έχει φτιαχτεί με στόχο να αποθηκεύεται μόνο η χρήσιμη πληροφορία. Εάν ένα Chunk, ενώ υπάρχει και περισσότερες φορές μέσα στο σύνολο των chunks, αποθηκευόταν περισσότερες φορές τότε στην Διανυσματική Βάση Δεδομένων θα συναντιόταν η ίδια πληροφορία παραπάνω φορές. Αυτό είναι ιδιαίτερα σημαντικό πλεονέκτημα καθώς δεσμεύει σημαντικό και χρήσιμο χώρο από την βάση δεδομένων διανυσμάτων. Αυτή η αχρείαση δέσμευση χώρου δυσχαιρένει και την ανάκτηση των αρχείων μετέπειτα. Οπότε σκοπός των embeddings είναι και η αποφυγή καποιού τέτοιου φαινομένου.

Snippet 12 Συνάρτηση dedupe : Τρόπος υλοποίησης της συνάρτησης dedupe η οποία είναι υπεύθυνη για την απομάκρυνση διπλότυπων chunks.

```
def dedupe_items(items: List[Dict[str, Any]]) -> List[Dict[str, Any]]:
    seen = set()
    out = []
    for it in items:
        key = (it.get("source"), it.get("chunk_index"), it.get("text"))
        if key in seen:
            continue
        seen.add(key)
        out.append(it)
    return out
```

Στο Snippet 12 παρουσιάζεται η συνάρτηση dedupe_items(), η οποία απομακρύνει τα διπλότυπα chunks.

Προτού ακολουθήσει η διαδικασία και η εφαρμογή του μοντέλου Embedding ακολουθεί και ένα ακόμα επιπλέον στάδιο. Στο συγκεκριμένο στάδιο συντίθεται το συνολικό κείμενο του chunk σε συνδυασμό των Metadata. Δηλαδή συνδυάζεται το κομμάτι του κειμένου με κάποια χαρακτηριστικά των metadata. Συγκεκριμένα με την θεματική ενότητα του κάθε chunk και με την πηγή αυτού. Αυτό το χαρακτηριστικό βοηθάει στην καλύτερη κατανόηση του πλαισίου πληροφορίας από τα Embeddings. Αυτό το κείμενο το οποίο φτιάχνεται και ονομάζεται retrieval text προστίθεται και στα

metadata. Αυτή η προσθήκη είναι χρήσιμη διότι θα μας χρειαστεί μετέπειτα και στην διαδικασία ανάκτησης.

Snippet 13 Πληροφορία που γίνεται Embedding : Στο παρακάτω πλαίσιο εμφανίζεται ο τρόπος με τον οποίο δημιουργείται το retrieval text το οποίο μετέπειτα θα δεχθεί κατάλληλη μετατροπή σε embeddings.

```
retrieval_text = (
    f"Section: {it.get('section')}\n\n"
    f"Source: {it.get('source')}\n\n"
    "Content:\n"
    f"{it.get('text')}"
)
```

Στο Snippet 13 παρουσιάζεται η δημιουργία του retrieval text από το περιεχόμενο, την ενότητα και την πηγή κάθε chunk.

Αφού έχει παραχθεί συνολικά το κείμενο το οποίο θα περάσει την διαδικασία των embeddings, αυτό το οποίο ακολουθεί είναι η ίδια διαδικασία των embeddings. Δηλαδή η μετατροπή των κειμενικών τμημάτων και στην συγκεκριμένη περίπτωση του retrieval text σε διάνυσμα πραγματικών αριθμών. Για την υλοποίηση αυτής της διαδικασίας χρησιμοποιήθηκε το μοντέλο infloat/multilingual-e5-base μέσω της βιβλιοθήκης Sentence Transform. Το συγκεκριμένο μοντέλο όπως φανερώνει και το όνομα του είναι ένα το οποίο χρησιμοποιείται και είναι κατάλληλο για διάφορες γλώσσες. Αυτό το χαρακτηριστικό είναι πολύ σημαντικό καθώς τα κείμενα και συνολικά η βάση γνώσης είναι στα ελληνικά. Δεν θα μπορούσαν δηλαδή να δημιουργηθούν τα κατάλληλα διανύσματα πραγματικών αριθμών χωρίς κάποια μέθοδο που να παρέχει πολυγλωσσικές δυνατότητες.

Snippet 14 Μοντέλο embeddings: Στο παρακάτω πλαίσιο εμφανίζεται το συγκεκριμένο μοντέλο το οποίο χρησιμοποιείται για την συνολική μετατροπή των chunks σε διανύσματα πραγματικών αριθμών.

```
def embed_texts(model_name: str, texts: List[str], batch_size: int = 32) -> np.ndarray:
    model = SentenceTransformer(model_name)
    embs = model.encode(
        texts,
        batch_size=batch_size,
        show_progress_bar=True,
        convert_to_numpy=True,
        normalize_embeddings=True,
    )
    return embs
```

Στο Snippet 14 γίνεται φανερό όλο το πλαίσιο της δημιουργίας των Embeddings. Γίνεται δηλαδή παράλληλα φανερή και η κανονικοποίηση η οποία συμβαίνει σε αυτό το στάδιο μέσω της παραμέτρου normalize_embeddings=True. Ο σκοπός της συγκεκριμένης κανονικοποίησης είναι η επίτευξη ενός κοινού μοναδιαίου μήκους διανυσμάτων. Αυτό έχει ιδιαίτερη σημασία καθώς το μέγεθος του Vector δεν μας επηρεάζει, αλλά μας επηρεάζει η κατεύθυνση αυτού. Όταν μετέπειτα μέσω cosine similarity συγκριθούν αυτά τα chunks με κάποιο ερώτημα του χρήστη το embedding θα επηρεάζεται μόνο από την κατεύθυνση στο χώρο των διανυσμάτων και η ομοιότητα υπολογίζεται μόνο βάσει αυτής. Αυτό το χαρακτηριστικό προσδίδει σημαντική βελτίωση στην απόδοση της αναζήτησης καθώς βοηθάει στον αποδοτικότερο υπολογισμό της ομοιότητας.

Αφου ολοκληρωθεί και η κανονικοποίηση και συνολικά η δημιουργία των embeddings αποθηκεύονται αυτά σε δύο αρχεία, στο embeddings.npy και στο metadata.json. Το αρχείο metadata.json το έχουμε συναντήσει και προηγουμένως και είναι υπεύθυνο για τα μεταδεδομένα κάθε chunks. Η σημαντική προσθήκη σε αυτό το αρχείο είναι το γεγονός ότι πλέον αντιστοιχείται με το αρχείο embeddings.npy. Αυτό συμβαίνει διότι το αντικείμενο i του metadata.json βρίσκεται στην θέση i του αρχείου embeddings.npy. Συγκεκριμένα αξίζει να αναφέρουμε ότι το συγκεκριμένο αρχείο περιέχει έναν πίνακα διαστάσεων $N * D$. Το N είναι το πλήθος των chunks ενώ το D αναφέρεται στην διάσταση των embeddings. Η δομή αποθήκευσης των embeddings και των μεταδεδομένων παρουσιάζεται στον Πίνακα 4.4.

Πίνακας 4.4: Δομή αποθήκευσης embeddings και μεταδεδομένων

Πεδίο	Περιγραφή
embeddings.npy	Πίνακας που περιέχει τα αριθμητικά διανύσματα των chunks
metadata.json	Πληροφορίες προέλευσης και περιεχομένου κάθε chunk
index i	Κοινή θέση embedding και metadata που εξασφαλίζει την αντιστοίχιση
retrieval_text	Το κείμενο που χρησιμοποιήθηκε για τη δημιουργία του embedding

Snippet 15 Δημιουργία και ενημέρωση αρχείων embeddings.npy και metadata.json: Στο παρακάτω πλαίσιο εμφανίζεται το συγκεκριμένο μοντέλο το οποίο χρησιμοποιείται για την συνολική μετατροπή των chunks σε διανύσματα πραγματικών αριθμών.

```
def save_embeddings_numpy(output_dir: Path, embs: np.ndarray, metadata: List[Dict[str, Any]]):
    output_dir.mkdir(parents=True, exist_ok=True)
    np.save(output_dir / "embeddings.npy", embs)
    with open(output_dir / "metadata.json", "w", encoding="utf-8") as f:
        json.dump(metadata, f, ensure_ascii=False, indent=2)
```

Στο Snippet 15 παρουσιάζεται η αποθήκευση των embeddings στο αρχείο embeddings.npy και των μεταδεδομένων στο αρχείο metadata.json

Μετα την δημιουργία των παραπάνω αρχείων δημιουργείται παράλληλα και ο διανυσματικός δείκτης μέσω της βιβλιοθήκης FAISS (Facebook AI Similarity Search). Η συγκεκριμένη βιβλιοθήκη δημιουργεί ένα ευνοϊκό περιβάλλον για να μπορούν να αναζητηθούν εύκολα και αποδοτικά οι αναπαραστάσεις των chunks μέσω των embeddings χάρις στην κατάλληλη οργάνωση των δεδομένων.

4.4 Hybrid Retrieval

4.4.1 Retrieval Pipeline

Σε προηγούμενα κομμάτια της εργασίας αναλύθηκε ο τρόπος λειτουργίας της ανάκτησης πληροφορίας. Στην συγκεκριμένη υλοποίηση του συστήματος η μέθοδος ανάκτησης με την χρήση των Embeddings είναι η τρίτη στην ιεραρχία. Προηγούνται πρώτα δύο επιπλέον επίπεδα ανάκτησης. Η

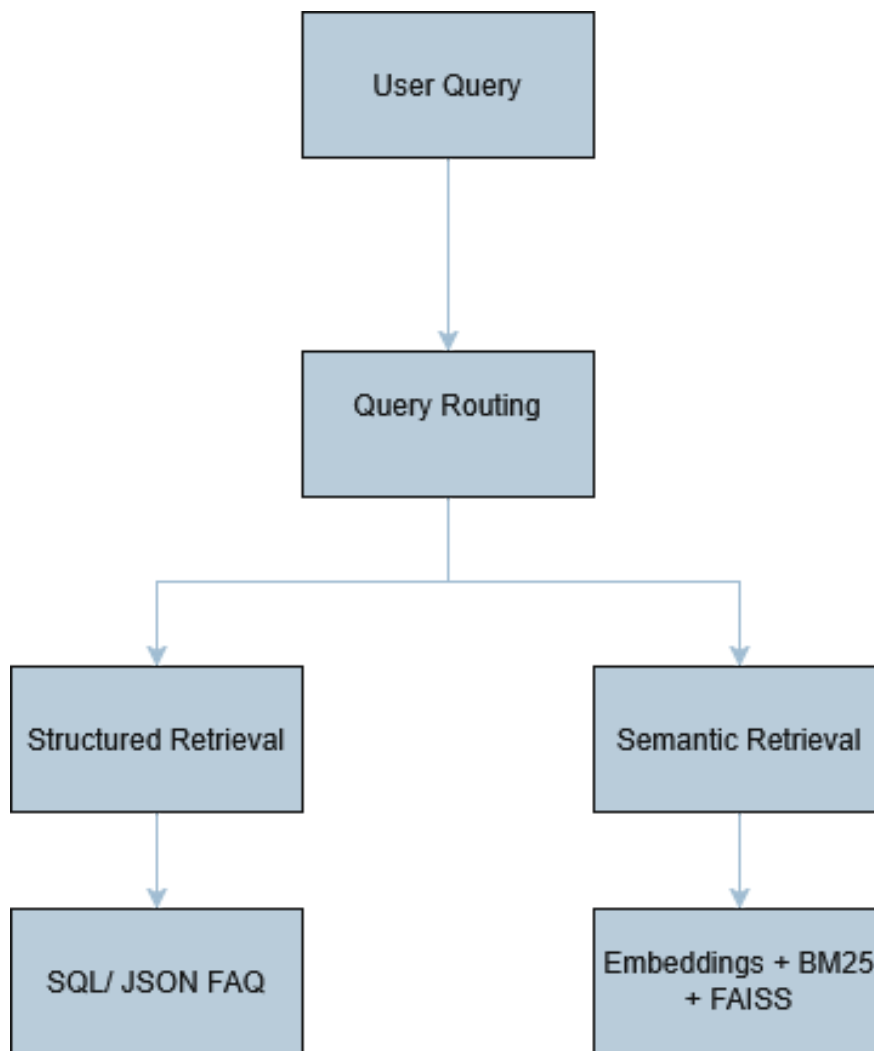
ανάκτηση πληροφορίας από τις δομημένες βάσεις δεδομένων και η ανάκτηση γνώσης από ειδικά αρχεία γνώσης, τα οποία περιέχουν διάφορες συχνές ερωτήσεις χρηστών. Έπειτα ακολουθεί το συγκεκριμένο επίπεδο ανάκτησης των μη δομημένων δεδομένων. Η συνολική διαδικασία δρομολόγησης του ερωτήματος και ανάκτησης των δεδομένων παρουσιάζεται στο Σχήμα 4.1.

Στο συγκεκριμένο στάδιο το ερώτημα του χρήστη μετατρέπεται κατάλληλα ώστε να γίνει η απαραίτητη αναζήτηση στις βάσεις γνώσης. Ο τελικός σκοπός της παραπάνω διαδικασίας είναι να δημιουργηθεί το κατάλληλο πλαίσιο πληροφορίας. Μετέπειτα το συγκεκριμένο πλαίσιο πληροφορίας θα δοθεί ως είσοδος για το μεγάλο γλωσσικό μοντέλο όπου θα παράγει την τελική απάντηση. Η διαδικασία και η υλοποίηση της ανάκτησης είναι ιδιαίτερης σημασίας καθώς κατευθύνει την τελική απάντηση από το LLM σε πολύ μεγάλο βαθμό.

4.4.2 Αρχικοποίηση Retrieval Engine

Στο συγκεκριμένο κομμάτι της υλοποίησης θα αναλύσουμε τον τρόπο της υλοποίησης της αρχικοποίησης. Σε αυτό το στάδιο θα γίνει περιγραφή του τρόπου με τον οποίο προετοιμάζεται το συνολικό σύστημα ώστε τελικά να γίνει η ανάκτηση της πληροφορίας.

Στα προηγούμενα κεφάλαια αναλύθηκαν οι έννοιες των embeddings, των μεταδεδομένων, του FAISS αλλά και του BM25. Τα παραπάνω είδαμε πως δημιουργούνται και πώς παράγονται. Όλα αυτά τα στοιχεία είναι ιδιαίτερα χρήσιμα στην τελική διαδικασία ανάκτησης καθώς είναι το πλαίσιο πληροφορίας στο οποίο γίνεται η αναζήτηση για να βρεθούν οι σχετικές πληροφορίες με το ερώτημα του χρήστη. Φορτώνονται τα embeddings.npy ένα αρχείο το οποίο περιέχει όλα τα διανύσματα πραγματικών αριθμών για το κάθε chunk. Παράλληλα φορτώνεται το metadata.json αρχείο, το οποίο



Σχήμα 4.1: Διαδικασία Query Routing και ανάκτησης δεδομένων

συνδέεται μέσω ενός δείκτη και με το αρχείο `embeddings.npy`. Το αρχείο αυτό περιέχει επιπλέον λεπτομέρειες για κάθε chunk. Γίνεται και η φόρτωση των αρχείων `faiss.index` και `bm25_index.json`. Το πρώτο αρχείο χρησιμοποιείται για γρήγορη αναζήτηση στις διανυσματικές βάσεις δεδομένων με την χρήση ενός διανυσματικού δείκτη το οποίο περιέχεται στο αρχείο. Ενώ το αρχείο `bm25_index.json` χρησιμοποιείται διότι παρέχεται ένα ευρετήριο για λεξική αναζήτηση. Τα παραπάνω χαρακτηριστικά περιγράφονται συγκεντρωτικά στον Πίνακα 4.5.

Πίνακας 4.5: Αρχεία που χρησιμοποιούνται για την αποθήκευση και ανάκτηση της γνώσης

Αρχείο	Περιγραφή
embeddings.npy	Περιέχει τα αριθμητικά διανύσματα των chunks
metadata.json	Περιέχει πληροφορίες προέλευσης και περιεχομένου
faiss.index	Διανυσματικός δείκτης για γρήγορη αναζήτηση
bm25_index.json	Sparse index για lexical αναζήτηση

4.5 Context Construction και LLM Generation

Στο συγκεκριμένο στάδιο του RAG συστήματος γίνεται η ανάλυση του τρόπου με τον οποίο συνδέεται το Context το οποίο δημιουργήθηκε στα προηγούμενα επίπεδα. Δηλαδή κατά την συγκεκριμένη διαδικασία δίνεται ως πλαίσιο πληροφορίας τα σχετικά chunks και παράλληλα δίνεται και ένα σύνολο κανόνων το οποίο υπάρχει με στόχο να προσδοθεί το κατάλληλο ύφος όταν δημιουργηθεί η απάντηση από το LLM.

Η διαδικασία αυτή αποτελείται από δύο στάδια. Στο πρώτο στάδιο της διαδικασίας δημιουργείται το συνολικό πλαίσιο πληροφορίας το οποίο θα οδηγηθεί στο LLM. Τα κομμάτια κειμένου τα οποία θεωρούνται σχετικά συγκεντρώνονται, συνδυάζονται με βάση το αρχικό αίτημα-ερώτημα του χρήστη. Αυτό το στάδιο ουσιαστικά χτίζει τα θεμέλια για να ακολουθήσει η επόμενη διαδικασία του LLM Generation. Αξίζει να σημειωθεί ότι για να συμπληρωθεί και να δημιουργηθεί το πλαίσιο πληροφορίας το οποίο θα φτάσει στο LLM Generation υπάρχουν κανόνες οι οποίοι καθορίζουν από τα ανακτημένα Chunks ποια είναι κατάλληλα για να οδηγηθούν για να καταλήξουν στο επόμενο στάδιο.

Snippet 16 Context Construction : Διαφαίνεται το τμήμα κώδικα το οποίο δέχεται ως είσοδο τα αποτελέσματα του Retrieval και τα μετατρέπει σε δομημένη πληροφορία κατάλληλη για το LLM.

```
def merge_chunks(chunks: List[Dict[str, Any]], max_chars: int = MAX_CONTEXT_CHARS) ->
    Tuple[str, int, bool]:
    parts = []
    total = 0
    included = 0
    truncated = False
    for c in chunks:
        meta = c.get("meta") if isinstance(c, dict) else {}
        if not isinstance(meta, dict):
            meta = {}
        text = (
            meta.get("retrieval_text")
            or meta.get("text")
            or (c.get("retrieval_text") if isinstance(c, dict) else "")
            or (c.get("text") if isinstance(c, dict) else "")
            or ""
        )
        if not text:
```

```

        continue
    remain = max_chars - total
    if remain <= 0:
        truncated = True
        break
    piece = text if len(text) <= remain else text[:remain] + "..."
    if len(text) > remain:
        truncated = True
    parts.append(f"[Source] {meta.get('source')}\n{piece}")
    total += len(piece)
    included += 1
return "\n\n".join(parts), included, truncated

```

Στο Snippet 16 παρουσιάζεται η διαδικασία συνένωσης των ανακτημένων chunks για τη δημιουργία του πλαισίου πληροφορίας.

Αφού το LLM λάβει ως είσοδο το παραπάνω πλαίσιο πληροφοριών, την ερώτηση του χρήστη και τις οδηγίες του συστήματος, συνδυάζει και επεξεργάζεται τις διαθέσιμες πληροφορίες, προκειμένου να δημιουργήσει την τελική απάντηση. Συγκεκριμένα στο στάδιο αυτό γίνεται η κατάλληλη σύνθεση όλων των παραπάνω στοιχείων ώστε να υπάρξει τελικά μία απάντηση η οποία να απαντάει στην ερώτηση και να προσαρμόζει ανάλογα με τις παραπάνω πληροφορίες το ύψος της απάντησης.

Snippet 17 LLM Prompt: Στο παρακάτω κομμάτι κώδικα φαίνεται το prompt το οποίο δίνεται στο LLM για να δημιουργηθεί μία φυσική απάντηση.

```

def build_prompt_from_structured_facts(query: str, structured_answer: dict) -> str:
    source_table = structured_answer.get("source_table")
    facts = structured_answer.get("row") or {}
    deterministic = structured_answer.get("answer") or ""
    facts_json = json.dumps(facts, ensure_ascii=False)

    return (
        "Θα απαντήσεις σε φοιτητή με φυσικό, σύντομο, ανθρώπινο ύφος.\n"
        "Χρησιμοποίησε ΑΠΟΚΛΕΙΣΤΙΚΑ τα FACTS που δίνονται παρακάτω.\n"
        "Απαγορεύεται να προσθέσεις πληροφορία που δεν υπάρχει στα FACTS.\n"
        "Απαγορεύεται να αντιφάσκεις με τα FACTS.\n"
        "Αν λείπει πεδίο, πες ότι δεν υπάρχει διαθέσιμη πληροφορία.\n"
        "Μέγιστο 3 προτάσεις.\n\n"
        f"Πηγή: {source_table}\n"
        f"FACTS(JSON): {facts_json}\n"
        f"Προκατασκευασμένη ασφαλής απάντηση: {deterministic}\n\n"
        f"Ερώτηση χρήστη: {query}\n"
        "Δώσε τελική απάντηση στα ελληνικά."
    )

```

Στο Snippet 17 παρουσιάζεται το prompt που διαμορφώνεται για τη δημιουργία της τελικής απάντησης από το LLM.

4.6 Επίλογος

Στην 4η ενότητα της Διπλωματικής εργασίας του Ρομποτικού Συμβούλου Σπουδών παρουσιάστηκε η συνολική υλοποίηση των επιμέρους μηχανισμών οι οποίοι συνθέτουν τελικά το συνολικό σύστημα RAG της παρούσας διπλωματικής εργασίας από την βάση γνώσης μέχρι την ανάκτηση πληροφορίας και τελικά την δημιουργία της τελικής απάντησης.

Συγκεκριμένα έγινε αναφορά στον τρόπο με τον οποίο οργανώθηκε η βάση γνώσης του συστήματος, η οποία χρησιμοποιεί τρία είδη πηγών. Η μία αφορά δομημένα δεδομένα, η δεύτερη αφορά μη δομημένα δεδομένα και η τρίτη αφορά ένα αρχείο συχνών ερωτήσεων.

Παράλληλα σε αυτό το κεφάλαιο γίνεται μία αναφορά στους τρόπους με τους οποίους εξάγονται οι πληροφορίες για τα μη δομημένα δεδομένα. Γίνεται αναφορά στην προεπεξεργασία, στην κανονικοποίηση, στην απομάκρυνση και στην διόρθωση αυτών των κειμένων με σκοπό να βελτιωθεί η δημιουργία των chunks. Αναλύθηκε επίσης η διαδικασία με την οποία δημιουργήθηκαν αυτά και οι θεωρητικοί σκοποί του κάθε chunk το οποίο δημιουργήθηκε.

Παράλληλα στο παρόν κεφάλαιο έγινε ανάλυση της υλοποίησης της διαδικασίας μετατροπής ενός chunk σε διανυσματικές αναπαραστάσεις μέσω των Embeddings. Τα embeddings τα οποία αναλύθηκαν εξίσου στην παρούσα ενότητα συνδυαστικά με τα μεταδεδομένα τα οποία προκύπτουν από κάθε chunk, του δείκτη FAISS αλλά και του ευρετηρίου BM25 δημιουργούν την υποδομή για ένα υβριδικό σύστημα ανάκτησης, συνδυάζοντας και λεξιλογική, μέσω του BM25, και σημασιολογική, μέσω των Embeddings, αντιστοίχιση.

Τέλος γίνεται αναφορά στον τρόπο υλοποίησης της δημιουργίας της τελικής απάντησης από το σύστημα RAG μέσω των ανακτημένων τμημάτων τα οποία προκύπτουν από τις προαναφερθείσες διαδικασίες. Ακολουθώντας τις επιμέρους οδηγίες του σταδίου της παραγωγής απάντησης μέσω μεγάλων γλωσσικών μοντέλων.

Κεφάλαιο 5ο: Σύστημα αναγνώρισης Φωνής

5.1 Αρχιτεκτονική και επιλογή μοντέλων αναγνώρισης φωνής

Ένας ρομποτικός σύμβουλος σπουδών για να υλοποιηθεί χρειάζεται πολύ καλή συνεργασία των συστημάτων του για να είναι αποδοτικός. Από τον τρόπο που κινείται, πως μιλάει αλλά και πως ακούει. Στο συγκεκριμένο κεφάλαιο θα γίνει μία περαιτέρω ανάλυση στον τρόπο με τον οποίο το σύστημα αντιλαμβάνεται τα αιτήματα του χρήστη μέσω της χρήσης των μικροφώνων της συσκευής ρομπότ. Η υλοποίηση ενός τέτοιου συστήματος αποτελείται από μία συγκεκριμένη αρχιτεκτονική και από διάφορες αποφάσεις οι οποίες κρίνουν την ποιότητα και την ταχύτητα της απάντησης. Το σύστημα πρέπει να αντιληφθεί όσο το δυνατόν καλύτερα την ερώτηση του χρήστη. Διότι αυτή μετέπειτα θα ακολουθήσει την διασωλήνωση του RAG και ανάλογα με την κατάλληλη αναγνώριση του κειμένου θα ανακτηθούν και πιο κατάλληλα αρχεία. Παράλληλα η αρχιτεκτονική πρέπει να είναι τέτοια έτσι ώστε όταν ο χρήστης κάνει κάποια ερώτηση αυτή να μετατραπεί σε κείμενο σε γρήγορο χρονικό διάστημα.

Στο συγκεκριμένο σύστημα η υλοποίηση αποτελείται και περιλαμβάνει δύο τέτοια μοντέλα μετατροπής φωνής σε κείμενο. Το πρώτο μοντέλο είναι αυτό του Vosk, είναι το εφεδρικό μοντέλο. Είναι ένα ελαφρύ μοντέλο το οποίο χρησιμοποιείται με σκοπό αν αποτύχει το κύριο μοντέλο σε θέματα λειτουργικότητας να υπάρχει ένα εναλλακτικό σχέδιο. Το συγκεκριμένο μοντέλο του Vosk το οποίο χρησιμοποιείται επίσης είναι εξειδικευμένο μόνο για τα ελληνικά, γεγονός το οποίο προσδίδει καλύτερη αναγνώριση φωνής στην ελληνική γλώσσα αλλά σε αγγλικούς όρους αντιμετωπίζει προβλήματα. Το κύριο μοντέλο όμως το οποίο χρησιμοποιείται είναι το Whisper (Faster-Whisper). Το συγκεκριμένο μοντέλο είναι αρκετά πιο βαρύ από το προηγούμενο με αποτέλεσμα να έχει μεγαλύτερη ανάγκη σε υπολογιστικούς πόρους αλλά προσφέρει πολύ μεγαλύτερη ακρίβεια. Το συγκεκριμένο μοντέλο επίσης αντιλαμβάνεται και ελληνικές και αγγλικές λέξεις. Τα βασικά χαρακτηριστικά και οι διαφορές των δύο μοντέλων παρουσιάζονται στον Πίνακα 5.1.

Πίνακας 5.1: Σύγκριση των μοντέλων αναγνώρισης φωνής που χρησιμοποιούνται στην υλοποίηση

Χαρακτηριστικό	Whisper (faster-whisper)	Vosk
Τύπος μοντέλου	Neural Speech Recognition Model	Offline Speech Recognition Framework
Εκτέλεση	Τοπικά μέσω faster-whisper	Τοπικά μέσω Vosk API
Γλώσσα	Ελληνικά μέσω multilingual μοντέλου	Ελληνικό acoustic model
Είσοδος	Αρχεία ήχου / PCM segments	WAV audio streams
Πλεονέκτημα	Υψηλή ακρίβεια και καλύτερη κατανόηση φυσικού λόγου	Χαμηλές απαιτήσεις πόρων και γρήγορη εκτέλεση
Χρήση στην εφαρμογή	Υψηλής ακρίβειας transcription	Lightweight real-time λειτουργία
Προσαρμογή λεξιλογίου	Δεν υποστηρίζεται άμεσα	Υποστηρίζεται μέσω custom vocabulary

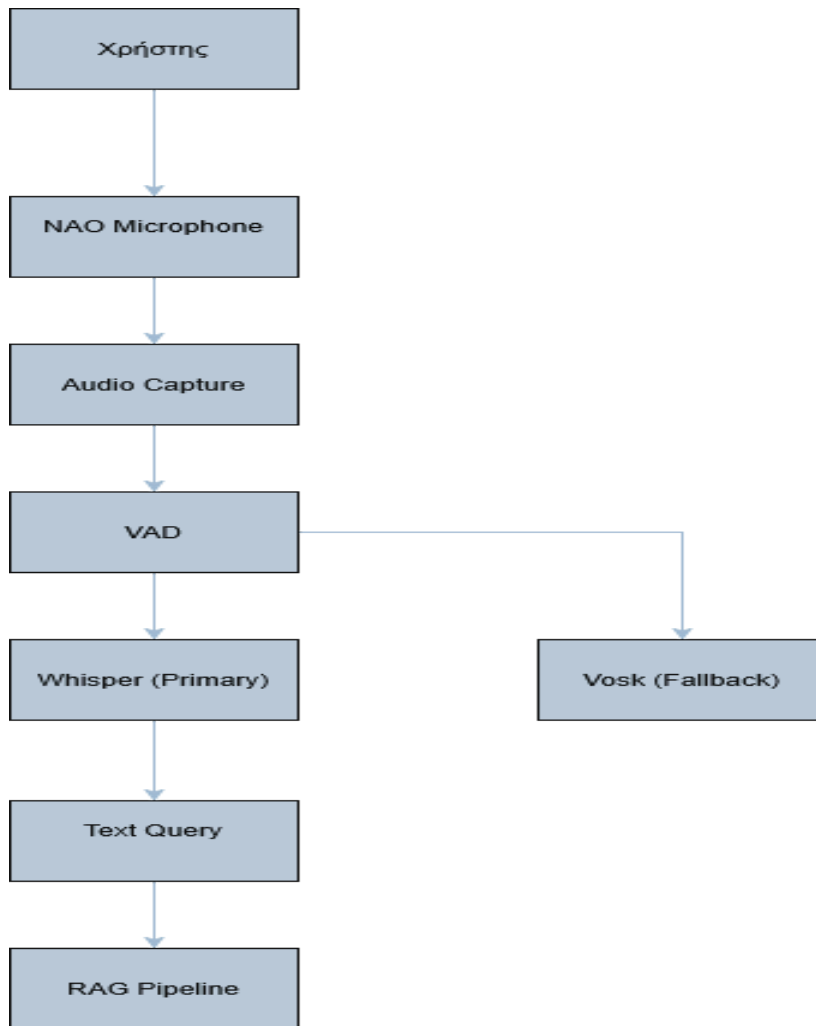
Στην συγκεκριμένη υλοποίηση του ρομποτικού συμβούλου σπουδών οι λειτουργίες που παρέχονται για αναγνώριση φωνής είναι τριών ειδών. Αρχικά παρέχεται μια εκδοχή της φωνητικής αναγνώρισης στην οποία ο χρήστης μπορεί να μεταφέρει ένα αρχείο ήχου στο σύστημα, όχι με κάποια διεπαφή αλλά μέσω του κώδικα και σε αυτό να γίνει μετατροπή από φωνή σε κείμενο. Αυτή η πρώτη εκδοχή και είδος χρήσης ήταν το πρώτο στάδιο με το οποίο έγινε ο έλεγχος για τον τρόπο λειτουργίας και την φωνητική αναγνώριση των μοντέλων. Στην συνέχεια αυτή η λειτουργία δεν απορρίφθηκε ή αφαιρέθηκε καθότι ήταν ένας πολύ βοηθητικός τρόπος για να γίνουν κάποιες μαζικές καταγραφές ήχου με διάφορες συνθήκες και να ελεγχθεί ο τρόπος λειτουργίας της αναγνώρισης φωνής από το σύστημα. Η δεύτερη εφαρμογή της φωνητικής καταγραφής είναι μέσω της λειτουργίας push to talk. Συγκεκριμένα και αυτή η λειτουργία προστέθηκε ώστε να διευκολύνει την διαδικασία ελέγχου των καταγραφών για το στάδιο του debugging και για μία πιο εύχρηστη λειτουργία καθώς είναι πιο οικονομική σε πόρους από την τρίτη εκδοχή αλλά και πιο ευχρήστη από την ανάγκη για δημιουργία καταγραφών και μετέπειτα πέρασμα αυτών για φωνητική αναγνώριση του πρώτου είδους φωνητικής καταγραφής. Η τρίτη εκδοχή είναι αυτή η οποία χρησιμοποιείται σχεδόν καθολικά στον ρομποτικό σύμβουλο σπουδών. Σε αυτήν την εκδοχή γίνεται συνεχής καταγραφή του ήχου. Δηλαδή ο χρήστης μιλάει, το σύστημα καταγράφει την φωνή του. Την μετατρέπει μετέπειτα σε κείμενο και αφού παρουσιάσει την απάντηση τότε ξανά χωρίς καμία επιπλέον ενέργεια καταγράφει εκ νέου τον ήχο.

5.2 Διαχείριση εισόδου ήχου και μηχανισμός Voice Activity Detection

Ο μηχανισμός ο οποίος είναι υπεύθυνος για την λειτουργία συστήματος σε συνεχή χρόνο το οποίο αντιλαμβάνεται πότε ο χρήστης μιλάει για να ενεργοποιηθεί τότε το σύστημα μετατροπής της ομιλίας σε κείμενο είναι το σύστημα VAD (Voice Activity Detection). Το συγκεκριμένο σύστημα χωρίζεται σε δύο μέρη. Στο πρώτο μέρος έχουμε το τμήμα στο οποίο συλλέγεται συνολικά ο ήχος και χωρίζεται σε επιπλέον τμήματα. Κάτι το οποίο καθιστά το VAD να είναι τύπου RMS. Ενώ στο δεύτερο μέρος υπάρχει το επίπεδο της φωνητικής αναγνώρισης.

Για τις ανάγκες του ελέγχου όπως και σε προηγούμενα επίπεδα το σύστημα δημιουργήθηκε έχοντας πολλαπλές πηγές εισόδου, ανάλογα με την επιλογή του χρήστη. Είτε αυτό του εξωτερικού μικροφώνου είτε του ενσωματωμένου μικροφώνου του NAO. Όπως και προηγουμένως έτσι και τώρα στην υλοποίηση του ρομποτικού συμβούλου σπουδών χρησιμοποιείται μόνο το ένα είδος. Το είδος το οποίο χρησιμοποιείται είναι το ενσωματωμένο μικρόφωνο του ρομπότ NAO V6. Το συγκεκριμένο μικρόφωνο καταγράφει ικανοποιητικές επιδόσεις όσον αφορά την συγκεκριμένη υλοποίηση με αποτέλεσμα να μην υπάρχει ανάγκη για κάποιο εξωτερικό μικρόφωνο.

Σε οποιαδήποτε περίπτωση όταν μιλήσει ο χρήστης τα δεδομένα που λαμβάνονται αποθηκεύονται σε μία thread safe ουρά. Αυτή η ουρά είναι το ενδιάμεσο στάδιο μεταξύ της διαδικασίας καταγραφής και μεταξύ της διαδικασίας μετατροπής της ομιλίας σε κείμενο. Ο λόγος χρήσης αυτής της τεχνοτροπίας είναι διότι καταφέρει και εξαλείφει ουσιαστικά τις περιπτώσεις απαγχονισμού των δύο διαδικασιών. Η ροή επεξεργασίας του φωνητικού ερωτήματος παρουσιάζεται στο Σχήμα 5.1.



Σχήμα 5.1: Ροή επεξεργασίας φωνητικού ερωτήματος

Αφού εξηγήσαμε τον τρόπο λειτουργίας και την αρχιτεκτονική που ακολουθεί το VAD, με στόχο την επίτευξη όσο το δυνατόν καλύτερου συγχρονισμού, αξίζει να αναφερθούμε και στον τρόπο με τον οποίο πραγματοποιείται η καταγραφή του ήχου από το μικρόφωνο του NAO, καθώς και στις τεχνικές που χρησιμοποιούνται για τη συνολική βελτίωση της απόδοσης του συστήματος στη μετατροπή της ομιλίας σε κείμενο.

Αρχικά μία μέθοδος η οποία χρησιμοποιείται για να βελτιωθεί το συνολικό σύστημα καταγραφής ήχου είναι η μέθοδος του Μονοφωνικού Ήχου (Mono Audio). Αρκετές εφαρμογές πολυμέσων τον ήχο τον καταγράφουν στερεοφωνικά. Όμως στην περίπτωση της μετατροπής ομιλίας σε κείμενο η καταγραφή της ομιλίας στερεοφωνικά δημιουργεί το πρόβλημα διότι αυξάνεται, σχεδόν δύο φορές, ο συνολικός όγκος δεδομένων καταγραφής. Στην περίπτωση μας αυτός ο όγκος δεδομένων είναι περιττός οπότε και στο σύστημα μας αλλά και σε πολλά συστήματα μετατροπής ομιλίας σε κείμενο χρησιμοποιείται η τεχνοτροπία Μονοφωνικού Ήχου.

Μία άλλη μέθοδος η οποία χρησιμοποιείται στην υλοποίηση είναι η μέθοδος PCM16 (16-bit Pulse Code Modulation). Η συγκεκριμένη μέθοδος χρησιμοποιείται σε διάφορες εφαρμογές οι οποίες περιέχουν ένα σύστημα μετατροπής ομιλίας σε κείμενο. Ουσιαστικά εξαιτίας του ότι η φωνή του ανθρώπου και γενικότερα πολλοί ήχοι είναι αναλογικοί δημιουργείται ένα πρόβλημα. Το πρόβλημα αυτό υπάρχει διότι ένας υπολογιστής δεν μπορεί να αντιληφθεί αναλογικά σήματα παρά μόνο

ψηφιακά. Αυτό το οποίο επιτυγχάνεται με αυτή την μέθοδο είναι πλέον αντί για το συνεχές αναλογικό σήμα αριθμών για την υλοποίηση να χρησιμοποιείται πεπερασμένος αριθμός δειγμάτων.

Επιπλέον στο σύστημα για να μην υπάρχει κενός χρόνος στον οποίο αυτό περιμένει ολόκληρο το κομμάτι από ομιλία να ολοκληρωθεί έχει προστεθεί μία τεχνική, αυτή των audio frames. Η συγκεκριμένη τεχνική ουσιαστικά δεν περιμένει ολόκληρο το κομμάτι ομιλίας για να ολοκληρωθεί αλλά την στιγμή την οποία το μοντέλο πιάσει ήχο τότε χωρίς να περιμένει να ολοκληρωθεί στέλνει για επεξεργασία αυτό το κομμάτι. Αυτή η διαδικασία συνεχίζεται μέχρις ότου σταματήσει ο ήχος. Στην υλοποίηση έχει χρησιμοποιηθεί το μέγεθος frame των 30 ms. Καθε κομμάτι, σπάει, δηλαδή στα 30ms. Οι βασικές παράμετροι εισόδου και τμηματοποίησης του ηχητικού σήματος παρουσιάζονται στον Πίνακα 5.2.

Πίνακας 5.2: Παράμετροι εισόδου και τμηματοποίησης του ηχητικού σήματος.

Παράμετρος	Τιμή υλοποίησης	Ρόλος στο σύστημα
Τύπος εισόδου ήχου	Mono PCM16	Μείωση δεδομένων και συμβατότητα με μοντέλα STT
Πηγή ήχου	Μικρόφωνο NAO / εξωτερικό audio stream	Υποστήριξη διαφορετικών τρόπων καταγραφής
Sample rate εισόδου	Native sample rate συσκευής	Διατήρηση αρχικού σήματος κατά την καταγραφή
Sample rate STT	16 kHz	Μετατροπή σε μορφή κατάλληλη για Whisper/Vosk
Μέγεθος audio frame	30 ms	Συνεχής επεξεργασία εισερχόμενου ήχου
Buffer αποθήκευσης	Thread-safe Queue (max 512 frames)	Ασφαλής επικοινωνία μεταξύ capture και VAD

5.3 Μηχανισμός Voice Activity Detection (VAD)

Στην συγκεκριμένη διπλωματική εργασία του Ρομποτικού Συμβούλου Σπουδών χρησιμοποιήθηκε ένα custom Voice Activity Detection (VAD) σύστημα. Το συγκεκριμένο σύστημα έχει ως σκοπό να αντιλαμβάνεται την στιγμή κατά την οποία ξεκινάει η ομιλία ενώ παράλληλα θα πρέπει να εντοπίζει και πότε σταματάει. Για να το πετύχει αυτό διαβάζει συνεχόμενα κομμάτια ήχου κάθε 30ms. Για κάθε κομμάτι το οποίο εντοπίζει μετράει και την ένταση αυτού. Χρησιμοποιώντας τους παρακάτω τύπους Ο τύπος του RMS είναι υπεύθυνος για να μετρήσει την ένταση του σήματος κατά πόσο μεγάλη και κατά πόσο μικρή είναι. Η μετατροπή αυτού του τύπου σε κλίμακα dBFS χρησιμοποιείται διότι μέσω αυτής γίνεται καλύτερη αναπαράσταση του ψηφιακού σήματος. Όταν αυτή η τιμή είναι μικρή σημαίνει ότι υπάρχει σιωπή ενώ όταν αυτή η τιμή μεγαλώνει υπάρχει και αντίστοιχη πιθανότητα για ομιλία ή θόρυβο.

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (5.1)$$

$$dBFS = 20 \log_{10}(RMS) \quad (5.2)$$

Πίνακας 5.3: Ενδεικτικά επίπεδα ηχητικού σήματος σε κλίμακα dBFS

Κατάσταση	dBFS
Σιωπή	-60 dBFS
Θόρυβος περιβάλλοντος	-40 dBFS
Ομιλία	-20 dBFS

Ο παραπάνω πίνακας 5.3 εκφράζει κάποια στοιχεία που καταγράφηκαν σε ένα δωμάτιο με ησυχία. Είναι από τις συνηθέστερες περιπτώσεις τιμών που καταγράφονται για τις παραπάνω καταστάσεις. Όμως αυτές οι τιμές μπορεί να διαφέρουν ανάλογα με τον περιβαλλοντικό ήχο του δωματίου. Κατά αυτή την λογική λοιπόν υλοποιήθηκε επιπλέον και ένα σύστημα στο οποίο όταν ξεκινάει η εφαρμογή του VAD συστήματος στο NAO V6, τότε αυτό κάνει μία γενική καταγραφή. Αυτή η γενική καταγραφή μετράει τον συνολικό θόρυβο του περιβάλλοντος. Μέσω αυτής της μέτρησης ορίζονται ανάλογα τα thresholds. Αυτό το χαρακτηριστικό είναι ιδιαίτερα χρήσιμο για μία εφαρμογή όπως ο ρομποτικός σύμβουλος σπουδών για το τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων και γενικότερα σε εφαρμογές Social Robotics. Διότι τα ρομπότ μπορεί να βρεθούν σε διάφορους χώρους. Από ένα εργαστήριο με ιδανικές συνθήκες μέχρι μία τάξη, μία αίθουσα πανεπιστημίου στην οποία οι συνθήκες μπορεί να διαφέρουν.

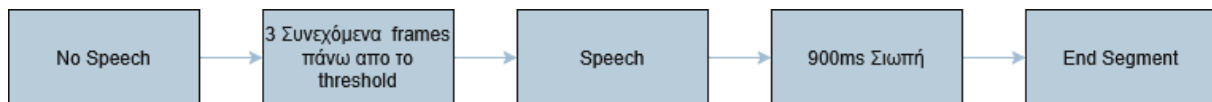
$$Threshold_{start} = NoiseFloor + 12 \text{ dB} \quad (5.3)$$

$$Threshold_{stop} = NoiseFloor + 6 \text{ dB} \quad (5.4)$$

Τα thresholds τα οποία αναφέρθηκαν είναι δύο ειδών. Το πρώτο είδος threshold είναι το start_threshold το οποίο καθορίζει πότε ξεκινάει η ομιλία και υπάρχει και το stop_threshold το οποίο ορίζει πότε υπάρχει παύση της ομιλίας. Για παράδειγμα εάν το Noise Floor είναι στα -50dB για να ενεργοποιηθεί η έναρξη καταγραφής μηνύματος πρέπει να καταγράψει το σύστημα ήχο -36dB. Αφού αρχίσει το σύστημα και καταγράφει ήχο τότε για να σταματήσει η καταγραφή θα πρέπει να καταγραφεί ήχος μεγέθους -44dB. Από το παραπάνω παράδειγμα βλέπουμε το φαινόμενο hysteresis. Στο συγκεκριμένο φαινόμενο για να ξεκινήσει η καταγραφή χρειάζεται ένας δυνατός ήχος αλλά ταυτόχρονα επιτρέπονται και οι μικρές παύσεις κατά την ομιλία.

Όμως ακόμα αυτή η μέθοδος υλοποίησης μπορεί να παρουσιάσει προβλήματα. Αν για παράδειγμα εντοπιστεί ένας ήχος ο οποίος είναι στιγμιαίος θα ενεργοποιηθεί το σύστημα και χωρίς κάποια ουσιαστική πληροφορία θα ακολουθηθεί η διασωλήνωση του RAG συστήματος. Αυτό είναι πολύ σημαντικό μειονέκτημα. Για να το αποφύγουμε στην συγκεκριμένη υλοποίηση αυτό το πρόβλημα χρησιμοποιείται η παρακάτω στρατηγική. Για να αναγνωριστεί κάτι ως ομιλία απαιτούνται

τουλάχιστον 3 συνεχόμενα Frames στα οποία ξεπερνιέται το Threshold Start. Αντίστοιχα και για την παύση της ομιλίας έχει οριστεί ως χρόνος έγκυρης παύσης τα 900ms. Η διαδικασία ανίχνευσης της ομιλίας μέσω του μηχανισμού VAD παρουσιάζεται στο Σχήμα 5.2. Αυτό γίνεται πάλι για αντίστοιχο λόγο. Όμως εδώ ο χρόνος είναι μεγαλύτερος. Αυτό συμβαίνει διότι κατά την διάρκεια της ομιλίας υπάρχει περίπτωση να υπάρξει μία φυσική παύση κατά την διάρκεια της φυσικής ροής του λόγου η οποία δεν θα έπρεπε να σταματήσει την καταγραφή και να κόψει το αίτημα και στην συγκεκριμένη περίπτωση την ερώτηση του χρήστη στην μέση.



Σχήμα 5.2: Διαδικασία ανίχνευσης ομιλίας μέσω VAD

Τέλος στην συγκεκριμένη υλοποίηση χρησιμοποιείται και ένα Preroll. Ο λόγος χρήσης αυτού είναι διότι ενδέχεται η πρώτη συλλαβή να μην ξεπερνάει ακόμα το όριο το threshold start. Αυτό το οποίο κάνει αυτός ο μηχανισμός είναι να αποθηκεύει προσωρινά προηγούμενα frames. Έτσι, κατά αυτόν τον τρόπο όταν εντοπίζει τελικά θόρυβο αρκετό ο οποίος ξεπερνάει το Threshold Start τότε η διαδικασία του Preroll προσθέτει στην αρχή αυτά τα προηγούμενα frames τα οποία έχουν αποθηκευτεί χάρη σε αυτή την μέθοδο αλλά δεν είχαν ξεπεράσει το Threshold Start.

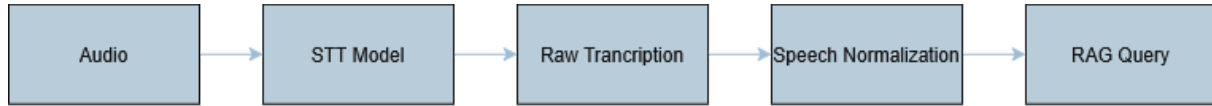
Στον παρακάτω πίνακα 5.4 φαίνονται συγκεντρωτικά όλες οι τιμές που αναφέρθηκαν και κάποιες επιπλέον οι οποίες χρειάζονται για τα παραπάνω στάδια της υλοποίησης.

Πίνακας 5.4: Παράμετροι του μηχανισμού ανίχνευσης και τμηματοποίησης ομιλίας

Παράμετρος	Τιμή	Ρόλος
Frame duration	30ms	Χρονικός διαχωρισμός ήχου
Calibration	1s	Υπολογισμός noise floor
Start threshold	+12dB	Έναρξη ομιλίας
Stop threshold	+6dB	Τερματισμός
Start trigger	3 frames	Αποφυγή false trigger
Silence duration	900ms	Κλείσιμο segment
Preroll	300ms	Διατήρηση αρχής ομιλίας
Max segment	12s	Προστασία από μεγάλα segments

5.4 Προεπεξεργασία και βελτίωση αποτελέσματος αναγνώρισης φωνής

Αφού γίνει η καταγραφή της φωνής προτού ακόμα σταλεί ως ερώτηση στη σωλήνωση του RAG το σύστημα κάνει μία προεπεξεργασία της καταγραφής. Σκοπός αυτής της προεπεξεργασίας είναι τυχόν λάθη κατά την καταγραφή του ήχου να διορθωθούν και να μην φτάσουν ως αίτημα του χρήστη με αποτέλεσμα να αποπροσανατολίζουν την ανάκτηση σχετικής πληροφορίας. Η ροή επεξεργασίας της ομιλίας και δημιουργίας του ερωτήματος για το RAG παρουσιάζεται στο Σχήμα 5.3.



Σχήμα 5.3: Ροή επεξεργασίας ομιλίας και δημιουργίας ερωτήματος RAG

Πολλές φορές τυχαίνει το STT μοντέλο το οποίο χρησιμοποιείται κάποιες λέξεις να μην τις περιέχει στο σύνολο εκπαίδευσης του. Αυτό μπορεί να δημιουργήσει σημαντικό πρόβλημα καθώς το νοηματικό πλαίσιο χάνεται όταν δεν υπάρχει αυτή η συγκεκριμένη λέξη. Ένα πολύ χαρακτηριστικό παράδειγμα το οποίο αποτέλεσε την σημαντικότερη αφορμή αυτού του στάδιου επεξεργασίας είναι η λέξη ΔΙΠΑΕ. Η συγκεκριμένη λέξη για έναν ρομποτικό σύμβουλο σπουδών είναι ιδιαίτερης σημασίας καθώς χρησιμοποιείται συχνά και καθορίζει συχνά και το νοηματικό πλαίσιο μίας ερώτησης. Η κανονικοποίηση για την διόρθωση τέτοιων προβλημάτων κρίθηκε απαραίτητη. Η συγκεκριμένη υλοποίηση αυτής αποτελείται από δύο στάδια.

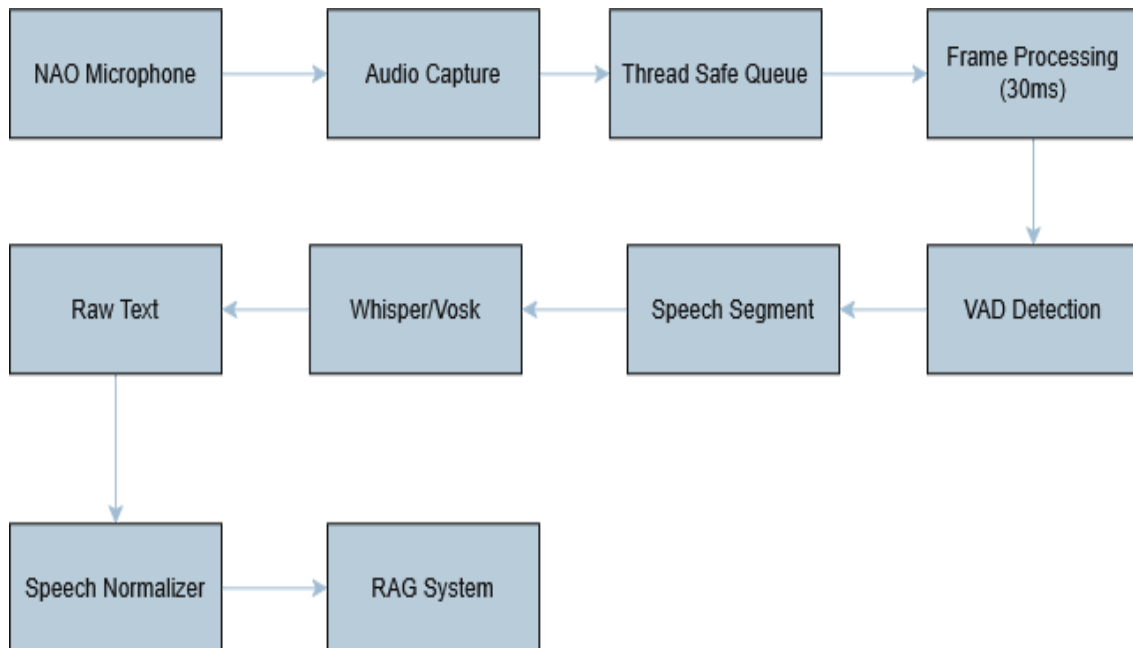
Το πρώτο στάδιο αυτής αφορά την διόρθωση με λεξικό οντοτήτων και χαρακτηρίζεται ως Deterministic. Σε αυτό το στάδιο χρησιμοποιείται ένα λεξικό οντοτήτων και κάποιοι κανόνες normalization, n-gram extraction και fuzzy similarity. Η αντικατάσταση των λέξεων δεν γίνεται αυθαίρετα αλλά γίνεται αναζήτηση κατάλληλου ταιριάσματος μέσω των παραπάνω κανόνων. Ενδεικτικά παραδείγματα διόρθωσης οντοτήτων μετά τη διαδικασία STT παρουσιάζονται στον Πίνακα 5.5.

Πίνακας 5.5: Παραδείγματα διόρθωσης οντοτήτων μετά το STT

Raw STT αποτέλεσμα	Canonical Entity
ζιπαε	ΔΙΠΑΕ
διπαε θεσσαλονικης	ΔΙΠΑΕ Θεσσαλονίκης
παπαδοπουλος	Παπαδόπουλος (καθηγητής)

Το δεύτερο στάδιο της επεξεργασίας αφορά ένα GPT Fallback τρόπο δηλαδή έναν τρόπο με τον οποίο εμπλέκεται ένα μεγάλο γλωσσικό μοντέλο. Αυτή η υλοποίηση μέσω ενός flag στον κώδικα μπορεί να οριστεί ως ενεργή ή ανενεργή αντίστοιχα. Αυτό που κάνει η συγκεκριμένη μέθοδος είναι να επεμβαίνει με βάση συγκεκριμένες εντολές και κανόνες στο μεγάλο γλωσσικό μοντέλο, οι οποίες προσαρμόζουν την καταγραφή κατάλληλα με τους κανόνες που έχουν οριστεί και με γνώση ότι αυτή η διόρθωση απευθύνεται σε εφαρμογές οι οποίες αφορούν μία βάση γνώσης και σχετικές ερωτήσεις με

το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Η συνολική αρχιτεκτονική επεξεργασίας της φωνητικής εισόδου παρουσιάζεται στο Σχήμα 5.4.



Σχήμα 5.4: Αρχιτεκτονική επεξεργασίας φωνητικής εισόδου του Ρομποτικού Συμβούλου Σπουδών

5.5 Επίλογος

Στο πέμπτο κεφάλαιο της διπλωματικής εργασίας γίνεται αναφορά στον τρόπο και στις διάφορες τεχνικές υλοποίησης οι οποίες χρησιμοποιούνται στο σύστημα αναγνώρισης φωνής του Ρομποτικού συμβούλου σπουδών. Το συγκεκριμένο κεφάλαιο είναι σημαντικό προς ανάλυση διότι επηρεάζει την συνολική αλληλεπίδραση μεταξύ ενός ανθρώπου και του ρομπότ NAO V6. Ουσιαστικά στο παρόν κεφάλαιο αναλύεται ο τρόπος με τον οποίο για την συγκεκριμένη εφαρμογή γίνεται η μετατροπή της ομιλίας σε κείμενο.

Η αρχιτεκτονική του επιμέρους συστήματος αυτού βασίζεται πάνω σε δύο μοντέλα αναγνώρισης φωνής. Και τα δύο αυτά μοντέλα εξειδικεύονται στην ελληνική γλώσσα καθώς η εφαρμογή σχεδιάστηκε για να γίνεται χρήση αυτής της γλώσσας. Το πρώτο και κύριο μοντέλο το οποίο χρησιμοποιείται είναι το faster-whisper. Ενώ αν αυτό το μοντέλο παρουσιάσει μη διαθεσιμότητα την απαιτούμενη χρονική στιγμή χρησιμοποιείται το μοντέλο Vosk. Το συγκεκριμένο μοντέλο υπάρχει στο συγκεκριμένο σύστημα αναγνώρισης ομιλίας και μετατροπή αυτής σε κείμενο ως μία αναπληρωματική λύση. Το συγκεκριμένο παρουσιάζει συνολικά χαμηλότερο επίπεδο φωνητικής αναγνώρισης όμως έχει το θετικό ότι είναι ιδιαίτερα ελαφρύ κάτι το οποίο το κάνει κατάλληλο για να χρησιμοποιηθεί ως μία λύση έκτακτης ανάγκης.

Ιδιαίτερη σημασία σε αυτό το κεφάλαιο δόθηκε και στον μηχανισμό VAD του συστήματος. Ο συγκεκριμένος μηχανισμός χρησιμοποιήθηκε για να εντοπίζει την έναρξη και την παύση της ομιλίας. Αυτό γίνεται για να αποφευχθεί η συνεχής καταγραφή ήχου από το συνολικό σύστημα κάτι το οποίο θα ήταν σημαντικά κοστοβόρο.

Τέλος για να αυξηθεί η αξιοπιστία του συστήματος της αναγνώρισης της ομιλίας του χρήστη χρησιμοποιείται λεξικό οντοτήτων, κανονικοποίηση και fuzzy matching. Ωστε η τελική αναγνώριση να περνάει από ένα επιπλέον φίλτρο πριν οδηγηθεί στο LLM για την δημιουργία της τελικής απάντησης. Τέλος, υπάρχει και η επιλογή της επέμβασης του ίδιου του LLM απευθείας στην καταγραφή της ομιλίας, με σκοπό να βοηθηθεί η νοηματική μετατροπή της σε κείμενο και να μειωθούν τα λάθη που προκύπτουν.

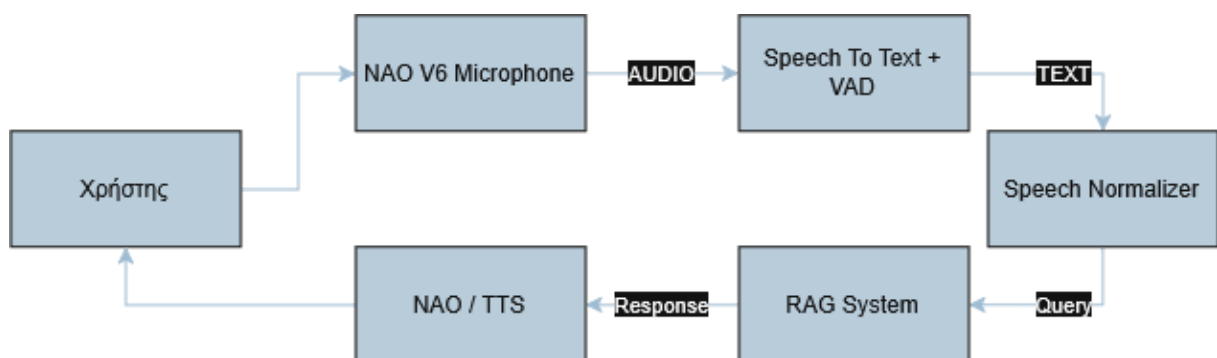
Κεφάλαιο 6ο: Τρόποι διασύνδεσης του NAO με το υπόλοιπο σύστημα

6.1 Εισαγωγή

Το έκτο κεφάλαιο της διπλωματικής εργασίας ασχολείται με τον τρόπο με τον οποίο γίνεται η συνολική διασύνδεση μεταξύ των υπόλοιπων δύο συστημάτων της εφαρμογής αλλά και με τα επιπλέον χαρακτηριστικά τα οποία υλοποιήθηκαν όσον αφορά το σύστημα του NAO V6. Ο τελικός σκοπός της εφαρμογής είναι η καλύτερη δυνατή αλληλεπίδραση μεταξύ ανθρώπου και ρομπότ. Για να επιτευχθεί λοιπόν ο συγκεκριμένος σκοπός απαιτείται η αρμονική συνεργασία των παραπάνω συστημάτων και η σύνδεση αυτών σε μία συσκευή η οποία θα λειτουργεί ως γέφυρα επικοινωνίας των συστημάτων με τον άνθρωπο. Αυτή η γέφυρα είναι το ρομπότ NAO V6 και στο παρόν κεφάλαιο θα αναλυθεί η συνολική συναρμολόγηση των συστημάτων.

Ο τρόπος υλοποίησης της διασύνδεσης ουσιαστικά αφορά την διασύνδεση μεταξύ του NAO V6, του συστήματος RAG αλλά και του συστήματος αναγνώρισης φωνητικών εντολών. Το ίδιο το ρομπότ χρησιμοποιείται για την παρουσίαση της τελικής απάντησης χάρη στην τεχνολογία μικροφώνου η οποία παρέχεται από την συσκευή. Παράλληλα αξιοποιείται επίσης το μικρόφωνο και το σύνολο της κινητικότητας του ρομπότ. Το μικρόφωνο και οι αισθητήρες ήχου χρησιμοποιούνται διότι μέσω αυτών το ρομποτικό σύστημα του NAO V6 καταγράφει τον ήχο του δωματίου αλλά και την ομιλία όταν αυτή εντοπιστεί. Ενώ το σύστημα της κινητικότητας της συγκεκριμένης συσκευής αξιοποιείται για να προσδοθεί μια επιπλέον φυσικότητα και μία συμπεριφορά κοντινή στην ανθρώπινη. Αυτό το χαρακτηριστικό είναι ιδιαίτερα σημαντικό καθώς συμβάλλει σημαντικά στην επίτευξη μίας επικοινωνίας η οποία δεν στηρίζεται μόνο στο κομμάτι του διαλόγου. Επιπλέον δηλαδή πέρα από την ανταλλαγή μηνυμάτων το ρομπότ NAO V6 χρησιμοποιεί και τις κινητικές του ικανότητες ανάλογα με την μορφή και με το είδος του μηνύματος.

Αυτό το οποίο περιγράφεται στο συγκεκριμένο σύστημα είναι ένα είδος υβριδικής υλοποίησης ενός συστήματος. Στο συγκεκριμένο σύστημα δηλαδή τις λειτουργίες που αφορούν την είσοδο και έξοδο τις αναλαμβάνει το ρομπότ NAO V6. Οι λειτουργίες οι οποίες απαιτούν περισσότερη υπολογιστική ισχύ αναλαμβάνονται από το υπολογιστικό σύστημα στο οποίο λειτουργεί η εφαρμογή. Στο συγκεκριμένο κεφάλαιο θα αναλυθεί ο τρόπος διασύνδεσης αυτών των συστημάτων.



Σχήμα 6.1: Ροή αλληλεπίδρασης χρήστη με τον Ρομποτικό Σύμβουλο Σπουδών

Όπως παρουσιάζεται στο Σχήμα 6.1, ο χρήστης ομιλεί στο NAO V6. Ο χρήστης ομιλεί στο NAO V6. Ο ήχος καταγράφεται από τα μικρόφωνα. Μετέπειτα αυτό στέλνεται σαν είσοδος στο σύστημα

μετατροπής φωνής σε κείμενο και γίνονται οι απαραίτητες κανονικοποιήσεις και επεξεργασίες με σκοπό να δημιουργηθεί ένα κατάλληλο query. Ύστερα αυτό το αίτημα στέλνεται ως είσοδος στην διασωλήνωση του RAG όπου ακολουθώντας διάφορες διαδικασίες προκύπτει τελικά μία έξοδος. Η έξοδος η οποία προκύπτει καταλήγει στο NAO V6 όπου τελικά είναι το ρομπότ το οποίο θα παρουσιάσει την τελική απάντηση στον χρήστη. Η συγκεκριμένη διαδικασία επαναλαμβάνεται κάθε φορά στην οποία ο χρήστης επικοινωνεί με το ρομπότ.

6.2 Διασύνδεση NAO V6 με Speech to Text και RAG

Η βασική διασωλήνωση του συστήματος ακολουθεί μία συγκεκριμένη ροή η οποία περιλαμβάνει τα εξής στάδια. Σε πρώτο στάδιο ο χρήστης υποβάλλει ερώτημα στο σύστημα μέσω ομιλίας, μετέπειτα η ομιλία του χρήστη καταγράφεται από τα μικρόφωνα του NAO. Ύστερα αυτή η ερώτηση από ομιλία μετατρέπεται σε κείμενο, όπου σε αυτό το κείμενο υπάρχει κάποιου είδους επεξεργασία. Στην συνέχεια αυτή η ερώτηση δίνεται ως είσοδος στο RAG, το οποίο παράγει τελικά μία απάντηση σε μορφή κειμένου. Η συγκεκριμένη απάντηση μετατρέπεται σε μορφή ομιλίας και εκφωνείται τελικά από τα ηχεία του NAO V6. Στο παρόν κεφάλαιο θα αναλυθεί ο τρόπος επικοινωνίας, δηλαδή τις υπηρεσίες οι οποίες παρέχονται από το σύστημα μέσω του αρχείου `nao_service` και αποσκοπούν στην διασύνδεση του ρομπότ με τα υπόλοιπα συστήματα.

6.2.1 Καταγραφή και μεταφορά Ήχου

Όπως έχει αναφερθεί και στα προηγούμενα κεφάλαια το ρομπότ NAO V6 καταγράφει τον ήχο με την χρήση των μικροφώνων τα οποία παρέχονται ενσωματωμένα στο ρομπότ. Ο ήχος αυτός μετέπειτα δίνεται για επεξεργασία, η οποία αναφέρθηκε και προηγουμένως, αποκλειστικά στον υπολογιστή. Για τις ανάγκες καταγραφής συνεχούς ήχου μέσω VAD ήταν απαραίτητη η συνεχής επικοινωνία μεταξύ υπολογιστή και του NAO V6, αυτό όμως δεν είναι εφικτό με τις προκαθορισμένες ρυθμίσεις. Με σκοπό λοιπόν να ξεπεραστεί το εμπόδιο αυτό και να ικανοποιηθεί η συγκεκριμένη ανάγκη έπρεπε να υπάρξει κατάλληλη επικοινωνία μεταξύ NAO V6 και υπολογιστή συστήματος.

Για να καλυφθεί λοιπόν αυτή η ανάγκη υλοποίησης ουσιαστικά από τον κεντρικό υπολογιστή του συστήματος ξεκινάει μία SSH συνεδρία προς το ρομπότ. Την στιγμή εκκίνησης αυτής της σύνδεσης εκτελείται και η εντολή `arecord` στην οποία υπάρχουν κατάλληλες παράμετροι. Ουσιαστικά η παραπάνω εντολή λαμβάνει τον ακατέργαστο ήχο ο οποίος καταγράφεται από τα μικρόφωνα του NAO V6 και έπειτα προκύπτει μία έξοδος σε PCM bytes. Τέλος αυτή η έξοδος συνδέεται μέσω του SSH με τον κεντρικό υπολογιστή του συστήματος. Ο σκοπός της συγκεκριμένης υλοποίησης είναι η ροή των δεδομένων πληροφορίας και συγκεκριμένα του ήχου σε πραγματικό χρόνο. Ουσιαστικά μέσω της παραπάνω εντολής σε πραγματικό χρόνο δεν μεταφέρεται ο ήχος σαν ένα αρχείο αλλά γίνεται stream αυτού σε πολλαπλά frames.

Snippet 16 Εντολή Arecord : Στο συγκεκριμένο κομμάτι κώδικα φαίνονται οι παράμετροι της εντολής που χρησιμοποιούνται.

```
def __enter__(self):
    arecord_parts = [
        "arecord",
        "-q",
        "-r",
        str(self.samplerate),
        "-f",
        "S16_LE",
        "-c",
        str(self.channels),
        "-t",
        "raw",
    ]
    if self.arecord_device:
        arecord_parts.extend(["-D", self.arecord_device])
```

Στο Snippet 16 παρουσιάζονται οι παράμετροι που χρησιμοποιούνται για την εκτέλεση της εντολής “arecord”. Οι παράμετροι της εντολής `arecord` και ο ρόλος τους παρουσιάζονται στον Πίνακα 6.1.

Πίνακας 6.1: Παράμετροι της εντολής arecord για την καταγραφή και μεταφορά ήχου από τον NAO V6

Παράμετρος	Τιμή/εντολή	Ρόλος
Πρόγραμμα	arecord	Καταγραφή ήχου από τη συσκευή εισόδου του NAO
-q	quiet	Περιορίζει τα μηνύματα που εμφανίζονται από την εντολή
-r	samplerate	Καθορίζει τη συχνότητα δειγματοληψίας
-f	S16_LE	Καθορίζει μορφή PCM 16-bit little-endian
-c	channels	Καθορίζει τον αριθμό καναλιών
-t	raw	Τα δεδομένα εξάγονται ως ακατέργαστο audio stream
-D	arecord_device	Επιλέγει συγκεκριμένη συσκευή ήχου, όταν απαιτείται

Στην συγκεκριμένη διαδικασία δεν αποθηκεύεται ως ένα ανεξάρτητο αρχείο ήχου το οποίο στέλνεται στον κεντρικό υπολογιστή του συστήματος αλλά μεταδίδονται συνέχεια κομμάτια πληροφορίας, τα audio frames, μέσω χρήσης της διαδικασίας SSH από τον NAO V6 προς τον κεντρικό υπολογιστή.

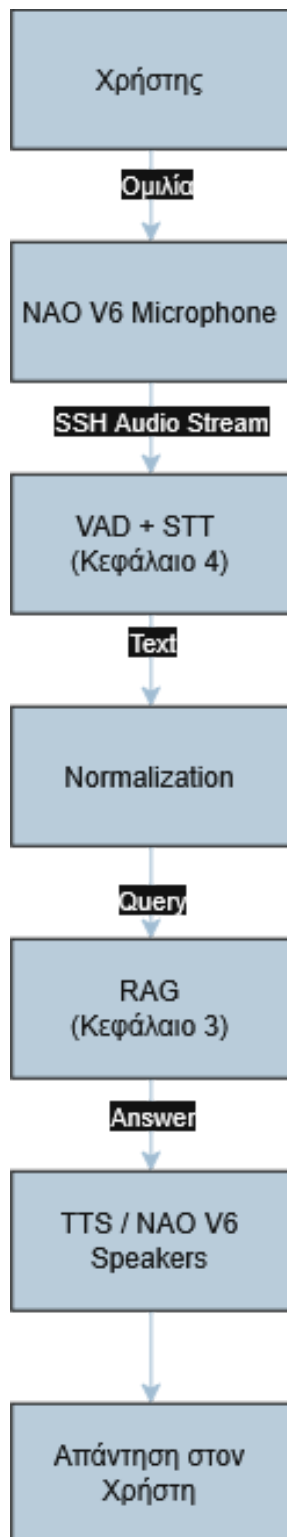
6.2.2 One Shot Επικοινωνία NAO V6 προς Υπολογιστή.

Η συγκεκριμένη υλοποίηση της διπλωματικής εργασίας παρέχει και ένα επιπλέον σύστημα μεταφοράς δεδομένων μεταξύ υπολογιστή και ρομπότ. Το σύστημα αυτό έχει αναφερθεί και σε προηγούμενα κεφάλαια της διπλωματικής εργασίας. Σε αυτό το σύστημα αντί για απλή επικοινωνία μέσω SSH υπάρχει η διαδικασία SSH+SCP. Το σύστημα πλέον χρειάζεται το SSH για συνεχή επικοινωνία μεταξύ υπολογιστή και ρομπότ αλλά πλέον τα δεδομένα δεν μεταφέρονται συνέχεια. Ουσιαστικά η μέθοδος SCP χρησιμοποιείται μόνο την στιγμή που ολοκληρωθεί το μήνυμα του χρήστη. Διότι πλέον αντί για συνεχή ανταλλαγή των audio frames υπάρχει ανταλλαγή μία φορά του αρχείου ήχου το οποίο καταγράφηκε από τα μικρόφωνα. Αυτή η συγκεκριμένη μέθοδος υλοποίησης χρησιμοποιήθηκε αποκλειστικά για να γίνουν οι απαραίτητοι έλεγχοι και οι απαραίτητες δοκιμές για το σύστημα. Αφορούσαν τον τρόπο διασύνδεσης μεταξύ υπολογιστή και ρομπότ, καθώς και την ποιότητα κατανόησης του κειμένου από το μοντέλο μετατροπής ομιλίας σε κείμενο. Οι δύο τρόποι επικοινωνίας μεταξύ του NAO V6 και του υπολογιστή συγκρίνονται στον Πίνακα 6.2.

Πίνακας 6.2: Σύγκριση των τρόπων επικοινωνίας NAO V6–υπολογιστή

Τρόπος	Ροή	Χρήση
Continuous	NAO → SSH stream → PC → VAD/STT	Κανονική λειτουργία
One-shot	NAO → WAV → SCP → PC → Whisper	Testing / debugging

6.3 Συνολική Αρχιτεκτονική Διασύνδεσης των Συστημάτων



Σχήμα 6.2: Συνολική Αρχιτεκτονική Διασύνδεσης των Συστημάτων

Η συνολική διασύνδεση των συστημάτων ουσιαστικά περιγράφεται στο Σχήμα 6.2. Ουσιαστικά το NAO V6 δέχεται την είσοδο του χρήστη μέσα από τα μικρόφωνα της συσκευής του ρομπότ μετέπειτα χάρη στο SSH υπάρχει διασύνδεση μεταξύ του ρομπότ και του κεντρικού υπολογιστή που

διαχειρίζεται το κομμάτι της αναγνώρισης φωνής, η οποία αναλύθηκε στο κεφάλαιο 4. Παράλληλα ακολουθείται μία επιπλέον σειρά διαδικασιών στην οποία το αναγνωρισμένο κείμενο φτάνει ως είσοδος στο σύστημα RAG. Θα προκύψει μετέπειτα μία νέα απάντηση όπου τελικά θα δοθεί ως είσοδος στο ρομπότ NAO ώστε να μετατραπεί το κείμενο σε ομιλία και να παρουσιαστεί τελικά στον χρήστη ως απάντηση.

6.4 Επίλογος

Στο 6ο κεφάλαιο της διπλωματικής εργασίας με θέμα την δημιουργία ενός ρομποτικού συμβούλου σπουδών για το τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων έγινε ανάλυση και παρουσίαση του τρόπου με τον οποίο γίνεται η συνολική διασύνδεση των συστημάτων του RAG και της φωνητικής αναγνώρισης με το ρομπότ NAO V6 σε ένα συγκεντρωτικό ενιαίο πλαίσιο περιβάλλοντος. Το παρόν κεφάλαιο ουσιαστικά ολοκληρώνει την διαδικασία αλληλεπίδρασης μεταξύ ανθρώπου και ρομπότ καθώς τα τμήματα τα οποία αναφέρθηκαν στα προηγούμενα κεφάλαια χρησιμοποιούν το NAO V6 για την ολοκληρωτική υλοποίηση της εφαρμογής.

Συγκεκριμένα το NAO V6 χρησιμοποιείται ως το αντικείμενο αλληλεπίδρασης ανάμεσα στον άνθρωπο και στο υπολογιστικό σύστημα το οποίο υπάρχει και εφαρμόζεται πάνω στο ρομπότ. Αξιοποιούνται τα ενσωματωμένα μικρόφωνα για την απαραίτητη καταγραφή της ομιλίας, τα ηχεία για την παρουσίαση της απάντησης στον τελικό χρήστη και παράλληλα χρησιμοποιούνται και οι κινητικές δυνατότητες του NAO V6 με σκοπό να εμπλουτιστεί με περισσότερο βάθος και να μιμείται ανθρώπινες κινητικές εκφράσεις με σκοπό να εμπλουτιστεί η τελική παρουσίαση της απόκρισης του συστήματος.

Παράλληλα στο συγκεκριμένο κεφάλαιο γίνεται ο διαχωρισμός των διεργασιών οι οποίες υλοποιούνται αξιοποιώντας τις ικανότητες του ρομπότ και στις λειτουργίες στις οποίες χρησιμοποιούνται οι δυνατότητες του κεντρικού υπολογιστικού συστήματος. Συγκεκριμένα οι διεργασίες οι οποίες αξιοποιούν τις ικανότητες και τα χαρακτηριστικά του ρομπότ στην παρούσα διπλωματική εργασία είναι εκείνα τα οποία αναφέρθηκαν στην προηγούμενη παράγραφο. Ενώ οι λειτουργίες οι οποίες χρησιμοποιούν το κεντρικό υπολογιστικό σύστημα είναι μεγαλύτερων απαιτήσεων. Οι συγκεκριμένες διεργασίες είναι η αναγνώριση της φωνής, η κανονικοποίηση τόσο της φωνής όσο και του κειμένου και η ανάκτηση της πληροφορίας μέσω του RAG. Με την χρήση αυτού του υβριδικού σχεδιασμού και διαχωρισμού των διεργασιών αξιοποιούνται σε πολύ μεγάλο βαθμό τόσο τα χαρακτηριστικά του ρομπότ, όσο και οι υπολογιστικές δυνατότητες του κεντρικού υπολογιστικού συστήματος.

Όμως για να υπάρξει κατάλληλη συνεργασία του υπολογιστικού συστήματος αλλά και του ρομπότ NAO V6 απαιτείται η συνεχής ανταλλαγή των δεδομένων μεταξύ των δύο μηχανών. Συγκεκριμένα για την μεταφορά του ήχου η ροή των δεδομένων υπάρχει χάρη της σύνδεσης SSH των δύο συστημάτων και της εντολής `arecord`, η οποία ορίζει κατάλληλα τις παραμέτρους ανταλλαγής δεδομένων. Αυτός ο συνδυασμός υπάρχει κυρίως για τις ανάγκες του VAD οι οποίες απαιτούν ανταλλαγή δεδομένων σε πραγματικό χρόνο. Παράλληλα όμως χρησιμοποιώντας το πρωτόκολλο SSH και SCP για σκοπούς ελέγχου του συστήματος δημιουργήθηκε μια λειτουργία κατάλληλη για δοκιμές του Speech to Text μοντέλου χωρίς να απαιτείται η συνεχής σύνδεση και ανταλλαγή δεδομένων όπως γίνεται στην πρώτη περίπτωση της παραγράφου.

Τέλος η συγκεκριμένη υλοποίηση διασύνδεσης αποσκοπεί ο χρήστης να μην αλληλεπιδράει με κάποιο απρόσωπο σύστημα, όπως θα μπορούσε να είναι μία ιστοσελίδα για παράδειγμα, αλλά να

αλληλεπιδρά απευθείας με ένα ανθρωπόμορφο ρομπότ, το ΝΑΟ V6, αξιοποιώντας έτσι και τα θετικά στοιχεία του ρομπότ και τα θετικά του υπολογιστικού συστήματος.

Κεφάλαιο 7ο: Συμπεράσματα και αποτίμηση της Υλοποίησης

7.1 Στόχοι και υλοποίηση

Ο συνολικός στόχος της παρούσας διπλωματικής εργασίας είναι η επίτευξη ενός ενιαίου συστήματος το οποίο θα μπορεί να αλληλεπιδρά με φυσικότητα με ανθρώπους απαντώντας στις ερωτήσεις τους με κατάλληλο ακαδημαϊκό ύφος. Αν διασπαστεί σε περισσότερα τμήματα ο παραπάνω στόχος διαχωρίζεται σε 5 επιμέρους τμήματα τα οποία αν συνδεθούν συμπληρώνουν τον τελικό στόχο επίτευξης ο οποίος είναι η δημιουργία ενός ρομποτικού συμβούλου σπουδών για το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Δι.Πα.Ε.

1. Στόχος Φωνητικής Επικοινωνίας και Αλληλεπίδρασης: Στον συγκεκριμένο στόχο παρουσιάζουμε τον τρόπο με τον οποίο ο χρήστης υποβάλλει μία ερώτηση και τον τρόπο με τον οποίο παρουσιάζει το ρομπότ την απάντηση στην ερώτηση του χρήστη.
2. Στόχος Κατανόησης Αιτήματος Χρήστη: Ο παρόν στόχος αφορά την κατάλληλη επεξεργασία και κατηγοριοποίηση της πρόθεσης του χρήστη ώστε να υπάρξει το απαραίτητο φιλτράρισμα στην βάση γνώσης από την οποία θα γίνει η ανάκτηση πληροφορίας.
3. Στόχος RAG: Ο συγκεκριμένος στόχος του συστήματος ασχολείται με την διαδικασία ανάκτησης της πληροφορίας του χρήστη ανάλογα με το αίτημα του.
4. Στόχος Παραγωγής Απάντησης: Το συγκεκριμένο τμήμα αφορά την δημιουργία απάντησης με την χρήση LLM και με τα αποτελέσματα της ανάκτησης ως είσοδο στο LLM. Με σκοπό να παραχθεί μία απάντηση η οποία αρμόζει στο ύφος ενός ρομποτικού συμβούλου και παράλληλα παρουσιάζει και τις σωστές ανακτημένες πληροφορίες.
5. Στόχος Διασύνδεσης NAO V6: Όλα αυτά πρέπει να ενσωματωθούν σε ένα ενιαίο σύστημα παρουσίασης. Αυτό συμβαίνει με την κατάλληλη διασύνδεση των διάφορων λειτουργιών με το ρομπότ NAO V6, το οποίο παρουσιάζει τελικά την απάντηση στο αίτημα του χρήστη.

Τα παραπάνω στοιχεία ήταν οι βασικοί στόχοι με τους οποίους σχεδιάστηκε συνολικά ο ρομποτικός σύμβουλος σπουδών και με τους οποίους υλοποιήθηκε τελικά η συγκεκριμένη διπλωματική εργασία. Συγκεκριμένα στην παρούσα διπλωματική εργασία έχει παρουσιαστεί κάθε τρόπος με τον οποίο επιτεύχθηκαν τελικά αυτοί οι στόχοι και οι οποίοι οδήγησαν στην τελική μορφή του Ρομποτικού Συμβούλου Σπουδών για το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Οι βασικοί στόχοι της υλοποίησης και τα αντίστοιχα αποτελέσματά τους παρουσιάζονται συνοπτικά στον Πίνακα 7.1.

Πίνακας 7.1: Συνοπτική παρουσίαση στόχων και αντίστοιχων υλοποιήσεων

Στόχος	Αποτέλεσμα
Αναγνώριση ομιλίας	Whisper + VAD
Ανάκτηση πληροφορίας	Dense + BM25 + RRF
Παραγωγή απάντησης	LLM
Φωνητική έξοδος	NAO TTS
Συνεχής διάλογος	State machine
Διασύνδεση NAO	SSH

7.2 Συνολική Λειτουργία Συστήματος

Για να γίνει κατανοητός ο τρόπος με τον οποίο λειτουργεί το συγκεκριμένο σύστημα του Ρομποτικού Συμβούλου Σπουδών θα γίνει μία αναφορά σε ένα παράδειγμα λειτουργίας του συστήματος από την αρχή μέχρι το τέλος, ακολουθώντας τα διάφορα στάδια της υλοποίησης. Ένας φοιτητής μπορεί να απευθύνει στο ρομπότ NAO V6 μία ερώτηση. Έστω ότι η ερώτηση αφορά εάν υπάρχουν εστίες για την στέγαση των φοιτητών. Αυτή η ερώτηση καταγράφεται από τα ενσωματωμένα μικρόφωνα του ρομπότ και μετατρέπεται από ομιλία σε κείμενο, το οποίο κείμενο δέχεται και ένα είδος επεξεργασίας προτού οδηγηθεί για ανάκτηση. Αφού συμβεί αυτό το σύστημα κάνει ανάκτηση της σχετικής πληροφορίας με βάση το Hybrid Retrieval και προσπαθεί να βρει τα σχετικά σημεία της βάσης γνώσης τα οποία αντιστοιχούν στο ερώτημα του χρήστη. Έπειτα αυτά δίνονται ως είσοδο και συνολικά ως το πλαίσιο πληροφορίας στο LLM, συνδυαστικά με την ίδια την ερώτηση του χρήστη. Αυτό συμβαίνει για να παραχθεί μία ολοκληρωτική απάντηση κατάλληλη για να εκφωνηθεί από το NAO V6 ως η τελική απάντηση η οποία θα ταιριάζει και με τον τρόπο με τον οποίο εκφώνησε ο χρήστης την εκάστοτε ερώτηση.

Ουσιαστικά το γεγονός το οποίο παρατηρείται από την συνολική υλοποίηση της διπλωματικής είναι ότι για να γίνει καλύτερη ανάκτηση πληροφορίας πρέπει να υπάρχει όσο το δυνατόν καλύτερα οργανωμένη η βάση γνώσης. Παράλληλα να χρησιμοποιούνται κατάλληλες μέθοδοι οι οποίες κατανοούν σε μεγάλο βαθμό το νοηματικό περιεχόμενο της ερώτησης και των κειμενικών τμημάτων ώστε να γίνει η καλύτερη δυνατή αντιστοίχιση αυτών και με ιδιαίτερη έμφαση στην ελληνική γλώσσα η οποία χρησιμοποιείται. Επίσης και η αναγνώριση ομιλίας και η μετατροπή αυτής σε κείμενο πρέπει να εξειδικεύονται στην ελληνική γλώσσα ενώ λέξεις οι οποίες δεν υπάρχουν αυτούσιες στο λεξιλόγιο του μοντέλου φωνητικής αναγνώρισης πρέπει να διαχειρίζονται κατάλληλα. Με σκοπό τελικά να φέρουν το κατάλληλο νοηματικό πλαίσιο πληροφορίας όταν φτάσουν στο σύστημα RAG και στο στάδιο ανάκτησης αυτού αρχικά και μετέπειτα στο LLM της παραγωγής απάντησης. Επιπρόσθετα για να μην ξεφεύγει και απαντάει μη έχοντας τις απαραίτητες πληροφορίες οι οποίες απαιτούνται για την δημιουργία μίας απάντησης έχουν ορισθεί κάποια όρια τα οποία οριοθετούν ποια κομμάτια κειμένου είναι όσο σχετικά χρειάζεται για να δημιουργηθεί η τελική απάντηση. Ενώ στην συνέχεια για να μην ξεφεύγουν οι απαντήσεις από την συνολική βάση γνώσης χρησιμοποιείται ένα σύστημα RAG ώστε το LLM να μην στηρίζεται στις γνώσεις τις οποίες κατέχει από την προηγούμενη εκπαίδευση του και να

στηρίζεται στην εξωτερική βάση γνώσης του. Τέλος η χρήση του ρομπότ NAO V6 καταφέρνει και προσδίδει μία επιπλέον φυσικότητα στο σύστημα η οποία είναι κατάλληλη για αλληλεπίδραση μεταξύ ανθρώπου και ρομπότ.

7.3 Πλεονεκτήματα και περιορισμοί της υλοποίησης

7.3.1 Πλεονεκτήματα

Συνδυασμός Πλήθους Τεχνολογιών: Ο συγκεκριμένος ρομποτικός σύμβουλος σπουδών δεν είναι ένα απλό Chatbot. Χρησιμοποιούνται διάφορες τεχνολογίες συνδυαστικά. Τεχνολογίες όπως η τεχνητή νοημοσύνη, τα μεγάλα γλωσσικά μοντέλα, τα συστήματα RAG, το Hybrid Retrieval, η μετατροπή φωνής σε κείμενο και το αντίστροφο για την κατανόηση του αιτήματος και για την παρουσίαση της απάντησης του και παράλληλα όλα αυτά υπάρχουν πάνω σε ένα κοινωνικό ρομπότ.

Δύο βάσεις Γνώσης: Η συγκεκριμένη υλοποίηση περιέχει δύο βάσεις γνώσης, η κάθε μία έχει δημιουργηθεί με σκοπό να έχει μεγαλύτερη εξειδίκευση στις ερωτήσεις της εκάστοτε θεματολογίας. Συγκεκριμένα κατά την ενεργοποίηση του συστήματος ο χρήστης επιλέγει εάν θα διαλέξει το σύστημα να του απαντάει για ερωτήσεις σχετικές με το προπτυχιακό πρόγραμμα σπουδών ή αν απαντάει για τα μεταπτυχιακά και διδακτορικά προγράμματα. Οι βάσεις γνώσης βασίζονται σε συγκεκριμένες πληροφορίες οι οποίες έχουν τοποθετηθεί στο RAG εσκεμμένα. Είναι επικαιροποιημένες και δεν υπάρχει ο κίνδυνος να βασίζεται η απάντηση σε εξωτερικές πηγές γνώσης πέρα από αυτές οι οποίες υπάρχουν στο RAG. Modular Τεχνική: Κάθε κομμάτι της υλοποίησης είναι ξεχωριστό και ανεξάρτητο και μπορεί για διάφορες ανάγκες του συστήματος να απομονωθεί, να τροποποιηθεί ή να αντικατασταθεί. Φυσικότητα στην Αλληλεπίδραση: Για την αλληλεπίδραση μεταξύ του ρομπότ και του ανθρώπου χρησιμοποιήθηκε το ρομπότ NAO V6 το οποίο ανήκει στην κατηγορία των Social Robots και παράλληλα η διεπαφή μεταξύ αυτού και των χρηστών γίνεται μέσω ομιλίας.

7.3.2 Περιορισμοί της υλοποίησης

Περιορισμένος όγκος δεδομένων στις Βάσεις Γνώσης: Οι βάσεις γνώσης ουσιαστικά περιορίζονται αποκλειστικά στην πληροφορία η οποία υπάρχει σε αυτές μέσω των αρχείων που τοποθετήθηκαν σε αυτές. Κάτι το οποίο περιορίζει διότι απορίες οι οποίες απαντώνται και βρίσκονται στο διαδίκτυο δεν απαντώνται με την συγκεκριμένη αρχιτεκτονική του συστήματος. Αστοχίες στην Αναγνώριση Φωνής: Για την επίτευξη της καλύτερης εμπειρίας του χρήστη επιλέχθηκε ένα μοντέλο αναγνώρισης φωνής το οποίο ενδέχεται να αναγνωρίσει μία λέξη λάθος καθώς δεν επιλέχθηκε το πιο ισχυρό μοντέλο διότι ένα τέτοιο μοντέλο θα καθυστερούσε την διαδικασία αναγνώρισης χρονικά. Γεγονός το οποίο θα επηρέαζε την συνολική λειτουργία της εφαρμογής. Επιλέχθηκε ένα μοντέλο το οποίο συνδυάζει και πολύ καλή φωνητική αναγνώριση αλλά και πολύ καλή ταχύτητα απάντησης. Το συγκεκριμένο σύστημα παρουσιάζει διάφορους περιορισμούς στα χαρακτηριστικά του υλικού με αποτέλεσμα ορισμένες μέθοδοι, πολλές φορές ίσως και πιο αποδοτικές να μην μπορούν να εφαρμοστούν στο συγκεκριμένο σύστημα και να οδηγούν σε χειρότερη εμπειρία του χρήστη. Έλλειψη Βάθους αξιολόγησης από μεγάλο πλήθος χρηστών: Στην συγκεκριμένη υλοποίηση το σύστημα δεν έχει ελεγχθεί από πολλούς χρήστες είτε με κάποια ενδεδειγμένη μέθοδο αξιολόγησης τέτοιων συστημάτων. Το συγκεκριμένο χαρακτηριστικό δεν συμβάλλει στην κατάλληλη και επίσημη αξιολόγηση του μοντέλου.

7.4 Έλεγχος γραπτής και φωνητικών αιτημάτων στο σύστημα

7.4.1 Σκοπός Αξιολόγησης

Στο υποκεφάλαιο 7.4 θα γίνει μία ανάλυση των αποτελεσμάτων, μία διερεύνηση αυτών όσον αφορά την ποιότητα των απαντήσεων για το RAG σύστημα του ρομποτικού συμβούλου σπουδών. Συγκεκριμένα θα συγκριθούν τα αποτελέσματα τα οποία προέκυψαν από την διενέργεια ελέγχων του συστήματος και σύγκριση αυτών σχετικά με την συνολική ποιότητα της απόδοσης ανάλογα με τον τρόπο με τον οποίο εκφράζεται ένα ερώτημα στο σύστημα. Είτε δηλαδή παρουσιάζεται με γραπτή μορφή αξιοποιώντας αυτή την δυνατότητα υλοποίησης της διπλωματικής εργασίας είτε εάν προκύπτει μέσω απευθείας φωνητικής αλληλεπίδρασης.

Η συγκεκριμένη σύγκριση είναι ιδιαίτερα υψίστης σημασίας καθώς μπορούν να διαπιστωθούν δύο σημαντικά πράγματα. Ο σκοπός της αξιολόγησης είναι να αποδείξει ότι σε πρώτο στάδιο το σύστημα λειτουργεί ικανοποιητικά και παρουσιάζει σχετικές απαντήσεις στον χρήστη ανάλογα με το ερώτημα του. Ενώ παράλληλα χρησιμεύει στην αξιολόγηση του συστήματος μετατροπής της ομιλίας σε κείμενο. Δηλαδή στο εάν η μέθοδος αυτή επιτυγχάνει εξίσου ικανοποιητικά αποτελέσματα και κατά πόσο αυτά τα αποτελέσματα επηρεάζονται συνολικά από την χρήση αυτού του επιπλέον σταδίου μετατροπής και επεξεργασίας συγκριτικά με τα ερωτήματα γραπτού λόγου.

7.4.2 Μεθοδολογία Αξιολόγησης

Για την αξιολόγηση και για την παραγωγή έγκυρων συμπερασμάτων έγινε χρήση 50 συνολικά ερωτημάτων σχετικά με τις γνώσεις του συστήματος όσον αφορά πληροφορίες οι οποίες βρίσκονται στον οδηγό σπουδών του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Τα ερωτήματα αυτά εκτελέστηκαν δύο φορές. Μία φορά εισήχθησαν στο σύστημα μέσω γραπτής μορφής ακολουθώντας μετέπειτα την διασωλήνωση του RAG συστήματος. Ενώ την δεύτερη φορά έγινε εισαγωγή των ερωτημάτων μέσω ομιλίας του χρήστη ,στην οποία τα ερωτήματα προτού ακολουθήσουν την διασωλήνωση η οποία προαναφέρθηκε προηγουμένως, έγινε αρχικά η ηχητική καταγραφή αυτών, μετέπειτα ακολούθησε η μετατροπή τους μέσω του συστήματος Speech to Text, ενώ επίσης ακολουθήθηκε και η μετέπειτα κανονικοποίηση και επεξεργασία της καταγραφής. Μετά από την συγκεκριμένη διαδικασία ακολουθήθηκε κανονικά η διασωλήνωση του συστήματος RAG.

Για κάθε είδος με το οποίο τέθηκε κάποιο ερώτημα, είτε γραπτώς, είτε με φωνητική εντολή, υπήρξε αξιολόγηση σε κλίμακα τριών βαθμών. Συγκεκριμένα με βαθμολογία 2 αντιστοιχίστηκαν τα ερωτήματα στα οποία το σύστημα κατάφερε να απαντήσει αρκετά ή και πλήρως ικανοποιητικά. Η βαθμολογία 1 αντιστοιχούσε στα ερωτήματα όπου η απάντηση ήταν μερικώς ικανοποιητική και είτε υπήρχαν μικροπροβλήματα σε θέματα σύγχυσης ή ψευδαισθήσεων συγκριτικά με μία πλήρως ικανοποιητική απάντηση είτε παρέλειπτε να αναφέρει κάποιο σημαντικό κομμάτι της απάντησης. Επιπρόσθετα υπήρχε και η βαθμολογία του 0 η οποία αφορούσε τα ερωτήματα για τα οποία είτε δεν προκύπτει απάντηση από το σύστημα είτε η απάντηση η οποία προκύπτει ήταν ολοκληρωτικά εκτός θέματος. Το συγκεκριμένο είδος κλίμακας χρησιμοποιήθηκε για να προσδώσει ένα βάθος αποτύπωσης των απαντήσεων για να διαχωριστούν οι απαντήσεις οι οποίες απαντούσαν μερικώς ικανοποιητικά το ερώτημα από αυτές που απαντούσαν σχεδόν απόλυτα σωστά το ερώτημα. Παράλληλα για την συνολική αξιολόγηση του συστήματος και για την ύπαρξη περισσότερης διαφάνειας καταγράφηκαν επίσης και οι επακριβείς απαντήσεις οι οποίες δόθηκαν από το σύστημα και στα δύο είδη εκπόνησης των ερωτήσεων. Η κλίμακα αξιολόγησης των απαντήσεων παρουσιάζεται στον Πίνακα 7.2.

Πίνακας 7.2: Κλίμακα αξιολόγησης απαντήσεων

Βαθμολογία	Κατηγορία	Περιγραφή
2	Ικανοποιητική/σωστή	Η απάντηση είναι σωστή, σχετική και καλύπτει το ερώτημα.
1	Μερικώς ικανοποιητική	Η απάντηση περιέχει σχετική πληροφορία, αλλά είναι ελλιπής, ασαφής ή δεν καλύπτει πλήρως το ερώτημα.
0	Μη ικανοποιητική	Η απάντηση είναι λανθασμένη, άσχετη ή δεν δίνει χρήσιμη πληροφορία.

7.4.3 Συνολικά αποτελέσματα

Πίνακας 7.3: Αποτελέσματα αξιολόγησης γραπτής και φωνητικής εισόδου

Κατηγορία αξιολόγησης	Γραπτή είσοδος	Φωνητική είσοδος
Ικανοποιητική/ολοκληρωτικά σωστή απάντηση	38/50 = 76%	37/50 = 74%
Μερικώς ικανοποιητική απάντηση	7/50 = 14%	8/50 = 16%
Αποδεκτές απαντήσεις (Βαθμός 0 ή 1)	45/50 = 90%	45/50 = 90%
Μη ικανοποιητική απάντηση	5/50 = 10%	5/50 = 10%
Μέση βαθμολογία	1,66/2 = 83%	1,64/2 = 82%

Στον παραπάνω πίνακα 7.3 παρουσιάζονται τα αποτελέσματα τα οποία προέκυψαν από τις διαφορετικές μορφές εισόδου του συστήματος. Το αποτέλεσμα το οποίο προκύπτει φανερώνει το γεγονός ότι το σύστημα παρουσιάζει αρκετά υψηλό ποσοστό ικανοποιητικών ή ολοκληρωτικά σωστών απαντήσεων ανεξάρτητα από το ποια είναι η μορφή εισόδου του συστήματος. Συγκεκριμένα

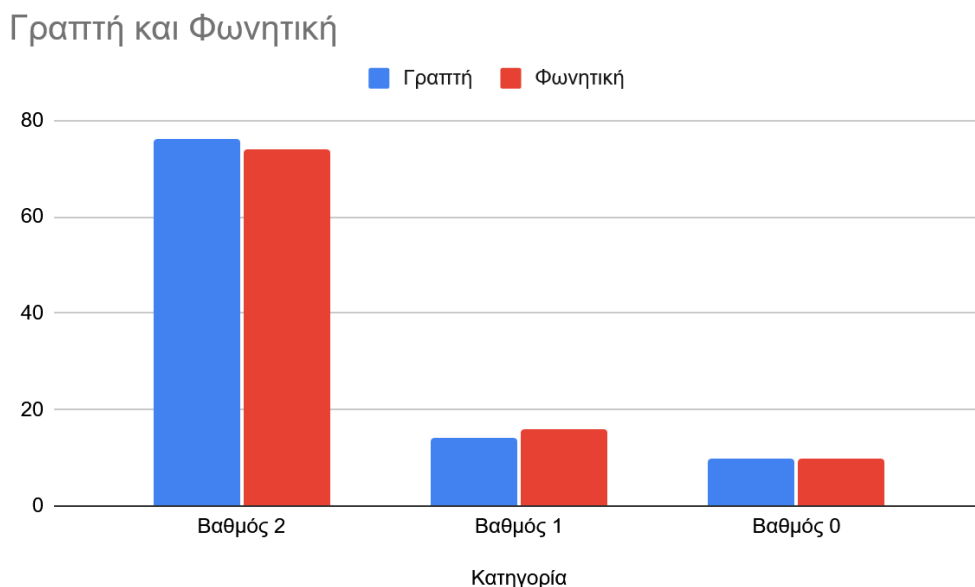
στην γραπτή μορφή εισόδου καταγράφηκαν 38 από τις συνολικά 50 απαντήσεις, γεγονός το οποίο αντιστοιχεί στο 76%, με βαθμολογία 2, το οποίο μεταφράζεται ως ικανοποιητική απάντηση. Στην μορφή φωνητικής εισόδου τα ποσοστά σωστών απαντήσεων είναι αντίστοιχα καθώς συναντάμε ένα ποσοστό της τάξης του 74% με συνολικά 37 ικανοποιητικές απαντήσεις στα 50 ερωτήματα της αξιολόγησης.

Η συνολική διαφορά των δύο μορφών εισόδου είναι ιδιαίτερα μικρή και εκφράζεται με 2 ποσοστιαίες μονάδες όσον αφορά τις απόλυτα ικανοποιητικές απαντήσεις. Παράλληλα η μέση βαθμολογία η οποία καταγράφεται είναι εξίσου μικρή διατηρώντας το πλεονέκτημα της γραπτής μορφής εισόδου έναντι της φωνητικής μορφής. Συγκεκριμένα για την γραπτή μορφή εισόδου παρατηρείται μέση βαθμολογία της τάξης του 1.66/2 το οποίο ισούται με 83%. Ενώ για την φωνητική μορφή εισόδου έχει καταγραφεί μέση βαθμολογία 1.64/2 το οποίο μεταφράζεται ως 82%. Επομένως το γενικό συμπέρασμα το οποίο προκύπτει είναι ότι η χρήση φωνητικής μορφής εισόδου δεν επιφέρει κάποια σημαντική επιβάρυνση και καταφέρνει να διατηρήσει την συνολική ποιότητα του συστήματος σε εργαστηριακές και ιδανικές συνθήκες.

Επιπλέον μία εξίσου σημαντική παρατήρηση είναι το γεγονός ότι το ποσοστό μη ικανοποιητικών απαντήσεων παραμένει σταθερό και ίδιο, στο 10%, ανεξάρτητα με το εάν γίνεται χρήση της γραπτής ή της φωνητικής μορφής εισόδου. Αυτό το στοιχείο είναι ιδιαίτερης αξίας καθώς αποτελεί μία ένδειξη ότι οι μη ικανοποιητικά απαντημένες ερωτήσεις σχετίζονται σε περιορισμούς της βάσης γνώσης, της ανάκτησης ή της δημιουργίας της απάντησης. Αυτή η συγκεκριμένη ένδειξη βοηθάει στο συμπέρασμα ότι το πιθανό σφάλμα μπορεί να υπάρχει αλλά τα στοιχεία μας βοηθούν να περιορίσουμε σημαντικά την ευθύνη στην διαδικασία καταγραφής και μετατροπής της ομιλίας του χρήστη σε μία κειμενική μορφή.

7.4.4 Διαγράμματα

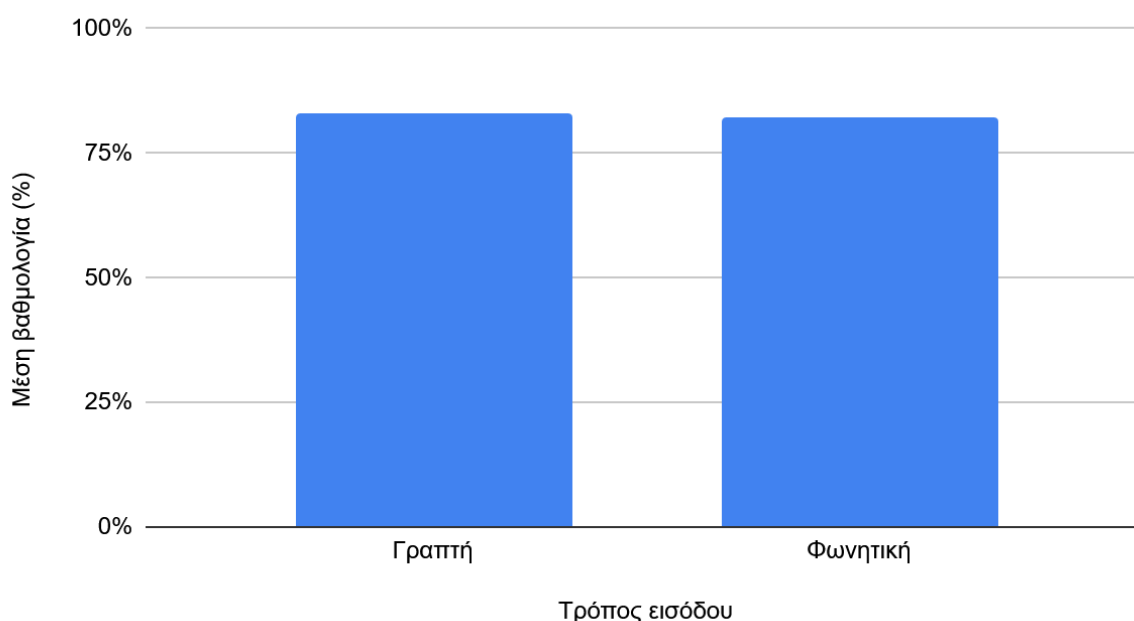
Σχήμα 7.1: Κατανομή αξιολόγησης απαντήσεων ανά τρόπο εισόδου



Στο διάγραμμα 7.1 φαίνεται η πανομοιότυπη κατανομή αξιολόγησης των δύο μορφών εισόδου του συστήματος. Η φωνητική είσοδος παρουσιάζει μικρότερη ακρίβεια στην βαθμολογία όμως αυτή η διαφορά καλύπτεται στον βαθμό 1 στο οποίο υπερτερεί έναντι της γραπτής μορφής εισόδου. Αυτά τα παραπάνω χαρακτηριστικά έχουν το θετικό αντίκτυπο ότι ουσιαστικά η τελική διαφορά ισοβαθμίζεται και το σύνολο των αποδεκτών απαντήσεων είναι ίδιο ανεξάρτητα από τη μορφή εισόδου. Γεγονός το οποίο φαίνεται και από το διάγραμμα καθώς το ποσοστό μη ικανοποιητικών απαντήσεων είναι αμετάβλητο.

Σχήμα 7.2: Μέση επίδοση του RAG συστήματος

Μέση βαθμολογία (%) έναντι Τρόπος εισόδου



Το διάγραμμα 7.2 βοηθάει στο να αναδειχθεί το γεγονός ότι η μέση απόδοση των δύο μεθόδων είναι πρακτικά ισοδύναμη με βάση τα στοιχεία των μετρήσεων. Η μικρή διαφορά της μέσης βαθμολογίας αποδεικνύει το γεγονός ότι η αποτελεσματικότητα διατηρείται ανεξάρτητα της μεθόδου εισόδου.

7.4.5 Ερμηνεία αποτελεσμάτων και περιορισμοί

Η μείωση της απόδοσης από την διαφορά της μορφής εισόδου είναι φυσιολογικό φαινόμενο, διότι το φωνητικό ερώτημα ακολουθεί διαφορετική διασωλήνωση από αυτό του γραπτού ερωτήματος. Σε αυτήν την διαφορετική διασωλήνωση υπάρχουν επιπλέον στάδια επεξεργασίας τα οποία υπάρχει η πιθανότητα να οδηγήσουν σε σφάλματα στην συνολική διασωλήνωση του ερωτήματος. Οι πιθανές αιτίες οι οποίες θα μπορούσαν να οδηγήσουν σε εσφαλμένη μετατροπή της ομιλίας σε κείμενο είναι η λάθος αναγνώριση των λέξεων, η μη αναγνώριση των ονομάτων, των τίτλων των μαθημάτων και των συντομογραφιών, η ύπαρξη θορύβου στον περιβάλλοντα χώρο και ο τρόπος αλλά και η ένταση της ομιλίας του αφηγητή του ερωτήματος.

Αξίζει όμως να σημειωθεί ότι η απόδοση του συστήματος μοιάζει συνολικά παρόμοια και τα σφάλματα τα οποία τελικά οδηγούν στην μη ικανοποιητική απάντηση του συστήματος κατά πάσα πιθανότητα δεν αφορούν τον τρόπο με τον οποίο τίθεται ένα ερώτημα αλλά αφορούν διαφορετικούς παράγοντες. Οι λόγοι οι οποίοι θα μπορούσαν να οδηγήσουν σε τέτοια αποτελέσματα είναι η απουσία

της απάντησης από την βάση γνώσης, η αποτυχία της ανάκτησης του κατάλληλου chunk από το σύστημα ή η υπερβολική πληροφορία η οποία καταφθάνει στο LLM με αποτέλεσμα αυτό να μην μπορεί να δημιουργήσει κατάλληλη τελική απάντηση.

Η συγκεκριμένη αξιολόγηση παρουσιάζει το εξής χαρακτηριστικό, βασίζεται σε 50 ερωτήσεις. Αυτό το στοιχείο θεωρείται κατάλληλο για μία πειραματική αποτίμηση του ρομποτικού συμβούλου σπουδών αλλά δεν είναι τόσο εύχρηστο για μία πλήρης στατιστική γενίκευση. Αυτό ισχύει διότι για να επεκταθεί αυτή η πειραματική αποτίμηση σε στατιστική γενίκευση απαιτείται μεγαλύτερο πλήθος ερωτήσεων, μεγαλύτερο πλήθος χρηστών και ποικίλα περιβάλλοντα χρήσης, όπως είναι για παράδειγμα ένας εξωτερικός χώρος με περισσότερο θόρυβο περιβάλλοντος. Οι πιθανές αιτίες μερικώς ή μη ικανοποιητικών απαντήσεων παρουσιάζονται στον Πίνακα 7.4.

Πίνακας 7.4: Πιθανές αιτίες μερικώς ή μη ικανοποιητικών απαντήσεων

Στάδιο	Πιθανός περιορισμός	Επίδραση
Καταγραφή ήχου	Θόρυβος ή χαμηλή ένταση ομιλίας	Αλλοίωση φωνητικού σήματος
Speech-to-Text	Λάθος σε όρους, ονόματα ή συντομογραφίες	Μη ακριβές query
Query normalization	Ανεπαρκής διόρθωση οντότητας	Διατήρηση αναγνώρισης λάθους
Retrieval	Ανάκτηση λιγότερο σχετικών chunks	Ελλιπές πλαίσιο στο LLM
Knowledge base	Απουσία ή παλαιότητα πληροφορίας	Αδυναμία απάντησης
LLM generation	Υπερβολική σύνοψη ή ασάφεια	Μερικώς πλήρης απάντηση

7.5 Συμπεράσματα

Η συγκεκριμένη υλοποίηση του συστήματος διερευνά την αξιοποίηση ενός κοινωνικού ρομπότ όπως το ρομπότ NAO V6 για τον σκοπό του ρομποτικού συμβούλου σπουδών για το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Ουσιαστικά εξερευνά τον αποδοτικό συνδυασμό διάφορων τεχνοτροπιών με σκοπό την παροχή υπηρεσιών για τους φοιτητές του συγκεκριμένου τμήματος. Υλοποιείται ένα ολοκληρωμένο σύστημα το οποίο αποτελείται από πολλά επιμέρους, τα οποία επιμέρους τμήματα μπορεί να περιέχουν και επιπλέον επίπεδα. Αυτό το ολοκληρωμένο σύστημα συνδυάζει τεχνικές αναγνώρισης ομιλίας, τεχνικές επεξεργασίας φωνής, ένα RAG σύστημα, ένα LLM, ένα σύστημα παραγωγής απάντησης και συνδυασμό όλων των παραπάνω με μία διεπαφή χρήστη μέσω του NAO V6. Όλα τα παραπάνω στοιχεία δημιουργούν ένα ενιαίο σύστημα το οποίο χάρη της τεχνητής νοημοσύνης σε συνδυασμό με τα κοινωνικά ρομπότ δημιουργείται τελικά μία

διεπαφή μεταξύ ρομπότ και ανθρώπων η οποία συμβάλλει στην φυσικότητα της επικοινωνίας αυτών και στην επίτευξη αλληλεπίδρασης.

7.6 Επίλογος

Στο κεφάλαιο αυτό αναλύθηκε συνολικά το σύστημα και η σχεδίαση του Ρομποτικού Συμβούλου Σπουδών για το τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙΠΑΕ. Η συνολική υλοποίηση αυτού του συστήματος βασίστηκε στην συνεργασία και διασύνδεση διαφόρων επιμέρους συστημάτων. Για παράδειγμα αξιοποιήθηκαν τεχνολογίες και μέθοδοι τεχνητής νοημοσύνης, μεγάλα γλωσσικά μοντέλα, συστήματα RAG, τεχνικές φωνητικής αναγνώρισης και αξιοποιήθηκαν και κοινωνικά ρομπότ και συγκεκριμένα το ρομπότ NAO V6. Συνδέοντας λοιπόν τις παραπάνω τεχνολογίες δημιουργήθηκε ένα ενιαίο σύστημα το οποίο λειτουργεί αυτόνομα και μπορεί να δεχτεί διάφορα ερωτήματα, είτε με την χρήση ομιλίας είτε ακόμη και με γραπτή μορφή εισόδου, να αναζητήσει και να ανάκτηση την κατάλληλη γνώση ανάλογα με το εκάστοτε ερώτημα και τελικά να παρουσιάσει στον χρήστη μέσω του ρομπότ το οποίο προαναφέρθηκε την τελική απάντηση σε μορφή ομιλίας.

Τα διακριτά στοιχεία τα οποία δημιουργούν αυτή την ενιαία αρχιτεκτονική αποτελούνται είναι πολλά. Αρχικά στην παρούσα αρχιτεκτονική το RAG σύστημα είναι ένα από τα βασικότερα στοιχεία της υλοποίησης και θέτει τις κατάλληλες βάσεις για να λειτουργήσει το υπόλοιπο σύστημα, καθώς είναι το σύστημα το οποίο χρησιμοποιείται για να δημιουργηθεί η τελική απάντηση. Πριν από την δημιουργία όμως της απάντησης στο ίδιο το RAG εμπεριέχεται και η μέθοδο ανάκτησης πληροφορίας, στην οποία το RAG χρησιμοποιεί μία υβριδική μέθοδο η οποία συνδυάζει αραιή και πυκνή αναζήτηση στα τμήματα κειμένου τα οποία ονομάζονται chunks.

Παράλληλα η αρχιτεκτονική του συστήματος εμπεριέχει και το σύστημα Speech To Text το οποίο μετατρέπει την ηχητική είσοδο σε κείμενο. Παράλληλα πριν και μετά την καταγραφή του ήχου ακολουθούνται διάφορες τεχνικές οι οποίες χρησιμοποιούνται για την καλύτερη αποκρυπτογράφηση της ομιλίας και για την αύξηση της χωρητικότητας όσον αφορά τον τρόπο με τον οποίο καταγράφεται. Όλα αυτά τα συστήματα όπως έχει αναφερθεί και στα προηγούμενα κεφάλαια συνδέονται με το ρομπότ NAO V6, το οποίο αποτελεί ουσιαστικά το μέσο αλληλεπίδρασης μεταξύ όλων των συστημάτων και του χρήστη.

Συνολικά σε αυτό το κεφάλαιο γίνεται και μία αξιολόγηση του συνολικού συστήματος σχετικά με τις απαντήσεις τις οποίες επιστρέφει τελικά το σύστημα συνολικά αλλά και εξειδικευμένα ανάλογα με το εάν χρησιμοποιείται ως μορφή εισόδου η ομιλία ή ο γραπτός λόγος. Χάρη σε αυτό το πείραμα αξιολογήθηκε το γεγονός ότι το 90% των ερωτημάτων που γίνονται στο σύστημα επιστρέφουν μία ικανοποιητική απάντηση ανεξάρτητα από την πηγή εισόδου της ερώτησης. Υπάρχει βέβαια ένα πλεονέκτημα του γραπτού λόγου καθώς παρατηρείται το γεγονός ότι παρουσιάζονται περισσότερες ολοκληρωτικά σωστές απαντήσεις συγκριτικά με την φωνητική μορφή εισόδου. Ουσιαστικά όμως οι αλλαγές είναι τόσο μικρές, οι οποίες επηρεάζουν τελικά ελάχιστα την ποιότητα της συνολικής λειτουργίας του RAG συστήματος.

Η παρούσα διπλωματική εργασία και τα συγκεκριμένα πειράματα παρουσιάζουν διάφορα θετικά αποτελέσματα όμως υπάρχουν αρκετές μελλοντικές βελτιώσεις οι οποίες θα βελτιώναν την εμπειρία του χρήστη από την αλληλεπίδραση με το σύστημα. Ενώ ακόμα υπάρχει η ανάγκη για επιπλέον ελέγχους στο σύστημα με σκοπό τα διάφορα πειραματικά συμπεράσματα τα οποία προέκυψαν να αναπτυχθούν σε μία επιστημονική στατιστική γενίκευση.

Επιπρόσθετα η εργασία αναδεικνύει το γεγονός ότι με την κατάλληλη διασύνδεση των συστημάτων μπορεί να δημιουργηθεί ένα σύστημα κατάλληλο για την εξυπηρέτηση ακαδημαϊκών αναγκών πληροφόρησης με φυσική και φιλική μορφή προς τον τελικό χρήστη. Μία τέτοια εφαρμογή μπορεί να θέσει τα θεμέλια ώστε μετέπειτα να γίνουν μελλοντικές επεκτάσεις οι οποίες θα αποσκοπούν στην πιο προσβάσιμη και άμεση απάντηση των αναγκών τους.

Κεφάλαιο 8ο: Μελλοντικές επεκτάσεις

8.1 Εισαγωγή

Στο συγκεκριμένο κεφάλαιο θα γίνουν αναφορές σε κάποιες μελλοντικές επεκτάσεις του συστήματος. Στους τρόπους με τους οποίους το ήδη υπάρχον τμήμα μπορεί να εξελιχθεί και να βελτιώσει την συνολική απόδοση της εφαρμογής.

8.2 Δυναμικός Εμπλουτισμός Βάσης Γνώσης

Μία από τις πιο βασικές μελλοντικές προσθήκες η οποία θα βοηθούσε συνολικά στην επίτευξη όσο το δυνατόν καλύτερης επικοινωνίας μεταξύ ανθρώπου και ρομπότ θα ήταν η προσθήκη πληροφοριών μέσω crawlers σε ιστοσελίδες της σχολής. Για παράδειγμα θα μπορούσε να έχει πρόσβαση στο Apps και η υλοποίηση να μπορεί να βλέπει της ανακοινώσεις και παράλληλα θα μπορεί ο χρήστης να ψάχνει με αντίστοιχο τρόπο όπως το Apps για συγκεκριμένες ανακοινώσεις καθηγητών ή για ανακοινώσεις για συγκεκριμένα μαθήματα. Κάτι τέτοιο σίγουρα θα εκτόξευε την πολυπλοκότητα της διπλωματικής εργασίας αλλά παράλληλα θα παρείχε πολλές χρήσιμες δυνατότητες στον χρήστη της εφαρμογής.

8.3 Μετάβαση σε υποδομή Server

Επιπλέον ένα πολύ σημαντικό κομμάτι το οποίο θα αναβαθμίσει ιδιαίτερα την ποιότητα της εφαρμογής αλλά και συνολικά την εμπειρία του χρήστη είναι η τοποθέτηση όλων των λειτουργιών από έναν κεντρικό υπολογιστή σε έναν κατάλληλο server. Μία τέτοια λειτουργία θα αναβάθμιζε σημαντικά την γενικότερη λειτουργία και τις δυνατότητες της εφαρμογής. Αυτό θα συνέβαινε καθώς αντί για τους περιορισμένους υπολογιστικούς πόρους του κεντρικού υπολογιστή θα χρησιμοποιούνταν ένας απομακρυσμένος υπολογιστής με καλύτερα τεχνικά χαρακτηριστικά. Αυτό το στοιχείο θα πρόσδιδε καλύτερους χρόνους απόκρισης. Επίσης μία τέτοια μελλοντική επέκταση θα έκανε το σύστημα πιο ανεξάρτητο καθώς αντί για την ανάγκη σύνδεσης με τον τοπικό υπολογιστή θα συνδεόταν το σύστημα με έναν απομακρυσμένο. Χωρίς να υπάρχει πλέον η ανάγκη διασύνδεσης μεταξύ του υπολογιστή και του NAO V6.

8.4 Επέκταση κινητικής συμπεριφοράς

Παράλληλα άλλο ένα στοιχείο το οποίο χρειάζεται βελτίωση στην συγκεκριμένη εφαρμογή είναι αυτό της κινητικής συμπεριφοράς του NAO V6. Μια σημαντική λοιπόν μελλοντική επέκταση είναι η κατάλληλη κινησιολογία ανάλογα με την απάντηση. Για παράδειγμα εάν ο χρήστης ρωτήσει κάτι στο οποίο ο ρομποτικός σύμβουλος σπουδών δεν μπορεί να δώσει την απάντηση τότε να υπάρχει διαφορετική αντίδραση στις κινήσεις του NAO V6 με όταν ο χρήστης ρωτήσει μία άλλη ερώτηση στην οποία το σύστημα μπορεί να δώσει μία απάντηση. Ένα τέτοιο χαρακτηριστικό προσδίδει ιδιαίτερα στην αξία της εφαρμογής καθώς η αλληλεπίδραση μεταξύ ρομπότ και ανθρώπου γίνεται περισσότερο πολυδιάστατη. Μία τέτοια προσθήκη είναι υψίστης σημασίας για μία διπλωματική εργασία ενός ρομποτικού συμβούλου σπουδών. Αυτό συμβαίνει διότι μία τέτοια διπλωματική εργασία βασίζεται σε μεγάλο βαθμό στα social robotics, στο RAG σύστημα και γενικότερα στην καλύτερη επίτευξη αλληλεπίδρασης μεταξύ ρομπότ και ανθρώπων.

8.5 Εκτενότερη Πειραματική Αξιολόγηση

Επιπρόσθετα μία σημαντική από τις ενδεχόμενες μελλοντικές επεκτάσεις αφορά την συνολική πειραματική αξιολόγηση του συστήματος ρομποτικού συμβούλου σπουδών του Τμήματος Μηχανικών

Πληροφορικής και Ηλεκτρονικών Συστημάτων. Ουσιαστικά μία τέτοια προσθήκη θα χωριζόταν σε τρία επιμέρους τμήματα. Το προτεινόμενο πλαίσιο μελλοντικής αξιολόγησης παρουσιάζεται στον Πίνακα 8.1.

- Το πρώτο τμήμα αφορά την αξιολόγηση της διαδικασίας ανάκτησης η οποία θα παρουσίαζε το Precision, το Recall και το MRR. Στο οποίο θα παρουσιαζόταν ο αριθμός των σχετικών αρχείων από τα πρώτα K αρχεία που επιστρέφονται, το συγκεκριμένο ονομάζεται Precision. Παράλληλα χάρις στην συγκεκριμένη μελλοντική προσθήκη μέσω του Recall θα βλέπαμε το πλήθος των σχετικών από τα συνολικά διαθέσιμα κατάφερε να επιστρέψει το σύστημα. Ενώ μέσω του MRR (Mean Reciprocal Rank) θα εμφανιζόταν η θέση και πόσο κοντά στις πρώτες είναι το πρώτο σημαντικά σχετικό αρχείο από αυτά τα οποία επιστράφηκαν συνολικά.
- Το δεύτερο τμήμα αφορά την αξιολόγηση της αναγνώρισης ομιλίας. Στο παρόν τμήμα θα εφαρμοζόταν μία μέθοδος Word Error Rate στην οποία θα γινόταν μία σύγκριση της μετατροπής της λέξης η οποία προέκυψε από το μοντέλο Speech To Text συγκριτικά με την πραγματική λέξη η οποία ήταν η πρόθεση να ειπωθεί. Ουσιαστικά αυτό το κομμάτι είναι ιδιαίτερης υψηλής σημασίας καθώς κατά αυτόν τον τρόπο θα εντοπιστούν σε βάθος τα κομμάτια στα οποία δυσκολεύεται το Speech To Text. Αυτού του είδους προσθήκη της αξιολόγησης συμβάλλει σημαντικά στην επέκταση των δυνατοτήτων της εφαρμογής ενώ μπορούν να προκύψουν κατάλληλα συμπεράσματα για την ικανότητα της εφαρμογής ώστε να εντοπίζει συγκεκριμένα μοτίβα στην μετατροπή αλλά και να παρατηρεί τις εκάστοτε αλλαγές όταν οι συνθήκες περιβάλλοντος αλλάζουν.
- Το τρίτο και τελευταίο τμήμα αξιολόγησης θα βασιζόταν στη συνολική εμπειρία των χρηστών και την ικανοποίηση τους από την απάντηση η οποία θα παρουσιάζεται από το ρομπότ. Ουσιαστικά θα υπήρχε ένα ερωτηματολόγιο το οποίο θα δινόταν στον χρήστη με σκοπό να γίνει μία αξιολόγηση στην συνολική λειτουργία του ρομπότ και της εφαρμογής με βάση μόνο την εμπειρία του εκάστοτε χρήστη. Για παράδειγμα το ερωτηματολόγιο μπορεί να περιέχει ερωτήσεις όσον αφορά το πόσο σωστή του φάνηκε η απάντηση. Με ποιόν βαθμό βαθμολογεί την αλληλεπίδραση του ρομπότ με αυτόν. Πόσο κατάλληλο ή ακατάλληλο είναι το ύφος του ρομπότ συγκριτικά με τις ανάγκες του ρομποτικού συμβούλου σπουδών. Πόσο πλήρης είναι η συνολική απάντηση. Αν η ταχύτητα απόκρισης είναι κατάλληλη ώστε να διατηρείται η φυσικότητα της αλληλεπίδρασης μεταξύ του ρομπότ και του ανθρώπου και σε τι βαθμό συμβαίνει αυτό. Τέλος ένα άλλο στοιχείο το οποίο θα εξετασθεί είναι ο βαθμός με τον οποίο μία τέτοια εφαρμογή αξίζει να υλοποιηθεί με την χρήση του NAO V6 ή αν αρκούσε ή όχι η υλοποίηση της με την χρήση ενός απλού Chatbot, το οποίο δεν συνδυάζει τα χαρακτηριστικά του NAO V6.

Πίνακας 8.1: Προτεινόμενο πλαίσιο μελλοντικής αξιολόγησης

Επίπεδο αξιολόγησης	Δείκτες ή μέθοδος	Επεξήγηση Μέτρησης
Ανάκτηση γνώσης	Precision@K, Recall@K, MRR	Ποιότητα και κατάταξη ανακτημένων chunks
Αναγνώριση ομιλίας	Word Error Rate	Ακρίβεια μετατροπής ομιλίας σε κείμενο
Εμπειρία χρήστη	Ερωτηματολόγιο Likert	Ικανοποίηση, χρησιμότητα και φυσικότητα
Απόκριση συστήματος	Χρόνος STT, retrieval, LLM και TTS	Καθυστέρηση σε κάθε στάδιο διαλόγου

8.6 Επέκταση κινητικής συμπεριφοράς

Πίνακας 8.2: Προτεραιοποίηση προτεινόμενων μελλοντικών επεκτάσεων

Επέκταση	Αναμενόμενο όφελος	Τεχνική δυσκολία	Προτεραιότητα
Εμπλουτισμός FAQ και βάσης γνώσης	Υψηλό	Χαμηλή	Υψηλή
Crawlers επίσημων ανακοινώσεων	Υψηλό	Μεσαία	Υψηλή
Αξιολόγηση με περισσότερους χρήστες	Υψηλό	Μεσαία	Υψηλή
Server / GPU υποδομή	Υψηλό	Υψηλή	Μεσαία
Κινησιολογία βάσει κατάστασης απάντησης	Μεσαίο προς υψηλό	Μεσαία	Μεσαία
Υποστήριξη πολλών NAO	Μεσαίο	Υψηλή	Χαμηλή

Οι επεκτάσεις οι οποίες προτάθηκαν στο συγκεκριμένο κεφάλαιο έχουν διαφορετική προτεραιότητα και η προτεραιότητα αυτών επηρεάζεται από διάφορους παράγοντες. Η επέκταση η οποία αναμένεται να προσδώσει το μεγαλύτερο όφελος είναι ο εμπλουτισμός της βάσης γνώσης, είτε μιλάμε για το αρχείο των συχνών ερωτήσεων, είτε μιλάμε για τους απευθείας crawlers στις ιστοσελίδες του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Οι συγκεκριμένες βελτιώσεις μπορούν να συμβάλλουν σημαντικά τόσο στην ποιότητα των απαντήσεων όσο και στις περιπτώσεις χρήσης της ίδιας της εφαρμογής.

Σε δεύτερο στάδιο αλλά εξίσου σημαντικό με την πρώτη ομάδα αλλαγών ακολουθεί η αξιολόγηση της εφαρμογής με περισσότερους χρήστες. Μία τέτοια προσθήκη θα μπορούσε να αυξήσει την αξιοπιστία της εφαρμογής σε σημαντικό βαθμό, ενώ παράλληλα δεν επηρεάζουν σημαντικά την αρχιτεκτονική του συνολικού συστήματος. Παράλληλα μία αλλαγή η εντάξη αυτής της αρχιτεκτονικής σε έναν server μπορεί να βελτιώσει την αυτονομία του συστήματος και τους χρόνους απόκρισης αυτού. Τέλος ακολουθούν οι αλλαγές οι οποίες αφορούν το ρομπότ ΝΑΟ όπου εάν προστεθούν θα βελτιωθεί σημαντικά η εμπειρία της αλληλεπίδρασης μεταξύ των χρηστών.

Στο σύνολο τους οι μελλοντικές επεκτάσεις αποσκοπούν να βελτιώσουν το συνολικό σύστημα το οποίο δημιουργήθηκε στην παρούσα διπλωματική εργασία και να του προσδώσουν περισσότερα στοιχεία αυτονομίας, μεγαλύτερο όγκο γνώσης αλλά και να προσθέσουν καταλληλότερα στοιχεία αλληλεπίδρασης του ρομπότ ως προς τις ανάγκες υποστήριξης των διαφόρων χρηστών της. Η προτεραιοποίηση των προτεινόμενων μελλοντικών επεκτάσεων παρουσιάζεται επίσης και στον Πίνακα 8.2.

BIBΛΙΟΓΡΑΦΙΑ

- [1] M. Yousef and M. A. Torad, “A Treatise On Conversational AI Agents: Learning From Humans’ Behaviour As A Design Outlook,” in *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, Ras Al Khaimah, United Arab Emirates: IEEE, Nov. 2019, pp. 1–4. doi: 10.1109/ICECTA48151.2019.8959585.
- [2] N. Fragakis, G. Trichopoulos, and G. Caridakis, “Embodied and Intelligent Approaches to Cultural Information Retrieval,” in *2024 19th International Workshop on Semantic and Social Media Adaptation & Personalization (SMAP)*, Athens, Greece: IEEE, Nov. 2024, pp. 178–181. doi: 10.1109/SMAP63474.2024.00041.
- [3] H.-T. Nguyen, Q.-D. Ngo, and Q.-D. Dang, “An Intent-Filtered Retrieval-Augmented Generation Chatbot for University Admissions in Vietnam,” in *2024 International Conference on Advanced Technologies for Communications (ATC)*, Ho Chi Minh City, Vietnam: IEEE, Oct. 2024, pp. 928–933. doi: 10.1109/ATC63255.2024.10908188.
- [4] Anna-Sophie Brauner, “Runaround by Isaac Asimov and the significance of the Three Laws of Robotics in today’s world,” 2016, doi: 10.13140/RG.2.2.16275.14884.
- [5] Md. Akhtaruzzaman and A. A. Shafie, “Evolution of Humanoid Robot and contribution of various countries in advancing the research and development of the platform,” in *ICCAS 2010*, Gyeonggi-do: IEEE, Oct. 2010, pp. 1021–1028. doi: 10.1109/ICCAS.2010.5669646.
- [6] C. Breazeal, *Designing Sociable Robots*. The MIT Press, 2002. doi: 10.7551/mitpress/2376.001.0001.
- [7] F. Hegel, C. Muhl, B. Wrede, M. Hielscher-Fastabend, and G. Sagerer, “Understanding Social Robots,” in *2009 Second International Conferences on Advances in Computer-Human Interactions*, Cancun, Mexico: IEEE, Feb. 2009, pp. 169–174. doi: 10.1109/ACHI.2009.51.
- [8] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, “The intelligent ASIMO: system overview and integration,” in *IEEE/RSJ International Conference on Intelligent Robots and System*, Lausanne, Switzerland: IEEE, 2002, pp. 2478–2483. doi: 10.1109/IRDS.2002.1041641.
- [9] D. Gouaillier *et al.*, “Mechatronic design of NAO humanoid,” in *2009 IEEE International Conference on Robotics and Automation*, Kobe: IEEE, May 2009, pp. 769–774. doi: 10.1109/ROBOT.2009.5152516.
- [10] M. A. Goodrich and A. C. Schultz, “Human–Robot Interaction: A Survey,” *Foundations and Trends® in Human–Computer Interaction*, vol. 1, no. 3, pp. 203–275, Jan. 2008, doi: 10.1561/11000000005.
- [11] M. A. Goodrich and D. R. Olsen, “Seven principles of efficient human robot interaction,” in *SMC’03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, Washington, DC, USA: IEEE, 2003, pp. 3942–3948. doi: 10.1109/ICSMC.2003.1244504.
- [12] J. W. Crandall, M. A. Goodrich, D. R. Olsen, and C. W. Nielsen, “Validating Human–Robot Interaction Schemes in Multitasking Environments,” *IEEE Trans. Syst., Man, Cybern. A*, vol. 35, no. 4, pp. 438–449, Jul. 2005, doi: 10.1109/TSMCA.2005.850587.
- [13] R. Karakaya, F. Camara, and S. Perinpanayagam, “Human Interactions with Autonomous Mobile Robots in Public Spaces: A Survey,” in *2025 34th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, Eindhoven, Netherlands: IEEE, Aug. 2025, pp. 1558–1565. doi: 10.1109/RO-MAN63969.2025.11217738.
- [14] A. Fujii and J. Kristiina, “Open Source System Integration Towards Natural Interaction with Robots,” in *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Sapporo, Japan: IEEE, Mar. 2022, pp. 768–772. doi: 10.1109/HRI53351.2022.9889582.
- [15] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, and N. Hagita, “Adapting Robot Behavior for Human–Robot Interaction,” *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 911–916, Aug. 2008, doi: 10.1109/TRO.2008.926867.
- [16] A. I. Kapteijns and M. De Graaf, “Gender-Emotion Stereotypes in HRI: The Effects of Robot Gender and Speech Act on Evaluations of a Robot,” in *2024 33rd IEEE International*

- Conference on Robot and Human Interactive Communication (ROMAN)*, Pasadena, CA, USA: IEEE, Aug. 2024, pp. 924–929. doi: 10.1109/RO-MAN60168.2024.10731294.
- [17] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*, Third edition, Global edition. in Prentice Hall series in artificial intelligence. Boston Columbus Indianapolis: Pearson, 2016.
 - [18] L. Aldhafeeri *et al.*, “Generative AI Chatbots Across Domains: A Systematic Review,” *Applied Sciences*, vol. 15, no. 20, p. 11220, Oct. 2025, doi: 10.3390/app152011220.
 - [19] S. Park, A. Kim, S. Lee, H. Lee, C. Kamyod, and C. G. Kim, “Design of REST API Client for Conversational Agent using Large Language Model with Open API System,” in *2024 IEEE/ACIS 22nd International Conference on Software Engineering Research, Management and Applications (SERA)*, Honolulu, HI, USA: IEEE, May 2024, pp. 55–58. doi: 10.1109/SERA61261.2024.10685639.
 - [20] D. Drašković, “Integration of Ai Tools into an Ai-Driven Software System to Make Learning Programming Easier,” in *2025 IEEE Global Engineering Education Conference (EDUCON)*, London, United Kingdom: IEEE, Apr. 2025, pp. 1–6. doi: 10.1109/EDUCON62633.2025.11016313.
 - [21] A. Bono, K. Brameld, L. D’Alfonso, and G. Fedele, “Open Access NAO (OAN): a ROS2-based software framework for HRI applications with the NAO robot,” in *2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN)*, Pasadena, CA, USA: IEEE, Aug. 2024, pp. 522–527. doi: 10.1109/RO-MAN60168.2024.10731252.
 - [22] Y. Ma *et al.*, “A Survey of Retrieval-Augmented Generation (RAG) for Large Language Models,” in *2025 International Conference on Trustworthy Big Data and Artificial Intelligence (ICTBAI)*, Tokyo, Japan: IEEE, Aug. 2025, pp. 7–13. doi: 10.1109/ICTBAI68361.2025.00008.
 - [23] P. Lewis *et al.*, “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” Apr. 12, 2021, *arXiv*: arXiv:2005.11401. doi: 10.48550/arXiv.2005.11401.
 - [24] R. Patil, S. Boit, V. Gudivada, and J. Nandigam, “A Survey of Text Representation and Embedding Techniques in NLP,” *IEEE Access*, vol. 11, pp. 36120–36146, 2023, doi: 10.1109/ACCESS.2023.3266377.
 - [25] H. Soudani, E. Kanoulas, and F. Hasibi, “Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge,” in *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, Tokyo Japan: ACM, Dec. 2024, pp. 12–22. doi: 10.1145/3673791.3698415.
 - [26] G. Budakoglu and H. Emekci, “Unveiling the Power of Large Language Models: A Comparative Study of Retrieval-Augmented Generation, Fine-Tuning, and Their Synergistic Fusion for Enhanced Performance,” *IEEE Access*, vol. 13, pp. 30936–30951, 2025, doi: 10.1109/ACCESS.2025.3542334.
 - [27] R. Ghaleb, M. Dharmaraj, S. Prayaga, and T. Banka, “Demo: Conversational AI Agent for ML Infrastructure Monitoring and Analysis,” in *2025 IEEE 33rd International Conference on Network Protocols (ICNP)*, Seoul, Korea, Republic of: IEEE, Sep. 2025, pp. 1–3. doi: 10.1109/ICNP65844.2025.11192362.
 - [28] K. Rajkumar, D. Devarajan, K. S J, and S. Usharani, “Towards Reliable AI-Based Assessment: Modular Scoring and Hallucination Control,” in *2025 Fourth International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, Villupuram, India: IEEE, Nov. 2025, pp. 1–6. doi: 10.1109/ICSTSN67075.2025.11398009.