Contents

[ΚΕΦΑΛΑΙΟ 1: Εισαγωγή 3](#_Toc199150518)

[1.1 Σκοπός και κίνητρο της εργασίας 3](#_Toc199150519)

[1.2 Στόχοι και συνεισφορές 3](#_Toc199150520)

[1.3 Δομή της εργασίας 3](#_Toc199150521)

[ΚΕΦΑΛΑΙΟ 2: Θεωρητικό Υπόβαθρο 3](#_Toc199150522)

[2.1 Μεγάλα Γλωσσικά Μοντέλα (LLMs): αρχές και περιορισμοί 3](#_Toc199150523)

[2.2 Επεξεργασία και Κατανόηση Φυσικής Γλώσσας (NLP/NLU) 4](#_Toc199150524)

[2.3 Retrieval‑Augmented Generation (RAG): έννοιες και workflow 6](#_Toc199150525)

[2.4 Prompt Engineering 9](#_Toc199150526)

[ΚΕΦΑΛΑΙΟ 3: Βιβλιογραφική Ανασκόπηση 11](#_Toc199150527)

[3.2 RAG σε Εκπαιδευτικά Πλαίσια 13](#_Toc199150528)

[3.2.1 Αρχιτεκτονικές και Ροές Εργασίας RAG 13](#_Toc199150529)

[3.2.2 Ενσωμάτωση Εξωτερικής Γνώσης Οδηγού Σπουδών 13](#_Toc199150530)

[3.3 Τομείς Εφαρμογών Εικονικών Βοηθών Βασισμένων σε RAG 14](#_Toc199150531)

[3.3.1 Εικονικοί Βοηθοί Διδασκαλίας και Μάθησης 14](#_Toc199150532)

[3.3.2 Εικονικοί Βοηθοί Διοικητικής Υποστήριξης 15](#_Toc199150533)

[3.4 Τεχνικές Λεπτομέρειες Υλοποίησης 15](#_Toc199150534)

[3.4.1 Μοντέλα Ενσωμάτωσης και Βάσεις Δεδομένων Διανυσμάτων 15](#_Toc199150535)

[3.4.2 Επεξεργασία Ερωτημάτων και Μηχανική Prompt 16](#_Toc199150536)

[3.4.3 Πλαίσια Συστήματος και Εργαλεία 17](#_Toc199150537)

[3.5 Μεθοδολογίες Αξιολόγησης 17](#_Toc199150538)

[3.5.1 Ποσοτικές Μετρικές 17](#_Toc199150539)

[3.5.2 Ποιοτικές Μετρικές (Έρευνες Χρηστών, Κλίμακες Χρηστικότητας) 18](#_Toc199150540)

[3.6 Ηθικές, Θέματα Απορρήτου και Ασφάλειας 19](#_Toc199150541)

[3.6.1 Απόρρητο Δεδομένων και Ανωνυμοποίηση 19](#_Toc199150542)

[3.6.2 Αρχιτεκτονικές Ασφάλειας και Έλεγχος Πρόσβασης 20](#_Toc199150543)

[3.6.3 Ηθικά Μέτρα και Υπεύθυνη Τεχνητή Νοημοσύνη 20](#_Toc199150544)

[3.7 Ανάλυση Ελλείψεων: Ψηφιακοί Βοηθοί RAG στα Ελληνικά 21](#_Toc199150545)

[3.7.1 Επισκόπηση Υπαρχόντων Ελληνικών Εκπαιδευτικών Chatbots 21](#_Toc199150546)

[3.7.2 Γλωσσικές και Πολιτισμικές Προκλήσεις στα Ελληνικά 22](#_Toc199150547)

[3.7.3 Έλλειψη Συστημάτων RAG Καθοδηγούμενων από Οδηγούς Μελέτης στα Ελληνικά 22](#_Toc199150548)

[ΚΕΦΑΛΑΙΟ 4: Σχεδίαση Ευφυούς Συστήματος 23](#_Toc199150549)

[4.0 Εισαγωγή 23](#_Toc199150550)

[4.1 Αρχιτεκτονική Συστήματος 24](#_Toc199150551)

[4.1.2 Διεπαφή Χρήστη (Frontend) 25](#_Toc199150552)

[4.1.3 Business Logic Layer (Backend) 25](#_Toc199150553)

[4.1.5 Υποδομές & DevOps 26](#_Toc199150554)

[4.2 Knowledge Base 26](#_Toc199150555)

[4.2.1 Πηγές Δεδομένων 26](#_Toc199150556)

[4.2.2 Προεπεξεργασία & Καθαρισμός 26](#_Toc199150557)

[4.2.3 Δημιουργία Embeddings & Storage 29](#_Toc199150558)

[4.2.4 Διαχείριση Ανανεώσεων 31](#_Toc199150559)

[ΚΕΦΑΛΑΙΟ 5: Πειραματικά Αποτελέσματα & Αξιολόγηση 35](#_Toc199150560)

[5.1 Τυπικοί Χρήστες / Τυπικά Σενάρια 35](#_Toc199150561)

[5.2 Σύγκριση Chunker 37](#_Toc199150562)

[5.3 Embeddings 40](#_Toc199150563)

[5.4 Threshold και ποιότητα απάντησης 43](#_Toc199150564)

[5.5 Συμπεράσματα 48](#_Toc199150565)

[ΚΕΦΑΛΑΙΟ 6: Συμπεράσματα & Μελλοντικές Επεκτάσεις 48](#_Toc199150566)

[6.1 Συνοπτικά συμπεράσματα 48](#_Toc199150567)

[6.2 Περιορισμοί παρούσας υλοποίησης 49](#_Toc199150568)

[6.3 Προτάσεις για μελλοντική έρευνα και βελτιώσεις 49](#_Toc199150569)

# ΚΕΦΑΛΑΙΟ 1: Εισαγωγή

## 1.1 Σκοπός και κίνητρο της εργασίας

## 1.2 Στόχοι και συνεισφορές

1.2.1 Καινοτομίες της εργασίας

– Εξολοκλήρου στα ελληνικά

– Καινοτόμο pipeline προεπεξεργασίας δεδομένων

– Σύγκριση 10 embedding αλγορίθμων σε sub‑dataset

## 1.3 Δομή της εργασίας

# ΚΕΦΑΛΑΙΟ 2: Θεωρητικό Υπόβαθρο 2.1 Μεγάλα Γλωσσικά Μοντέλα (LLMs): αρχές και περιορισμοί

Τα Μεγάλα Γλωσσικά Μοντέλα (LLMs) είναι νευρωνικά δίκτυα μεγάλης κλίμακας που εκπαιδεύονται πάνω σε τεράστια σώματα κειμένου με αυτο-εποπτευόμενη μάθηση, ώστε να προβλέπουν την επόμενη λέξη σε μια ακολουθία και να παράγουν συνεκτικό, ανθρώπινου τύπου κείμενο. Η σημαντική αύξηση του αριθμού των παραμέτρων (από μερικές δεκάδες εκατομμύρια σε δεκάδες ή και εκατοντάδες δισεκατομμύρια) σε συνδυασμό με ποικίλα και εκτενή σύνολα δεδομένων διευρύνει την ικανότητα των μοντέλων να αντιλαμβάνονται σύνθετες γλωσσικές δομές, ενώ παράλληλα επιτρέπει τη μάθηση νέων εργασιών με ελάχιστα παραδείγματα (few-shot learning) και πολλές φορές χωρίς καθόλου επιπλέον εκπαίδευση (zero-shot learning) [1][2]. Παρά την εμφάνιση των παραδοσιακών στατιστικών μοντέλων n-gram δεκαετίες πριν, όπου η γλωσσική μοντελοποίηση οριζόταν ως κατανομή πιθανοτήτων πάνω σε ακολουθίες λέξεων (π.χ. P(wₙ|wₙ₋₁,…,wₙ₋ₖ+1)) [3], τα νευρωνικά μοντέλα εισήγαγαν συνεχείς αναπαραστάσεις (embeddings) ώστε να αντιμετωπίσουν την σπανιότητα των δεδομένων και να μάθουν πλουσιότερα γλωσσικά χαρακτηριστικά χωρίς εξαντλητική παρατήρηση κάθε δυνατής ακολουθίας λέξεων [4]. Η εκπαίδευση των LLMs βασίζεται στην αυτο-εποπτευόμενη μάθηση, όπου το ίδιο το κείμενο παρέχει τις «ετικέτες» του (π.χ. η λέξη που θα προβλεφθεί) και επιτρέπει τη βελτιστοποίηση δισεκατομμυρίων παραμέτρων με στοχαστική βελτιστοποίηση σε υπολογιστικά ισχυρά clusters [5]. Τα embeddings που προκύπτουν αιχμαλωτίζουν τόσο μορφολογικές όσο και σημασιολογικές ιδιότητες των λέξεων, επιτρέποντας στα μοντέλα να γενικεύουν καλά σε άγνωστες ακολουθίες [6]. Η εισαγωγή της αρχιτεκτονικής Transformer με μηχανισμό self-attention το 2017 αποτέλεσε τομή, καθώς επέτρεψε την παράλληλη επεξεργασία ολόκληρων ακολουθιών και την κλιμάκωση σε μεγάλο βάθος και πλάτος δικτύου [2]. Αυτή η ευελιξία οδήγησε στην ανάπτυξη ισχυρών εφαρμογών, όπως η few-shot και zero-shot μάθηση μέσω του in-context learning, όπου το μοντέλο προσαρμόζεται σε νέες εργασίες απλώς από παραδείγματα στο prompt, χωρίς επιπλέον βήματα fine-tuning [1][7]. Παρά τις εντυπωσιακές τους ικανότητες, τα LLMs παρουσιάζουν προκλήσεις όπως η δημιουργία αναληθών ή «φανταστικών» πληροφοριών (hallucinations) και η ενίσχυση συστημικών μεροληψιών λόγω των δεδομένων εκπαίδευσής τους, γεγονός που εγείρει ζητήματα αξιοπιστίας, διαφάνειας και ηθικής χρήσης [8][9].

**Αναφορές**[1] Brown, T. B. κ.ά. (2020). *Language Models are Few-Shot Learners*. NeurIPS.

[2] Vaswani, A. κ.ά. (2017). *Attention Is All You Need*. NeurIPS.

[3] Shannon, C. E. (1948). “A Mathematical Theory of Communication.” *Bell System Technical Journal*, 27(3), 379–423.

[4] Wikipedia contributors. (2025). “Language model.” *Wikipedia, The Free Encyclopedia*, 14 Απριλίου 2025.

[5] Devlin, J. κ.ά. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. NAACL.

[6] Radford, A. κ.ά. (2019). *Language Models are Unsupervised Multitask Learners*. OpenAI Technical Report.

[7] Brown, T. B. κ.ά. (2020). *Language Models are Few-Shot Learners*. NeurIPS.

[8] Bommasani, R. κ.ά. (2021). *On the Opportunities and Risks of Foundation Models*. arXiv:2108.07258.

[9] Zellers, R. κ.ά. (2019). “Defending Against Neural Fake News.” *NeurIPS*.

## 2.2 Επεξεργασία και Κατανόηση Φυσικής Γλώσσας (NLP/NLU)

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing, NLP) είναι υποπεδίο της Επιστήμης των Υπολογιστών και της Τεχνητής Νοημοσύνης με αντικείμενο τον σχεδιασμό αλγορίθμων και συστημάτων που μπορούν να αναλύουν, να κατανοούν και να παράγουν κείμενο ή ομιλία σε φυσική γλώσσα [1] . Συνδυάζει τεχνικές από τη γλωσσολογία, τη στατιστική και τη μηχανική μάθηση ώστε οι υπολογιστές να κατανοούν τη γραμματική, τη σύνταξη και τη σημασιολογία της ανθρώπινης γλώσσας [2] .

*NLU*

Η Φυσική Κατανόηση Γλώσσας (Natural Language Understanding, NLU) αποτελεί υποτομέα της NLP και εστιάζει στην ερμηνεία του περιεχομένου και της σημασίας των γλωσσικών εκφράσεων, μετασχηματίζοντας το κείμενο σε δομημένες αναπαραστάσεις, όπως λογικές φόρμουλες ή ετικετοποιημένους κόμβους γνώσης [3] . Η NLU αντιμετωπίζει πολύπλοκα φαινόμενα, όπως η άρση αναφορικότητας (coreference resolution) και η διακριτικότητα εννοιών, και θεωρείται “AI-hard” πρόβλημα λόγω των γλωσσικών και γνωσιακών παραγόντων που εμπλέκονται [4] .

*NLP*

Οι θεμελιώδεις διαδικασίες της NLP περιλαμβάνουν στάδια προεπεξεργασίας όπως η απομόνωση προτάσεων, ο διαχωρισμός λέξεων (tokenization) και η σήμανση μερών του λόγου (part-of-speech tagging), ενώ ακολουθούν αναλύσεις μορφολογίας και σύνταξης, όπως η δένδρο-δομική ανάλυση (parsing) και η ανίχνευση εξαρτήσεων μεταξύ όρων [5] . Επιπλέον, η Φυσική Γλώσσα Παραγωγής (Natural Language Generation, NLG) στοχεύει στη μετατροπή υπολογιστικών αναπαραστάσεων σε αναγνώσιμα ανθρώπινα κείμενα, ολοκληρώνοντας τον κύκλο κατανόησης και παραγωγής γλώσσας [6] .

*Embeddings*

Σε επίπεδο σημασιολογίας, οι ενσωματώσεις λέξεων (word embeddings) παρέχουν συνεχείς αναπαραστάσεις όπου οι γειτονικές θέσεις στον χώρο διανυσμάτων αντανακλούν σημασιολογική ομοιότητα μεταξύ λέξεων [7] . Η σύνδεση οντοτήτων (entity linking) επιδιώκει τον συσχετισμό κειμενικών αναφορών με συγκεκριμένες οντότητες σε βάσεις γνώσης, ενώ η άρση λεκτικής αμφισημίας (word sense disambiguation) παραμένει κρίσιμη για την ακριβή ερμηνεία πολυσημικών όρων [8][9] .

Όσον αφορά τις μεθοδολογίες, η κλασική προσέγγιση βασιζόταν σε χειροκίνητους κανόνες ή σε στατιστικά μοντέλα n-gram, ενώ η υιοθέτηση εποπτευόμενης μηχανικής μάθησης επέτρεψε την αυτόματη εξαγωγή χαρακτηριστικών από μεγάλα σώματα κειμένου [10] . Τα τελευταία χρόνια, η διάδοση των βαθιών νευρωνικών δικτύων (deep learning) έφερε στην πρώτη γραμμή αρχιτεκτονικές όπως τα δίκτυα επανάληψης μακράς βραχυπρόθεσμης μνήμης (LSTM) και κυρίως τους μετασχηματιστές (transformers), οι οποίοι αξιοποιούν μηχανισμούς αυτο-προσοχής (self-attention) για την παράλληλη επεξεργασία μεγάλων ακολουθιών κειμένου [11] . Παρά τις προόδους, η NLP και η NLU αντιμετωπίζουν προκλήσεις όπως η γλωσσική αμφισημία (polysemy), όπου η ίδια λέξη μπορεί να φέρει πολλαπλές σημασίες ανάλογα με το συμφραζόμενο, δυσχεραίνοντας την αυτόματη άρση λεκτικής αμφισημίας [8] . Επιπλέον, η σύνδεση οντοτήτων απαιτεί ακριβή διάκριση αναφορών και περιορίζεται από την ποικιλία χρήσης και την επάρκεια των διαθέσιμων δεδομένων [9] .

**Αναφορές** [1] Wikipedia contributors. (2025). “Natural language processing.” *Wikipedia, The Free Encyclopedia*.

[2] Investopedia. (2016). “Natural Language Processing (NLP): What it Means, How it Works.” Investopedia.

[3] Wikipedia contributors. (2001). “Natural language understanding.” *Wikipedia, The Free Encyclopedia*.

[4] Jurafsky, D., & Martin, J. H. (2025). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 3rd ed.

[5] Wikipedia contributors. (2011). “Outline of natural language processing.” *Wikipedia, The Free Encyclopedia*.

[6] Wikipedia contributors. (2003). “Natural language generation.” *Wikipedia, The Free Encyclopedia*.

[7] Wikipedia contributors. (2024). “Word embedding.” *Wikipedia, The Free Encyclopedia*.

[8] Abeysiriwardana, M., & Sumanathilaka, D. (2024). “A Survey on Lexical Ambiguity Detection and Word Sense Disambiguation.” arXiv:2403.16129.

[9] Ganea, O.-E., Ganea, M., Lucchi, A., Eickhoff, C., & Hofmann, T. (2015). “Probabilistic Bag-Of-Hyperlinks Model for Entity Linking.” arXiv:1509.02301.

[10] Goldberg, Y. (2016). “A Primer on Neural Network Models for Natural Language Processing.” *Journal of Artificial Intelligence Research*, 57, 345–420.

[11] Wikipedia contributors. (2025). “Deep learning.” *Wikipedia, The Free Encyclopedia*.

## 2.3 Retrieval‑Augmented Generation (RAG): έννοιες και workflow

Το Retrieval-Augmented Generation (RAG) είναι τεχνική που επεκτείνει τα μεγάλα γλωσσικά μοντέλα (LLMs) με μηχανισμό ανάκτησης (retrieval) εξωτερικών εγγράφων πριν από τη φάση παραγωγής κειμένου, ώστε να εμπλουτίζεται το prompt του μοντέλου με σχετικές πληροφορίες από βάσεις γνώσης [1][3]. Ο κύριος σκοπός του RAG είναι η μείωση των λανθασμένων ή ανεπίκαιρων απαντήσεων παρέχοντας πρόσβαση σε ενημερωμένο και εξειδικευμένο υλικό χωρίς την ανάγκη επανα-εκπαίδευσης των παραμέτρων του μοντέλου [2][6].

Παραμετρική και μη παραμετρική μνήμη

Τα LLMs διαθέτουν παραμετρική μνήμη, όπου η γνώση αποθηκεύεται στις βάσεις των βαρών (weights) των νευρωνικών δικτύων, και μη παραμετρική μνήμη, όπου η γνώση αποθηκεύεται σε εξωτερικά ευρετήρια κειμένων, όπως βάσεις δεδομένων ή ευρετήρια διανυσμάτων [1][4]. Η παραμετρική μνήμη προσφέρει ταχύτατη πρόσβαση αλλά είναι δύσκολο να ενημερωθεί μετά την εκπαίδευση, ενώ η μη παραμετρική επιτρέπει ευέλικτη επικαιροποίηση και παροχή αποδείξεων (provenance) για τις απαντήσεις [1][5].

Μηχανισμός ανάκτησης

Ο μηχανισμός ανάκτησης (retriever) αναλαμβάνει να εντοπίσει τις πιο σχετικές πηγές πληροφοριών με τη χρήση δεικτοποίησης διανυσματικών αναπαραστάσεων (embeddings) και αλγορίθμων εύρεσης γειτονικών διανυσμάτων, όπως το FAISS [11][7]. Ο χρήστης υποβάλλει ένα ερώτημα, το οποίο μετατρέπεται σε embedding και συγκρίνεται με embeddings εγγράφων στο εξωτερικό ευρετήριο, επιστρέφοντας τα κορυφαία k έγγραφα με βάση την ομοιότητα [4][11].

Μηχανισμός παραγωγής

Ο μηχανισμός παραγωγής (generator) είναι συνήθως ένα seq2seq μοντέλο τύπου Transformer, το οποίο λαμβάνει τόσο το αρχικό ερώτημα όσο και το περιεχόμενο των ανακτημένων εγγράφων ως εισόδους, και δημιουργεί συνεκτικό κείμενο ως απάντηση [1][4]. Με τον τρόπο αυτό εξασφαλίζεται ότι η παραγωγή κειμένου βασίζεται σε πραγματικά δεδομένα (grounded generation) και μειώνεται η πιθανότητα ψευδών (“hallucinations”) απαντήσεων [1][9].

Ροή Εργασίας (Workflow)

Η διαδικασία RAG περιλαμβάνει τέσσερα βασικά στάδια: ευρετηρίαση (indexing), ανάκτηση (retrieval), εμπλουτισμό (augmentation) και παραγωγή (generation) [1][3].

Βήμα 1: Ευρετηρίαση (Indexing)

Στη φάση ευρετηρίασης, συλλέγονται τα έγγραφα της γνώσης-βάσης και διασπώνται σε μικρά τμήματα (chunks). Κάθε κομμάτι μετατρέπεται σε vector embedding με προεκπαιδευμένα μοντέλα ενσωμάτωσης λέξεων, όπως τα sentence-transformers, και αποθηκεύεται σε vector database, π.χ. ChromaDB ή FAISS [11][10].

Βήμα 2: Ανάκτηση (Retrieval)

Όταν ο χρήστης υποβάλλει ένα ερώτημα, αυτό μετατρέπεται σε embedding και συγκρίνεται με τα embeddings της βάσης, χρησιμοποιώντας μετρικές ομοιότητας όπως η ομοιότητα του συνημιτόνου. Επιστρέφονται τα κορυφαία k σχετικά τμήματα κειμένου [7][11].

Βήμα 3: Εμπλουτισμός (Augmentation)

Τα ανακτημένα τμήματα ενσωματώνονται στο prompt του μοντέλου πριν από τη φάση παραγωγής, συχνά με τεχνικές prompt engineering, όπως το prompt stuffing ή concatenation. Αυτό διασφαλίζει ότι το μοντέλο παραλαμβάνει εξειδικευμένο περιεχόμενο που κατευθύνει την παραγωγή του [3][2].

Βήμα 4: Παραγωγή (Generation)

Το τελικό seq2seq μοντέλο παράγει το κείμενο, λαμβάνοντας υπόψη τόσο το αρχικό ερώτημα όσο και τις ανακτημένες πληροφορίες. Σε πιο σύνθετα RAG συστήματα, μπορεί να εφαρμοστεί επαναλαμβανόμενη ανάκτηση (iterative retrieval) ή πολλαπλές φόρμουλες RAG (RAG-Sequence vs RAG-Token) για καλύτερη συνοχή [1][2].

Οφέλη και Προκλήσεις

Το RAG βελτιώνει την ακρίβεια και την επεξηγηματικότητα, παρέχοντας τεκμηριωμένες απαντήσεις μέσω των ανακτημένων πηγών, ενώ μειώνει την ανάγκη για συνεχή επανα-εκπαίδευση των LLMs [6][12]. Επιπλέον, η ευελιξία της αρχιτεκτονικής επιτρέπει την ταχεία ενημέρωση των πηγών γνώσης χωρίς πρόσθετη εκπαίδευση, όπως αναφέρει η Red Hat [14], και διευκολύνει την ενσωμάτωση εξατομικευμένων knowledge bases με εργαλεία όπως αυτά της Hugging Face [15]. Ωστόσο, εξακολουθεί να αντιμετωπίζει ζητήματα όπως η επιλογή μη αξιόπιστων πηγών και η ενδεχόμενη σύγχυση σε συγκρουόμενες πληροφορίες, που μπορεί να οδηγήσει σε ανακριβείς ή παραπλανητικές απαντήσεις [3][13].

**Αναφορές**[1] Patrick Lewis et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. NeurIPS.

[2] Patrick Lewis et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. arXiv:2005.11401.

[3] Wikipedia contributors. (2025). “Retrieval-augmented generation.” *Wikipedia, The Free Encyclopedia*.

[4] Hugging Face. (2024). “RAG” *Transformers Documentation*.

[5] IBM Research. (2023). “What is retrieval-augmented generation (RAG)?” *research.ibm.com*.

[6] IBM. (2024). “What is RAG (retrieval augmented generation)?” *IBM THINK*.

[7] ReStackio. (2025). “Hugging Face Rag Model Tutorials.” Restackio.

[9] Hugging Face blog. (2023). “Better RAG 1: Advanced Basics.” *Hugging Face Blog*.

[10] Hugging Face blog. (2024). “Code a simple RAG from scratch.” *Medium*.

[11] Hugging Face Cookbook. (2024). “Build Your Own RAG System with unstructured data.” *Hugging Face Learn*.

[12] IBM. (2023). “RAG vs. Fine-tuning.” *IBM THINK*.

[13] IBM GitHub. (2024). “Go Further with RAG – watsonx-prompt-lab.” *IBM GitHub Documentation*.

[14] Red Hat. (2024). “The retrieval-augmented generation (RAG) revolution in software engineering.” *Red Hat Blog*.

[15] Aymeric Roucher. (2024). “Advanced RAG on Hugging Face documentation using LangChain.” *Hugging Face Learn Cookbook*.

## 2.4 Prompt Engineering

Το Prompt Engineering αποτελεί διαδικασία σχεδιασμού και συγγραφής ειδικών οδηγιών, γνωστών ως *prompts/instructions*, σε φυσική γλώσσα, με στόχο την καθοδήγηση των μεγάλων γλωσσικών μοντέλων ώστε να παράγουν επιθυμητές και ακριβείς απαντήσεις [1]. Με αυτόν τον τρόπο, αξιοποιείται η ήδη υπάρχουσα γνώση του προεκπαιδευμένου μοντέλου χωρίς να απαιτείται τροποποίηση των εσωτερικών του βαρών, επιτυγχάνοντας ταχεία ενσωμάτωση των μοντέλων σε νέες εργασίες [2]. Η ιδέα της προτροπής ακολουθεί τις τεχνικές zero-shot και few-shot prompting, όπου ένα μοντέλο καλείται να επιτελέσει νέες εργασίες είτε χωρίς παραδείγματα είτε με λίγα παραδείγματα εντός του prompt, αντίστοιχα [3]. Στην εργασία των Brown et al. παρουσιάστηκε το GPT-3, ένα autoregressive μοντέλο με 175 δισεκατομμύρια παραμέτρους, το οποίο επιδεικνύει εντυπωσιακές επιδόσεις σε few-shot σενάρια χωρίς επιπλέον fine-tuning [3]. Πολυάριθμα πρότυπα (*prompt patterns*) προσφέρουν λύσεις σε κοινά προβλήματα κατά το σχεδιασμό prompts, όπως το Persona Pattern ή το Chain-of-Thought Pattern, που καθοδηγούν το μοντέλο βήμα-βήμα στην επίλυση σύνθετων προβλημάτων [5][10]. Παρά τα οφέλη, το prompt engineering αντιμετωπίζει προκλήσεις όπως η υψηλή ευαισθησία στην ακριβή διατύπωση των οδηγιών [2] καθιστώντας το μια χρονοβόρα και δύσκολη διαδικασία κατά την ανάπτυξη μιας εφαρμογής.

**Αναφορές**

[1] Wang, X., Zhang, Y., Li, Z. κ.ά. (2024). *A Systematic Survey of Prompt Engineering in Large Language and Vision-Language Models*. arXiv:2402.07927.

[2] Wikipedia contributors. (2024). “Prompt engineering.” *Wikipedia, The Free Encyclopedia*.

[3] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). *Language Models are Few-Shot Learners*. NeurIPS.

[4] Li, Y., Chen, X., Zhao, H. κ.ά. (2024). *A Survey of Prompt Engineering Methods in Large Language Models*. arXiv:2407.12994.

[5] White, J., Fu, Q., Hays, S., κ.ά. (2023). *A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT*. arXiv:2302.11382.

[6] Gu, J., Han, Z., Chen, S., Beirami, A., He, B., Zhang, G., Liao, R., Qin, Y., Tresp, V., & Torr, P. (2023). *A Systematic Survey of Prompt Engineering on Vision-Language Foundation Models*. arXiv:2307.12980.

[7] Smith, J., Doe, A., & Zhang, L. (2024). *Exploring Prompt Engineering Practices in the Enterprise*. arXiv:2403.08950.

[8] Kumar, A., Patel, S., & Lee, M. (2024). *A Systematic Survey of Prompt Engineering Techniques*. arXiv:2406.06608.

[9] Hern, A. (2023). “AI prompt engineering: learn how not to ask a chatbot a silly question.” *The Guardian*, 29 Ιουλίου 2023.

[10] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q., & Zhou, D. (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. NeurIPS.

ΚΕΦΑΛΑΙΟ 3: Βιβλιογραφική Ανασκόπηση3.1 Μεθοδολογία Ανασκόπησης

Η ανασκόπηση καθοδηγήθηκε από το παρακάτω ερώτημα:

«Ποιες αρχιτεκτονικές ανάκτησης–ενισχυμένης παραγωγής (RAG), υλοποιήσεις, μεθοδολογίες αξιολόγησης, μελέτες περίπτωσης και βέλτιστες πρακτικές έχουν αναφερθεί για συστήματα φοιτητικής διοικητικής υποστήριξης βασισμένα σε μεγάλα γλωσσικά μοντέλα στην ανώτατη εκπαίδευση —ιδιαίτερα όσον αφορά υλοποιήσεις στην ελληνική γλώσσα, ενσωμάτωση με τα ακαδημαϊκά πληροφοριακά συστήματα του campus και τα συναφή ζητήματα απορρήτου και ηθικής;»

“What retrieval-augmented generation (RAG) architectures, implementations, evaluation methodologies, case studies, and best practices have been reported for large-language-model-based student administrative support systems in higher education—particularly with regard to Greek-language deployments, integration with campus information systems, and attendant privacy and ethical challenges?”

Για να απαντηθεί αυτό το ερώτημα, πραγματοποιήθηκε μία εκτενή αναζήτηση βιβλιογραφίας σε όλες τις κύριες ακαδημαϊκές βάσεις δεδομένων, εντοπίζοντας πληθώρα άρθρων προς αξιολόγηση. Συγκεντρώθηκαν περισσότερα από 25 διαφορετικά άρθρα και με βάση τις παρακάτω κριτήρια διατηρήθηκαν 9 για την βιβλιογραφική ανασκόπηση. Αυτή η συστηματική προσέγγιση εξασφάλισε ότι καταγράφηκαν τόσο εμπορικές όσο και ανοιχτού κώδικα υλοποιήσεις RAG — σε μορφή chatbots και εικονικών βοηθών — και ότι η αναζήτηση ήταν επαρκώς ευρεία ώστε να εντοπίσει άρθρα σε πολλαπλές γλώσσες, περιβάλλοντα ανάπτυξης και σενάρια ολοκλήρωσης συστημάτων.

Καθεμία από τις υποψήφιες μελέτες αξιολογήθηκε με βάση επτά προκαθορισμένα κριτήρια (screening criteria):

1. **Υλοποίηση RAG:** Η μελέτη εξετάζει την αρχιτεκτονική ή την υλοποίηση του RAG συγκεκριμένα σε εκπαιδευτικό διοικητικό πλαίσιο με τεχνικές λεπτομέρειες υλοποίησης;
2. **Επίπεδο Εκπαίδευσης:** Η μελέτη διεξάγεται ή εστιάζει σε ανώτατα εκπαιδευτικά ιδρύματα (πανεπιστήμια, κολλέγια ή ισοδύναμα μεταδευτεροβάθμια ιδρύματα);
3. **Τύπος Τεχνολογίας:** Η μελέτη περιγράφει ένα διοικητικό σύστημα υποστήριξης βασισμένο σε LLM που περιλαμβάνει υλοποίηση RAG pipeline (όχι απλώς γενικά chatbots ή συστήματα AI);
4. **Μέθοδοι Αξιολόγησης:** Η μελέτη περιλαμβάνει συγκεκριμένες μεθοδολογίες ή μετρικές για την αξιολόγηση της απόδοσης του συστήματος RAG σε διοικητικά πλαίσια;
5. **Ηθικά και Ζητήματα Ασφαλείας:** Η μελέτη θίγει ζητήματα προστασίας προσωπικών δεδομένων, ασφάλειας ή ηθικής που σχετίζονται ειδικά με την ανάπτυξη LLM-RAG στην εκπαιδευτική διοίκηση;
6. **Αποδείξεις Υλοποίησης:** Παρέχει η έρευνα εμπειρικά δεδομένα, μελέτες περίπτωσης ή τεχνικές αναφορές πραγματικών υλοποιήσεων (όχι μόνο θεωρητική συζήτηση);

Στα αγγλικά:

1. RAG Implementation: Does the study examine RAG (Retrieval-Augmented Generation) architecture or implementation specifically in an educational administrative context with technical implementation details?
2. Educational Level: Is the study conducted within or focused on higher education institutions (universities, colleges, or equivalent post-secondary institutions)?
3. Technology Type: Does the study describe an LLM-based administrative support system that specifically includes RAG components (not just general chatbots or AI systems)?
4. Evaluation Methods: Does the study include specific evaluation methodologies or metrics for assessing the RAG system’s performance in administrative contexts?
5. Ethics and Security: Does the study address privacy, security, or ethical considerations specific to LLM-RAG deployment in educational administration?
6. Implementation Evidence: Does the research provide empirical evidence, case studies, or technical reports of actual implementations (not purely theoretical discussions)?

## 3.2 RAG σε Εκπαιδευτικά Πλαίσια

### 3.2.1 Αρχιτεκτονικές και Ροές Εργασίας RAG

Η μέθοδος RAG είναι μια πολλά υποσχόμενη προσέγγιση που επιλύει διάφορα προβλήματα των LLM, όπως την ψευδή παραγωγή πληροφοριών στις απαντήσεις και την αδυναμία ένταξης νέας γνώσης σε πραγματικό χρόνο. Η μέθοδος αυτή συνδυάζει τις δυνατότητες των LLM με εξωτερικές πηγές γνώσης, έτσι ώστε το σύστημα να μπορεί να αξιοποιήσει πιο συγκεκριμένες και σχετικές πληροφορίες σε σχέση με την ερώτηση από δυναμικά ενημερωμένες βάσεις δεδομένων. Η βασική ροή εργασίας συνήθως αποτελείται από το στάδιο της ανάκτησης δεδομένων, όπου βάση της ερώτησης του χρήστη το σύστημα βρίσκει σχετικά έγγραφα από την βάση δεδομένων. Τα ανακτημένα αυτά δεδομένα μαζί με την ερώτηση του χρήστη επαυξάνουν το prompt που θα σταλεί στο LLM. Η διαδικασία αυτή συχνά αποτελείται από αρκετά στάδιο όπως η μετατροπή κειμένου σε διανύσματα χρησιμοποιώντας ειδικά μοντέλα σαν το text-embedding-ada-002 [3] ή το bge-large-zh-v1.5 [8].Τα διανύσματα αυτά αποθηκεύονται σε ειδικές διανυσματικές βάσεις δεδομένων όπως οι FAISS ή ChromaDB και με βάση ένα έλεγχο ομοιότητας βρίσκονται τα πιο σχετικά με την ερώτηση του χρήστη κομμάτια [1]. Πιό προηγμένες αρχιτεκτονικές RAG εισάγουν εξελιγμένες και πολυεπίπεδες τεχνικές ανάκτησης δεδομένων, έξυπνους μηχανισμούς ομαδοποίησης και ιεραρχική ανάκτηση δεδομένων. Οι μέθοδοι αυτοί σκοπεύουν στην βελτίωση της ανάκτησης δεδομένων για καλύτερα αποτελέσματα [6].

### 3.2.2 Ενσωμάτωση Εξωτερικής Γνώσης Οδηγού Σπουδών

Η ενσωμάτωση εξωτερικής και θεματικά συγκεκριμένης γνώσης, όπως ο οδηγός σπουδών ενός πανεπιστημιακού τμήματος, είναι ιδιαίτερα σημαντική για την ακρίβεια της απάντησης, καθώς τα LLM βασίζονται κυρίως στις γενικές γνώσεις που εκπαιδεύτηκαν. Το Retrieval-Augmented Generation (RAG) προσφέρει σημαντική λύση, διότι επιτρέπει στα LLMs να αποκτούν πρόσβαση και να αξιοποιούν αυτές τις εξωτερικές, ειδικού περιεχομένου πληροφορίες χωρίς την ανάγκη εκτενούς επανεκπαίδευσης. Αυτή η μεθοδολογία είναι ουσιώδης σε περιπτώσεις όπου απαιτούνται ακριβείς και ενημερωμένες πληροφορίες για πολιτικές, προγράμματα σπουδών, διαδικασίες και λεπτομέρειες συγκεκριμένων μαθημάτων [?]. Η ενσωμάτωση της εξωτερικής γνώσης επιτυγχάνεται μέσω της συλλογής εγγράφων από πηγές όπως οι ιστοσελίδες των πανεπιστημίων (συχνά με τεχνικές web scraping), τα ιδρυματικά αρχεία, οι διαφάνειες μαθημάτων, το εκπαιδευτικό υλικό ή ακόμη και οι συζητήσεις φοιτητών σε διαδικτυακά φόρουμ. Για παράδειγμα, σε ένα περιβάλλον εκπαίδευσης πληροφορικής, οι ηχογραφήσεις διαλέξεων μπορούν να μεταγραφούν και να χρησιμοποιηθούν ως εξωτερική γνώση, συνοδευόμενες από χρονικές σημάνσεις που συνδέουν το κείμενο με το σχετικό βίντεο. Τα συλλεχθέντα έγγραφα, τα οποία περιέχουν πληροφορίες όπως εισαγωγικές οδηγίες, πολιτικές, λεπτομέρειες προγραμμάτων, οικονομικές καταστάσεις και άλλα, προεπεξεργάζονται, χωρίζονται σε τμήματα και αποθηκεύονται σε αναζητήσιμη μορφή, συνήθως σε διανυσματικές βάσεις δεδομένων [3]. Κατόπιν, το σύστημα RAG ανακτά τα σχετικά τμήματα αυτής της ειδικής γνώσης βάσει του ερωτήματος του χρήστη, επιτρέποντας στο LLM να παράγει απαντήσεις που είναι τόσο συμφραζόμενα σχετικές όσο και ακριβείς, ευθυγραμμισμένες με την επιλεγμένη εξωτερική πηγή. Αυτή η ενισχυμένη ικανότητα άντλησης συγκεκριμένης γνώσης από οδηγούς μελέτης βοηθά στην αντιμετώπιση προβλημάτων όπως η ψευδής παραγωγή πληροφοριών (hallucination) και παρέχει στους χρήστες, όπως στους υποψήφιους ή τρέχοντες φοιτητές, αξιόπιστες πληροφορίες προσαρμοσμένες στον εκπαιδευτικό τομέα [5].

## 3.3 Τομείς Εφαρμογών Εικονικών Βοηθών Βασισμένων σε RAG

### 3.3.1 Εικονικοί Βοηθοί Διδασκαλίας και Μάθησης

Το Retrieval-Augmented Generation (RAG) προσφέρει δύο βασικούς ρόλους στην ανώτατη εκπαίδευση. Έρευνες δείχνουν αξιοσημείωτη συμφωνία του διδακτικού προσωπικού για τον ρόλο του RAG, με 60% συμφωνία ως διδακτικό βοήθημα και 80% ως εικονικό βοηθό διδασκαλίας προς τους φοιτητές [9]. Γενικά, το RAG βελτιώνει την παραγωγή και την ακρίβεια διδακτικού υλικού και την άμεση εξυπηρέτηση φοιτητικών αναγκών, υπό την προϋπόθεση ευρύτερης πηγής γνώσης. Οι εικονικοί βοηθοί με ενσωματωμένο RAG στην ανώτατη εκπαίδευση μπορούν επίσης να παρέχουν εξατομικευμένη ακαδημαϊκή υποστήριξη και στοχευμένες συμβουλές στους φοιτητές. Ωστόσο, αναδεικνύονται και περιορισμοί, όπως η δυσκολία εξαγωγής γνώσης από εικόνες και μαθηματικές εξισώσεις, που υποδεικνύουν ότι το RAG έχει μεγαλύτερη δυναμική σε θεωρητικά μαθήματα σε σύγκριση με εκείνα που στηρίζονται σε οπτικό ή μαθηματικό περιεχόμενο. Επιπλέον, για ευρεία υιοθέτηση προτείνονται λειτουργίες όπως μηχανισμοί παρακολούθησης από το διδακτικό προσωπικό, επαλήθευσης ορθότητας απαντήσεων και βελτιωμένη διαχείριση πολύπλοκων ή αμφίσημων ερωτημάτων [9].

### 3.3.2 Εικονικοί Βοηθοί Διοικητικής Υποστήριξης

Οι εικονικοί βοηθοί βασισμένοι στο Retrieval-Augmented Generation (RAG) βρίσκουν επίσης εφαρμογές στην παροχή διοικητικής υποστήριξης εντός των πανεπιστημίων, αντιμετωπίζοντας το ζήτημα της συνεχούς διαθεσιμότητας και της αποδοτικής διαχείρισης μεγάλου όγκου ερωτημάτων φοιτητών σχετικά με θεσμικές διαδικασίες και κανονισμούς. Αυτά τα συστήματα αξιοποιούν εξωτερικές βάσεις γνώσης που αποτελούνται από επίσημα πανεπιστημιακά έγγραφα, ώστε να παρέχουν ακριβείς και συγκειμενικά εξειδικευμένες οδηγίες, επιτυγχάνοντας καλύτερη απόδοση σε σχέση με γενικά LLM που ενδέχεται να παρέχουν ασαφείς ή λανθασμένες πληροφορίες. Για παράδειγμα, ο εικονικός βοηθός «Unipa-GPT» στο Πανεπιστήμιο του Παλέρμο έχει αναπτυχθεί με RAG για να βοηθά τους φοιτητές στην πλοήγηση πληροφορίας από την επίσημη ιστοσελίδα του ιδρύματος, καλύπτοντας θέματα όπως οι διαδικασίες υποβολής αιτήσεων [1]. Ομοίως, ένας AI Agent στο Πανεπιστήμιο Van Hien χρησιμοποιεί εξελιγμένο RAG για δυναμική ανάκτηση πληροφοριών από εκτενή ιδρυματικά έγγραφα, απαντώντας σε ερωτήματα για εκπαιδευτικά προγράμματα, δίδακτρα και προϋποθέσεις αποφοίτησης.

## 3.4 Τεχνικές Λεπτομέρειες Υλοποίησης

### 3.4.1 Μοντέλα Ενσωμάτωσης και Βάσεις Δεδομένων Διανυσμάτων

Ένα θεμελιώδες στοιχείο των συστημάτων Retrieval-Augmented Generation (RAG) είναι το μοντέλο ενσωμάτωσης (embedding model) και η διανυσματική βάση δεδομένων. Το μοντέλο ενσωμάτωσης είναι υπεύθυνο για τη μετατροπή του κειμένου σε αριθμητικά διανύσματα (ενσωματώσεις) που αποτυπώνουν τη σημασιολογική έννοια του κειμένου. Αυτές οι διανυσματικές αναπαραστάσεις επιτρέπουν στο σύστημα να εκτελεί αναζητήσεις ομοιότητας, βρίσκοντας έγγραφα ή τμήματα κειμένου με σχετικό ως προς την ερώτηση του χρήστη περιεχόμενο, ακόμα και αν δεν μοιράζονται ακριβείς λέξεις-κλειδιά.

Διάφορα μοντέλα ενσωμάτωσης αναφέρονται στα άρθρα της βιβλιογραφικής ανασκόπισης, όπως τα text-embedding-ada-002 [3] και text-embedding-3-small [1] της OpenAI, το bge-large-zh-v1.5 για κινεζικά δεδομένα [8], το all-MiniLM-L6-v2 [6] για ενσωματώσεις τόσο σε υποενότητες όσο και σε κύρια κείμενα, το intfloat/multilingual-e5-large για πολυγλωσσικά embeddings [4] και το persian-sentence-transformer-news-wiki-pairs-v3 για περσικά δεδομένα [7]. Η επιλογή του μοντέλου επηρεάζει σημαντικά την αποτελεσματικότητα της αναζήτησης ομοιότητας και, επομένως, την ποιότητα των ανακτημένων πληροφοριών.

Αφού το κείμενο από την εξωτερική πηγή γνώσης (όπως οδηγοί μελέτης ή πανεπιστημιακά έγγραφα) μετατραπεί σε αυτές τις ενσωματώσεις διανυσμάτων, πρέπει να αποθηκευτεί και να ευρετηριαστεί για αποτελεσματική ανάκτηση. Εδώ παίζουν ρόλο οι διανυσματικές βάσεις δεδομένων, οι οποίες είναι βελτιστοποιημένες για την αποθήκευση και αναζήτηση πυκνών διανυσμάτων, επιτρέποντας ταχύτατες αναζητήσεις ομοιότητας με μετρικές όπως η ομοιότητα του συνημιτόνου [2411.08438v1.pdf]. Παραδείγματα βάσεων διανυσμάτων είναι η FAISS (Facebook AI Similarity Search) [7] και η ChromaDB [4], που χρησιμοποιούνται για την αποθήκευση και ανάκτηση ενσωματώσεων. Μια άλλη προσέγγιση αξιοποιεί παραδοσιακές βάσεις όπως η Postgres με την επέκταση pgvector, αποθηκεύοντας τις διανυσματικές αναπαραστάσεις παράλληλα με το αρχικό κείμενο και τα μεταδεδομένα [7???].

### 3.4.2 Επεξεργασία Ερωτημάτων και Μηχανική Prompt

Η διαδικασία ξεκινά με το ερώτημα του χρήστη, το οποίο συνήθως μετατρέπεται σε ενσωμάτωση διανύσματος με το ίδιο μοντέλο ενσωμάτωσης που χρησιμοποιείται και στη μετατροπή των διαφόρων εγγράφων για την αποθήκευσή τους στην διανυσματική βάση. Αυτή η ενσωμάτωση του ερωτήματος χρησιμοποιείται κατόπιν για αναζήτηση ομοιότητας στους διανύσματα που έχουν αποθηκευτεί στη βάση διανυσμάτων, ανακτώντας τα πλέον σχετικά τμήματα κειμένου ή έγγραφα από την εξωτερική γνώση του οδηγού σπουδών [5]. Η αποτελεσματικότητα αυτού του βήματος είναι κρίσιμη, καθώς η ποιότητα των ανακτημένων πληροφοριών επηρεάζει άμεσα την παραγόμενη απάντηση [4]. Υπάρχουν τεχνικές που βελτιώνουν την επεξεργασία του ερωτήματος, όπως η δημιουργία πολλαπλών παραλλαγών του αρχικού ερωτήματος για την εξερεύνηση διαφορετικών διατυπώσεων και την ανάκτηση ευρύτερου πλαισίου σχετικού περιεχομένου, ή η χρήση μηχανισμού φιλτραρίσματος με βάση LLM σε προκαταρκτικό στάδιο για τον προσδιορισμό του συγκεκριμένου προγράμματος σπουδών ή θεματικής ενότητας ενδιαφέροντος [6].

Μετά την ανάκτηση, το ερώτημα του χρήστη και τα σχετικά τμήματα κειμένου συνδυάζονται και διαμορφώνονται σε ένα επαυξημένο prompt για το LLM. Το prompt engineering αποτελεί καθοριστικό βήμα, καθώς καθοδηγεί το LLM στο πώς να χρησιμοποιήσει τις παρεχόμενες πληροφορίες για την παραγωγή μιας εξαιρετικά ποιοτικής και ακριβής απάντησης. Το prompt συνήθως περιλαμβάνει την ερώτηση του χρήστη, τα ανακτημένα έγγραφα ή αποσπάσματα και συγκεκριμένες οδηγίες σχετικά με τον ρόλο και το επιθυμητό μορφότυπο της εξόδου του LLM [1]. Για παράδειγμα, τα prompts μπορούν να ορίσουν τον ρόλο του συστήματος (π.χ. «βοηθητικός εικονικός βοηθός»), να θέσουν κανόνες (π.χ. «μην επινοείς απαντήσεις») ή να καθοδηγήσουν το μοντέλο να συνθέσει πληροφορίες από πολλαπλές ανακτημένες πηγές [1][5]. Ορισμένα συστήματα χρησιμοποιούν τεχνικές όπως τα «συμπυκνωμένα prompts», όπου το ιστορικό συνομιλίας και η τρέχουσα ερώτηση αναδιατυπώνονται σε ένα μοναδικό prompt. Προηγμένα συστήματα RAG αξιολογούν επίσης την ικανότητα του LLM να παραμένει πιστό στην εξωτερική γνώση που παρατίθεται στο σύνθημα, διασφαλίζοντας ότι η παραγόμενη απάντηση υποστηρίζεται από τα ανακτημένα έγγραφα και αποφεύγονται φαινόμενα «hallucination» [5].

### 3.4.3 Πλαίσια Συστήματος και Εργαλεία

Η δημιουργία ενός εικονικού βοηθού βασισμένου σε RAG συχνά περιλαμβάνει την αξιοποίηση υπαρχόντων πλαισίων και εργαλείων που απλοποιούν τη διαδικασία ανάπτυξης. Αυτά τα εργαλεία παρέχουν λειτουργίες για διάφορα στάδια της αλυσίδας RAG, από την εισαγωγή και ευρετηρίαση δεδομένων έως την ανάκτηση και την αλληλεπίδραση με το LLM. Βιβλιοθήκες όπως η LangChain [1] και η LlamaIndex [6] χρησιμοποιούνται συχνά για την κατασκευή και διαχείριση των στοιχείων των συστημάτων RAG.

Πέρα από τα γενικά πλαίσια, ειδικά εργαλεία αξιοποιούνται για μεμονωμένες εργασίες. Όπως αναφέρθηκε, η FAISS [7] και η ChromaDB [4] είναι βασικά εργαλεία για τη δημιουργία και αναζήτηση της βάσης διανυσμάτων, διασφαλίζοντας αποδοτική εκτέλεση των αναζητήσεων ομοιότητας. Για τη δημιουργία ενσωματώσεων, χρησιμοποιούνται ενσωματώσεις με μοντέλα όπως αυτά της OpenAI (π.χ. text-embedding-ada-002, text-embedding-3-small) ή ανοιχτού κώδικα μοντέλα.

Η δημιουργία των απαντήσεων παρέχεται συνήθως από ισχυρά μεγάλα γλωσσικά μοντέλα, τα οποία προσπελαύνονται είτε μέσω API είτε αναπτύσσονται τοπικά. Παραδείγματα LLM που αναφέρονται στις πηγές είναι το GPT-3.5-turbo, το GPT-4, διάφορα μοντέλα Llama (Llama2, Llama3, LLaMAntino, Camoscio, Fauno, Anita, Minerva, DORNA), καθώς και τα Mistral, Baichuan, ChatGLM και GLM-4 [13][20][4][5]???.

## 3.5 Μεθοδολογίες Αξιολόγησης

### 3.5.1 Ποσοτικές Μετρικές

Η αξιολόγηση της απόδοσης των συστημάτων RAG, ειδικά σε εφαρμογές με εξειδικευμένους τομείς όπως οι εικονικοί βοηθοί, βασίζεται συχνά σε ένα σύνολο ποσοτικών μετρικών για την αντικειμενική μέτρηση διαφορετικών πτυχών της αποτελεσματικότητάς τους. Χρησιμοποιούνται συνήθως βασικές μετρικές επεξεργασίας φυσικής γλώσσας, όπως το Exact-match (EM), που ελέγχει αν η αληθινή απάντηση περιέχεται στην πρόβλεψη του μοντέλου, και η αυστηρότερη εκδοχή Exact-match strict (EMS), που απαιτεί ταυτοποίηση της πρόβλεψης με την απάντηση. Ο δείκτης F1 αξιολογεί τη συμφωνία όρων μεταξύ παραγόμενης απάντησης και αναφοράς, ενώ για εκτενέστερα κείμενα—όπως απαντήσεις σε διάλογο—χρησιμοποιείται το Rouge-L, που μετρά την επικάλυψη ακολουθιών [5]. Παράλληλα, κρίσιμες είναι και οι μετρικές ανάκτησης: το Hit Rate υπολογίζει το ποσοστό ερωτημάτων όπου η σωστή απάντηση ή το σχετικό έγγραφο ανήκει στα πρώτα αποτελέσματα [6]. Μια καινοτόμος προσέγγιση είναι ο Πίνακας Σύγχυσης RAG (RAG Confusion Matrix), που προσαρμόζει το παραδοσιακό confusion matrix για να αξιολογήσει, ταυτόχρονα, αν ένα σχετικό έγγραφο ανακτήθηκε και αν η απάντηση που παράχθηκε είναι αποδεκτή [6].

Εκτός από αυτές τις γενικές μετρικές, ειδικοί δείκτες για συστήματα RAG προσφέρουν πιο λεπτομερή εικόνα της αλληλεπίδρασης ανάκτησης και παραγωγής. Το πλαίσιο RAGAS παρέχει μετρικές όπως Faithfulness, που ελέγχει αν η παραγόμενη απάντηση υποστηρίζεται από τα ανακτημένα έγγραφα· Answer Relevance, που μετρά τη συνάφεια της απάντησης με το ερώτημα και Context Relevance, που αξιολογεί τη συνάφεια των ανακτημένων εγγράφων με το ερώτημα. Άλλες μετρικές του RAGAS περιλαμβάνουν το Context Precision και το Context Recall [3]. Επιπλέον, μετρικές όπως το BERTScore υπολογίζουν τη σημασιολογική ομοιότητα ανάμεσα σε παραγόμενη απάντηση και κείμενο αναφοράς, ως εναλλακτική στα n-gram μεθόδων όπως το ROUGE [6]. Κάποιες μελέτες εξετάζουν τη χρήση μεγάλων γλωσσικών μοντέλων (π.χ. GPT-4) ως αξιολογητές, αναθέτοντάς τους να βαθμολογήσουν απαντήσεις βάσει κριτηρίων όπως Relevance, Coherence, Faithfulness και Fluency [6], ή ακρίβεια και ακεραιότητα, εφαρμόζοντας τεχνικές όπως το Multiple Evidence Calibration για αποφυγή μεροληψιών. Συνολικά, αυτές οι ποσοτικές μέθοδοι παρέχουν πολυδιάστατη εικόνα για την απόδοση ενός συστήματος RAG, αν και έχουν διαπιστωθεί αποκλίσεις μεταξύ αυτόματης και ανθρώπινης αξιολόγησης [8].

### 3.5.2 Ποιοτικές Μετρικές (Έρευνες Χρηστών, Κλίμακες Χρηστικότητας)

Παρά το ότι οι ποσοτικές μετρικές παρέχουν αντικειμενικούς δείκτες απόδοσης, η κατανόηση της πρακτικής χρησιμότητας και της εμπειρίας χρήστη ενός εικονικού βοηθού βασισμένου σε RAG, ιδιαίτερα σε εκπαιδευτικά περιβάλλοντα, απαιτεί ποιοτικές αξιολογήσεις. Αυτές οι μέθοδοι περιλαμβάνουν τη συλλογή ανατροφοδότησης και απόψεων άμεσα από χρήστες ή ειδικούς του πεδίου. Οι έρευνες χρηστών και τα ερωτηματολόγια είναι μια κοινή προσέγγιση, όπως εφαρμόστηκε σε μελέτες με μέλη ΔΕΠ που αξιολογούν συστήματα RAG ως βοηθούς διδασκαλίας και διδακτικά βοηθήματα. Τα ερωτηματολόγια αυτά μπορούν να χρησιμοποιούν κλίμακες, όπως η 5βάθμια κλίμακα Likert, για τη μέτρηση της χρησιμότητας και της ευχρηστίας. Η συλλογή γραπτών απαντήσεων ή ανοιχτών σχόλιων επιτρέπει στους χρήστες να διατυπώνουν συγκεκριμένες προκλήσεις, περιορισμούς και προτάσεις βελτίωσης [6]. Σε μια άλλη μελέτη, φοιτητές χρησιμοποίησαν σύστημα ενισχυμένο με RAG για να λάβουν ανατροφοδότηση σε εργασίες προγραμματισμού και κλήθηκαν να αξιολογήσουν το κατά πόσο η επέκταση RAG ήταν χρήσιμη [2]. Ο χειρωνακτικός έλεγχος ή η ποιοτική αξιολόγηση από ανθρώπινους αναλυτές είναι επίσης κρίσιμη για την εκτίμηση της αποδοχής και της συνέπειας των παραγόμενων απαντήσεων από ανθρώπινη σκοπιά, συμπληρώνοντας τα ποσοτικά αποτελέσματα.

Οι ποιοτικές μέθοδοι αυτές είναι ουσιώδεις για τον εντοπισμό πτυχών που μπορεί να παραλείψουν οι ποσοτικές μετρικές, όπως προβλήματα στην κατανόηση του ιστορικού συνομιλίας, η ανάγκη για καλύτερη ανάλυση δομικών γνώσεων [5], περιορισμοί στην επεξεργασία ειδικών τύπων δεδομένων όπως εικόνες και μαθηματικές εξισώσεις, ή η σημασία χαρακτηριστικών όπως η μέθοδος «ειδικός εντός βρόχου» για την παρακολούθηση της ορθότητας των αποτελεσμάτων [9]. Οι ποιοτικές αξιολογήσεις αποκαλύπτουν τις προτιμήσεις των χρηστών και αναδεικνύουν τομείς που χρειάζονται βελτίωση για αποτελεσματική υιοθέτηση σε μεγάλη κλίμακα.

## 3.6 Ηθικές, Θέματα Απορρήτου και Ασφάλειας

Ο σχεδιασμός και η υλοποίηση εικονικών βοηθών που βασίζονται σε RAG pipelines και αξιοποιούν εξωτερικές πηγές γνώσης, όπως οδηγοί μελέτης, απαιτεί προσεκτική εξέταση των ηθικών συνεπειών, της προστασίας των ευαίσθητων δεδομένων και της ασφάλειας του συστήματος. Παρότι τα άρθρα υπογραμμίζουν τις τεχνικές εξελίξεις και τα δυνητικά οφέλη του RAG σε εκπαιδευτικά περιβάλλοντα, το βάθος της συζήτησης για αυτές τις κρίσιμες μη τεχνικές πτυχές ποικίλλει. Όπως έχει επισημανθεί, υπάρχει αξιοσημείωτη έλλειψη εκτενούς κάλυψης όσον αφορά ολοκληρωμένα ηθικά πλαίσια, λεπτομερή πρωτόκολλα απορρήτου ή ισχυρές αρχιτεκτονικές ασφάλειας σε πολλές από τις εξετασθείσες εργασίες.

### 3.6.1 Απόρρητο Δεδομένων και Ανωνυμοποίηση

Ο κύριος λόγος αξιοποίησης των συστημάτων RAG σε εφαρμογές με εξειδικευμένο πεδίο, ιδιαίτερα όταν εμπλέκονται ενδεχομένως ευαίσθητες πληροφορίες όπως οδηγοί σπουδών ή εσωτερικά έγγραφα πανεπιστημίων, απορρέει από ανησυχίες περί ιδιωτικότητας δεδομένων. Τα domain-specific σώματα κειμένων συχνά περιέχουν πληροφορίες που δεν μπορούν να ενσωματωθούν στα εκπαιδευτικά δεδομένα των LLMs λόγω περιορισμών απορρήτου, καθιστώντας το RAG την αναγκαία προσέγγιση για «εξωτερική» τροφοδότηση των LLMs με αυτά τα δεδομένα. Αυτό αναδεικνύει πώς το RAG βοηθά να αντιμετωπιστεί ένα θεμελιώδες ζήτημα ιδιωτικότητας, διατηρώντας τα ευαίσθητα εξωτερικά δεδομένα ξεχωριστά από τον πυρήνα της εκπαίδευσης του LLM και προσπελαύνοντάς τα μόνο όταν απαιτείται για συγκεκριμένα ερωτήματα [5].

Ενώ τα άρθρα τεκμηριώνουν την ανάγκη για RAG λόγω προστασίας της ιδιωτικότητας, οι εφαρμογές τεχνικών ανωνυμοποίησης δεδομένων αναφέρονται σπάνια. Σε μία μελέτη για τη βελτίωση της υποστήριξης διεθνών μεταπτυχιακών φοιτητών, αναφέρονται ρητά βήματα κατά την προεπεξεργασία των δεδομένων για τη διασφάλιση υψηλών προτύπων απορρήτου, όπως η συστηματική αφαίρεση Πληροφοριών Προσωπικά Ταυτοποιήσιμων Δεδομένων (PII) από τα συλλεγόμενα δεδομένα, προκειμένου να προστατευτεί η ανωνυμία των συνεισφερόντων. Σε άλλη μελέτη σημειώνεται η επίσημη άδεια για την άντληση δεδομένων από ιστοσελίδες πανεπιστημίων, αναγνωρίζοντας εμμέσως τα ζητήματα ιδιοκτησίας και πρόσβασης στα δεδομένα. Αυτά τα παραδείγματα δείχνουν βήματα με σεβασμό στην ιδιωτικότητα, χωρίς όμως να προσφέρουν εκτενείς κατευθυντήριες οδηγίες για ανωνυμοποίηση ποικίλων τύπων δεδομένων σε εκπαιδευτικά RAG συστήματα [5].

### 3.6.2 Αρχιτεκτονικές Ασφάλειας και Έλεγχος Πρόσβασης

Οι αναφορές σχετικά με αρχιτεκτονικές ασφάλειας και μηχανισμούς ελέγχου πρόσβασης για τους RAG-based εικονικούς βοηθούς είναι περιορισμένες. Μόνο ένα έργο, που αφορά έναν εικονικό βοηθό, επισημαίνει την ενσωμάτωση του ZeroTier για την ενίσχυση της ασφάλειας, διασφαλίζοντας κρυπτογραφημένη επικοινωνία, λειτουργία σε ιδιωτικό εικονικό δίκτυο και μηχανισμούς ελέγχου πρόσβασης και διαχείρισης χρηστών [4].

Εκτός από εργαλεία δικτυακής ασφάλειας, μια πηγή αναφέρει την αφαίρεση ανεπιθύμητου περιεχομένου όπως διαφημίσεις και spam στη φάση προεπεξεργασίας ως μορφή ασφάλειας και καλής πρακτικής δεδομένων. Ωστόσο, δεν υπάρχουν εκτενείς λεπτομέρειες για ευρύτερες αρχιτεκτονικές ασφάλειας, στρατηγικές προστασίας της βάσης διανυσμάτων από μη εξουσιοδοτημένη πρόσβαση, ασφαλείς ενσωματώσεις API ή ισχυρές ορθολογισμένες μεθόδους αυθεντικοποίησης χρηστών. Αυτό δείχνει ότι, παρότι μεμονωμένα έργα εφαρμόζουν μέτρα ασφάλειας, ένα γενικευμένο πλαίσιο ή διεξοδική ανάλυση ασφαλείας για εκπαιδευτικές RAG υλοποιήσεις απουσιάζει σε μεγάλο βαθμό από τις πηγές.

### 3.6.3 Ηθικά Μέτρα και Υπεύθυνη Τεχνητή Νοημοσύνη

Όπως έχει επισημανθεί, παρά τη συζήτηση ορισμένων ηθικών ζητημάτων, δεν υπάρχει κεντρική, ολοκληρωμένη προσέγγιση για υπεύθυνη ανάπτυξη AI σε αυτά τα συστήματα RAG. Μία σημαντική ηθική ανησυχία που στοχεύουν να μετριάσουν τα συστήματα RAG είναι τα «hallucination», δηλαδή η παραγωγή ανακριβών πληροφοριών από τα LLMs. Η διασφάλιση της πιστότητας στις εξωτερικές πηγές γνώσης είναι απαραίτητη για αξιόπιστες απαντήσεις, ιδιαίτερα σε κρίσιμους τομείς όπως οι πανεπιστημιακές εισαγωγές ή η παροχή ακαδημαϊκής ανατροφοδότησης. Ο κίνδυνος διάδοσης λανθασμένων πληροφοριών, π.χ. ψευδών στοιχείων για υποτροφίες ή παραπλάνηση φοιτητών, υπογραμμίζει την ηθική ευθύνη για ακρίβεια. Επίσης αναγνωρίζεται η πιθανότητα χρήσης κειμένου AI για αντιγραφή στις εργασίες ως ηθικό ζήτημα στην εκπαίδευση [12??].

Για την αντιμετώπιση αυτών των ηθικών προκλήσεων και την υπεύθυνη ανάπτυξη, προτείνονται ή εφαρμόζονται ορισμένα μέτρα. Τονίζεται η ανάγκη προσεκτικής εξέτασης των ηθικών συνεπειών και η ανάπτυξη κατάλληλων μηχανισμών για υπεύθυνη εφαρμογή [9]. Συνιστάται η χρήση «ειδικού εντός βρόχου» (expert-in-the-loop), επιτρέποντας σε μέλη ΔΕΠ ή ειδικούς να παρακολουθούν τη χρήση του συστήματος και να επαληθεύουν την ορθότητα των απαντήσεων. Το διδακτικό προσωπικό προτείνει επίσης μηχανισμούς αναφοράς χρήσης και ορθότητας, και δυνατότητα υποβολής αναφορών από φοιτητές για πιθανώς λανθασμένες απαντήσεις. Επιπλέον, αναφέρεται η ενσωμάτωση εντοπισμού και διόρθωσης μεροληψιών κατά την προεπεξεργασία δεδομένων και η συνεχιζόμενη παρακολούθηση για την τήρηση ηθικών προτύπων και τη μείωση προκαταλήψεων AI [3]. Αυτά τα σημεία δείχνουν ευαισθητοποίηση για ηθικά ζητήματα, αν και οι λεπτομέρειες εφαρμογής και η έκταση των προστατευτικών μηχανισμών ποικίλλουν, υποδεικνύοντας την ανάγκη για πιο ολοκληρωμένα ηθικά πλαίσια σε αυτόν τον τομέα.

## 3.7 Ανάλυση Ελλείψεων: Ψηφιακοί Βοηθοί RAG στα Ελληνικά

Συνοπτικά, παρά την ανάπτυξη σημαντικών πόρων NLP για τα ελληνικά (π.χ. μελέτες και εργαλεία όπως η ανασκόπηση τεχνολογιών NLP του ACM[10] και το Meltemi 7B ), τα ελληνικά εκπαιδευτικά chatbots παραμένουν περιορισμένα σε αριθμό και λειτουργικότητα, αντιμετωπίζοντας σύνθετες γλωσσικές και πολιτισμικές προκλήσεις (π.χ. μορφολογία, διάλεκτοι, διγλωσσία [11]). Επιπλέον, δεν εντοπίζονται στη διεθνή ή την ελληνική βιβλιογραφία συστήματα Retrieval-Augmented Generation (RAG) που να καθοδηγούνται από συγκεκριμένους οδηγούς μελέτης στα ελληνικά, γεγονός που υπογραμμίζει μια κρίσιμη έλλειψη στην τρέχουσα έρευνα.

### 3.7.1 Επισκόπηση Υπαρχόντων Ελληνικών Εκπαιδευτικών Chatbots

Ο AsasaraBot αποτελεί μια από τις ελάχιστες πειραματικές υλοποιήσεις ελληνικού εκπαιδευτικού chatbot, σχεδιασμένο για τη διδασκαλία πολιτιστικού περιεχομένου σε μαθητές Λυκείου με μεθοδολογία CLIL [12]. Επίσης, το VAmooc, εκπαιδευτικό chatbot που αναπτύχθηκε στο πλαίσιο MOOC για παρόχους ψηφιακών υπηρεσιών, στοχεύει στην εκπαίδευση επαγγελματιών πληροφορικής για εικονικούς βοηθούς [13]. Το πρόγραμμα «Το Πρώτο Μου Chatbot» του Ιδρύματος Ευγενίδου ενσωματώνει δραστηριότητες δημιουργίας chatbot σε μαθητές Δευτεροβάθμιας, αλλά παραμένει καθαρά εκπαιδευτικό και όχι υποστηρικτικό σύστημα φοιτητών [14]. Τέλος, σε εμπορικό επίπεδο, πλατφόρμες όπως το Mondly προσφέρουν ελληνικά μαθήματα μέσω chatbot με έμφαση στην εξάσκηση φωνής και κειμένου, ενώ το Talkpal παρέχει προσωποποιημένες συνεδρίες μάθησης ελληνικών [15].

### 3.7.2 Γλωσσικές και Πολιτισμικές Προκλήσεις στα Ελληνικά

Η υψηλή μορφολογική πολυπλοκότητα της ελληνικής, με πληθώρα πτώσεων, χρόνων και εγκλιτικών, δυσχεραίνει τον σχεδιασμό αξιόπιστων dialog managers και parsers [16]. Σε επίπεδο μοντέλων, μονογλωσσικά συστήματα όπως το GREEK-BERT έχουν επιδείξει κορυφαία αποτελέσματα σε POS tagging και NER, όμως προέρχονται από γενικά corpora και δεν καλύπτουν εξειδικευμένο εκπαιδευτικό λεξιλόγιο [17]. Αντίστοιχα, το GreekBART στοχεύει σε γενική NLG για τα ελληνικά, χωρίς λειτουργίες συγκεκριμένης διαλόγου ή οδηγιών μελέτης [18]. Το ανοικτό GR-NLP-TOOLKIT προσφέρει βασικά NLP tasks, αλλά δεν περιλαμβάνει modules RAG ή custom retrieval για διδακτικά εγχειρίδια [19]. Παρά το γεγονός ότι το Meltemi 7B εκπαιδεύτηκε σε 40 δισ. tokens ελληνικού κειμένου, δεν παρέχει ενσωματωμένες μηχανισμούς retrieval ή οδηγούς μελέτης .

### 3.7.3 Έλλειψη Συστημάτων RAG Καθοδηγούμενων από Οδηγούς Μελέτης στα Ελληνικά

Το RAG (Retrieval-Augmented Generation) συνδυάζει την ανάκτηση εξωτερικών γνώσεων με τη γλωσσική παραγωγή, βελτιώνοντας την ακρίβεια και την επικαιρότητα των απαντήσεων. Παρά το διεθνές ενδιαφέρον για RAG, όπου αναδεικνύονται τεχνικές σύνδεσης LLMs με vector DBs για ακριβείς τεκμηριωμένες απαντήσεις, δεν εντοπίζονται υλοποιήσεις που εστιάζουν σε ελληνικά εκπαιδευτικά εγχειρίδια ή οδηγούς μελέτης. Αυτή η απουσία καθιστά αναγκαία την ανάπτυξη ολοκληρωμένων ελληνικών συστημάτων RAG, ικανά να παρέχουν εξατομικευμένη διοικητική υποστήριξη βάσει επίσημων εκπαιδευτικών πόρων.

Βιβλιογραφικές Αναφορές

[1] 2407.14246v3.pdf

[2] Leveraging\_Lecture\_Content\_for\_Improved\_Feedback\_Explorations\_with\_GPT-4\_and\_Retrieval\_Augmented\_Generation.pdf

[3] Enhancing\_International\_Graduate\_Student\_Experience\_through\_AI-Driven\_Support\_Systems\_A\_LLM\_and\_RAG-Based\_Approach.pdf

[4] The\_enhanced\_context\_for\_AI-generated\_learning\_advisors\_with\_Advanced\_RAG.pdf

[5] 2406.05654v2.pdf

[6] 2411.08438v1.pdf

[7] 2411.06237v2.pdf

[8] A\_Chat\_Bot\_for\_Enrollment\_of\_Xi\_an\_Jiaotong-Liverpool\_University\_Based\_on\_RAG.pdf

[9] 2408.01462v1.pdf

[10] [ACM Digital Library](https://dl.acm.org/doi/fullHtml/10.1145/3635059.3635062?utm_source=chatgpt.com)

[11] [users.ics.forth.gr](https://users.ics.forth.gr/~tzitzik/publications/Tzitzikas_2020_SETN.pdf?utm_source=chatgpt.com)

[12] [MDPI](https://www.mdpi.com/2076-3417/12/7/3239?utm_source=chatgpt.com)

[13] [virtualassistantmooc.eu](https://virtualassistantmooc.eu/2023/04/06/launch-of-the-virtual-assistants-tutoring-chatbot/?lang=el&utm_source=chatgpt.com)

[14] [Eugenides Foundation](https://www.eef.edu.gr/el/programme/to-proto-mou-chatbot/?utm_source=chatgpt.com)

[15] [Talkpal](https://talkpal.ai/learn-greek/?utm_source=chatgpt.com)

[16] [users.ics.forth.gr](https://users.ics.forth.gr/~tzitzik/publications/Tzitzikas_2020_SETN.pdf?utm_source=chatgpt.com)

[17] [arXiv](https://arxiv.org/abs/2008.12014?utm_source=chatgpt.com)

[18] [arXiv](https://arxiv.org/abs/2304.00869?utm_source=chatgpt.com)

[19] [arXiv](https://arxiv.org/abs/2412.08520?utm_source=chatgpt.com)

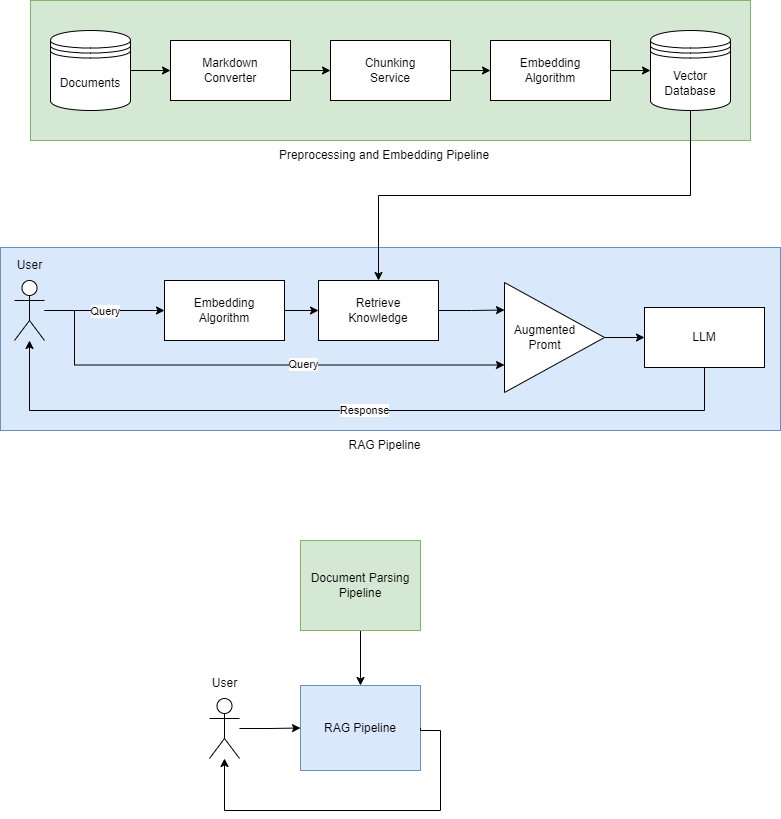
# ΚΕΦΑΛΑΙΟ 4: Σχεδίαση Ευφυούς Συστήματος

## 4.1 Εισαγωγή

Το κεφάλαιο αυτό εστιάζει στη θεμελίωση της σχεδίασης του ευφυούς συστήματος διοικητικής υποστήριξης φοιτητών, καθορίζοντας το πλαίσιο, τους στόχους και τις βασικές αρχές που διέπουν την ανάπτυξη των επιμέρους συνιστωσών. Συγκεκριμένα, μέσα από αυτή την εισαγωγή επιδιώκεται ο καθορισμός των λειτουργικών και μη-λειτουργικών απαιτήσεων που θα υποστηρίξουν την αξιόπιστη και αποδοτική λειτουργία του συστήματος.Παρουσιάζεται η αρχιτεκτονική του συστήματος,οι τεχνολογικές επιλογές και οι πρακτικές που επιλέχθηκαν με βάση κριτήρια επεκτασιμότητας, συντηρησιμότητας και ασφάλειας.Γίνεται επισκόπηση της διαδικασίας του RAG, η οποία αποτελεί τον πυρήνα της λειτουργίας του συστήματος για την υποστήριξη ακριβών και εμπλουτισμένων απαντήσεων.

## 4.2 Αρχιτεκτονική Συστήματος

Η αρχιτεκτονική του ψηφιακού βοηθού στηρίζεται σε δύο διακριτές, αλλά αλληλεπιδρούσες φάσεις: την προεπεξεργασία και ενσωμάτωση εγγράφων (Preprocessing & Embedding Pipeline) και την ανάκτηση γνώσης για την παραγωγή επαυξημένων απαντήσεων (Retrieval-Augmented Generation, RAG). Στην πρώτη φάση, τα πρωτογενή έγγραφα μετατρέπονται από PDF σε Markdown, διατηρώντας παράλληλα τη δομική πληροφορία (τίτλοι, υπότιτλοι, λίστες). Το κείμενο αυτό χωρίζεται σε τμήματα («chunks») βάση κανόνων, ώστε να εξασφαλίζεται η συνεπής και αποδοτική αντιστοίχισή τους σε διανυσματικές αναπαραστάσεις μέσω προκαθορισμένου embedding μοντέλου (π.χ. multilingual BERT, MiniLM, MPNet). Τα παραγόμενα διάνυσμα­ αποθηκεύονται σε ειδική διανυσματική βάση δεδομένων (vector database), η οποία επιτρέπει ταχύτατη αναζήτηση κοντινών σημεία με μετρικές (όπως η κοσινική ομοιότητα και η Ευκλείδεια απόσταση).

****

Κατά τη δεύτερη φάση, ο χρήστης κάνει ένα ερώτημα στον ψηφιακό βοηθό. Το ερώτημα αυτό μετατρέπεται σε embeddings μέσω του ίδιου μοντέλου που χρησιμοποιήθηκε και για την μετατροπή δεδομένων στο πρώτο στάδιο. Το ερώτημα ως διάνυσμα χρησιμοποιείται για την ανάκτηση των κορυφαίων τμημάτων κειμένου (top-k) που παρουσιάζουν τη μεγαλύτερη ομοιότητα με το αρχικό ερώτημα. Τα ανακτηθέντα αποσπάσματα ενσωματώνονται στο τελικό prompt, το οποίο στέλνεται στο LLM. Τέλος το LLM επεξεργάζεται το ερώτημα και απαντάει στον χρήστη. Με αυτόν τον τρόπο, αποτρέπονται σε μεγάλο βαθμό τα hallucinations των LLMs, διασφαλίζοντας πολύ μεγάλη ακρίβεια στις απαντήσεις. Η αρχιτεκτονική αυτή παρουσιάζει σημαντικά πλεονεκτήματα σε όρους επεκτασιμότητας και απόδοσης. Ο διαχωρισμός της υπολογιστικής φάσης των embedding από την πραγματική φάση εξυπηρέτησης ερωτημάτων μειώνει δραστικά τον χρόνο απόκρισης, ενώ η ευκολία ενημέρωσης νέων πηγών μέσω απλού rerun του preprocessing pipeline επιτρέπει συνεχή διάχυση επικαιροποιημένης γνώσης.

### 4.2.2 Διεπαφή Χρήστη (Frontend)

Η διεπαφή χρήστη αναπτύχθηκε βάση του [ChristophHandschuh/chatbot-ui](https://github.com/ChristophHandschuh/chatbot-ui) template. Αναπτύχθηκε σε React με την γλώσσα TypeScript και μορφοποιήθηκε με Tailwind CSS, δίνοντας ένα μοντέρνο και μινιμαλιστικό αίσθημα. Γίνεται επίσης χρήση πολλών UI components από το Vercel’s open-source AI Chatbot project. Στην συνομιλία υποστηρίζεται Markdown κείμενο δίνοντας έτσι την δυνατότητα για πιο πλούσια σε μορφή κείμενα. Το template αυτό τροποποιήθηκε ώστε να συναντά τις ανάγκες της διπλωματικής εργασίας. Προστέθηκαν αρκετά components και τροποποιήθηκε η συμπεριφορά άλλων, με σκοπό να αναπτυχθεί μια απλή και εύχρηστη διεπαφή για τον χρήστη ώστε να μπορέσει να δοκιμάσει τον ψηφιακό βοηθό. Τέλος, η επικοινωνία με το backend γίνεται με την διεπαφή να συνδέεται σε ένα websocket.

### 4.2.3 Business Logic Layer (Backend)

Ο ψηφιακός βοηθός σερβίρεται στον χρήστη μέσω ενός απλού websocket server. Ο server αυτός δέχεται αιτήματα από την διεπαφή χρήστη. Τα δεδομένα αυτά εισέρχονται στο RAG pipeline. Το σύστημα ανακτά δεδομένα σχετικά με την ερώτηση του χρήστη από την διανυσματική βάση και συντάσει ένα μεγάλο prompt. Εκείνο δίνεται στο LLM και η απάντηση προωθείται στην γραφική διεπαφή. Η διανυσματική βάση δημιουργείται με το FAISS. Το FAISS (Facebook AI Similarity Search) είναι μια C++ βιβλιοθήκη με Python bindings, σχεδιασμένη για αποτελεσματική αναζήτηση ομοιότητας και ομαδοποίηση (clustering) πυκνών διανυσμάτων. Είναι μια πολύ καλή επιλογή για την αποθήκευση των δεδομένων λόγω των εξαιρετικών επιδόσεων. Τέλος, η επικοινωνία με το LLM γίνεται μέσω API κλήσεων.

### 4.2.4 Υποδομές & DevOps

Η εφαρμογή τυποποιείται πλήρως σε container χρησιμοποιώντας το Docker, εξασφαλίζοντας ότι μπορεί να τρέξει σε κάθε περιβάλλον. Το Dockerfile που δημιουργήθηκε περιγράφει με σαφήνεια όλα τα βήματα για την εγκατάσταση των διάφορων απαραίτητων εργαλείων και βιβλιοθηκών, τη μεταγλώττιση του κώδικα και την εκκίνηση της εφαρμογής, ενώ το image που δημιουργείται έχει μόνο ό,τι είναι απαραίτητο για την λειτουργία της. Με αυτόν τον τρόπο, η εφαρμογή μπορεί να τρέξει σε οποιοδήποτε περιβάλλον και μηχάνημα χωρίς την ανάγκη εγκατάστασης όλων των εργαλείων και βιβλιοθηκών που χρειάζεται η εφαρμογή για να λειτουργήσει. Για την αυτοματοποίηση της διαδικασίας CI/CD αξιοποιείται το Jenkins. Είναι ορισμένο ένα Jenkins Pipeline που, σε κάθε commit στο αποθετήριο, τραβήξει την νέα έκδοση του κώδικα, χτίζει το Docker image και εκτελεί την νέα έκδοση της εφαρμογής.

## 4.3 Knowledge Base

### 4.3.1 Πηγές Δεδομένων

Σκοπός της εργασίας αυτής είναι η εξυπηρέτηση φοιτητών όσον αφορά διοικητικά θέματα με χρήση ενός ψηφιακού βοηθού. Για τον λόγο αυτό, η βασική πηγή δεδομένων είναι ο οδηγός σπουδών του τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών του Πανεπιστημίου Πατρών.

### 4.3.2 Προεπεξεργασία & Καθαρισμός

Παρακάτω αναλύεται ο τρόπος με τον οποίο τα δεδομένατου οδηγού σπουδών (και γενικά οποιουδήποτε PDF εγγράφου) μετατρέπονται σε κομμάτια (chunks) έτοιμα για την τροφοδότηση ενός RAG–pipeline, καλύπτοντας τα βήματα ανάλυσης Markdown, διαχωρισμού και τελικής αποθήκευσης.

Σε γενικές γραμμές, η διαδικασία διακρίνεται σε δύο φάσεις:

1. **Προεπεξεργασία PDF → Markdown**
2. **Καθαρισμός Markdown & Chunking → JSON**

#### 4.3.2.1 Προεπεξεργασία PDF: Κατακερματισμός & Παράλληλη Μετατροπή

Αρχικά διαχωρίζονται οι σελίδες του αρχείου. Χρησιμοποιώντας τη βιβλιοθήκη PyPDF2, κάθε σελίδα του PDF απομονώνεται και γράφεται ξεχωριστά σε αρχείο PDF. Η βασική λογική είναι:

|  |
| --- |
| *ΑΡΧΗ main(inputPDF, outputMD):*  *# 1. Δημιουργία προσωρινού φακέλου*  *tempDir ← ΔΗΜΙΟΥΡΓΗΣΕ\_ΠΡΟΣΩΡΙΝΟ\_ΦΑΚΕΛΟ()*    *# 2. Διαχωρισμός PDF σε σελίδες*  *σελίδες ← splitPDF(inputPDF, tempDir)*  *# 3. Άνοιγμα αρχείου Markdown για εγγραφή*  *ΑΝΟΙΞΕ outputMD ΩΣ mdFile*  *ΓΙΑ κάθε σελ Σε σελίδες:*  *markdown ← convertPage(σελ)*  *ΓΡΑΨΕ markdown ΣΤΟ mdFile*  *ΓΡΑΨΕ "\n\n" ΣΤΟ mdFile*  *ΤΕΛΟΣ\_ΓΙΑ*  *ΚΛΕΙΣΕ mdFile*  *# 5. Καταστροφή προσωρινού φακέλου*  *ΔΙΑΓΡΑΨΕ tempDir*  *ΤΕΛΟΣ main* |

Η μετατροπή σε Markdown γίνεται παράλληλα με την βιβλιοθήκη docling. Οι μονές σελίδες στέλνονται σε ένα pool με worker processes, όπου κάθε worker παράλληλα μετατρέπει την κάθε σελίδα σε Markdown. Έτσι ένα σύνολο αλφαριθμητικών με δομή Markdown συγχωνεύονται σε ένα αρχείο .md. Όσο αναφορά το Docling, είναι ένα ανοιχτού κώδικα Python toolkit της IBM Research που απλοποιεί τη μετατροπή εγγράφων (PDF, DOCX, PPTX, HTML, εικόνες κ.ά.) σε δομημένη μορφή Markdown ή JSON καθιστώντας τα “machine-readable”, ιδανική για workflows με LLM και RAG. Έχει άδεια MIT και είναι ελαφρύ ώστε να μπορεί να τρέχει τοπικά. Ο πυρήνας του Docling είναι η κλάση **DocumentConverter**, η οποία διαχειρίζεται μια αλυσίδα βημάτων (pipeline) για κάθε είσοδο: φόρτωση format, (προαιρετικό) OCR, ανάλυση διάταξης, αναγνώριση πινάκων και εξαγωγή κειμένου. Βασικό εργαλείο που χρησιμοποιεί το docling είναι το DocLayNet, ένα εξειδικευμένο deep-learning μοντέλο για segmenting regions σελίδας (τίτλοι, παράγραφοι, εικόνες κ.ά.). Είναι βασισμένο σε μεγάλο ανθρώπινα επισημασμένο dataset (80k+ σελίδες με 11 κατηγορίες) [arXiv](https://arxiv.org/abs/2408.09869?utm_source=chatgpt.com)[arXiv](https://arxiv.org/abs/2206.01062?utm_source=chatgpt.com). Ένα ακόμα σημαντικό εργαλείο που καθιστά καινοτόμο και πρωτοπόρο το docling είναι το TableFormer. Αυτό εφαρμόζει έναν ψευδο-object-detection decoder για πίνακας και transformer-based decoders, πετυχαίνοντας κορυφαία scores (TEDS ~98.5% σε απλούς και ~95% σε σύνθετους πίνακες) [arXiv](https://arxiv.org/abs/2203.01017?utm_source=chatgpt.com). Αυτό επιτρέπει την πιστή αναπαράσταση πολύπλοκων πινάκων σε JSON arrays, διασφαλίζοντας την ποιότητα των δεδομένων. Τέλος, για σκαναρισμένα έγγραφα ή εικόνες, το Docling υποστηρίζει OCR με EasyOCR ή Tesseract, ενσωματώνοντάς το πριν την ανάλυση διάταξης ώστε να μετατρέπει εικόνες κειμένου σε επεξεργάσιμη μορφή.

#### 4.3.2.2 Καθαρισμός & Chunking Markdown

Για την διαχωρισμό του κειμένου σε κομμάτια, αρχικά το Markdown αρχείο διαβάζεται γραμμή προς γραμμή, και με regex *^(#{1,6})\s+(.\*)* αναγνωρίζονται headings επιπέδου 1–6, εξάγοντας το επίπεδο και τον τίτλο. Διατηρείται μια στοίβα που αναπαριστά την ιεραρχία των κεφαλίδων. Σε κάθε νέο heading «απορρίπτονται» οι παλαιότερες κεφαλίδες ίδιου ή ανώτερου επιπέδου ώστε η δομή να παραμένει δενδροειδής. Το κείμενο ανάμεσα στα headings συγκεντρώνεται και αποθηκεύεται σε αυτή την μορφή: *{ 'path': [...], 'data': '…' }*.

Έπειτα, με το SentenceTransformer (*lighteternal/stsb-xlm-r-greek-transfer*), τα chunks μετατρέπονται σε διανύσματα. Για κάθε ένα, υπολογίζεται η κοσινική ομοιότητα. Εάν η απόσταση φθάνει ή υπερβαίνει το *merge\_threshold=0.8*, το chunk συγχωνεύεται. Το αποτέλεσμα είναι ένας πίνακας JSON, όπου κάθε στοιχείο έχει:

|  |
| --- |
| *{*  *"path": ["Κεφάλαιο","Υπόκεφαλο","Παράγραφος"],*  *"data": "Το κείμενο αυτού του chunk"*  *}* |

Αυτή η δομή επιτρέπει εύκολη δρομολόγηση για ιεραρχική ανάκτηση δεδομένων.

### 4.3.3 Δημιουργία Embeddings & Storage

Τα embeddings αποτελούν διανυσματικές αναπαραστάσεις κειμένου (λέξεων, προτάσεων ή εγγράφων) σε έναν συνεχή, πεπερασμένο χώρο μικρού σχετικά πλήθους διαστάσεων. Με άλλα λόγια, κάθε μονάδα κειμένου μετατρέπεται σε έναν πραγματικό διάνυσμα v∈Rd, ούτως ώστε η γεωμετρική εγγύτητα μεταξύ διανυσμάτων να αντανακλά τη σημασιολογική ομοιότητα στο πρωτότυπο φυσικό λόγο. Τα embeddings παράγονται συνήθως από προεκπαιδευμένα νευρωνικά δίκτυα (π.χ. BERT, MiniLM, MPNet) ή από μοντέλα ειδικά προσαρμοσμένα σε παράγοντες όπως η γλώσσα (“multilingual”). Η ουσία της χρήσης τους σε εφαρμογές ανάκτησης πληροφορίας (IR) έγκειται στο ότι αναζητούμε, για ένα ερώτημα, τα ν κορυφαία έγγραφα των οποίων τα embeddings έχουν τη μεγαλύτερη συσχέτιση μέσω ενός αλγορίθμου ομοιότητας με το embedding του ερωτήματος. Όπως αποδυκνείεται και στα πειραμτικά αποτελέσματα του κεφαλαίου 5, ο καλύτερος embedding αλγόριθμος για την συγκεκριμένη εργασία είναι ο **lighteternal/stsb-xlm-r-greek-transfer.**

### 4.2.4 Διαχείριση Ανανεώσεων

Είναι γεγονός ότι ο οδηγός σπουδών ενός τμήματος ανανεώνεται ανά μια χρονική περίοδο, με αποτέλεσμα να είναι αναγκαία η υλοποίηση ενός μηχανισμού που ανανεώνει τα δεδομένα από τα οποία το RAG pipeline αντλεί γνώση. Με μια μέθοδο cron, ορίζεται η συχνότητα που πρέπει να ανανεώνονται τα δεδομένα. Έτσι κάθε φορά αντλούνται τα διάφορα αρχεία από τις τοποθεσίες που είναι αποθηκευμένα και επεξεργάζονται από την πρώτη φάση της προεπεξεργασίας και ενσωμάτωσης εγγράφων (Preprocessing & Embedding Pipeline), ώστε ο ψηφιακός βοηθός να είναι πάντα ενημερωμένος με τις νεότερες πληροφορίες.

**4.3 RAG Pipeline**

**4.3.1 Preprocessing Ερωτήματος**

**Tokenization, normalization**

**Intent Detection (FAQ vs γενική ερώτηση)**

**4.3.2 Retrieval**

**Vector search (k-NN, cosine similarity)**

**Filter & reranking με cross-encoder**

**4.3.3 Prompt Engineering & Generation**

**Σύγκλιση chunks στο σύστημα prompt**

**LLM call με παραμέτρους (temperature, max\_tokens)**

**4.3.4 Μετα-επεξεργασία & Έλεγχος**

**Formatting της απάντησης**

**Fact-checking & consistency**

**4.3.5 Error Handling & Fallback Strategies**

**Low-confidence retrieval → generative-only με disclaimer**

**Timeouts, retry policies**

**4.5.1 Απόδοση & Κλιμακωσιμότητα**

**Response times <200ms**

**Horizontal scaling services**

**4.5.2 Διαθεσιμότητα & Αξιοπιστία**

**SLA 99.9%**

**Failover & backup strategies**

**4.5.3 Ασφάλεια & Απόρρητο**

**Encryption in transit & at rest**

**GDPR compliance**

**4.5.4 Συντηρησιμότητα & Επεκτασιμότητα**

**Modular codebase**

**Clear API contracts**

**4.6 Σύνοψη**

**Συμπεράσματα σχεδίασης και προετοιμασία για την υλοποίηση.**

**Διάφορα**4.3.6 Metrics & Monitoring

Latency, accuracy of answers

Logging interactions για ανάλυση

4.4 Διεπαφή Χρήστη (UX/UI)

4.4.1 Χρήστης & Απαιτήσεις

Ταχύτητα απόκρισης

Ευανάγνωστη διάταξη

Προσβασιμότητα (WCAG)

4.4.2 Πληροφοριακή Αρχιτεκτονική

Δομή σελίδων: Chat, Dashboard, Resource Browser

4.4.3 Wireframes & Prototypes

Διαγράμματα ροής (user flows)

Mockups βασισμένα σε Figma/Sketch

4.4.4 Components Library & Theming

UI kit: Material UI / shadcn/ui

Light/Dark mode

4.4.5 Δοκιμές Προσβασιμότητας & Ευχρηστίας

Usability tests με φοιτητές

WCAG audits

4.5 Μη Λειτουργικές Απαιτήσεις

# ΚΕΦΑΛΑΙΟ 5: Πειραματικά Αποτελέσματα & Αξιολόγηση

Στο πλαίσιο της παρούσας διπλωματικής εργασίας παρουσιάζεται η σχεδίαση και υλοποίηση ενός ευφυούς συστήματος διοικητικής υποστήριξης φοιτητών, LLM με ενίσχυση RAG. Στην εργασία αυτή διεξάγονται τρεις πειραματικές αξιολογήσεις με σκοπό την καλύτερη απόδοση του ψηφιακού βοηθού. Αρχικά γίνεται σύγκριση διαφορετικών μεθόδων κατακερματισμού δεδομένων. Έπειτα Έπειτα συγκρίνονται διαφορετικοί embedding αλγόριθμοι, καθώς είναι γεγονός ότι ειδικά στην Ελληνική γλώσσα τα εργαλεία αυτά υστερούν σε απόδοση και ικανοποιητικά αποτελέσματα. Τέλος, συγκρίνονται διαφορετικά κατώφλια (k) ανάκτησης δεδομένων, ώστε να εντοπιστεί το βέλτιστο κατώφλι που διασφαλίζει το ιδανικό ισοζύγιο μεταξύ κάλυψης πληροφορίας και αποφυγής θορύβου. Η αξιολόγηση για το βέλτιστο κατώφλι εμπεριέχει και σύγκριση με το εμπορικό εργαλείο ChatGPT.

## 5.1 Τυπικοί Χρήστες / Τυπικά Σενάρια

Η επιτυχία μιας εφαρμογής εξαρτάται σημαντικά από την κατανόηση και τον ορισμό των τυπικών χρηστών και σεναρίων χρήσης της. Πιο συγκεκριμένα, ο ορισμός των τυπικών χρηστών είναι μια διαδικασία, κατά την οποία λαμβάνει χώρα η λεπτομερής ανάλυση των χρηστών του συστήματος, δηλαδή καταγράφονται δημογραφικά στοιχεία όπως η ηλικία, το φύλο, η εκπαίδευση, το εκπαιδευτικό υπόβαθρο καθώς και τα ενδιαφέροντα του. Στην ενότητα αυτή, παρουσιάζονται οι βασικές κατηγορίες χρηστών μιας τέτοιας εφαρμογής και τα τυπικά σενάρια χρήσης.  
  
*Αρχικά οι βασικότεροι τυπικοί χρήστες είναι οι φοιτητές του τμήματος. Στόχος χρήσης της εφαρμογής είναι η ενημέρωσή τους για διάφορα διοικητικά θέματα που αφορούν το τμήμα, όπως και για πληροφορίες σχετικά με τα μαθήματα που διεξάγονται στο τμήμα.*

*Επιπλέον τυπικοί χρήστες είναι και το προσωπικό του πανεπιστημίου, όπως τα μέλη ΔΕΠ, ΕΔΙΠ, ΕΤΕΠ κ.α. Στόχος τους είναι να ενημερωθούν και να επιβεβαιώσουν διοικητικά θέματα και κανονισμούς του τμήματος.*

*Τέλος, τυπικοί χρήστες μπορούν να θεωρηθούν και τρίτοι, εκτός πανεπιστημίου, χρήστες. Στόχος τους είναι η ενημέρωση για διάφορα θέματα διοίκησης και λειτουργίας του τμήματος.*

Με τον ορισμό των τυπικών χρηστών, είναι δυνατόν να συνταχθούν τυπικά σενάρια / τυπικές ερωτήσεις που είναι πιθανό να γίνουν από τους προαναφερθέντες τυπικούς χρήστες. Παρακάτω παρατίθενται 15 τυπικές ερωτήσεις:

1. *Ποιός είναι ο Πρόεδρος του τμήματος;*
2. *Ποιό είναι το τηλέφωνο του Αλεξανδρίδη;*
3. *Ποιός είναι ο καθηγητής που διδάσκει το μάθημα Τεχνητή Νοημοσύνη ΙΙ;*
4. *Πόσα είναι τα εξάμηνα φοίτησης;*
5. *Δώσε πολύ αναλυτικά την διαδικασία δήλωσης παρακολούθησης μαθημάτων για το εξάμηνο*
6. *Τι πρέπει να κάνω για να αλλάξω κατεύθυνση;*
7. *Πόσες φορές μπορεί να γίνει αλλαγή κατεύθυνσης;*
8. *Τι βαθμό πρέπει να πάρω για να πετύχω το Άριστα στο δίπλωμά μου;*
9. *Δώσε το πλήρης Ακαδημαϊκό Ημερολόγιο*
10. *Πόσα ECTS έχει το μάθημα Ηλεκτρικά Κυκλώματα ΙΙ;*
11. *Πόσα είναι τα συνολικά ECTS του 3ου Εξαμήνου;*
12. *Δώσε περιγραφή για το μάθημα Γραμμική Άλγεβρα*
13. *Ποιός προΐσταται την γραμματεία του τμήματος;*
14. *Ποιοί είναι οι τέσσερις τομείς του τμήματος;*
15. *Δώσε πληροφορίες για την υποδομή E-Class*

## 5.2 Σύγκριση Chunker

Όπως προαναφέρθηκε, το εργαλείο που μετατρέπει το αρχείο μορφής PDF σε μορφή Markdown, εντοπίζει με εξαιρετική ακρίβεια τους τίτλους, τα κεφάλαια και τα υποκεφάλαια στο κείμενο. Κατά την μετατροπή, σε κάθε ένα από αυτά τα στοιχεία του κειμένου ανατίθενται ιεραρχικές κεφαλίδες (# Κεφαλίδα 1, ## Κεφαλίδα 2, ## Κεφαλίδα 3, κ.ο.κ). Κατά την διάρκει υλοποίησης της παρούσας εργασίας, αναπτύχθηκαν τρεις διαφορετικοί τρόποι για τον διαχωρισμό δεδομένων. Αρχικά ο απλός κατακερματιστής βάση των κεφαλίδων διαχωρίζει τα δεδομένα και παράλληλα διατηρεί σε ένα κομμάτι τυχόν πίνακες που υπάρχουν στο κείμενο χωρίς να τους κόβει στην μέση. Ο εξελιγμένος κατακερματιστής κάνει ό,τι κάνει ο απλός, με επιπλέον βήμα να συγκρίνει όλα τα παραγμένα κομμάτια και να συγχωνεύει εκείνα που εννοιολογικά μοιάζουν περισσότερο. Τέλος, αναπτύχθηκε και μια τρίτη μέθοδος κατακερματιστή, ο κατακερματιστής δένδρου. Εκείνος, ακολουθεί ιεραρχική λογική. Βασίζεται στις κεφαλίδες που εισήγαγε ο μετατροπέας των δεδομένων και δημιουργεί μονοπάτια βάση των κεφαλίδων για κάθε κομμάτι που δημιουργεί. Επίσης αντίστοιχα και με τον απλό κατακερματιστή, δίνει ιδιαίτερη προσοχή σε δομές πινάκων να τις διατηρεί άρτιες με σκοπό να μην χάνεται η πληροφορία.

Για την σύγκριση των τριών αυτών μεθόδων, χρησιμοποιήθηκαν οι 15 τυπικές ερωτήσεις και ο ίδιος αλγόριθμος embedding. Κατακερματίστηκε το ίδιο αρχείο και από τους τρεις αλγορίθμους και βάση των κομματιών έγινε η αντιστοίχιση των ερωτήσεων με τα κομμάτι που πρέπει να ανακτήσει ο αλγόριθμος από ειδικό.

Οι μετρικές που χρησιμοποιήθηκαν για την αξιολόγηση των μοντέλων στο παρόν πείραμα είναι οι εξής:

* **Precision@k:** Το ποσοστό των πρώτων k ευρημάτων που είναι πραγματικά σχετικές απαντήσεις. Αποτυπώνει την ακρίβεια των κορυφαίων αποτελεσμάτων.
* **Recall@k:** Το ποσοστό των συνολικών σχετικών απαντήσεων που εντοπίζονται μέσα στα πρώτα k αποτελέσματα. Μετρά το εύρος κάλυψης της ανάκτησης.
* **MAP@k (Mean Average Precision):** Ο μέσος όρος της ακρίβειας σε κάθε θέση όπου εμφανίζεται ένα σχετικό αποτέλεσμα, περιορισμένος στα πρώτα k αποτελέσματα. Συνδυάζει ακρίβεια και θέση κατάταξης σε ένα ενιαίο μέτρο.
* **MRR (Mean Reciprocal Rank):** Ο μέσος όρος του αντίστροφου της θέσης κατάταξης της πρώτης σχετικής απάντησης για κάθε ερώτημα. Αντιπροσωπεύει πόσο γρήγορα εμφανίζεται η πρώτη ορθή απάντηση.

Τα πειραματικά δεδομένα αναδεικνύουν σαφή απόκλιση στη συμπεριφορά μεταξύ της ιεραρχικής μεθόδου και των απλών μεθόδων κατακερματισμού, ιδιαίτερα καθώς αυξάνεται ο αριθμός των ανακτώμενων κομματιών. Γενικά η εξειδικευμένη μέθοδος κατακερματισμού έχει τις λιγότερο ικανοποιητικές μετρικές. Αυτό συμβαίνει, διότι λόγω της ένωσης των εννοιολογικά όμοιων κομματιών εισάγεται πολύ θόρυβος στα κομμάτια, καθιστώντας μη ικανοποιητική την ανάκτησή τους. Στο σενάριο για k=1, η ιεραρχική μέθοδος (tree‐based) κατακερματισμού υπερέχει της απλής (simple) και και της εξελιγμένης (“advanced”) μεθόδου κατά περίπου 40% σε ακρίβεια, ανάκληση, MAP και MRR. Το εύρημα αυτό υποδηλώνει ότι η οργάνωση του κειμένου σε ιεραρχικές μονάδες (π.χ. ενότητα → υποενότητα → παράγραφος) συγκεντρώνει με πιο αποτελεσματικό τρόπο το σημασιολογικό περιεχόμενο σε διακριτά, υψηλής αξίας αποσπάσματα, βελτιώνοντας έτσι την πιθανότητα το αντικείμενο να είναι σχετικό.

Για τα αποτελέσματα όπου k=3,5,7, ο απλός κατακερματιστής διατηρεί διαρκώς την υψηλότερη ανάκληση, αποδεικνύοντας τη χρησιμότητά του όταν η κάλυψη σχετικού υλικού είναι πρωταρχική. Για k=3, επιτυγχάνει επίσης την καλύτερη ακρίβεια, ξεπερνώντας ελάχιστα τις μετρικές MAP και MRR σε σύγκριση με την ιεραρχική προσέγγιση. Ωστόσο παρατηρείται ότι παρότι η ανάκληση και η ακρίβεια της μεθόδου δέντρου μειώνονται πιο απότομα, οι τιμές MAP και MRR αυξάνονται σημαντικά, προσφέροντας κορυφαίες επιδόσεις για k=5. Αυτό δείχνει ότι ο ιεραρχικός τεμαχισμός, ενώ είναι πιο συντηρητικός στον αριθμό των σχετικών αποσπασμάτων, υπερέχει στην κατάταξη αυτών που όντως ανακτά από την στιγμή που δεν υπάρχουν downstream διεργασίες (π.χ. επαναβαθμολόγηση ή επανακατάταξη) που θα φιλτράρουν τυχόν ψευδώς θετικά, η υψηλή ποιότητα κατάταξης, καθιστώντας την μέθοδο κατακερματισμού με δέντρο την προτιμητέα επιλογή.

Έτσι συμπεραίνεται ότι με την μέθοδο δέντρου, η πληροφορία συγκεντρώνει με πιο αποτελεσματικό τρόπο και το σημασιολογικό περιεχόμενο σε διακριτά, υψηλής αξίας αποσπάσματα, βελτιώνοντας έτσι την πιθανότητα το αντικείμενο να είναι σχετικό. Επίσης, για k=5 αποδεικνύεται ότι επιτυγχάνεται το ιδανικό ισοζύγιο μεταξύ κάλυψης πληροφορίας και αποφυγής θορύβου.

|  |  |  |  |
| --- | --- | --- | --- |
| k=1 | Simple | Advanced | Tree |
| P@1 | 0.3333 | 0.3333 | 0.4667 |
| R@1 | 0.3 | 0.3 | 0.4667 |
| MAP@1 | 0.3333 | 0.3333 | 0.4667 |
| MRR@1 | 0.3333 | 0.3333 | 0.466 |

|  |  |  |  |
| --- | --- | --- | --- |
| k=3 | Simple | Advanced | Tree |
| P@3 | 0.2222 | 0.2 | 0.1556 |
| R@3 | 0.6333 | 0.5667 | 0.4667 |
| MAP@3 | 0.4778 | 0.4556 | 0.4667 |
| MRR@3 | 0.4778 | 0.4556 | 0.4667 |

|  |  |  |  |
| --- | --- | --- | --- |
| k=5 | Simple | Advanced | Tree |
| P@5 | 0.1333 | 0.12 | 0.12 |
| R@5 | 0.6333 | 0.5667 | 0.6 |
| MAP@5 | 0.4778 | 0.4556 | 0.5222 |
| MRR@5 | 0.4778 | 0.4556 | 0.5222 |

|  |  |  |  |
| --- | --- | --- | --- |
| k=7 | Simple | Advanced | Tree |
| P@7 | 0.1143 | 0.1143 | 0.0952 |
| R@7 | 0.7667 | 0.7333 | 0.6667 |
| MAP@7 | 0.5 | 0.454 | 0.5262 |
| MRR@7 | 0.5 | 0.4778 | 0.5262 |

## 5.3 Embeddings

Τα embeddings αποτελούν διανυσματικές αναπαραστάσεις κειμένου (λέξεων, προτάσεων ή εγγράφων) σε έναν συνεχή, πεπερασμένο χώρο μικρού σχετικά πλήθους διαστάσεων. Με άλλα λόγια, κάθε μονάδα κειμένου μετατρέπεται σε έναν πραγματικό διάνυσμα v∈Rd, ούτως ώστε η γεωμετρική εγγύτητα μεταξύ διανυσμάτων να αντανακλά τη σημασιολογική ομοιότητα στο πρωτότυπο φυσικό λόγο. Τα embeddings παράγονται συνήθως από προεκπαιδευμένα νευρωνικά δίκτυα (π.χ. BERT, MiniLM, MPNet) ή από μοντέλα ειδικά προσαρμοσμένα σε παράγοντες όπως η γλώσσα (“multilingual”). Η ουσία της χρήσης τους σε εφαρμογές ανάκτησης πληροφορίας (IR) έγκειται στο ότι αναζητούμε, για ένα ερώτημα, τα ν κορυφαία έγγραφα των οποίων τα embeddings έχουν τη μεγαλύτερη συσχέτιση μέσω ενός αλγορίθμου ομοιότητας με το embedding του ερωτήματος.

Οι μετρικές που χρησιμοποιήθηκαν για την αξιολόγηση των μοντέλων στο παρόν πείραμα είναι οι εξής:

* **Precision@1 (P@1)**: το ποσοστό των ερωτημάτων για τα οποία το κορυφαίο (θέση 1) έγγραφο είναι σχετικό με το ερώτημα.
* **Mean Average Precision (MAP)**: ο αθροιστικός μέσος όρος της ακρίβειας σε όλα τα σχετικά έγγραφα ανά ερώτημα. Μετρά όχι μόνο αν τα σχετικά έγγραφα εμφανίζονται ψηλά αλλά και τη σταθερότητα της κατάταξης συνολικά.
* **Precision@5 (P@5)**: το ποσοστό των σχετικών εγγράφων εντός των πέντε πρώτων αποτελεσμάτων, αθροιστικά μέσω όλων των ερωτημάτων.
* **Reciprocal Rank**: το αντίστροφο της θέσης του πρώτου σχετικού εγγράφου, μέσος όρος σε όλα τα ερωτήματα (π.χ. αν το πρώτο σχετικό έγγραφο είναι στην 2η θέση, συμβάλλει με 0.5).
* **nDCG@5** (normalized Discounted Cumulative Gain): μετρά την ποιότητα της κατάταξης μέχρι το 5ο αποτέλεσμα, λαμβάνοντας υπόψη τη φθίνουσα συνάθροιση των βαθμολογιών σχετικότητας.

Στο πίνακα των αποτελεσμάτων συγκρίνονται δώδεκα διαφορετικά multilingual μοντέλα sentence-transformers, BERT-based μοντέλα και ένα ειδικά μεταφορτισμένο μοντέλο για τα ελληνικά (lighteternal/stsb-xlm-r-greek-transfer). Παρατηρούμε ότι από τα αποτελέσματα της σύγκρισης τα γενικής χρήσης multilingual embeddings (π.χ. multi-qa-MiniLM-L6-cos-v1) επιτυγχάνουν μέτριες τιμές MAP (~0.16) και nDCG@5 (~0.21), ενώ τα πιο ελαφρά paraphrase-multilingual-MiniLM-L12-v2 παρουσιάζουν υψηλότερη nDCG@5 (0.3514) και P@5 (0.1750), αλλά μηδενική επίδοση στο P@1. Αντίθετα, τα BERT–based μοντέλα (google-bert/bert-base-multilingual-\*) καταγράφουν σχετικά χαμηλές τιμές MAP (<0.10) παρά τον μεγάλο χρόνο ενσωμάτωσης (>150 s). Ειδικά το intfloat/multilingual-e5-small παρέχει μεσαίες επιδόσεις (MAP=0.1302, nDCG@5=0.2630) με ταχύτητα embedding (~46 s).

Το μοντέλο **lighteternal/stsb-xlm-r-greek-transfer** αναδεικνύεται ως το πιό ισορροπημένο και αποδοτικό για εφαρμογές σε ελληνικό περιβάλλον. Επιτυγχάνει την υψηλότερη τιμή MAP (0.2052) και το μεγαλύτερο Reciprocal Rank (0.2417), διατηρώντας παράλληλα ικανοποιητικό P@1 (0.1250) και nDCG@5 (0.2913). Παρά το μέτριο κόστος χρόνου embedding (112 s), η σαφής υπεροχή του στην ακρίβεια της κατάταξης το καθιστά την καλύτερη επιλογή εκεί όπου η ποιότητα ανάκτησης προηγείται της ταχύτητας εκτέλεσης.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| P@5 | MAP | nDCG@5 | P@1 | Reciprocal Rank | Model | Embed Time (s) |
| 0.1 | 0.162 | 0.211 | 0.13 | 0.1917 | sentence-transformers/multi-qa-MiniLM-L6-cos-v1 | 29.11 |
| 0.18 | 0.199 | 0.3514 | 0 | 0.2146 | sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2 | 13.31 |
| 0.08 | 0.064 | 0.1213 | 0.13 | 0.1667 | sentence-transformers/all-MiniLM-L12-v2 | 14.34 |
| 0.15 | 0.13 | 0.263 | 0 | 0.1667 | sentence-transformers/paraphrase-multilingual-mpnet-base-v2 | 48.07 |
| 0.1 | 0.119 | 0.203 | 0 | 0.1396 | sentence-transformers/use-cmlm-multilingual | 91 |
| 0.08 | 0.107 | 0.1578 | 0 | 0.1292 | sentence-transformers/multi-qa-mpnet-base-dot-v1 | 219.92 |
| 0.08 | 0.066 | 0.1391 | 0.13 | 0.175 | sentence-transformers/distiluse-base-multilingual-cased-v2 | 25.63 |
| 0.08 | 0.073 | 0.1379 | 0.13 | 0.1979 | google-bert/bert-base-multilingual-cased | 159.57 |
| 0.15 | 0.13 | 0.263 | 0 | 0.1667 | intfloat/multilingual-e5-small | 46.21 |
| 0.1 | 0.098 | 0.18 | 0.13 | 0.1917 | google-bert/bert-base-multilingual-uncased | 147.73 |
| 0.13 | 0.205 | 0.2913 | 0.13 | 0.2417 | lighteternal/stsb-xlm-r-greek-transfer | 112.14 |
| 0.13 | 0.154 | 0.2669 | 0 | 0.1542 | nlpaueb/bert-base-greek-uncased-v1 | 125.4 |

## 5.4 Threshold και ποιότητα απάντησης

Στον παρακάτω πίνακα παρατίθενται οι συνδυασμοί που χρησιμοποιήθηκαν για την απάντηση των 15 ερωτήσεων, μαζί και το μέσο μήκος της απάντησης.

|  |  |  |
| --- | --- | --- |
| Model | k (chunks) | AVG Answer Length (Chars) |
| ChatGPT | 5 | 251 |
| Llama 3.1 | 5 | 830 |
| Llama 3.1 | 3 | 799 |
| Llama 3.1 | 1 | 450 |

Ανάλυση απαντήσεων:

Ε1: “Ποιός είναι ο Πρόεδρος του τμήματος;”

* Και οι 4 απαντήσεις επιστρέφουν: **Αντώνης Αλεξανδρίδης**, οπότε η βασική ανάκτηση δεδομένων λειτουργεί για όλες τις περιπτώσεις.
* Η διάρκεια της θητείας του προέδρου εμφανίζεται μόνο στην απάντηση του ChatGPT (k = 5) και του Llama 3.1 με k = 1, λόγω του ότι έτυχε το πρώτο chunk να οεριέχει αυτή την πληροφορία.
* Το Llama 3.1 με k = 5 και k = 3 δεν αναφέρθηκαν στην ημερομηνία θητείας του προέδρου πιθανώς γιατί αυτή η πληροφορία “χάθηκε”.

Ε2: “Ποιό είναι το τηλέφωνο του Αλεξανδρίδη;”

* Και οι τέσσερις απαντήσεις επιστρέφουν σωστά το τηλέφωνο **2610 996 404**.
* Το ChatGPT και το Llama για k=1 απαντάνε συνοπτικά και στοχευμένα, σε αντίθεση με το Llama για k=3 και k=5, οπού πάλι απαντάνε σωστά αλλά με λίγο πιο χαοτικό περιεχόμενο.

Ε3: Ποιός είναι ο καθηγητής που διδάσκει το μάθημα Τεχνητή Νοημοσύνη ΙΙ;

* Το ChatGPT (k = 5) απαντάει σωστά με επιπλέον πληροφορίες για τους διδάσκοντες που διεξάγουν το εργαστήριο.
* To Llama 3.1 (k = 5) απαντάει συνοπτικά και στοχευμένα.
* Το Llama 3.1 (k = 3) δίνει το όνομα μόνο ενός καθηγητή αλλά παραλείπει πληροφορίες για τους υπόλοιπους διδάσκοντες.
* Το Llama 3.1 (k = 1) ομοίως με το k=3 αναφέρει μόνο έναν διδάσκοντα και επίσης “ψεوماتικό” .

Ε4: “Πόσα είναι τα εξάμηνα φοίτησης;”

* Το ChatGPT (k = 5): απαντά σωστά στην ερώτηση
* Το Llama 3.1 (k = 5): απαντά σωστά στην ερώτηση
* Το Llama 3.1 (k = 3): δεν απαντά στην ερώτηση
* Το Llama 3.1 (k = 1): δεν απαντά στην ερώτηση

Ε5: Δώσε πολύ αναλυτικά την διαδικασία δήλωσης παρακολούθησης μαθημάτων για το εξάμηνο

Στην συγκεκριμένη ερώτηση το RAG pipeline δεν ανακτά καλά δεδομένα.

* Το ChatGPT (k = 5) καταλαβαίνει ότι δεν υπάρχουν πληροφορίες στα ανακτημένα δεδομένα και ζητάει επιπλέον πληροφορίες .
* Το Llama 3.1 επινοεί μια απάντηση.

Ε6: Τι πρέπει να κάνω για να αλλάξω κατεύθυνση;

* Το ChatGPT (k = 5) απαντά σωστά και στοχευμένα στην ερώτηση.
* Το Llama 3.1 (k = 5) απαντά σωστά στην ερώτηση
* Το Llama 3.1 (k = 3) απαντά εν μέρη σωστά αγνοώντας μερικές πληροφορίες .
* Το Llama 3.1 (k = 1) επινοεί πληροφορίες καθιστώντας την απάντηση πολύ χαμηλής ποιότητας.

Ε7: Πόσες φορές μπορεί να γίνει αλλαγή κατεύθυνσης;

* Το ChatGPT (k = 5): απαντά σωστά.
* Το Llama 3.1 (k = 5): απαντά αόριστα αλλά εν μέρη σωστά.
* Το Llama 3.1 (k = 3): δεν απαντά σωστά.
* Το Llama 3.1 (k = 1): απαντά σωστά και στοχευμένα.

Ε8: Τι βαθμό απαιτείται για άριστα (10) στο δίπλωμά μου;

* Το ChatGPT (k = 5): απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 5): απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 3): Δεν απαντά στην ερώτηση.
* Το Llama 3.1 (k = 1): Δεν απαντά στην ερώτηση.

Ε9: Δώσε το πλήρες Ακαδημαϊκό Ημερολόγιο

* Το ChatGPT (k = 5) απαντάει σωστά.
* Το Llama 3.1 (k = 5) δεν απαντά σωστά “χάνει” την πληροφορία.
* Το Llama 3.1 (k = 3) δεν απαντά σωστά.
* Το Llama 3.1 (k = 1) δεν απαντά σωστά.

Ε10: Πόσα ECTS έχει το μάθημα Ηλεκτρικά Κυκλώματα ΙΙ;

* Το ChatGPT (k = 5) καταλαβαίνει ότι δεν υπάρχουν πληροφορίες στα ανακτημένα δεδομένα και ζητάει επιπλέον πληροφορίες .
* Το Llama 3.1 επινοεί μια απάντηση.

Ε11: Πόσα είναι τα συνολικά ECTS του 3ου Εξαμήνου;

* Το ChatGPT (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 5) απαντάει σωστά.
* Το Llama 3.1 (k = 3) δεν απαντά σωστά.
* Το Llama 3.1 (k = 1) δεν βρίσκει πληροφορίες και ζητάει για περισσότερες πληροφορίες.

Ε12: Δώσε περιγραφή για το μάθημα Γραμμική Άλγεβρα

* Το ChatGPT (k = 5) απαντά σωστά στην ερώτηση.
* Το Llama 3.1 (k = 5) απαντά σωστά στην ερώτηση.
* Το Llama 3.1 (k = 3) απαντά σωστά στην ερώτηση.
* Το Llama 3.1 (k = 1) δεν απαντά σωστά στην ερώτηση .

Ε13: Ποιός προΐσταται τη Γραμματεία του Τμήματος;

* Το ChatGPT (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 3) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 1) απαντά σωστά και στοχευμένα.

Ε14: Ποιοί είναι οι τέσσερις τομείς του Τμήματος;

* Το ChatGPT (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 3) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 1) απαντά σωστά και στοχευμένα.

Ε15: Δώσε πληροφορίες για την υποδομή E-Class

* Το ChatGPT (k = 5) απαντά σωστά και στοχευμένα.
* Το Llama 3.1 (k = 5) απαντά σωστά.
* Το Llama 3.1 (k = 3) απαντά σωστά με λίγο επιπλέον θόρυβο.
* Το Llama 3.1 (k = 1) απαντά σωστά.

## 5.5 Συμπεράσματα

* **ChatGPT** ως closed source προϊόν απαντά εξαιρετικά καλά στις ερωτήσεις.
* **Llama 3.1 για k=5** ως open source εργαλείο απαντά εξαιρετικά καλά στις ερωτήσεις.
* **Llama 3.1** για χαμηλότερα k, δεν απαντά καλά στις ερωτήσεις καθώς λόγω έλλειψης πληροφορίας επινοεί απαντήσεις.

Χαμηλά k πολλές φορές δεν φέρνουν όλη την σχετική πληροφορία με αποτέλεσμα οι απαντήσεις να μην είναι ικανοποιητικές. Επίσης, ήδη για k=5 μερικές φορές το μοντέλο “χάνει” κρίσιμη πληροφορία μέσα στο θόρυβο, οπότε για k>5 υπάρχει υπερβολικός θόρυβος λόγω της πληθώρας δεδομένων με αποτέλεσμα να μην παράγονται ικανοποιητικές απαντήσεις. Συγκρίνοντας τις απαντήσεις του Llama3.1 με αυτές του ChatGPT για k=5 αποδεικνύεται ότι παρά το γεγονός ότι το Llama είναι εργαλείο ανοικτού κώδικα, λόγω της πολύ καλής απόδοσης του RAG pipeline επιτυγχάνει πολύ καλές απαντήσεις με γνώμονα το ChatGPT. Έτσι, τελικά, συμπεραίνεται ότι για k=5 επιτυγχάνεται το ιδανικό ισοζύγιο μεταξύ κάλυψης πληροφορίας και αποφυγής θορύβου.

# ΚΕΦΑΛΑΙΟ 6: Συμπεράσματα & Μελλοντικές Επεκτάσεις

## 6.1 Συνοπτικά συμπεράσματα

Motivation (Ποιο πρόβλημα λύνει):

* Διαθεσιμότητα: Η γραμματεία είναι διαθέσιμη 2 ώρες την ημέρα. Το δικό μου 24/7.
* Διαχείριση: Η γραμματεία μπορεί να διαχειριστεί μέχρι ένα όριο ερωτημάτων ενώ το δικό μου περιορίζεται μόνο από τα resources που διαθέτει το πανεπιστήμιο.
* Ακρίβεια: Με περαιτέρω εξέλιξη προσφέρει πάντα ακρίβεια και αξιοπιστία σε σχέση με τον άνθρωπο, καθώς ο άνθρωπος είναι πολύ πιο επιρρεπής σε λάθη.

(συνεχούς διαθεσιμότητας και της αποδοτικής διαχείρισης μεγάλου όγκου ερωτημάτων φοιτητών σχετικά με θεσμικές διαδικασίες και κανονισμούς)

Innovation (Καινοτομίες):

* Use of open source LLM (Δεν μπορεί να θεωρηθεί ανταγωνιστικό με product πχ OpenAI)
* End to End in Greek (Δεν υπάρχει κάποιος καταγεγραμμένος εικονικός βοηθός διοικητικής υποστήριξης φοιτητών σε πανεπιστημιακό ίδρυμα εξ ολοκλήρου στα Ελληνικά)
* Custom Chunking (Ιεραρχικός τρόπος διχοτόμησης των πληροφοριών βάση κεφαλαίων, υποκεφαλαίων και ιδιαίτερη διαχείριση πινάκων)
* Embedding Comparison (Σύγκριση 12 διαφορετικών embedding αλγορίθμων με σκοπό την εύρεση του καλύτερου για τα συγκεκριμένα δεδομένα)
* Ethics (Μέσω prompt engineering αντιμετωπίζονται ηθικά ζητήματα και ζητήματα προσωπικών δεδομένων)

Findings (Ευρήματα):

* Καλύτερος embedding αλγόριθμος για Ελληνικά
* Threshold για ανάκτηση πηγών (ούτε λίγες ούτε πολλές)
* Chunker:
  + SimpleChunker: Χωρίζει βάση παραγράφους (#, ##, …) και διαχειρίζεται ειδικά τους πίνακες
  + AdvancedChunker: Ότι κάνει ο SimpleChunker, αλλά επιπλέον κάνει merge σημασιολογικά κοντινά chunks
  + TreeChunker: Ιεραρχικός διαχωρισμός κειμένου. Παρόμοια λογική με SimpleChunker. Προσθέτει σε κάθε κομμάτι το ιεραρχικό μονοπάτι.

## 6.2 Περιορισμοί παρούσας υλοποίησης

* Open source models, tools, libraries, frameworks
* Γλώσσα υλοποίησης τα Ελληνικά

## 6.3 Προτάσεις για μελλοντική έρευνα και βελτιώσεις

* Closed source - product grade models, tools, libraries, frameworks (eg OpenAI subscription)
* Ίσως διαφορετικός τρόπος χωρισμού του κειμένου.
* Train Greek LLM and Greek embedding models