



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών &
Μηχανικών Γυπολογιστών
Τομέας Ηλεκτρονικής και Γυπολογιστών

Διπλωματική Εργασία

Αρχιτεκτονική και Βελτιστοποίηση Συστημάτων
Επαυξημένης Παραγωγής μέσω Ανάκτησης σε
Τεχνικά Ερωτήματα Μηχανικής Λογισμικού

Εκπόνηση:
Χατζηγεωργίου Σπύρος
ΑΕΜ: 10527

Επίβλεψη:
Καθ. Συμεωνίδης Ανδρέας

*Φτάσε όπου δεν μπορείς παιδί μου.
Μην ντραπείς αν έπαιξες καλά κι' έχασες.
Να ντραπείς αν έπαιξες κακά και κέρδισες*
— Νίκος Καζαντζάκης

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία αποτελεί το επιστέγασμα της ακαδημαϊκής μου πορείας. Μέσα από αυτήν επιδιώκω να ξεδιπλώσω τις ιδέες μου και να τις παρουσιάσω με την επιστημονική μέθοδο, όπως μου δίδαξαν όλα αυτά τα χρόνια οι καθηγητές μου. Η επιτυχής ολοκλήρωση της εργασίας συνιστά ένα μεγάλο ορόσημο στη ζωή μου. Μπορώ πλέον να κυνηγήσω τους επόμενους στόχους μου με θάρρος, αξιοπρέπεια και πολύτιμες γνώσεις, εκφράζοντας παράλληλα την ειλικρινή μου ευγνωμοσύνη προς όλους όσους με στήριξαν σε αυτή την όμορφη, και ενίστε δύσκολη, διαδρομή.

Πρωτίστως, θα ήθελα να εκφράσω την ευγνωμοσύνη και την εκτίμηση μου προς τον επιβλέποντα καθηγητή μου, Ανδρέα Συμεωνίδη, για την καθοδήγηση και τη στήριξή του καθ' όλη τη διάρκεια της εκπόνησης.

Ιδιαίτερες ευχαριστίες οφείλω στη NET2GRID και ειδικότερα στους Δημήτριο Δούκα και Νικόλαο Βιρτσιώνη, χάρη στους οποίους είχα την ευκαιρία να ασχοληθώ με το συγκεκριμένο θέμα.

Από καρδιάς εκφράζω την απεριόριστη αγάπη και εκτίμησή μου προς τους γονείς μου, την Άννα και τον Νικόλαο, για τη στήριξη και την αγάπη τους όλα αυτά τα χρόνια. Ένα μεγάλο ευχαριστώ αξίζει επίσης στα υπόλοιπα μέλη της οικογένειάς μου, τον Παναγιώτη, την Κατερίνα και την Ευαγγελία, για την αμέριστη αγάπη τους, τη συνεχή ενθάρρυνση και τη χαρά που μου προσφέρουν καθημερινά. Ιδιαίτερη μνεία οφείλω στον παππού μου, τον Σπύρο, ο οποίος υπήρξε για εμένα πρότυπο επιστήμονα και ανθρώπου, εμπνέοντάς με με το ήθος και τη συνέπειά του.

Οι φίλοι μου, που έχω την τύχη να είναι πολλοί και ξεχωριστοί, στάθηκαν δίπλα μου σε κάθε βήμα, και τους ευχαριστώ ολόψυχα.

Τέλος, αφιερώνω την παρούσα εργασία στην κοπέλα μου, τη Μαρία, για την πίστη της σε εμένα και στις ικανότητές μου, την ατελείωτη υποστήριξή της και τη μοναδική της ικανότητα να με εμπνέει. Της εύχομαι «να φτάσει εκεί όπου νομίζει ότι δεν μπορεί».

Περίληψη

Η εκθετική αύξηση του όγκου των διαθέσιμων πληροφοριών στη σύγχρονη φημιακή εποχή, σε συνδυασμό με την ανάπτυξη προηγμένων μοντέλων τεχνητής νοημοσύνης, έχει δημιουργήσει νέες δυνατότητες αλλά και σημαντικές προκλήσεις στον τομέα της επεξεργασίας φυσικής γλώσσας. Τα μεγάλα γλωσσικά μοντέλα (LLMs), παρά τις εντυπωσιακές τους δυνατότητες στην παραγωγή φυσικού κειμένου, αντιμετωπίζουν κρίσιμους περιορισμούς που σχετίζονται με την ακρίβεια και την επικαιρότητα των πληροφοριών που παρέχουν. Το φαινόμενο των παραισθήσεων (hallucinations), όπου τα μοντέλα παράγουν φανομενικά αξιόπιστες αλλά ανακριβείς πληροφορίες, καθώς και η αδυναμία πρόσβασης σε γνώση πέρα από τα δεδομένα εκπαίδευσής τους, περιορίζουν σημαντικά τη χρήση τους σε συστήματα που απαιτούν υψηλή αξιοπιστία.

Ανταποκρινόμενη σε αυτές τις προκλήσεις, η παρούσα εργασία εισάγει ένα επεκτάσιμο και ευπροσάρμοστο πλαίσιο πειραματισμού για συστήματα Επαυξημένης Παραγωγής μέσω Ανάκτησης (Retrieval-Augmented Generation, RAG), χρησιμοποιώντας ως πεδίο μελέτης τα τεχνικά ερωτήματα μηχανικής λογισμικού και αξιολογώντας πραγματικά δεδομένα. Το προτεινόμενο σύστημα σχεδιάστηκε με έμφαση στην επεκτασιμότητα, την αναπαραγωγισμότητα και την πλήρη ιχνηλασιμότητα των πειραμάτων, επιτρέποντας τη συστηματική διερεύνηση διαφορετικών μεθόδων ανάκτησης και παραγωγής.

Πέρα από το ανωτέρω τεχνικό πλαίσιο, η εργασία προτείνει μια μεθοδολογία ιεραρχικής βελτιστοποίησης και αξιολόγησης από άκρη σε άκρη (end-to-end) που συνδυάζει μετρικές ανάκτησης με ποιοτική αποτίμηση των παραγόμενων απαντήσεων μέσω της προσέγγισης «μεγάλο γλωσσικό μοντέλο ως κριτής» (LLM-as-a-Judge). Εισάγεται επίσης αρχιτεκτονική αυτοδιορθούμενης παραγωγής μέσω ανάκτησης (Self-RAG), η οποία ενσωματώνει βρόγχο επαλήθευσης και αναθεώρησης για τη βελτίωση της πιστότητας των απαντήσεων. Η συγκεκριμένη προσέγγιση συμβάλλει στην κωδικοποίηση της μεθοδολογίας κατασκευής συστημάτων RAG που θα επιφέρουν τη βέλτιστη απόδοση σε παραγωγικό περιβάλλον.

Συνολικά, η εργασία προτείνει στοχευμένες ενέργειες βελτιστοποίησης που συμβαδίζουν με την πρόσφατη βιβλιογραφία και ικανοποιούν τις σχεδιαστικές απαιτήσεις ενός παραγωγικού συστήματος.

Title

Architecture and Optimization of Retrieval Augmented Generation Systems in Technical Software Engineering Questions

Abstract

The exponential growth in the volume of available information in the modern digital age, combined with the development of advanced artificial intelligence models, has created new opportunities but also significant challenges in the field of natural language processing. Large language models (LLMs), despite their impressive capabilities in generating natural text, face critical limitations related to the accuracy of the information they provide. The phenomenon of hallucinations , where models generate seemingly reliable but inaccurate information, as well as the inability to access knowledge beyond their training data, significantly limit their use in systems that require high reliability.

Responding to these challenges, this thesis introduces an extensible and adaptable experimentation framework for Retrieval-Augmented Generation (RAG) systems, using technical software engineering queries as a testing ground and evaluating real-world data. The proposed system was designed with an emphasis on extensibility, reproducibility, and full traceability of experiments, allowing for the systematic exploration of different retrieval and generation methods.

Beyond the above technical framework, the thesis proposes a hierarchical optimization and end-to-end evaluation methodology that combines retrieval metrics with qualitative assessment of the generated responses through the "large language model as a judge" (LLM-as-a-Judge) approach. It also introduces a self-correcting retrieval-based generation (Self-RAG) architecture, which incorporates a verification and review loop to improve response faithfulness. This approach contributes to the codification of the methodology for constructing RAG systems that will deliver optimal performance in a production environment.

Overall, the thesis proposes targeted optimization actions that are in line with recent literature and meet the design requirements of a production system.

Spiros Chatzigeorgiou
Electrical & Computer Engineering Department,
Aristotle University of Thessaloniki, Greece
October 2025

Περιεχόμενα

Ευχαριστίες	iii
Περίληψη	v
Abstract	vii
Ακρωνύμια	1
1 Εισαγωγή	2
1.1 Περιγραφή του Προβλήματος	3
1.2 Συνεισφορά της Εργασίας	4
1.3 Διάρθρωση της Αναφοράς	5
2 Θεωρητικό Υπόβαθρο	7
2.1 Θεμελιώδεις έννοιες Νευρωνικών Δικτύων	7
2.1.1 Τεχνητός Νευρώνας και Βασικές Αρχές	7
2.1.2 Συναρτήσεις Ενεργοποίησης και ο Ρόλος τους	8
2.1.3 Το Πολυστρωματικό Perceptron (MLP)	10
2.1.4 Η Διαδικασία Μάθησης στα Νευρωνικά Δίκτυα	12
2.1.5 Περιορισμοί των Πολυστρωματικών Perceptron	15
2.2 Διακριτοποίηση σε λεκτικές μονάδες	16
2.2.1 Λεκτική μονάδα	16
2.2.2 Διακριτοποίηση σε λεκτικές μονάδες	16
2.2.3 Λεξιλόγιο	18
2.2.4 Είδη αναλυτών λεκτικών μονάδων	19
2.2.5 Κωδικοποίηση Ζεύγους Byte	19
2.3 Ενσωματώσεις	22
2.3.1 One-hot κωδικοποίηση	22
2.3.2 Συνεχείς Ενσωματώσεις	23
2.3.3 Σημασιολογικές Ενσωματώσεις	24
2.4 Μετασχηματιστές	25
2.4.1 Πολυκεφαλική Προσοχή	26
2.4.2 Η Αρχιτεκτονική του Μετασχηματιστή	27
2.5 Μεγάλα Γλωσσικά Μοντέλα	29
2.5.1 Η Αρχιτεκτονική Αποκωδικοποιητή	30
2.5.2 Μηχανισμός Πρόβλεψης και Αυτοπαλίνδρομη Παραγωγή	31
2.5.3 Εκπαίδευση Μεγάλων Γλωσσικών Μοντέλων	34
2.5.4 Προσαρμογή	35
2.6 Επαυξημένη Παραγωγή μέσω Ανάκτησης	36
2.6.1 Αρχιτεκτονική του RAG	37

ΠΕΡΙΕΧΟΜΕΝΑ

2.6.2	Βάσεις Δεδομένων Διανυσμάτων (Vector Databases)	37
2.6.3	Προκλήσεις και Υπερπαράμετροι στα Συστήματα RAG	39
3	Κριτική Ανασκόπηση Μεθοδολογιών Επαυξημένης Παραγωγής μέσω Ανάκτησης	40
3.1	Εισαγωγή	40
3.2	Αρχικό RAG (Naive RAG)	40
3.2.1	Αρχιτεκτονική και Αρχές Λειτουργίας	40
3.2.2	Περιορισμοί και Προκλήσεις	41
3.3	Προηγμένο RAG (Advanced RAG)	42
3.3.1	Βελτιστοποίησεις στη Διαδικασία Ανάκτησης	42
3.3.2	Μηχανισμοί Επεξεργασίας και Βελτίωσης Ανακτηθέντων Εγγράφων	45
3.3.3	Προσαρμοστικές Στρατηγικές Ανάκτησης	47
3.4	Αρθρωτό RAG (Modular RAG)	48
3.4.1	Ιεραρχική Ανάκτηση και το Παράδειγμα RAPTOR	48
3.4.2	Πολυβηματική Συλλογιστική και Λογική Ανάκτηση	49
3.4.3	Μετρικές Ανάκτησης Πληροφοριών	50
3.4.4	Μετρικές Αξιολόγησης Παραγωγής Κειμένου	52
3.4.5	Εξειδικευμένες Μετρικές για Συστήματα RAG	53
3.5	Το Πλαίσιο AutoRAG: Αυτοματοποιημένη Βελτιστοποίηση Συστημάτων RAG	55
3.5.1	Αρχιτεκτονική και Φιλοσοφία Σχεδιασμού	55
3.5.2	AutoRAG-HP: Online Βελτιστοποίηση Υπερπαραμέτρων	56
3.5.3	Πειραματικά Αποτελέσματα και Αξιολόγηση Απόδοσης	57
3.6	Ενσωμάτωση σε Παραγωγικά Συστήματα	59
3.6.1	Προκλήσεις ένταξης του RAG στην παραγωγή	59
3.6.2	Η Συνεισφορά του AutoRAG σε περιβάλλοντα παραγωγής	60
3.7	Ερευνητικό Κενό και Συνεισφορά της Παρούσας Εργασίας	61
3.7.1	Μεθοδολογικά και Πρακτικά Εμπόδια στην Εφαρμογή Υφιστάμενων Προσεγγίσεων	61
4	Υλοποίηση	63
4.1	Αρχιτεκτονικές Επιλογές	63
4.1.1	Αγωγός Εισαγωγής Δεδομένων	64
4.1.2	Αγωγός Ανάκτησης Πληροφορίας	67
4.1.3	Σύστημα Ευφυούς Πράκτορα	69
4.2	Τεχνολογικές Επιλογές	70
4.2.1	Διανυσματική Βάση Δεδομένων	70
4.2.2	Πλαίσιο Ανάπτυξης	70
4.2.3	Υποδομή Γλωσσικού Μοντέλου	70
5	Πειράματα και Αποτελέσματα	71
5.1	Σύνολο Δεδομένων και Τομέας Εφαρμογής	71
5.1.1	Προέλευση και Χαρακτηριστικά Δεδομένων	71
5.1.2	Επιλογή συνόλου δεδομένων	72

5.2	Διεξαγωγή Πειραμάτων	74
5.3	Πείραμα 1: Επιλογή Μεθόδου Ανάκτησης	75
5.3.1	Μεθοδολογία Αξιολόγησης	76
5.3.2	Στρατηγικές Ανάκτησης	76
5.3.3	Επιλογή Μετρικών	77
5.3.4	Αποτελέσματα	77
5.4	Πείραμα 2: Βελτιστοποίηση Γραμμικής Ανάκτησης	82
5.4.1	Μεθοδολογία Βελτιστοποίησης	82
5.4.2	Αποτελέσματα	85
5.4.3	Συμπεράσματα Πειράματος 2	89
5.5	Πείραμα 3: Αξιολόγηση Παραγωγής Απαντήσεων	90
5.5.1	Μεθοδολογία	91
5.5.2	Αποτελέσματα	92
5.5.3	Πρόταση Βελτίωσης: Μηχανισμός Αυτοδιορθούμενης Παραγωγής	94
6	Συμπεράσματα και μελλοντική εργασία	100
6.1	Σύνθεση Πειραματικών Ευρημάτων	100
6.2	Περιορισμοί της Παρούσας Έρευνας	101
6.3	Μελλοντικές Κατευθύνσεις Έρευνας	102

Κατάλογος Σχημάτων

2.1	Τυπικές συναρτήσεις ενεργοποίησης και οι αντίστοιχες παράγωγοί τους	9
2.2	Εποπτική αναπαράσταση της softmax ως χαρτογράφηση από \mathbb{R}^K στο simplex πιθανοτήτων. Πηγή: [1].	10
2.3	Αρχιτεκτονική Πολυστρωματικού Perceptron με δύο κρυφά επίπεδα (feedforward).	12
2.4	Απαιτήσεις μνήμης του GPT-J σε συνάρτηση με το sequence length [2].	18
2.5	Παράδειγμα One-hot κωδικοποίησης	22
2.6	Σύγκριση One-hot κωδικοποίησης με σημασιολογική αναπαράσταση	23
2.7	Σχηματικό διάγραμμα πολυκεφαλικής προσοχής (προσαρμογή από [3])	26
2.8	Η αρχιτεκτονική του μετασχηματιστή όπως παρουσιάστηκε στη δημοσίευση "Attention Is All You Need" [4]	28
2.9	Απεικόνιση της διαδικασίας πρόβλεψης σε ένα Μεγάλο Γλωσσικό Μοντέλο (LLM). Οι ενσωματώσεις εισόδου (token + positional encodings) περνούν από μια στοίβα L στρωμάτων μετασχηματιστή με αυτοπροσοχή, κανονικοποίηση και Feed-Forward δίκτυα, ώστε να παραχθεί η κρυφή αναπαράσταση h_t για τη θέση t . Η αναπαράσταση αυτή ενσωματώνει πληροφορία από όλο το προηγούμενο πλαίσιο και τροφοδοτεί τον γραμμικό μετασχηματισμό και τη συνάρτηση softmax, με αποτέλεσμα την κατανομή πιθανοτήτων στο λεξιλόγιο. Τέλος, μέσω δειγματοληψίας, επιλέγεται το επόμενο token εξόδου.	34
2.10	Αρχιτεκτονική ροή ενός συστήματος Επαυξημένης Παραγωγής μέσω Ανάκτησης (RAG). Η διαδικασία περιλαμβάνει δύο διακριτά στάδια: (i) τη δεικτοδότηση, όπου τα έγγραφα τεμαχίζονται, μετατρέπονται σε διανύσματα μέσω μοντέλου ενσωματώσεων και αποθηκεύονται σε διανυσματική βάση δεδομένων, και (ii) την online ανάκτηση και παραγωγή, όπου το ερώτημα του χρήστη ενσωματώνεται, αναζητά σχετικές εγγραφές στη βάση και, σε συνδυασμό με τις οδηγίες (prompt), τροφοδοτεί το LLM για την παραγωγή της τελικής απάντησης.	38
3.1	Τα γλωσσικά μοντέλα εμφανίζουν μεροληφία θέσης (U-shaped performance): βρίσκουν ευκολότερα τις πληροφορίες όταν βρίσκονται στην αρχή ή στο τέλος του κειμένου, ενώ η απόδοσή τους πέφτει σημαντικά όταν η απάντηση βρίσκεται στη μέση [5].	42
4.1	Αρχιτεκτονική του προτεινόμενου συστήματος RAG	64
4.2	Τυπικό αρχείο YAML για ορισμό του αγωγού εισαγωγής δεδομένων	66

4.3 Υβριδικό σύστημα ανάκτησης από διανυσματική βάση δεδομένων	68
4.4 Τυπικό αρχείο YAML για ορισμό του αγωγού ανάκτησης	69
 5.1 Το σύνολο δεδομένων SOSum περιληπτικά	73
5.2 Κατανομή μήκους κειμένου ερωτήσεων και απαντήσεων στο σύνολο δεδομένων	73
5.3 Συνολική συνοπτική απόδοση του συστήματος	77
5.4 Σύγκριση των διαμορφώσεων ως προς την μετρική: ακρίβεια	78
5.5 Σύγκριση των διαμορφώσεων ως προς την μετρική: ανάκληση	79
5.6 Σύγκριση των διαμορφώσεων ως προς την μετρική: F1	79
5.7 Συμβιβασμός μεταξύ ανάκλησης και ακρίβειας	80
5.8 Σύγκριση των διαμορφώσεων ως προς την μετρική: NDCG	80
5.9 Σύγκριση των διαμορφώσεων ως προς την μετρική: καθυστέρηση	81
5.10 Χάρτης σύνθετης βαθμολογίας: η βέλτιστη διαμόρφωση ($\alpha = 0.8$, $k = 20$) αντιστοιχεί σε υβριδική ανάκτηση με ισχυρή έμφαση στο πυκνό σήμα. Η απόδοση αυξάνεται μονοτονικά με το α , ενώ το k έχει αμελητέα επίδραση.	85
5.11 Επιμέρους μετρικές: Success@3 και Precision@3 βελτιώνονται δραματικά με το α , η Recall@10 παραμένει σταθερή (0.51), και η καθυστέρηση έμφανίζει μικρές διακυμάνσεις. Το k δεν επηρεάζει συστηματικά τις μετρικές.	86
5.12 Τρισδιάστατη επιφάνεια σύνθετης βαθμολογίας στον χώρο υπερπαραμέτρων. Η επιφάνεια έμφανίζει μονοτονική αύξηση με το α και επιπεδότητα κατά μήκος του k . Το βέλτιστο σημείο ($\alpha = 0.8$, $k = 20$) επισημαίνεται με αστέρι.	87
5.13 Ανάλυση ευαισθησίας υπερπαραμέτρων. Αριστερά: η απόδοση αυξάνεται μονοτονικά με το α , με όλες τις καμπύλες (διαφορετικά k) να συγκλίνουν. Δεξιά: η απόδοση παραμένει σταθερή για όλα τα k , με διαστρωμάτωση βάσει του α . Το βέλτιστο $\alpha = 0.8$ επισημαίνεται με διακεκομένη γραμμή.	88
5.14 Επικύρωση απόδοσης στο test set: σύγκριση κύριων μετρικών και Success@ k για διαφορετικές τιμές k . Το train-test gap είναι ελάχιστο, επιβεβαιώνοντας την ικανότητα γενίκευσης του βέλτιστου μοντέλου.	89
5.15 Αρχιτεκτονική ευφυούς πράκτορα για την αξιολόγηση του υποσυστήματος γεννεσιογρίας. Το σύστημα επεξεργάζεται ερωτήματα μέσω αλληλουχίας κόμβων: εκκίνηση, ανάκτηση, γεννεσιογρία, και καταγραφή αποτελεσμάτων.	91
5.16 Κατανομές συχνότητας των βαθμολογιών αξιολόγησης για τις τέσσερις μετρικές. Οι κόκκινες διακεκομένες γραμμές υποδεικνύουν τον μέσο όρο ενώ οι πράσινες διακεκομένες γραμμές τη διάμεσο κάθε κατανομής. Παρατηρείται έντονη ασυμμετρία στη μετρική της Συνάφειας (skewness προς υψηλές τιμές) και ευρύτερη διασπορά στην Πιστότητα.	93

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

5.17 Προτεινόμενη αρχιτεκτονική ευφυούς πράκτορα με μηχανισμό Self-RAG. Προστίθενται οι κόμβοι query_analyzer για την αποσύνθεση του ερωτήματος και self_rag_generator για την επαναληπτική βελτίωση της απάντησης μέσω επαλήθευσης.	95
5.18 Εσωτερικός βρόχος του κόμβου self_rag_generator. Το σύστημα παράγει μια αρχική απάντηση, την επαληθεύει έναντι του πλαισίου, και αναθεωρεί επαναληπτικά την απάντηση μέχρι να επιτευχθεί ικανοποιητική πιστότητα ή να εξαντληθεί ο μέγιστος αριθμός επαναλήψεων.	96
5.19 Κατανομένες συχνότητας των βαθμολογιών αξιολόγησης για τις τέσσερις μετρικές χρησιμοποιώντας την αρχιτεκτονική self RAG.	98

Κατάλογος πινάκων

2.1 Συγχριτική ανάλυση επιπέδων διακριτοποίησης	17
5.1 Προδιαγραφές του υπολογιστικού περιβάλλοντος που χρησιμοποιήθηκε για τη διεξαγωγή των πειραμάτων.	74
5.2 Ρυθμίσεις λογισμικού και παραμέτρων του πειραματικού περιβάλλοντος.	75
5.3 Αποτελέσματα αξιολόγησης retrievers: μέσος όρος \pm τυπική απόκλιση για Precision@5, Recall@5, F1@5, MAP, MRR, NDCG@5 και Latency.	77
5.4 Περιγραφικά στατιστικά των μετρικών αξιολόγησης γεννεσιοναργίας (N=500). Οι τιμές παρουσιάζονται ως μέσος όρος \pm τυπική απόκλιση. Το μοντέλο GPT-5 χρησιμοποιήθηκε ως αξιολογητής μέσω της μεθοδολογίας LLM-as-a-Judge.	92
5.5 Περιγραφικά στατιστικά των μετρικών αξιολόγησης γεννεσιοναργίας χρησιμοποιώντας την αρχιτεκτονική self RAG (N=500). Οι τιμές παρουσιάζονται ως μέσος όρος \pm τυπική απόκλιση. Το μοντέλο GPT-5 χρησιμοποιήθηκε ως αξιολογητής μέσω της μεθοδολογίας LLM-as-a-Judge.	97

Κατάλογος Αλγορίθμων

2.1	Εκπαίδευση με mini-batch στοχαστική επικλινή κάθοδο και backpropagation	14
2.2	Byte Pair Encoding (BPE)	20
3.1	Συγχώνευση Κατατάξεων με Reciprocal Rank Fusion (RRF)	45

Ακρωνύμια Εγγράφου

Παρακάτω παρατίθενται ορισμένα από τα πιο συχνά χρησιμοποιούμενα ακρωνύμια της παρούσας διπλωματικής εργασίας:

AI	→ Artificial Intelligence
ML	→ Machine Learning
NN	→ Neural Network
MLP	→ Multilayer Perceptron
GPT	→ Generative Pre-trained Transformer
ReLU	→ Rectified Linear Unit
LLM	→ Large Language Model
RAG	→ Retrieval Augmented Generation
FFN	→ Feed Forward Network
OOV	→ Out of Vocabulary
RRF	→ Reciprocal Rank Fusion
BPE	→ Byte Pair Encoding
CBOW	→ Continuous Bag-of-Words
RNN	→ Recurrent Neural Networks
LSTM	→ Long Short-Term Memory
GRU	→ Gated Recurrent Unit (GRU)
GPU	→ Graphics Processing Unit
YAML/YML	→ YAML ain't markup language

1

Εισαγωγή

Η ανάπτυξη της αρχιτεκτονικής του μετασχηματιστή (Transformer) το 2017 [4] σηματοδότησε την απαρχή μιας νέας εποχής στην επεξεργασία φυσικής γλώσσας, θέτοντας τα θεμέλια για την εμφάνιση των Μεγάλων Γλωσσικών Μοντέλων. Από τα πρώτα μοντέλα της οικογένειας GPT με εκατομμύρια παραμέτρους [6], η εξέλιξη οδήγησε σε αρχιτεκτονικές δισεκατομμυρίων παραμέτρων όπως το GPT-3 [7] και το GPT-4 [8], τα οποία επέδειξαν πρωτόγνωρες ικανότητες στη σύνθεση κειμένου, την επίλυση προβλημάτων και τη γλωσσική κατανόηση.

Η μετάβαση από τα εξειδικευμένα μοντέλα επεξεργασίας γλώσσας σε γενικής χρήσης συστήματα τεχνητής νοημοσύνης έχει επαναπροσδιορίσει τις δυνατότητες αυτοματοποίησης γνωστικών εργασιών. Τα σύγχρονα LLMs έχουν ενσωματωθεί σε εφαρμογές που εκτείνονται από την αυτοματοποιημένη συγγραφή κώδικα και την επιστημονική έρευνα, μέχρι την εκπαιδευτική υποστήριξη και την ιατρική διάγνωση [9]. Η ικανότητά τους να προσαρμόζονται σε νέες εργασίες με ελάχιστα παραδείγματα (few-shot learning) και να εκτελούν σύνθετους συλλογισμούς έχει ανοίξει νέους ορίζοντες στην αλληλεπίδραση ανθρώπου-μηχανής.

Παρά τις εντυπωσιακές τους δυνατότητες, η μετάβαση από πειραματικές εφαρμογές σε παραγωγικά συστήματα αποκαλύπτει θεμελιώδεις περιορισμούς που απειλούν την αξιοπιστία και την ασφάλεια των εφαρμογών. Η τεχνική της Επαυξημένης Παραγωγής μέσω Ανάκτησης έχει προταθεί ως λύση για την άμβλυνση αυτών των περιορισμών, επιτρέποντας στα μοντέλα να αξιοποιούν δυναμικά εξωτερικές πηγές γνώσης κατά την παραγωγή απαντήσεων [10]. Συστήματα όπως το RETRO της DeepMind [11] και το Atlas της Meta [12] έχουν αποδείξει ότι η ενσωμάτωση μηχανισμών ανάκτησης μπορεί να βελτιώσει δραματικά την ακρίβεια και την αξιοπιστία των παραγόμενων απαντήσεων.

Η εξέλιξη των συστημάτων επαυξημένης παραγωγής μέσω ανάκτησης (RAG) έχει ακολουθήσει μια πορεία συνεχούς βελτίωσης, από απλούς μηχανισμούς ανάκτησης βασισμένους σε λέξεις-κλειδιά μέχρι σύνθετες αρχιτεκτονικές που συνδυάζουν πολλαπλές στρατηγικές αναζήτησης και προηγμένους αλγορίθμους κατάταξης.

1.1. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

Παράλληλα η πρόσφατη ανάπτυξη πλαισίων όπως το LangChain έχει απλοποιήσει την υλοποίηση τέτοιων συστημάτων.

Ωστόσο, η κατασκευή αποτελεσματικών συστημάτων RAG απαιτεί την επίλυση πολύπλοκων τεχνικών προκλήσεων που εκτείνονται πέρα από την απλή σύνδεση ενός μοντέλου με μια βάση δεδομένων. Υπάρχει μια αλυσίδα αλληλοεξαρτούμενων μηχανισμών, που χρειάζονται βελτιστοποίηση προκειμένου να διασφαλιστεί η ποιότητα της ανάκτησης και η αξιοπιστία των παραγόμενων απαντήσεων.

Η παρούσα διπλωματική εργασία στοχεύει στη συστηματική διερεύνηση και επίλυση αυτών των προκλήσεων μέσω του σχεδιασμού και της υλοποίησης ενός ολοκληρωμένου συστήματος RAG. Η έρευνα εστιάζει στην ανάπτυξη λύσεων, βελτιστοποιώντας κάθε στάδιο της επεξεργασίας από την εισαγωγή των ακατέργαστων δεδομένων μέχρι την παραγωγή της τελικής απάντησης. Για να αποκτήσει η έρευνα πρακτική αξία, η βελτιστοποίηση εφαρμόζεται σε ένα σύνολο δεδομένων τεχνικών ερωτήσεων από το StackOverflow.

1.1 ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

Τα σύγχρονα μεγάλα γλωσσικά μοντέλα έχουν επιδείξει εξαιρετικές ικανότητες στην κατανόηση και παραγωγή φυσικής γλώσσας, καθιστώντας τα πολύτιμα εργαλεία για ένα ευρύ φάσμα εφαρμογών. Ωστόσο, η αξιοπιστία των πληροφοριών που παράγουν παραπλένει ένα κρίσιμο και άλυτο πρόβλημα που περιορίζει σημαντικά την εφαρμογή τους σε κρίσιμα συστήματα λήψης αποφάσεων [13].

Το φαινόμενο της παραισθητικότητας (hallucination) αποτελεί την πλέον σημαντική πρόκληση στη χρήση των γλωσσικών μοντέλων. Τα μοντέλα αυτά παράγουν συχνά απαντήσεις που φαίνονται συντακτικά και σημασιολογικά ορθές, αλλά περιέχουν ανακριβείς ή εντελώς κατασκευασμένες πληροφορίες. Σύμφωνα με μελέτες της OpenAI, τα ποσοστά παραισθήσεων στα πρώιμα μοντέλα όπως το GPT-3 έφταναν στο 42%, ενώ για μικρότερα μοντέλα τα πράγματα γίνονται ακόμα χειρότερα[14]. Έκτοτε έχουν γίνει αξιοσημείωτες προσπάθειες για την ελαχιστοποίηση των παραισθήσεων, καθώς η επιστημονική κοινότητα της τεχνητής νοημοσύνης έχει επικεντρώσει σε αυτόν τον στόχο τις προσπάθειές της. Παρ' όλα αυτά, ακόμα και σήμερα τα φαινόμενα παραισθήσεων δεν έχουν εξαλειφθεί, με αποτέλεσμα η χρήση σε εξειδικευμένους τομείς όπως η νομική ή η ιατρική να κρίνεται ακατάλληλη, τουλάχιστον χωρίς περαιτέρω διερεύνηση και επικύρωση των παραγόμενων πληροφοριών.

Επιπλέον, τα γλωσσικά μοντέλα λειτουργούν με στατική γνώση που προέρχεται αποκλειστικά από τα δεδομένα εκπαίδευσής τους. Αυτό σημαίνει ότι δεν μπορούν να έχουν πρόσβαση σε πληροφορίες που δημοσιεύθηκαν μετά την ημερομηνία διακοπής της εκπαίδευσής τους (knowledge cutoff), ούτε σε εξειδικευμένες ή ιδιωτικές πηγές δεδομένων που δεν συμπεριλήφθηκαν στο σώμα εκπαίδευσης. Η επανεκπαίδευση ενός μεγάλου μοντέλου όπως το GPT-4 απαιτεί χιλιάδες GPU-hours και εκτιμάται ότι κοστίζει δεκάδες έως εκατοντάδες εκατομμύρια δολάρια [15], καθιστώντας την πρακτικά ανέφικτη για τακτική ενημέρωση.

Παράλληλα, οι παραδοσιακές μέθοδοι ανάκτησης πληροφοριών που βασίζονται σε λεξικολογική αντιστοίχιση (όπως ο αλγόριθμος BM25) αδυνατούν να κατανοήσουν τη σημασιολογική σχέση μεταξύ διαφορετικών όρων [16]. Αντίστοιχα, τα συ-

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

στήματα που βασίζονται αποκλειστικά σε σημασιολογική αναζήτηση μέσω πυκνών διανυσματικών αναπαραστάσεων μπορεί να χάσουν σημαντικές λεπτομέρειες όταν η ακριβής αντιστοίχιση όρων είναι κρίσιμη [17]. Πρόσφατες μελέτες έχουν δείξει ότι η υβριδική προσέγγιση που συνδυάζει και τις δύο μεθόδους επιτυγχάνει καλύτερα αποτελέσματα από κάθε μέθοδο μεμονωμένα [18].

Η πολυπλοκότητα αυξάνεται περαιτέρω λόγω της ετερογένειας των πηγών δεδομένων που πρέπει να διαχειριστεί ένα σύγχρονο σύστημα πληροφοριών. Επιστημονικές δημοσιεύσεις, τεχνική τεκμηρίωση, φόρουμ συζητήσεων και βάσεις δεδομένων έχουν διαφορετική δομή, ύφος και απαιτήσεις επεξεργασίας [19]. Ένα ενιαίο σύστημα που μπορεί να διαχειριστεί αποτελεσματικά όλες αυτές τις πηγές απαιτεί σύνθετους μηχανισμούς προσαρμογής και επεξεργασίας.

Τέλος, η έλλειψη ολοκληρωμένων πλαισίων που συνδυάζουν αποτελεσματικά την ανάκτηση πληροφοριών με την ικανότητα παραγωγής κειμένου των γλωσσικών μοντέλων δημιουργεί ένα σημαντικό κενό στην πρακτική εφαρμογή αυτών των τεχνολογιών. Οι περισσότερες υπάρχουσες υλοποιήσεις παραμένουν είτε πειραματικές και δύσκολες στην ανάπτυξη είτε εμπορικές και κλειστού κώδικα, γεγονός που δυσχεραίνει τη δημιουργία προσαρμοσμένων λύσεων.

1.2 ΣΥΝΕΙΣΦΟΡΑ ΤΗΣ ΕΡΓΑΣΙΑΣ

Η παρούσα διπλωματική εργασία συνεισφέρει στην επίλυση των προαναφερθέντων προκλήσεων μέσω της ανάπτυξης ενός επεκτάσιμου πλαισίου πειραματισμού για συστήματα επαυξημένης παραγωγής μέσω ανάκτησης, συνοδευόμενου από συστηματική πειραματική αξιολόγηση στο πεδίο των τεχνικών ερωτημάτων μηχανικής λογισμικού.

Το σύστημα που αναπτύχθηκε αποτελεί ένα προσαρμόσιμο και επεκτάσιμο περιβάλλον πειραματισμού που επιτρέπει τη δημιουργία και αξιολόγηση διαφορετικών αγωγών επεξεργασίας (pipelines) μέσω δηλωτικών αρχείων YAML και τερματικής διεπαφής (CLI). Η αρχιτεκτονική βασίζεται σε αφηρημένες διεπαφές που διευκολύνουν την προσαρμογή σε διαφορετικά σύνολα δεδομένων και πεδία εφαρμογής, ενώ παρέχει έτοιμη υποστήριξη για πολλαπλές μεθόδους ανάκτησης (BM25, SPLADE, Dense, Hybrid) από διαφορετικούς παρόχους (OpenAI, VoyageAI, HuggingFace, Google). Η προσαρμοστικότητα του συστήματος επιτρέπει στον ερευνητή να πειραματιστεί με διαφορετικές σχεδιαστικές επιλογές χωρίς σημαντικές τροποποιήσεις στον πυρήνα του κώδικα, διασφαλίζοντας παράλληλα την αναπαραγωγισμότητα των πειραμάτων.

Πέρα από το πλαίσιο υλοποίησης, η εργασία παρέχει τεκμηριωμένη μεθοδολογία πολυεπίπεδης και ιεραρχικά δομημένης βελτιστοποίησης, από τη μέθοδο ανάκτησης έως την αξιολόγηση από άκρη σε άκρη (end-to-end). Μέσω συστηματικής συγκριτικής αξιολόγησης πέντε μεθόδων ανάκτησης, εξαντλητικής αναζήτησης βέλτιστων υπερπαραμέτρων και αξιολόγησης μηχανισμού αυτοδιορθούμενης παραγωγής απαντήσεων σε τρεις διαστάσεις ποιότητας, η εργασία διερευνά τις σχέσεις μεταξύ των χαρακτηριστικών του συνόλου δεδομένων και της αποτελεσματικότητας των μεθόδων ανάκτησης και παραγωγής. Σε αντίθεση με προσεγγίσεις που επιδιώκουν την ολική βελτιστοποίηση συστημάτων επαυξημένης παραγωγής μέσω ανάκτησης,

η παρούσα εργασία εστιάζει στη στοχευμένη βελτιστοποίηση επιμέρους στοιχείων όπου κρίνεται απαραίτητη, τεκμηριώνοντας κάθε σχεδιαστική επιλογή με βάση τις πρακτικές και τους συμβιβασμούς που ορίζει η βιβλιογραφία. Ο συνδυασμός επεκτάσιμου πλαισίου πειραματισμού και συστηματικής πειραματικής μεθοδολογίας στοχεύει να αποτελέσει χρήσιμο οδηγό για την ένταξη τέτοιων συστημάτων σε παραγωγικά περιβάλλοντα, παρέχοντας τόσο την τεχνική υποδομή όσο και την εμπειρική τεκμηρίωση που απαιτείται για τη λήψη τεκμηριωμένων σχεδιαστικών αποφάσεων.

1.3 ΔΙΑΡΘΡΩΣΗ ΤΗΣ ΑΝΑΦΟΡΑΣ

Η παρούσα διπλωματική εργασία διαρθρώνεται σε έξι κεφάλαια, τα οποία ακολουθούν συστηματική οργάνωση από τη θεωρητική θεμελίωση έως την πειραματική επαλήθευση των προτεινόμενων μεθόδων:

- **Κεφάλαιο 1 - Εισαγωγή:** Το πρώτο κεφάλαιο εισάγει τον αναγνώστη στην ερευνητική περιοχή των συστημάτων επαυξημένης παραγωγής με ανάκτηση, παρουσιάζοντας το επιστημονικό κίνητρο που υπαγορεύει την ανάγκη ανάπτυξης βελτιωμένων αρχιτεκτονικών. Αναλύεται το πρόβλημα της αναξιοπιστίας των σύγχρονων γλωσσικών μοντέλων και διατυπώνονται οι ερευνητικοί στόχοι που θέτει η εργασία για την αντιμετώπιση των εγγενών περιορισμών αυτών των συστημάτων.
- **Κεφάλαιο 2 - Θεωρητικό Υπόβαθρο:** Το δεύτερο κεφάλαιο συγκροτεί το θεωρητικό και τεχνολογικό υπόβαθρο που απαιτείται για την κατανόηση της προτεινόμενης προσέγγισης. Εξετάζονται οι θεμελιώδεις αρχές λειτουργίας των αρχιτεκτονικών μετασχηματιστών, η εξέλιξη των μεγάλων γλωσσικών μοντέλων, οι μέθοδοι διανυσματικής αναπαράστασης σημασιολογικού περιεχομένου, και οι αλγόριθμοι ανάκτησης πληροφοριών που συνθέτουν τα σύγχρονα συστήματα RAG.
- **Κεφάλαιο 3 - Κριτική Ανασκόπηση Μεθοδολογιών Επαυξημένης Παραγωγής μέσω Ανάκτησης:** Το τρίτο κεφάλαιο παρουσιάζει κριτική ανασκόπησης της υπάρχουσας βιβλιογραφίας, αναλύοντας τις μεθοδολογικές προσεγγίσεις που έχουν προταθεί για την επίλυση του προβλήματος της επαυξημένης παραγωγής μέσω ανάκτησης. Εξετάζονται συστηματικά οι αρχιτεκτονικές που έχουν αναπτυχθεί από ερευνητικά κέντρα και οργανισμούς, αξιολογούνται τα πλεονεκτήματα, οι περιορισμοί τους και εντοπίζονται τα κενά που η παρούσα εργασία επιδιώκει να καλύψει.
- **Κεφάλαιο 4 - Γλοποίηση:** Το τέταρτο κεφάλαιο αναπτύσσει τη μεθοδολογία που ακολουθήθηκε για τον σχεδιασμό και την υλοποίηση του συστήματος. Περιγράφονται αναλυτικά οι αρχιτεκτονικές επιλογές, οι αλγόριθμοι που εφαρμόστηκαν και η διαδικασία ενορχήστρωσης των επιμέρους δομοστοιχείων σε ένα ενιαίο λειτουργικό σύνολο.

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

- **Κεφάλαιο 5 - Πειράματα και Αποτελέσματα:** Το πέμπτο κεφάλαιο παρουσιάζει το πειραματικό πρωτόκολλο και τα αποτελέσματα της αξιολόγησης. Αναλύονται συστηματικά τα δεδομένα που προέκυψαν από τη σύγχριση διαφορετικών στρατηγικών ανάκτησης, εξετάζεται βελτιστοποίηση του αλγορίθμου RRF και παρουσιάζονται ποιοτικές μετρικές παραγωγής απαντήσεων.
- **Κεφάλαιο 6 - Συμπεράσματα και Μελλοντική Εργασία:** Το έκτο και τελευταίο κεφάλαιο συνθέτει τα συμπεράσματα που απορρέουν από την ερευνητική διαδικασία, αξιολογεί την επίτευξη των αρχικών στόχων, αναγνωρίζει τους περιορισμούς της προτεινόμενης προσέγγισης, και υποδεικνύει κατευθύνσεις για μελλοντική έρευνα που θα μπορούσε να επεκτείνει και να βελτιώσει τα αποτελέσματα της παρούσας εργασίας.

2

Θεωρητικό Υπόβαθρο

2.1 ΘΕΜΕΛΙΩΔΕΙΣ ΕΝΝΟΙΕΣ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

Η κατανόηση των νευρωνικών δικτύων αποτελεί απαραίτητη προϋπόθεση για τη μελέτη των σύγχρονων μοντέλων επεξεργασίας φυσικής γλώσσας. Οι θεμελιώδεις έννοιες που παρουσιάζονται στο κεφάλαιο αυτό, από τον μεμονωμένο τεχνητό νευρώνα έως τη δομή και τη διαδικασία εκπαίδευσης των πολυστρωματικών δικτύων, διαμορφώνουν το εννοιολογικό πλαίσιο που επιτρέπει την κατανόηση προηγμένων αρχιτεκτονικών, όπως οι μηχανισμοί προσοχής (attention mechanisms) και τα μοντέλα μετασχηματιστών που αποτελούν τον πυρήνα των σύγχρονων συστημάτων γλωσσικής μοντελοποίησης. Η ανάλυση ξεκινά από τη βασική υπολογιστική μονάδα, τον τεχνητό νευρώνα, συνεχίζει με τις συναρτήσεις ενεργοποίησης και τη δομή του Πολυστρωματικού Perceptron (MLP), και ολοκληρώνεται με τη διαδικασία μάθησης και τους περιορισμούς που ανακύπτουν κατά την εκπαίδευση βαθιών δικτύων.

2.1.1 Τεχνητός Νευρώνας και Βασικές Αρχές

Ο τεχνητός νευρώνας αποτελεί την στοιχειώδη υπολογιστική μονάδα των νευρωνικών δικτύων. Λαμβάνει εισόδους x_1, \dots, x_n , καθεμία σταθμισμένη με συντελεστή βάρους w_i . Το γραμμικό άθροισμα των σταθμισμένων εισόδων και ο όρος πόλωσης b (bias term) γράφονται συνοπτικά ως

$$z = \sum_{i=1}^n w_i x_i + b = w^\top x + b, \quad (2.1)$$

όπου w και x παριστάνουν τα διανύσματα των βαρών και των εισόδων αντίστοιχα. Η τελική έξοδος του νευρώνα προκύπτει εφαρμόζοντας μια συνάρτηση ενεργοποίησης $\sigma(\cdot)$ στο γραμμικό συνδυασμό, δηλαδή

$$h = \sigma(z). \quad (2.2)$$

Η διασύνδεση πολλαπλών τέτοιων μονάδων σε διαδοχικά επίπεδα σχηματίζει ένα νευρωνικό δίκτυο. Το επίπεδο εισόδου δέχεται τα αρχικά δεδομένα, τα ενδιάμεσα κρυφά επίπεδα (hidden layers) υλοποιούν διαδοχικούς μετασχηματισμούς που εξάγουν ιεραρχικές αναπαραστάσεις των δεδομένων, και το επίπεδο εξόδου παράγει την τελική πρόβλεψη.

2.1.2 Συναρτήσεις Ενεργοποίησης και ο Ρόλος τους

Η εισαγωγή μη γραμμικότητας μέσω των συναρτήσεων ενεργοποίησης αποτελεί κρίσιμο στοιχείο για την εκφραστική ισχύ των νευρωνικών δικτύων. Εάν δεν χρησιμοποιηθούν μη γραμμικές συναρτήσεις, η σύνθεση γραμμικών μετασχηματισμών παραμένει γραμμική. Συγκεκριμένα, για δύο διαδοχικά επίπεδα με γραμμικές μόνο πράξεις, έχουμε

$$h^{(2)} = W^{(2)}(W^{(1)}x + b^{(1)}) + b^{(2)} = W_{\text{eq}}x + b_{\text{eq}}, \quad (2.3)$$

όπου $W_{\text{eq}} = W^{(2)}W^{(1)}$ και $b_{\text{eq}} = W^{(2)}b^{(1)} + b^{(2)}$. Συνεπώς, ελλείψει μη γραμμικής συναρτησης ενεργοποίησης, το δίκτυο παραμένει αδύνατον να αναπαραστήσει μη γραμμικά διαχωρίσιμα προβλήματα.

Οι πλέον διαδεδομένες συναρτήσεις ενεργοποίησης περιλαμβάνουν τη σιγμοειδή (*sigmoid*), η οποία ορίζεται ως

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \quad \sigma'(x) = \sigma(x)(1 - \sigma(x)), \quad (2.4)$$

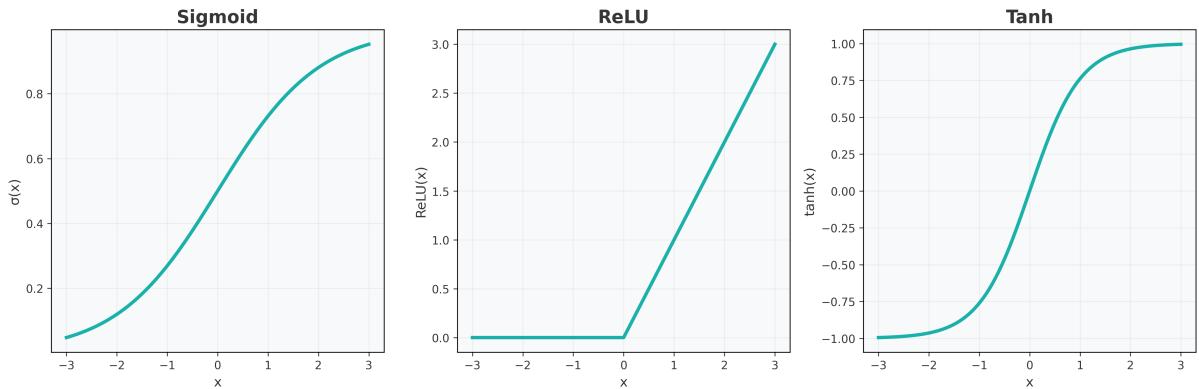
και χαρτογραφεί την είσοδο στο διάστημα $(0, 1)$, καθιστώντας την κατάλληλη για προβλήματα δυαδικής ταξινόμησης. Η υπερβολική εφαπτομένη (hyperbolic tangent) δίνεται από τη σχέση

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad \tanh'(x) = 1 - \tanh^2(x), \quad (2.5)$$

και χαρτογραφεί την είσοδο στο διάστημα $(-1, 1)$, παρέχοντας συμμετρική μη γραμμικότητα γύρω από το μηδέν. Μια σύγχρονη και ευρέως χρησιμοποιούμενη επιλογή είναι η *Rectified Linear Unit (ReLU)*,

$$\text{ReLU}(x) = \max(0, x), \quad (2.6)$$

η οποία είναι υπολογιστικά λιτή, εφαρμόζεται με απλή σύγκριση και κατωφλίωση, και μετριάζει σημαντικά το πρόβλημα της εξαφάνισης των κλίσεων για θετικές τιμές εισόδου, καθώς η παράγωγός της είναι σταθερή και ίση με τη μονάδα στην θετική περιοχή. Παρά τα πλεονεκτήματά της, η ReLU μπορεί να οδηγήσει στο φαινόμενο των «νεκρών νευρώνων» (dying ReLU), όπου νευρώνες παύουν να ενεργοποιούνται για οποιαδήποτε είσοδο λόγω της μηδενικής κλίσης στην αρνητική περιοχή, ζήτημα που έχει οδηγήσει στην ανάπτυξη παραλλαγών όπως η Leaky ReLU και η Parametric ReLU.



Σχήμα 2.1: Τυπικές συναρτήσεις ενεργοποίησης και οι αντίστοιχες παράγωγοι τους.

Η συνάρτηση softmax

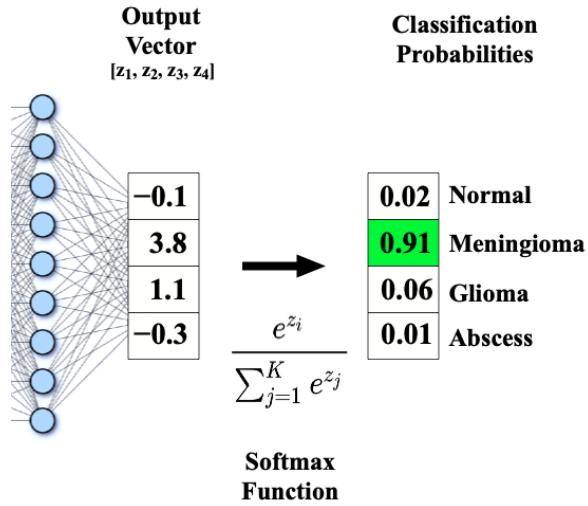
Για προβλήματα πολυταξινόμησης, όπου απαιτείται η ανάθεση κάθε εισόδου σε μία από πολλές διακριτές κατηγορίες, χρησιμοποιείται η συνάρτηση softmax. Δοθέντος ενός διανύσματος ετικέτας (logits) $z \in \mathbb{R}^K$, η softmax ορίζει μια κατανομή πιθανότητας επί των K κλάσεων ως

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \quad i = 1, \dots, K, \quad (2.7)$$

εξασφαλίζοντας ότι $\sum_{i=1}^K \text{softmax}(z_i) = 1$ και $\text{softmax}(z_i) \in (0, 1)$ για κάθε i . Η softmax συνδυάζεται φυσικά με τη συνάρτηση απώλειας διασταυρωμένης εντροπίας (cross-entropy loss) για την εκπαίδευση μοντέλων πολυταξινόμησης. Για έναν στόχο y κωδικοποιημένο ως one-hot διάνυσμα (δηλαδή $y_c = 1$ για τη σωστή κλάση c και $y_i = 0$ για $i \neq c$), η απώλεια διασταυρωμένης εντροπίας γράφεται

$$L_{\text{CE}} = - \sum_{i=1}^K y_i \log(\text{softmax}(z_i)) = - \log(\text{softmax}(z_c)), \quad (2.8)$$

όπου η τελευταία ισότητα προκύπτει από το γεγονός ότι μόνο ο όρος της σωστής κλάσης συνεισφέρει στο άθροισμα. Ένα σημαντικό πλεονέκτημα αυτής της σύζευξης είναι ότι η παράγωγος της απώλειας ως προς τις ετικέτες απλοποιείται σημαντικά: εφαρμόζοντας τον κανόνα της αλυσίδας και αξιοποιώντας τις μαθηματικές ιδιότητες της σύνθεσης, προκύπτει $\partial L / \partial z = \hat{y} - y$, όπου $\hat{y} = \text{softmax}(z)$. Αυτή η κομψή μορφή της κλίσης καθιστά την εκπαίδευση ιδιαίτερα αποδοτική και αριθμητικά σταθερή.



Σχήμα 2.2: Εποπτική αναπαράσταση της softmax ως χαρτογράφηση από \mathbb{R}^K στο simplex πιθανοτήτων. Πηγή: [1].

2.4.3 Το Πολυστρωματικό Perceptron (MLP)

Αφού έγινε αντιληπτή η λειτουργία των μεμονωμένων νευρώνων και των συναρτήσεων ενεργοποίησης, δύναται να εξεταστεί η οργανωμένη διασύνδεσή τους σε επίπεδα που δημιουργεί δίκτυα με σημαντικά αυξημένη υπολογιστική και αναπαραστασιακή ισχύ. Το Πολυστρωματικό Perceptron αποτελεί τη θεμελιώδη αρχιτεκτονική που επιτρέπει την εκμάθηση πολύπλοκων μη γραμμικών απεικονίσεων.

Ορισμός και ορολογία. Ένα Πολυστρωματικό Perceptron (MultiLayer Perceptron, MLP) είναι ένα προωθητικής ροής (feedforward) νευρωνικό δίκτυο: η πληροφορία ρέει μονοδρομικά από την είσοδο προς την έξοδο, χωρίς αναδράσεις ή εσωτερική μνήμη κατάστασης. Αυτή η μονοκατευθυντική ροή διαφοροποιεί τα MLP από τα αναδρομικά δίκτυα (Recurrent Neural Networks), τα οποία διαθέτουν κυκλικές συνδέσεις και εσωτερική μνήμη. Το MLP αποτελείται από διαδοχικά πλήρως συνδεδεμένα (fully connected ή dense) επίπεδα που υλοποιούν εναλλάξ γραμμικούς και μη γραμμικούς μετασχηματισμούς.

Μαθηματικό μοντέλο ανά επίπεδο. Για είσοδο $h^{(0)} \equiv x \in \mathbb{R}^{n_0}$ και επίπεδα $l = 1, \dots, L$, κάθε επίπεδο εφαρμόζει τη μετασχηματιστική σχέση

$$z^{(l)} = W^{(l)} h^{(l-1)} + b^{(l)}, \quad h^{(l)} = \sigma^{(l)}(z^{(l)}), \quad (2.9)$$

όπου $W^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ είναι ο πίνακας βαρών, $b^{(l)} \in \mathbb{R}^{n_l}$ το διάνυσμα πόλωσης, και $\sigma^{(l)} : \mathbb{R} \rightarrow \mathbb{R}$ μια συνάρτηση ενεργοποίησης που εφαρμόζεται κατά στοιχείο (elementwise), δηλαδή $\sigma^{(l)}(z) = [\sigma^{(l)}(z_1), \dots, \sigma^{(l)}(z_{n_l})]^\top$. Η σ εισάγει την αναγκαία μη γραμμικότητα που εμποδίζει τη σύνθεση των στρώσεων να καταρρεύσει σε έναν ενιαίο γραμμικό μετασχηματισμό, όπως αναλύθηκε προηγουμένως.

2.1. ΘΕΜΕΛΙΩΔΕΙΣ ΕΝΝΟΙΕΣ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

Στα κρυφά επίπεδα, συνηθέστερες επιλογές για σ αποτελούν οι ReLU και οι σύγχρονες παραλλαγές της όπως η GELU (Gaussian Error Linear Unit), λόγω της απλότητας της παραγώγου τους και της καλής αριθμητικής συμπεριφοράς κατά την εκπαίδευση. Στο επίπεδο εξόδου, η επιλογή της συνάρτησης ενεργοποίησης εξαρτάται από το πρόβλημα: για πολυταξινόμηση χρησιμοποιείται η softmax, για δυαδική ταξινόμηση η sigmoid, ενώ για προβλήματα παλινδρόμησης συνήθως χρησιμοποιείται η γραμμική ταυτότητα (δηλαδή καμία μη γραμμικότητα).

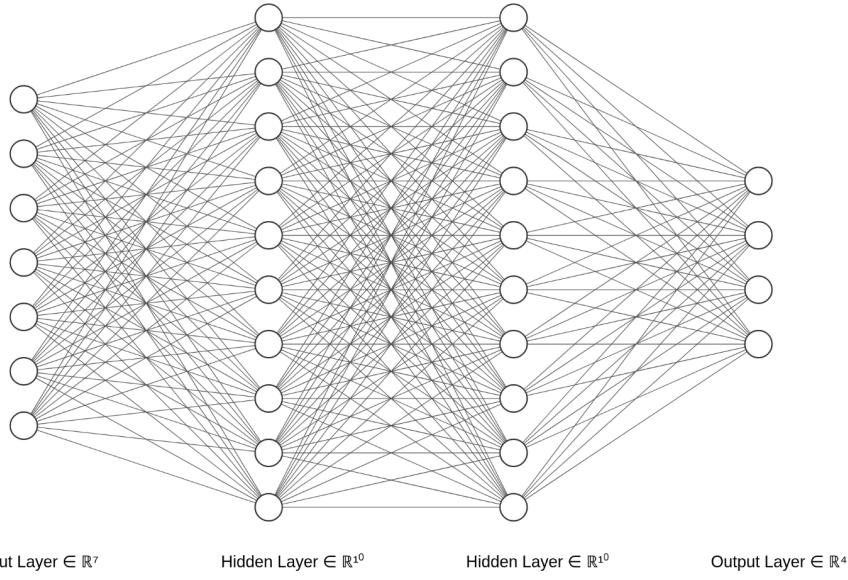
Διανυσματική μορφή και πολυπλοκότητα. Συμβολικά, ένα MLP ορίζει μια παραμετρική συνάρτηση

$$f_\theta : \mathbb{R}^{n_0} \rightarrow \mathbb{R}^{n_L}, \quad \theta = \{(W^{(l)}, b^{(l)})\}_{l=1}^L,$$

μέσω της επαναληπτικής εφαρμογής της σχέσης (2.9). Η συνολική πολυπλοκότητα παραμέτρων είναι $\sum_{l=1}^L (n_l n_{l-1} + n_l)$, όπου n_l είναι ο αριθμός νευρώνων στο επίπεδο l και n_{l-1} στο προηγούμενο επίπεδο. Ο πρώτος όρος $n_l n_{l-1}$ προέρχεται από τα βάρη μεταξύ των επιπέδων, ενώ ο δεύτερος όρος n_l από τους αντίστοιχους bias όρους. Αυτή η έκφραση αποτυπώνει τον βασικό συμβιβασμό μεταξύ πλάτους και βάθους της αρχιτεκτονικής: η αύξηση του πλάτους (δηλαδή του αριθμού νευρώνων ανά επίπεδο) αυξάνει άμεσα την εκφραστικότητα αλλά με τετραγωνικό κόστος παραμέτρων, ενώ το πρόσθετο βάθος (δηλαδή περισσότερα επίπεδα) επιτρέπει ιεραρχικούς, συνθετικούς μετασχηματισμούς με γραμμική αύξηση παραμέτρων ανά επίπεδο. Αυτή η παρατήρηση έχει σημαντικές πρακτικές συνέπειες: βαθύτερα δίκτυα τείνουν να επιτυγχάνουν υψηλότερη απόδοση με λιγότερες παραμέτρους σε σύγκριση με ρηχά αλλά πλατιά δίκτυα, υπό την προϋπόθεση ότι η εκπαίδευσή τους παραμένει σταθερή.

Εκφραστικότητα και Θεώρημα Καθολικής Προσέγγισης. Το θεώρημα καθολικής προσέγγισης (universal approximation theorem) [20] εγγυάται ότι ένα MLP με τουλάχιστον ένα κρυφό επίπεδο επαρκούς πλάτους και μη πολυωνυμική, μη γραμμική ενεργοποίηση μπορεί να προσεγγίσει αυθαίρετα καλά κάθε συνεχή συνάρτηση σε συμπαγή υποσύνολα του \mathbb{R}^n . Το θεώρημα αυτό αποτελεί θεμελιώδες αποτέλεσμα για την κατανόηση της θεωρητικής δυναμικής των νευρωνικών δικτύων, καθώς διασφαλίζει την εκφραστικότητα του μοντέλου, δηλαδή τι μπορεί να αναπαρασταθεί διοθέντων επαρκών πόρων.

Ωστόσο, είναι κρίσιμο να κατανοηθεί ότι το θεώρημα δεν παρέχει εγγυήσεις σχετικά με την εκμάθηση: δεν εγγυάται ότι οι αλγόριθμοι βελτιστοποίησης θα εντοπίσουν στην πράξη τις κατάλληλες παραμέτρους, ούτε ότι ο απαιτούμενος αριθμός νευρώνων θα είναι πρακτικά εύλογος. Στα βαθιά δίκτυα με ReLU ενεργοποίηση, το μοντέλο ορίζει κομματικά γραμμικές συναρτήσεις: το βάθος αυξάνει εκθετικά τον αριθμό των διακριτών περιοχών γραμμικότητας, βελτιώνοντας την ικανότητα μοντελοποίησης πολύπλοκων αποφασιστικών ορίων με σχετική οικονομία παραμέτρων. Αυτή η ιδιότητα εξηγεί εν μέρει γιατί τα σύγχρονα βαθιά δίκτυα έχουν επιτύχει τόσο εντυπωσιακά αποτελέσματα σε πρακτικές εφαρμογές.



Σχήμα 2.3: Αρχιτεκτονική Πολυστρωματικού Perceptron με δύο αρυφά επίπεδα (feedforward).

2.1.4 Η Διαδικασία Μάθησης στα Νευρωνικά Δίκτυα

Η εκπαίδευση ενός νευρωνικού δικτύου αποτελεί επαναληπτική διαδικασία προσαρμογής των παραμέτρων με στόχο την ελαχιστοποίηση μιας συνάρτησης απώλειας που ποσοτικοποιεί την απόκλιση μεταξύ των προβλέψεων του μοντέλου και των επιθυμητών εξόδων. Κάθε κύκλος εκπαίδευσης περιλαμβάνει τρία βασικά στάδια: την πρόδρομη διάδοση (forward propagation) για την παραγωγή πρόβλεψης, τον υπολογισμό της απώλειας μέσω σύγκρισης με την πραγματική έξοδο, την οπισθοδιάδοση (backpropagation) για τον αποδοτικό υπολογισμό των κλίσεων, και την ενημέρωση των παραμέτρων σύμφωνα με έναν κανόνα βελτιστοποίησης. Η διαδικασία επαναλαμβάνεται επί πολλαπλών κύκλων (epochs) διέλευσης του συνόλου εκπαίδευσης.

Το πρόβλημα της εκπαίδευσης ως βελτιστοποίηση

Δοθέντος συνόλου δεδομένων εκπαίδευσης $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, όπου x_i αναπαριστά την είσοδο και y_i την αντίστοιχη επιθυμητή έξοδο, επιδιώκεται η εύρεση παραμέτρων θ που ελαχιστοποιούν την εμπειρική συνάρτηση κινδύνου (empirical risk function) [24]:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N L(f_\theta(x_i), y_i) + \lambda R(\theta), \quad (2.10)$$

όπου L είναι η συνάρτηση απώλειας που μετρά την απόκλιση μεταξύ πρόβλεψης $f_\theta(x_i)$ και πραγματικής εξόδου y_i , ενώ $R(\theta)$ αποτελεί όρο κανονικοποίησης (regularization term) με συντελεστή $\lambda \geq 0$. Ο όρος κανονικοποίησης, συνήθως της μορφής ℓ_2 (weight decay), δηλαδή $R(\theta) = \|\theta\|_2^2$, περιορίζει την πολυπλοκότητα του

μοντέλου και λειτουργεί ως μηχανισμός προστασίας έναντι της υπερπροσαρμογής (overfitting), συμπληρώνοντας άλλες πρακτικές όπως η έγκαιρη διακοπή της εκπαίδευσης (early stopping).

Επικλινής κάθοδος και στοχαστικές παραλλαγές

Η επικλινής κάθοδος (Gradient Descent, GD) αποτελεί τον κλασικό αλγόριθμο βελτιστοποίησης που ενημερώνει τις παραμέτρους προς την αρνητική κατεύθυνση της κλίσης της συνάρτησης κινδύνου:

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} J(\theta_t), \quad (2.11)$$

όπου $\eta > 0$ είναι ο ρυθμός μάθησης (learning rate), υπερπαράμετρος που ελέγχει το μέγεθος του βήματος ενημέρωσης. Στα σύγχρονα συστήματα βαθιάς μάθησης, η πλήρης επικλινής κάθοδος υποκαθίσταται από τη στοχαστική επικλινή κάθοδο (Stochastic Gradient Descent, SGD), κατά την οποία η κλίση εκτιμάται από υποσύνολα δειγμάτων σταθερού μεγέθους, τα οποία ονομάζονται mini-batches. Για ένα mini-batch \mathcal{B} μεγέθους $B \ll N$, η εκτίμηση της κλίσης γράφεται

$$\nabla_{\theta} J_{\mathcal{B}} = \frac{1}{B} \sum_{i \in \mathcal{B}} \nabla_{\theta} L(f_{\theta}(x_i), y_i),$$

πράγμα που μειώνει δραστικά το υπολογιστικό κόστος ανά ενημέρωση και εισάγει ελεγχόμενη στοχαστικότητα στη διαδικασία βελτιστοποίησης. Η στοχαστικότητα αυτή, παρά την πρώτη εντύπωση, μπορεί να είναι ευεργετική: βοηθά το μοντέλο να διαφύγει από ρηγά τοπικά ελάχιστα και να εξερευνήσει ευρύτερες περιοχές του χώρου παραμέτρων. Για επαρκώς μικρά βήματα η , το ανάπτυγμα Taylor εγγυάται τοπική μείωση του κόστους [22].

Backpropagation: αποδοτικός υπολογισμός κλίσεων

Ο αλγόριθμος backpropagation [23] αποτελεί τον πυρήνα της εκπαίδευσης των νευρωνικών δικτύων. Εφαρμόζει συστηματικά τον κανόνα της αλυσίδας ώστε να υπολογιστούν οι παράγωγοι της συνάρτησης απώλειας ως προς κάθε παράμετρο του δικτύου, με υπολογιστική πολυπλοκότητα που είναι συγχρίσιμη με αυτή της πρόδρομης διάδοσης. Η βασική ιδέα συνίσταται στον αναδρομικό υπολογισμό των ευαισθησιών κάθε επιπέδου, ξεκινώντας από το επίπεδο εξόδου και κινούμενοι προς την είσοδο.

Θέτοντας $z^{(l)} = W^{(l)} h^{(l-1)} + b^{(l)}$ για το προ-ενεργοποιήσης διάνυσμα και $h^{(l)} = \sigma^{(l)}(z^{(l)})$ για το διάνυσμα ενεργοποιήσεων του επιπέδου l , ορίζουμε τις ευαισθησίες $\delta^{(l)}$ ως την παράγωγο της απώλειας ως προς το $z^{(l)}$. Οι ευαισθησίες υπολογίζονται αναδρομικά ξεκινώντας από το τελικό επίπεδο:

$$\delta^{(L)} = \nabla_{h^{(L)}} L \odot (\sigma^{(L)})'(z^{(L)}), \quad (2.12)$$

$$\delta^{(l)} = (W^{(l+1)})^{\top} \delta^{(l+1)} \odot (\sigma^{(l)})'(z^{(l)}), \quad l = L-1, \dots, 1, \quad (2.13)$$

όπου το σύμβολο \odot παριστά το γινόμενο κατά στοιχείο (elementwise ή Hadamard product) των διανυσμάτων. Αφού υπολογιστούν οι ευαισθησίες, οι κλίσεις ως προς

ΚΕΦΑΛΑΙΟ 2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

τις παραμέτρους κάθε επιπέδου προκύπτουν άμεσα ως:

$$\frac{\partial J}{\partial W^{(l)}} = \delta^{(l)} (h^{(l-1)})^\top, \quad \frac{\partial J}{\partial b^{(l)}} = \delta^{(l)}. \quad (2.14)$$

Η κομψότητα του αλγορίθμου έγκειται στο ότι, με μία πρόδρομη και μία ανάδρομη διάδοση, υπολογίζονται όλες οι απαιτούμενες παράγωγοι, καθιστώντας την εκπαίδευση βαθιών δικτύων πρακτικά εφικτή.

Αλγόριθμος 2.1 Εκπαίδευση με mini-batch στοχαστική επικλινή κάθοδο και backpropagation

- 1: **Input:** Σύνολο εκπαίδευσης \mathcal{D} , ρυθμός μάθησης η , μέγεθος mini-batch B , αριθμός εποχών E
 - 2: **Output:** Βελτιστοποιημένες παράμετροι θ^*
 - 3: Αρχικοποίησε θ # π.χ. Xavier/Glorot
 - 4: **for** epoch = 1 **To** E **do**
 - 5: **for all** mini-batches \mathcal{B} μεγέθους B από το \mathcal{D} **do**
 - 6: Υπολόγισε προβλέψεις $\hat{y}_i = f_\theta(x_i)$ για $(x_i, y_i) \in \mathcal{B}$ # Forward pass
 - 7: Υπολόγισε $J_{\mathcal{B}}$ και κλίσεις $g = \nabla_{\theta} J_{\mathcal{B}}$ μέσω backpropagation
 - 8: Ενημέρωσε παραμέτρους: $\theta \leftarrow \theta - \eta g$
 - 9: **end for**
 - 10: Προαιρετικά: Αξιολόγηση κριτηρίου διακοπής σε σύνολο επικύρωσης
 - 11: **end for**
 - 12: **return** θ
-

Σύγχρονες μέθοδοι βελτιστοποίησης και τακτικές εκπαίδευσης

Πέραν της βασικής επικλινούς καθόδου, στα σύγχρονα συστήματα βαθιάς μάθησης εφαρμόζονται προσαρμοστικές μέθοδοι βελτιστοποίησης που εκτιμούν και αξιοποιούν ροπές των κλίσεων. Ο αλγόριθμος Adam (Adaptive Moment Estimation) [24] αποτελεί ίσως τον πλέον διαδεδομένο αλγόριθμο στην πράξη, καθώς συνδυάζει όρο ορμής και προσαρμοστική κλιμάκωση του ρυθμού μάθησης ανά παράμετρο:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \quad (2.15)$$

$$\theta_{t+1} = \theta_t - \eta \frac{m_t / (1 - \beta_1^t)}{\sqrt{v_t / (1 - \beta_2^t)} + \epsilon}, \quad (2.16)$$

όπου $g_t = \nabla_{\theta} J(\theta_t)$ είναι η κλίση στο βήμα t , m_t και v_t είναι εκτιμητές της πρώτης και δεύτερης ροπής αντίστοιχα, $\beta_1, \beta_2 \in (0, 1)$ είναι παράμετροι εκθετικής απόσβεσης (τυπικά $\beta_1 = 0.9$ και $\beta_2 = 0.999$), και ϵ είναι μικρή σταθερά για αριθμητική σταθερότητα. Ο Adam προσαρμόζει το ρυθμό μάθησης για κάθε παράμετρο ξεχωριστά, επιτρέποντας ταχύτερη σύγκλιση και καλύτερη διαχείριση παραμέτρων με διαφορετικές κλίμακες μεγεθών.

Η επιλογή της αρχικοποίησης των βαρών αποτελεί κρίσιμο παράγοντα για τη σταθερότητα και την επιτυχία της εκπαίδευσης. Η αρχικοποίηση Glorot/Xavier [25] επιλέγει τα αρχικά βάρη από κατανομή με διασπορά που εξαρτάται από τις διαστάσεις εισόδου και εξόδου του επιπέδου, με στόχο τη διατήρηση της διασποράς

2.1. ΘΕΜΕΛΙΩΔΕΙΣ ΕΝΝΟΙΕΣ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

των ενεργοποιήσεων και των κλίσεων σε παρόμοια επίπεδα καθ' ύψος του δικτύου. Αυτή η ισορροπία είναι ουσιώδης για την αποφυγή προβλημάτων εξαφάνισης ή έκρηξης των κλίσεων στα πρώτα στάδια της εκπαίδευσης.

Ως τεχνικές ταχτοποίησης και προστασίας από την υπερπροσαρμογή περιλαμβάνονται η ποινικοποίηση ℓ_2 των βαρών (weight decay), η τεχνική dropout που απενεργοποιεί τυχαία κλάσματα νευρώνων κατά την εκπαίδευση, και η έγκαιρη διακοπή (early stopping) που τερματίζει την εκπαίδευση όταν η απόδοση στο σύνολο επικύρωσης αρχίσει να υποβαθμίζεται. Αυτές οι μέθοδοι συμβάλλουν συλλογικά στη βελτίωση της ικανότητας γενίκευσης του μοντέλου σε δεδομένα που δεν έχει συναντήσει κατά την εκπαίδευση.

2.1.5 Περιορισμοί των Πολυστρωματικών Perceptron

Παρά την εκφραστική τους ισχύ και την ευρεία χρήση τους, τα Πολυστρωματικά Perceptron εμφανίζουν ορισμένους εγγενείς περιορισμούς που καθιστούν δύσκολη την εφαρμογή τους σε συγκεκριμένες κατηγορίες προβλημάτων.

Πρώτον, η εκπαίδευση πολύ βαθιών MLPs δυσχεραίνεται από τα φαινόμενα των εξαφανιζόμενων και των εκρηκτικών κλίσεων (vanishing and exploding gradients) [26]. Κατά την ανάδρομη διάδοση, η κλίση προς τα πρώτα επίπεδα του δικτύου προκύπτει από το γινόμενο των κλίσεων όλων των ενδιάμεσων επιπέδων. Όταν οι τοπικές κλίσεις είναι κατά μέσο όρο μικρότερες της μονάδας, το γινόμενο τους τείνει να συρρικνώνεται εκθετικά καθώς το βάθος αυξάνεται, με αποτέλεσμα οι παράμετροι των πρώτων επιπέδων να εκπαιδεύονται εξαιρετικά αργά ή καθόλου. Αντίθετα, όταν οι τοπικές κλίσεις υπερβαίνουν την μονάδα, το γινόμενο διογκώνεται εκθετικά, προκαλώντας αριθμητική αστάθεια και απόκλιση της βελτιστοποίησης. Αυτά τα προβλήματα περιορίζουν στην πράξη το βάθος των MLPs και ενθάρρυνται την ανάπτυξη εξελιγμένων αρχιτεκτονικών με παράκαμψη συνδέσεων (residual connections), όπως τα ResNets, που μετριάζουν σημαντικά αυτά τα ζητήματα.

Δεύτερον, τα MLPs προϋποθέτουν είσοδο σταθερού μεγέθους. Κάθε επίπεδο έχει προκαθορισμένο αριθμό εισόδων και εξόδων, γεγονός που καθιστά προβληματική την επεξεργασία ακολουθιών μεταβλητού μήκους, όπως προτάσεις φυσικής γλώσσας ή χρονοσειρές.

Τρίτον, λόγω της πλήρους διασύνδεσης των επιπέδων, τα MLPs δεν είναι ικανά να μοντελοποιήσουν αποδοτικά εξαρτήσεις μεγάλης εμβέλειας σε ακολουθιακά ή χωρικά δεδομένα. Για παράδειγμα, σε μια πρόταση, η σημασία μιας λέξης μπορεί να εξαρτάται από λέξεις που βρίσκονται πολλές θέσεις μακριά. Ενώ ένα αρκετά βαθύ MLP θεωρητικά θα μπορούσε να συλλάβει τέτοιες εξαρτήσεις, στην πράξη η εκμάθησή τους απαιτεί τεράστιο αριθμό παραμέτρων και δεδομένων εκπαίδευσης.

Αυτοί οι περιορισμοί οδήγησαν στην ανάπτυξη εξειδικευμένων αρχιτεκτονικών που αντιμετωπίζουν συγκεκριμένα προβλήματα: τα Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks, RNNs) και οι παραλλαγές τους όπως τα LSTM και GRU εισάγουν εσωτερική κατάσταση για την επεξεργασία ακολουθιών, τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks, CNNs) αξιοποιούν τοπικές δομές σε χωρικά δεδομένα, και τελικά οι μηχανισμοί προσοχής (attention mechanisms) και τα μοντέλα μετασχηματιστών επιτρέπουν την άμεση μοντελοποίηση εξαρτήσεων αυθαίρετης εμβέλειας σε ακολουθίες, αποτελώντας το θεμέλιο των σύγχρονων μο-

ντέλων επεξεργασίας φυσικής γλώσσας που θα εξεταστούν στα επόμενα κεφάλαια της παρούσας εργασίας.

2.2 ΔΙΑΚΡΙΤΟΠΟΙΗΣΗ ΣΕ ΛΕΚΤΙΚΕΣ ΜΟΝΑΔΕΣ

Η διαδικασία της διακριτοποίησης σε λεκτικές μονάδες (tokenization) αποτελεί βασικό στοιχείο στη σύγχρονη επεξεργασία φυσικής γλώσσας και εν γένει στα συστήματα τεχνητής νοημοσύνης που χειρίζονται γλωσσικά δεδομένα. Η θεμελιώδης αυτή διαδικασία εισάγει μια μετατροπή του φυσικού κειμένου σε επεξεργάσιμες φημιακές οντότητες, επιτρέποντας στους υπολογιστικούς αλγορίθμους τον χειρισμό και την ανάλυση της γλώσσας μέσω μαθηματικών και στατιστικών μεθόδων.

2.2.1 Λεκτική μονάδα

Ως λεκτική μονάδα (token) ορίζεται μια διακριτή μονάδα κειμένου με αυθαίρετα καθορισμένο μέγεθος. Αυθαίρετα γιατί, ανάλογα με τη στρατηγική διακριτοποίησης, μπορεί να θεωρηθεί ως λεκτική μονάδα μια λέξη, μία υπολέξη, ένας χαρακτήρας ή ακόμη και ένα byte, ανάλογα με την επιθυμητή κοκκομετρία (granularity). Παρακάτω θα παρουσιαστεί η ανάγκη ύπαρξης διαφορετικών αλγορίθμων παραγωγής λεκτικών μονάδων καθώς και η ανάγκη εξισορρόπησης της κοκκομετρίας και του μεγέθους του λεξιλογίου με την πολυπλοκότητα της επεξεργασίας.

2.2.2 Διακριτοποίηση σε λεκτικές μονάδες

Ο όρος διακριτοποίηση σε λεκτικές μονάδες (tokenization) αναφέρεται στην αποσύνθεση του κειμένου σε λεκτικές μονάδες όπου κάθε μία αντιστοιχεί σε μία διακριτή λέξη. Η προσέγγιση αυτή είναι εύκολα κατανοητή, καθώς ευθυγραμμίζεται με τη φυσική ανθρώπινη γλωσσική αντίληψη. Ωστόσο, αυτή η μέθοδος δημιουργεί εκτεταμένα λεξιλόγια και αποτυγχάνει σε εκτός λεξιλογίου (out-of-vocabulary, OOV) περιπτώσεις, περιορίζοντας τη γενίκευση.

Διακριτοποίηση σε Επίπεδο Υπο-Λέξεων

Η αποσύνθεση των λέξεων σε μικρότερες μονάδες αποτέλεσε σημείο καμπής στην Επεξεργασία Φυσικής Γλώσσας. Μεθοδολογίες όπως η κωδικοποίηση ζεύγους byte (Byte Pair Encoding, BPE) [27], το WordPiece [28] και το SentencePiece [29] μειώνουν το λεξιλόγιο, διατηρώντας ωστόσο σημασιολογικά σημαντικές ρίζες. Για παράδειγμα, η φράση “hazel eyes” μπορεί να αναλυθεί ως [haz] [el] [eyes]. Η διακριτοποίηση σε επίπεδο υπολέξεων θεωρείται σήμερα ο χρυσός κανόνας σε LLMs, καθώς μειώνει τα σφάλματα εκτός λεξιλογίου και ενισχύει τη γενίκευση.

Διακριτοποίηση σε Επίπεδο Χαρακτήρων

Η ανάλυση σε χαρακτήρες [30] εγγυάται πλήρη κάλυψη του λεξιλογίου με το μικρότερο δυνατό λεξιλόγιο, καθώς κάθε χαρακτήρας γίνεται λεκτική μονάδα. Η

2.2. ΔΙΑΚΡΙΤΟΠΟΙΗΣΗ ΣΕ ΛΕΚΤΙΚΕΣ ΜΟΝΑΔΕΣ

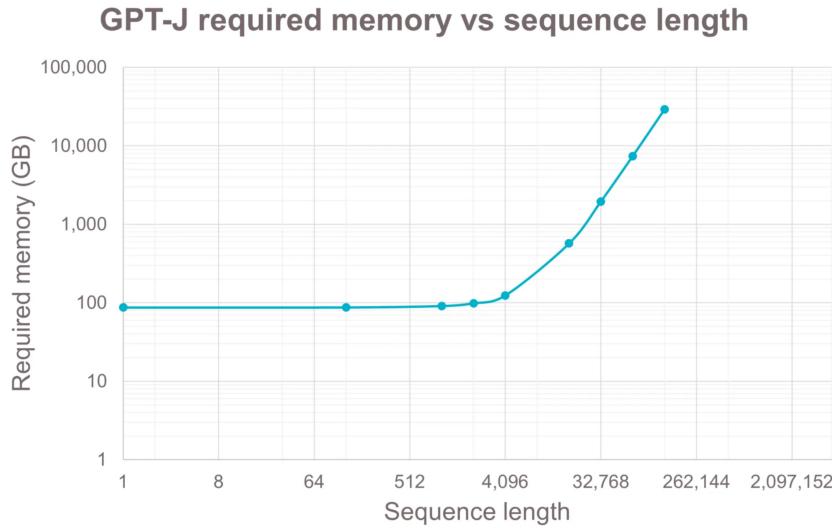
φράση “hazel eyes” παράγει την ακολουθία [h] [a] [z] [e] [l] [] [e] [y] [e] [s]. Ωστόσο, αυτή η προσέγγιση αυξάνει δραματικά το μήκος των ακολουθιών, επιβαρύνοντας την εκπαίδευση και την απόδοση κατά το στάδιο απόφασης (inference).

Διακριτοποίηση σε Επίπεδο Bytes

Η διακριτοποίηση σε επίπεδο byte προτάθηκε ευρέως με την ανάπτυξη των GPT-2 μοντέλων, όπου χρησιμοποιήθηκε κωδικοποίηση ζεύγους byte στο επίπεδο των byte (byte-level BPE)[31]. Κάθε byte (0–255) αποτελεί token, π.χ. η φράση “hazel eyes” σε UTF-8 δίνει [104] [97] [122] [101] [108] [32] [101] [121] [101] [115]. Η μέθοδος εξασφαλίζει καθολική συμβατότητα με οποιοδήποτε κείμενο, αλλά οδηγεί σε πολύ μεγαλύτερες ακολουθίες χωρίς εγγυημένη σημασιολογική συνοχή.

Επίπεδο	Χαρακτηριστικά	Παράδειγμα
Λέξεων	<p>Κάθε λέξη αντιμετωπίζεται ως token.</p> <p><i>Πλεονεκτήματα:</i> Εύκολη κατανόηση, ταχύτερη εκτέλεση.</p> <p><i>Μειονεκτήματα:</i> Μεγάλο λεξιλόγιο, άγνωστες λέξεις.</p>	[hazel] [eyes]
Υπο-λέξεων	<p>Αποσύνθεση λέξεων σε υπομονάδες (BPE, WordPiece).</p> <p><i>Πλεονεκτήματα:</i> Μειωμένο λεξιλόγιο, καλύτερος χειρισμός άγνωστων λέξεων.</p> <p><i>Μειονεκτήματα:</i> Πιο πολύπλοκη αναπαράσταση.</p>	[haz] [el] [eyes]
Χαρακτήρων	<p>Κάθε χαρακτήρας αντιμετωπίζεται ως token.</p> <p><i>Πλεονεκτήματα:</i> Ελάχιστο λεξιλόγιο, απόλυτη κάλυψη.</p> <p><i>Μειονεκτήματα:</i> Πολύ μακρές ακολουθίες, αυξημένη πολυπλοκότητα.</p>	[h] [a] [z] [e] [l] [] [e] [y] [e] [s]
Bytes	<p>Κάθε byte αντιμετωπίζεται ως token.</p> <p><i>Πλεονεκτήματα:</i> Σταθερό λεξιλόγιο 256 tokens, πλήρης συμβατότητα.</p> <p><i>Μειονεκτήματα:</i> Μέγιστο μήκος ακολουθιών, απώλεια σημασιολογίας.</p>	[104] [97] [122] [101] [108] [32] [101] [121] [101] [115]

Πίνακας 2.1: Συγχριτική ανάλυση επιπέδων διακριτοποίησης



Σχήμα 2.4: Απαιτήσεις μνήμης του GPT-J σε συνάρτηση με το sequence length [2].

Η επιλογή της μεθόδου tokenization παίζει καθοριστικό ρόλο στη συνολική απόδοση ενός γλωσσικού μοντέλου. Από τη μία πλευρά, η χρήση tokenization επιπέδου λέξης μειώνει το μήκος των ακολουθιών, αλλά οδηγεί σε υπερβολικά μεγάλο λεξιλόγιο με υψηλά ποσοστά σφαλμάτων εκτός λεξιλογίου. Από την άλλη, η χρήση χαρακτήρων n bytes εξαλείφει τα προβλήματα εκτός λεξιλογίου, αλλά δημιουργεί ακολουθίες πολύ μεγαλύτερου μήκους. Το μήκος αυτό έχει άμεσο αντίκτυπο στο υπολογιστικό κόστος, καθώς οι μηχανισμοί αυτοπροσοχής κλιμακώνονται με πολυπλοκότητα $O(n^2)$ ως προς το μήκος της ακολουθίας n [4].

Το γεγονός αυτό αποτυπώνεται χαρακτηριστικά στο Σχήμα 2.4, όπου φαίνεται ότι οι απαιτήσεις μνήμης για το GPT-J αυξάνονται εκθετικά με το μήκος ακολουθίας [2]. Επομένως, η επιλογή του tokenizer αποτελεί κρίσιμη εξισορρόπηση μεταξύ μεγέθους λεξιλογίου και μήκους ακολουθίας, επηρεάζοντας τόσο την ικανότητα γενίκευσης όσο και την αποδοτικότητα των μοντέλων.

2.2.3 Λεξιλόγιο

Αφού έχει γίνει κατανοητή η διαδικασία τεμαχισμού του κειμένου, απαιτείται η κατανόηση της έννοιας του λεξιλογίου. Το λεξιλόγιο V είναι ένα ορισμένο σύνολο από λεκτικές μονάδες. Χρησιμοποιείται ως αναφορά για την μετατροπή του κειμένου εισόδου σε λεκτικές μονάδες. Το μέγεθος $|V|$ εξαρτάται από δύο παράγοντες όπως προαναφέρθηκε.

- Την κοκκομετρία (granularity): λεπτή κοκκομετρία (επιπέδου byte) σημαίνει μικρότερο λεξιλόγιο ενώ μεγαλύτερη κοκκομετρία (επιπέδου λέξης) έχει ως αποτέλεσμα μεγαλύτερο λεξιλόγιο.
- Αρχικό corpus κειμένων: Για παράδειγμα αν για την εκπαίδευση έγινε χρήση πολυγλωσσικών κειμένων τότε θα χρειαστούν πολύ περισσότερα λεκτικές μονάδες για να αναπαραστήσουν το τελικό λεξιλόγιο συγχριτικά με μονογλωσσικά corpuses.

2.2.4 Είδη αναλυτών λεκτικών μονάδων

Ο αναλυτής λεκτικών μονάδων, για ευκολία εφεξής θα χρησιμοποιείται ο όρος tokenizer, χωρίζεται σε δύο βασικές κατηγορίες με βάση την μεθοδολογία κατασκευής του[32]. Πιο συγκεκριμένα:

Tokenizers βασισμένοι σε κανόνες

Το πρώτο είδος είναι ο tokenizer βασισμένος σε κανόνες (rule-based tokenizer). Αυτό το είδος tokenizer είναι ιδιαίτερα απλό, καθώς συστηματικά τεμαχίζει το κείμενο σύμφωνα με τους κανόνες που έχουν οριστεί. Ο tokenizer επιπέδου λέξης και ο tokenizer επιπέδου χαρακτήρα είναι δύο πολύ κλασσικά παραδείγματα αυτής της οικογένειας. Η συγκεκριμένη μέθοδος δεν χρειάζεται εκπαίδευση και αυτό αποτελεί σημαντικό πλεονέκτημα.

Ωστόσο, εμφανίζει έναν θεμελιώδη περιορισμό: οι συγκεκριμένοι tokenizers βασίζονται μόνο στα μοτίβα που υπάρχουν στα δεδομένα και δεν λαμβάνουν καθόλου υπόψη τους τη σημασιολογία της κάθε λέξης. Ως εκ τούτου δεν μπορούν να γενικεύσουν σωστά και τα σφάλματα εκτός λεξιλογίου (out-of-vocabulary, OOV) είναι συχνά.

Στατιστικοί Tokenizers

Για τον παραπάνω λόγο τα σύγχρονα γλωσσικά μοντέλα καταφεύγουν σε στατιστικούς tokenizers (learned tokenizers). Το δεύτερο είδος εμφανίζει αρκετά βελτιωμένα αποτελέσματα και πετυχαίνει δύο πολύ σημαντικά πράγματα:

- Εκμεταλλεύεται την σημασιολογία των λέξεων, λέξεις όπως **κολύμπι, κολυμβητήριο, κολυμβητής** έχουν όλες κοινή ρίζα. Ένας αποτελεσματικός tokenizer πρέπει να το λαμβάνει αυτό υπόψη τεμαχίζοντας το κείμενο με τέτοιο τρόπο ώστε να κρατηθεί η κοινή ρίζα, για παράδειγμα η υπολέξη [κολυμ].
- Μειώνει σημαντικά τα σφάλματα εκτός λεξιλογίου. Στο προηγούμενο παράδειγμα αν στο αρχικό corpus κειμένου υπήρχαν οι λέξεις **κολύμπι, κολυμβητήριο** και δεν υπήρχε η λέξη **κολυμβητής** και χρησιμοποιούταν rule-based tokenizer, όταν το μοντέλο συναντούσε την πρόταση Ο **κολυμβητής πήδηξε από το βάθρο**, τότε το μοντέλο δεν θα μπορούσε να μετατρέψει σωστά την πρόταση.

2.2.5 Κωδικοποίηση Ζεύγους Byte

Η κωδικοποίηση ζεύγους byte (Byte Pair Encoding, BPE) είναι ένας από τους πλέον δημοφιλής αλγορίθμους tokenization. Αφορά tokenizers σε επίπεδο υπολέξεων και παραλλαγές αυτού χρησιμοποιούνται από μερικά από τα δημοφιλέστερα Μεγάλα Γλωσσικά Μοντέλα (LLM) όπως το ChatGPT της OpenAI και το Llama της Meta [31]. Ο συγκεκριμένος αλγόριθμος κατασκευάζει το λεξιλόγιο από τις πιο κοινές οντότητες στο corpus εκπαίδευσης.

Ο αλγόριθμος BPE υιοθετήθηκε από την OpenAI για την εκπαίδευση του GPT και στη συνέχεια εφαρμόστηκε ευρέως σε μοντέλα όπως GPT-2, RoBERTa, BART

και DeBERTa [27]. Η θεμελιώδης ιδέα του BPE έγκειται στην επαναληπτική συγχώνευση των πιο συχνών ζευγών χαρακτήρων ή συμβόλων στο corpus εκπαίδευσης, δημιουργώντας σταδιακά ένα λεξιλόγιο υπο-λέξεων.

Περιγραφή Αλγορίθμου

Ο αλγόριθμος BPE εκτελείται σε δύο διακριτά στάδια: την εκπαίδευση (training phase) και την κωδικοποίηση (encoding phase). Κατά το στάδιο της εκπαίδευσης, το σύστημα αναλύει το corpus κειμένου και μαθαίνει τις βέλτιστες στρατηγικές συγχώνευσης χαρακτήρων. Κατά το στάδιο της κωδικοποίησης, εφαρμόζει τις μαθημένες στρατηγικές για την τεμαχισμό νέου κειμένου.

Η διαδικασία ξεκινά με τη δημιουργία ενός αρχικού λεξιλογίου που περιέχει όλους τους μοναδικούς χαρακτήρες που απαντώνται στο corpus. Στη συνέχεια, ο αλγόριθμος επαναληπτικά:

1. Ύπολογίζει τη συχνότητα εμφάνισης όλων των δυνατών ζευγών διαδοχικών συμβόλων
2. Επιλέγει το ζεύγος με τη μεγαλύτερη συχνότητα
3. Συγχωνεύει αυτό το ζεύγος σε ένα νέο σύμβολο
4. Ενημερώνει το λεξιλόγιο προσθέτοντας το νέο σύμβολο
5. Επαναλαμβάνει τη διαδικασία μέχρι να επιτευχθεί το επιθυμητό μέγεθος λεξιλογίου

Αλγόριθμος 2.2 Byte Pair Encoding (BPE)

```

1: Input: Σώμα κειμένου  $C$ , μέγιστο μέγεθος λεξιλογίου  $V_{\max}$ 
2: Output: Λεξιλόγιο  $V$ , λίστα συγχωνεύσεων  $M$ 
3:  $W \leftarrow$  λίστα λέξεων από  $C$  (με δείκτη τέλους  $\langle /w \rangle$ )
4:  $V \leftarrow$  σύνολο όλων των αρχικών χαρακτήρων (με  $\langle /w \rangle$ )
5:  $M \leftarrow []$  # κενή ακολουθία συγχωνεύσεων
6: while  $|V| < V_{\max}$  do
7:    $pairs \leftarrow$  μετρήσεις συχνότητας γειτονικών συμβόλων σε όλες τις λέξεις  $W$ 
8:   if  $pairs$  είναι κενό then break
9:   end if
10:   $(a, b) \leftarrow \arg \max (x, y) \in pairs \text{ freq}(x, y)$  # πιο συχνό ζεύγος
11:   $W \leftarrow$  αντικατάστησε σε κάθε λέξη όλες τις εμφανίσεις του  $a$   $b$  με το νέο σύμβολο  $ab$ 
12:   $V \leftarrow V \cup \{ab\}$ 
13:  προσάρτησε  $(a, b)$  στο  $M$ 
14: end while
15: return  $V, M$ 

```

Παράδειγμα Εκτέλεσης

Για την καλύτερη κατανόηση της λειτουργίας του BPE, παρουσιάζεται ένα απλοποιημένο παράδειγμα.¹ Εστω ότι διαθέτουμε το ακόλουθο corpus:

`["low", "lower", "newest", "widest"]`

¹Το παράδειγμα είναι προσαρμοσμένο από [2]

Βήμα 1: Αρχικοποίηση Δημιουργία του αρχικού λεξιλογίου με όλους τους μοναδικούς χαρακτήρες: $\text{vocab} = \{\text{l, o, w, e, r, n, s, t, i, d}\}$

Προσθήκη ειδικού συμβόλου τέλους λέξης ($</w>$):

- $\text{low}</w>$
- $\text{lower}</w>$
- $\text{newest}</w>$
- $\text{widest}</w>$

Βήμα 2: Υπολογισμός συχνοτήτων ζευγών

- es : 2 εμφανίσεις (newest, widest)
- st : 2 εμφανίσεις (newest, widest)
- lo : 2 εμφανίσεις (low, lower)

Βήμα 3: Συγχώνευση Επιλογή του ζεύγους με μεγαλύτερη συχνότητα (π.χ. "es") και δημιουργία νέου συμβόλου: $\text{vocab} = \text{vocab} \cup \{\text{es}\}$

Η διαδικασία συνεχίζεται επαναληπτικά μέχρι την επίτευξη του επιθυμητού μεγέθους λεξιλογίου.

Παραλλαγές του Byte Pair Encoding

Παρότι ο αλγόριθμος BPE [27] αποτέλεσε σημείο καμπής στη δημιουργία υπολέξεων, στη συνέχεια εμφανίστηκαν αρκετές παραλλαγές που βελτίωσαν συγκεκριμένα μειονεκτήματα:

- **WordPiece [28]**: Εισήχθη αρχικά στη φωνητική αναγνώριση ιαπωνικών και κορεατικών. Σε αντίθεση με το BPE που συγχωνεύει με βάση τη συχνότητα, το WordPiece επιλέγει συγχωνεύσεις με βάση τη μεγιστοποίηση της πιθανότητας (likelihood) σε ένα γλωσσικό μοντέλο. Αποτελεί τον tokenizer των BERT μοντέλων.
- **Unigram Language Model [29]**: Πρόκειται για στοχαστική μέθοδο, όπου ξεκινάμε με ένα μεγάλο αρχικό λεξιλόγιο υπο-λέξεων και στη συνέχεια αφαιρούμε υποψήφια tokens με βάση την πιθανότητα να εξηγήσουν καλύτερα το corpus. Αυτή η μέθοδος υλοποιείται στη βιβλιοθήκη SentencePiece και χρησιμοποιείται από μοντέλα όπως XLNet και T5.

Η ύπαρξη διαφορετικών προσεγγίσεων υπογραμμίζει το γεγονός ότι η επιλογή του tokenizer επηρεάζει σημαντικά την απόδοση του εκάστοτε γλωσσικού μοντέλου. Η απόφαση για το ποια μέθοδος θα υιοθετηθεί εξαρτάται από παραμέτρους όπως η γλώσσα, το μέγεθος του corpus και οι απαιτήσεις σε ακρίβεια ή ταχύτητα.

2.3 ΕΝΣΩΜΑΤΩΣΕΙΣ

Από την προηγούμενη ενότητα έγινε κατανοητό πώς το κείμενο τεμαχίζεται σε tokens, μια διαδικασία που αποτελεί το αρχικό στάδιο επεξεργασίας σε κάθε γλωσσικό μοντέλο. Το επόμενο βήμα είναι η απόδοση σε κάθε token μιας διανυσματικής αναπαράστασης. Η αναπαράσταση αυτή, γνωστή ως **ενσωμάτωση** (*embedding*), αποτυπώνει τις σημασιολογικές και συντακτικές σχέσεις των tokens και λειτουργεί ως η βασική είσοδος σε ένα LLM, καθιστώντας δυνατή την αριθμητική επεξεργασία της γλώσσας [33].

2.3.1 One-hot κωδικοποίηση

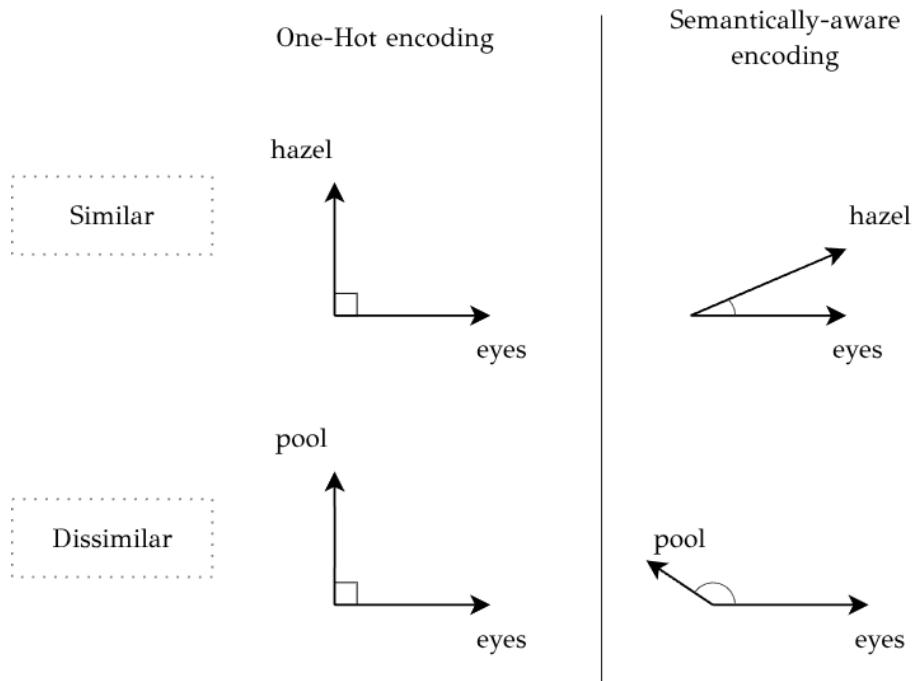
Η ανάλυση ξεκινά από την πιο απλή, αλλά ταυτόχρονα προβληματική μέθοδο: την One-Hot κωδικοποίηση (One-hot Encoding, OHE). Σε αυτήν, κάθε token που βρίσκεται στο λεξιλόγιο V αναπαρίσταται με ένα διάνυσμα μεγέθους $|V|$, στο οποίο όλα τα στοιχεία είναι 0, εκτός από εκείνο που αντιστοιχεί στο συγκεκριμένο token, το οποίο έχει τιμή 1.



Σχήμα 2.5: Παράδειγμα One-hot κωδικοποίησης

Η μέθοδος αυτή παρουσιάζει δύο σημαντικά μειονεκτήματα, και για τον λόγο αυτό δεν χρησιμοποιείται στα σύγχρονα γλωσσικά μοντέλα:

- **Απουσία σημασιολογικής πληροφορίας:** Στον χώρο των ενσωματώσεων, σημασιολογικά παρόμοια tokens θα πρέπει να βρίσκονται κοντά μεταξύ τους. Στην παραπάνω αναπαράσταση, όμως, όλα τα tokens του λεξιλογίου σχηματίζουν μεταξύ τους ορθή γωνία. Ως εκ τούτου, το μοντέλο αδυνατεί να κατανοήσει τη νοηματική συνάφεια μεταξύ παρόμοιων tokens.



Σχήμα 2.6: Σύγκριση One-hot κωδικοποίησης με σημασιολογική αναπαράσταση

- **Υψηλή διαστατικότητα:** Η διάσταση κάθε ενσωμάτωσης είναι ίση με το μέγεθος του λεξιλογίου ($|V|$). Αυτό συνεπάγεται πολύ υψηλό υπολογιστικό κόστος, καθώς ένα τυπικό λεξιλόγιο έχει μέγεθος της τάξης $10^4 - 10^5$.

2.3.2 Συνεχείς Ενσωματώσεις

Οι μέθοδοι που παράγουν συνεχείς ενσωματώσεις δημιουργήθηκαν από την ανάγκη αποτύπωσης της σημασιολογίας κάθε λέξης στην αντίστοιχη ενσωμάτωση. Παρακάτω παρουσιάζονται συνοπτικά οι μέθοδοι αναπαράστασης, που οδήγησαν στον μηχανισμό αυτοπροσοχής, ο οποίος χρησιμοποιείται από τα σύγχρονα LLM.

Word2Vec

Μια από τις πρώτες μεθόδους για τη δημιουργία συνεχών ενσωματώσεων ήταν το Word2Vec [34]. Η βασική του ιδέα είναι ότι οι λέξεις που εμφανίζονται σε παρόμοια συμφραζόμενα τείνουν να έχουν παρόμοιες σημασίες, και επομένως μπορούν να αναπαρασταθούν με κοντινά διανύσματα σε έναν πολυδιάστατο χώρο.

Το Word2Vec προτάθηκε σε δύο παραλλαγές:

- **Continuous Bag-of-Words (CBOW):** Το μοντέλο προβλέπει τη λέξη-στόχο με βάση τα γειτονικά της tokens. Για παράδειγμα, από τα συμφραζόμενα “*the cat _ on the mat*”, το μοντέλο μαθαίνει να προβλέπει τη λέξη “*sits*”.
- **Skip-gram:** Η αντίστροφη διαδικασία, όπου από μία λέξη-στόχο το μοντέλο προβλέπει τα γειτονικά της tokens.

Και στις δύο περιπτώσεις, η εκπαίδευση γίνεται με έναν απλό νευρωνικό ταξινομητή ενός χρυφού επιπέδου, ο οποίος, μετά από πολλές επαναλήψεις, μαθαίνει βάρη που λειτουργούν ως διανυσματικές αναπαραστάσεις των λέξεων. Έτσι, λέξεις με παρόμοια χρήση αποκτούν παρόμοια embeddings. Ένα κλασικό παράδειγμα αυτής της ιδιότητας είναι ότι τα διανύσματα ικανοποιούν σχέσεις αναλογιών, όπως:

$$\text{king} - \text{man} + \text{woman} \approx \text{queen}.$$

Σε αντίθεση με τη one-hot κωδικοποίηση, το Word2Vec αποτυπώνει σημασιολογικές σχέσεις και οδηγεί σε πολύ πιο χρήσιμες αναπαραστάσεις. Ωστόσο, εξακολουθεί να παράγει στατικά embeddings: κάθε λέξη έχει μία και μοναδική διανυσματική αναπαράσταση, ανεξάρτητα από τα συμφραζόμενα μέσα στα οποία εμφανίζεται.

Αναδρομικά Νευρωνικά Δίκτυα (RNNs)

Μετά τις στατικές ενσωματώσεις όπως το Word2Vec, εμφανίστηκαν τα **αναδρομικά νευρωνικά δίκτυα** (RNNs) ως μια λύση για την επεξεργασία ακολουθιακών δεδομένων. Η βασική ιδέα ήταν ότι κάθε χρυφή κατάσταση h_t ενημερώνεται με βάση την προηγούμενη κατάσταση h_{t-1} και την τρέχουσα είσοδο x_t :

$$h_t = f(Wx_t + Uh_{t-1}), \quad (2.17)$$

όπου W, U είναι βάρη που μαθαίνονται και f μια μη γραμμική συνάρτηση ενεργοποίησης όπως η \tanh [35]. Με αυτόν τον τρόπο, το δίκτυο μπορεί θεωρητικά να «θυμάται» πληροφορίες από το παρελθόν.

Ωστόσο, κατά την εκπαίδευση μέσω οπισθοδιάδοσης στο χρόνο (*backpropagation through time*), οι καλίσεις (gradients) είτε εξαφανίζονται είτε εκρήγνυνται, καθιστώντας δύσκολη την εκμάθηση μακροπρόθεσμων εξαρτήσεων. Έτσι, τα απλά RNNs είναι αποτελεσματικά μόνο για βραχυπρόθεσμα συμφραζόμενα.

Long Short-Term Memory (LSTM) και Gated Recurrent Unit (GRU)

Για να ξεπεραστούν οι αδυναμίες των RNNs, αναπτύχθηκαν πιο σύνθετες μονάδες, όπως τα **LSTMs** [36] και οι **GRUs** [37]. Τα LSTMs εισάγουν μια κυψέλη μνήμης και μηχανισμούς «πυλών» (gates) που ρυθμίζουν ποια πληροφορία αποθηκεύεται, ποια ξεχνιέται και ποια εξάγεται σε κάθε χρονικό βήμα. Οι GRUs αποτελούν μια απλούστερη παραλλαγή με λιγότερες πύλες, αλλά παρόμοια αποτελεσματικότητα. Παρά τη σημαντική βελτίωση, τα μοντέλα αυτά παραμένουν εγγενώς σειριακά: η επεξεργασία κάθε στοιχείου εξαρτάται από το προηγούμενο. Αυτό περιορίζει δραστικά την παραλληλοποίηση και καθιστά αργή την εκπαίδευση σε μεγάλα σύνολα δεδομένων. Επιπλέον, αν και οι LSTMs μπορούν να διατηρήσουν πληροφορίες σε μεγαλύτερο χρονικό βάθος, στην πράξη το εύρος μνήμης παραμένει περιορισμένο.

2.3.3 Σημασιολογικές Ενσωματώσεις

Οι παραδοσιακές στατικές ενσωματώσεις, όπως η Word2Vec, αποτυπώνουν γενικές σημασιολογικές σχέσεις, αλλά αποδίδουν στην ίδια λέξη την ίδια αναπαράσταση ανεξαρτήτως συμφραζόμενων. Για παράδειγμα, η λέξη *bank* θα έχει την ίδια

ενσωμάτωση τόσο στη φράση *river bank* όσο και στη φράση *bank account*, παρότι η σημασία της είναι εντελώς διαφορετική.

Για να ξεπεραστεί αυτός ο θεμελιώδης περιορισμός, τα σύγχρονα γλωσσικά μοντέλα χρησιμοποιούν τον μηχανισμό προσοχής (attention) για να παράγουν **σημασιολογικές ενσωματώσεις** (*contextualized embeddings*). Αυτός ο μηχανισμός επιτρέπει σε κάθε token να αναπροσαρμόζει την αναπαράστασή του λαμβάνοντας υπόψη όλα τα υπόλοιπα tokens της ακολουθίας και το συγκεκριμένο πλαίσιο στο οποίο εμφανίζεται.

Η διαδικασία της αυτοπροσοχής (*self-attention*) λειτουργεί ως εξής: για κάθε token i σε μια ακολουθία μήκους T , δημιουργούνται τρία διανύσματα μέσω γραμμικών μετασχηματισμών της αρχικής ενσωμάτωσης: το διάνυσμα ερώτησης (*query*) q_i , το διάνυσμα κλειδιού (*key*) k_i και το διάνυσμα τιμής (*value*) v_i . Τα διανύσματα αυτά υπολογίζονται για όλες τις ενσωματώσεις της ακολουθίας.

Ο βαθμός προσοχής μεταξύ του token i και κάθε άλλου token j στην ακολουθία υπολογίζεται με εσωτερικό γινόμενο των αντίστοιχων διανυσμάτων ερώτησης και κλειδιού, ακολουθούμενο από κανονικοποίηση με softmax (2.7):

$$\alpha_{i,j} = \frac{\exp\left(\frac{q_i \cdot k_j}{\sqrt{d_k}}\right)}{\sum_{l=1}^T \exp\left(\frac{q_i \cdot k_l}{\sqrt{d_k}}\right)} \quad (2.18)$$

όπου d_k είναι η διάσταση των διανυσμάτων κλειδιού και χρησιμοποιείται για λόγους αριθμητικής σταθερότητας.

Η νέα σημασιολογική ενσωμάτωση του token i προκύπτει ως σταθμισμένος μέσος όλων των διανυσμάτων τιμής, όπου τα βάρη καθορίζονται από τους βαθμούς προσοχής:

$$z_i = \sum_{j=1}^T \alpha_{i,j} v_j \quad (2.19)$$

2.4 ΜΕΤΑΣΧΗΜΑΤΙΣΤΕΣ

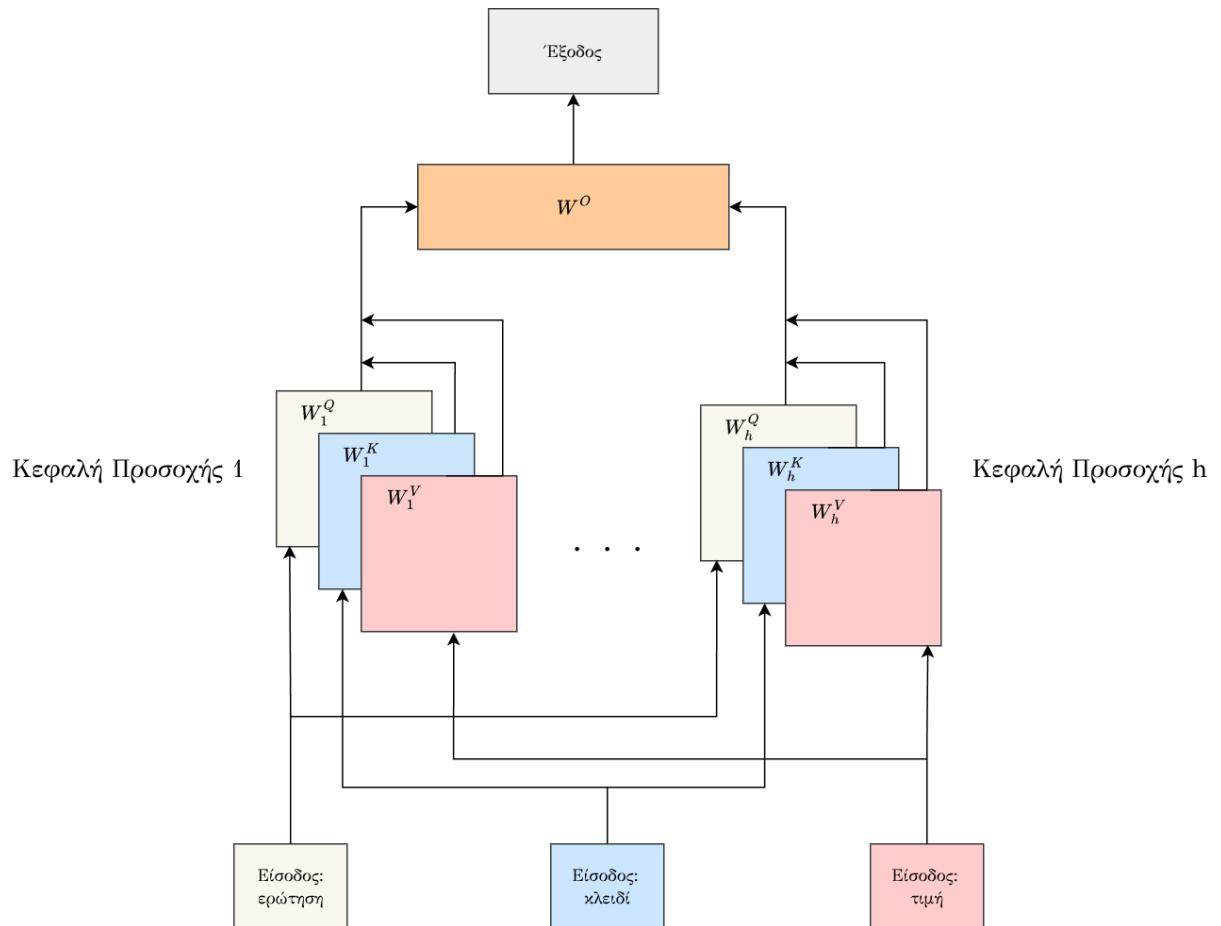
Έχοντας πλέον ορίσει την έννοια της προσοχής ως θεμελιώδη μηχανισμό στην επεξεργασία φυσικής γλώσσας, καθίσταται αναγκαία η ανάλυση της αρχιτεκτονικής του μετασχηματιστή (Transformer), η οποία αποτέλεσε καθοριστική καινοτομία στην υπολογιστική γλωσσολογία. Η συγκεκριμένη αρχιτεκτονική [4], επέφερε σημαντική μεταστροφή στον τρόπο προσέγγισης των γλωσσικών μοντέλων, καταργώντας την εξάρτηση από τα επαναληπτικά νευρωνικά δίκτυα και βασιζόμενη αποκλειστικά σε μηχανισμούς προσοχής.

Οι μετασχηματιστές κατάφεραν να υπερβούν τους εγγενείς υπολογιστικούς περιορισμούς των προηγούμενων αρχιτεκτονικών [37], ιδίως τη σειριακή φύση των LSTM και GRU δικτύων που εμπόδιζε την αποτελεσματική παραλληλοποίηση των υπολογισμών, καθώς και το πρόβλημα της υποβάθμισης των μακρινών εξαρτήσεων σε εκτεταμένες ακολουθίες [26]. Η θεωρητική αναλυτική ικανότητα των μετασχηματιστών έχει τεκμηριωθεί από τους Chulhee Yun, et al.[38], οι οποίοι απέδειξαν ότι, υπό συγκεκριμένες συνθήκες, οι μετασχηματιστές μπορούν να λειτουργήσουν

ως καθολικοί προσεγγιστές (universal approximators) για συναρτήσεις ακολουθιών. Δηλαδή, διαθέτουν επαρκή εκφραστική δύναμη ώστε να προσεγγίσουν οποιαδήποτε απεικόνιση μεταξύ ακολουθιών εισόδων και εξόδων, εφόσον διαθέτουν επαρκές μέγεθος και κατάλληλη παραμετροποίηση. Η αρχιτεκτονική αυτή αποτελεί τη θεωρητική και τεχνολογική βάση των σύγχρονων μεγάλων γλωσσικών μοντέλων, καθιστώντας την κατανόησή της θεμελιώδες στοιχείο για την κατανόηση της σύγχρονης τεχνητής νοημοσύνης.

2.4.1 Πολυκεφαλική Προσοχή

Η πολυκεφαλική προσοχή (multihead attention) αποτελεί εξελιγμένη μορφή του μηχανισμού αυτοπροσοχής και συνιστά θεμελειώδες στοιχείο της αρχιτεκτονικής των μετασχηματιστών [39]. Η βαθιά κατανόηση του συγκεκριμένου μηχανισμού κρίνεται απαραίτητη για την αντίληψη της λειτουργίας των σύγχρονων γλωσσικών μοντέλων.



Σχήμα 2.7: Σχηματικό διάγραμμα πολυκεφαλικής προσοχής (προσαρμογή από [3])

Σε κάθε κεφαλή προσοχής i , οι είσοδοι ερώτηση, κλειδί και τιμή πολλαπλασιάζονται με διαφορετικούς πίνακες βαρών W_i^Q, W_i^K, W_i^V αντίστοιχα. Μέσω αυτής της

διαδικασίας προκύπτουν οι πίνακες:

$$Q_i = XW_i^Q, \quad K_i = XW_i^K, \quad V_i = XW_i^V,$$

όπου X αναπαριστά τον πίνακα εισόδων, συνήθως τα διανύσματα ενσωμάτωσης των tokens.

Στη συνέχεια εφαρμόζεται ο μηχανισμός προσοχής σύμφωνα με τη σχέση:

$$\text{Attention}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i K_i^\top}{\sqrt{d_k}}\right) V_i, \quad (2.20)$$

όπου d_k αναφέρεται στη διάσταση των διανυσμάτων K_i . Το παραγόμενο αποτέλεσμα συνιστά έναν πίνακα Z_i που εκφράζει τον τρόπο με τον οποίο κάθε token αλληλεπιδρά με τα υπόλοιπα tokens της ακολουθίας.

Η καινοτομία της πολυκεφαλικής προσοχής έγκειται στο γεγονός ότι η προ-αναφερθείσα διαδικασία πραγματοποιείται παράλληλα από h διακριτές κεφαλές προσοχής, παράγοντας τα αποτελέσματα Z_1, Z_2, \dots, Z_h . Η εμπειρική έρευνα των Kevin Clark et al.[40] και των Jesse Vig et al. [41] έχει αποκαλύψει ότι κάθε κεφαλή αναπτύσσει εξειδίκευση σε διαφορετικές γλωσσικές πτυχές. Ενδεικτικά, συγκεκριμένες κεφαλές εμφανίζουν ευαισθησία σε συντακτικές σχέσεις όπως η συμφωνία ρήματος-υποκειμένου, ενώ άλλες επικεντρώνονται σε σημασιολογικές συσχετίσεις μεταξύ εννοιολογικά συναφών όρων.

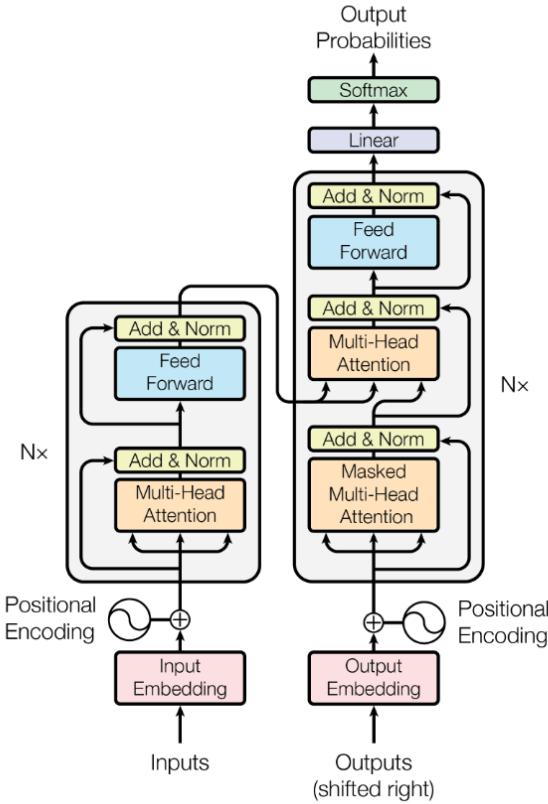
Στο τέλος της διαδικασίας, τα αποτελέσματα όλων των κεφαλών συνενώνονται σε έναν ενοποιημένο πίνακα και προβάλλονται μέσω ενός επιπλέον πίνακα βαρών W^O :

$$\text{MultiHead}(Q, K, V) = \text{Concat}(Z_1, Z_2, \dots, Z_h)W^O. \quad (2.21)$$

Κάθε κεφαλή προσοχής παρέχει μια διακριτή προοπτική επεξεργασίας των δεδομένων, ενώ ο συνδυασμός τους οδηγεί σε πλουσιότερη και εκφραστικότερη αναπαράσταση της πληροφορίας.

2.4.2 Η Αρχιτεκτονική του Μετασχηματιστή

Η αρχιτεκτονική του μετασχηματιστή αναπτύχθηκε το 2017 από ερευνητική ομάδα της Google και παρουσιάστηκε στη θεμελιώδη εργασία "Attention Is All You Need" [4]. Η συγκεκριμένη αρχιτεκτονική επέτυχε αποτελέσματα τελευταίας τεχνολογίας σε εργασίες μηχανικής μετάφρασης, αποδεικνύοντας την αποτελεσματικότητα του μηχανισμού προσοχής.



Σχήμα 2.8: Η αρχιτεκτονική του μετασχηματιστή όπως παρουσιάστηκε στη δημοσίευση "Attention Is All You Need" [4]

Η αρχιτεκτονική του μετασχηματιστή διαχρίνεται σε δύο κύρια τμήματα: τον κωδικοποιητή (encoder) που απεικονίζεται στην αριστερή πλευρά του διαγράμματος, και τον αποκωδικοποιητή (decoder) που παρουσιάζεται στη δεξιά πλευρά.

Ο Κωδικοποιητής του Μετασχηματιστή Η διαδικασία της επεξεργασίας αρχίζει από την ακολουθία των tokens ενός κειμένου. Κάθε token αντιστοιχίζεται σε ένα διάνυσμα ενσωμάτωσης μέσω ενός εκπαιδεύσιμου πίνακα token-ενσωματώσεων. Δεδομένου ότι ο μηχανισμός αυτοπροσοχής δεν ενσωματώνει πληροφορία σχετικά με τη θέση του κάθε token, προστίθενται τα διανύσματα κωδικοποίησης θέσης (positional encodings). Το άθροισμα των ενσωματώσεων και των διανυσμάτων θέσης συνιστά την είσοδο του κωδικοποιητή.

Κάθε μπλοκ κωδικοποιητή ενσωματώνει τρία θεμελιώδη υπολογιστικά στοιχεία. Το στρώμα αυτοπροσοχής παράγει από τις ενσωματώσεις τα διανύσματα ερωτήσεων, κλειδιών και τιμών. Μέσω του μηχανισμού αυτοπροσοχής, κάθε token ενημερώνει την αναπαράστασή του βασιζόμενο στη συνάφειά του με όλα τα υπόλοιπα tokens της ακολουθίας. Το Feed-Forward δίκτυο εφαρμόζεται ανεξάρτητα σε κάθε αναπαράσταση, υλοποιώντας τη ιδιότητα καθολικής προσέγγισης, που επιτρέπει την προσέγγιση οποιασδήποτε συνεχούς συνάρτησης με επαρκή ακρίβεια. Οι υπολειμματικές συνδέσεις και η κανονικοποίηση στρώματος εφαρμόζονται μετά την

προσοχή και το Feed-Forward, βελτιώνοντας τη σταθερότητα της εκπαίδευσης σύμφωνα με την ανάλυση των Kaiming He et al. [42].

Η στοίβα N τέτοιων κωδικοποιητών δημιουργεί διαδοχικά εμπλουτισμένες αναπαραστάσεις, καθώς κάθε επίπεδο επεξεργάζεται και βελτιώνει τα αποτελέσματα του προηγούμενου.

Ο Αποκωδικοποιητής του Μετασχηματιστή Ο αποκωδικοποιητής αναλαμβάνει την αυτοπαλίνδρομη (autoregressive) διαδικασία παραγωγής της ακολουθίας εξόδου. Σε αντίθεση με τον κωδικοποιητή που επεξεργάζεται ολόκληρη την ακολουθία εισόδου ταυτόχρονα, ο αποκωδικοποιητής λειτουργεί διαδοχικά, παράγοντας κάθε token με βάση τα token που παράχθηκαν προηγουμένως.

Κάθε μπλοκ αποκωδικοποιητή ενσωματώνει τέσσερα θεμελιώδη υπολογιστικά στοιχεία. Το στρώμα μασκαρισμένης αυτοπροσοχής επιτρέπει σε κάθε token της ακολουθίας εξόδου να αλληλεπιδρά αποκλειστικά με τα token που αποκωδικοποιήθηκαν προηγουμένως. Η μάσκα εξασφαλίζει ότι η ροή της πληροφορίας ακολουθεί αιτιακή (causal) κατεύθυνση, δηλαδή κάθε token μπορεί να εξαρτάται μόνο από όσα έχουν ήδη παραχθεί, ποτέ από μελλοντικά. Το στρώμα διασταυρούμενης προσοχής εκτελεί τη σύνδεση μεταξύ της ακολουθίας εισόδου και εξόδου. Οι αναπαραστάσεις από το στρώμα μασκαρισμένης προσοχής χρησιμοποιούνται ως ερωτήσεις, ενώ τα κλειδιά και οι τιμές προέρχονται από τις τελικές αναπαραστάσεις του κωδικοποιητή.

Το Feed-Forward δίκτυο εφαρμόζει μη-γραμμικούς μετασχηματισμούς στις συνδυασμένες αναπαραστάσεις. Οι υπολειμματικές συνδέσεις και η κανονικοποίηση στρώματος εφαρμόζονται μετά από κάθε υπο-στρώμα, διασφαλίζοντας τη σταθερότητα της εκπαίδευσης.

Η Έξοδος του Μετασχηματιστή Η τελική έξοδος του μετασχηματιστή περιλαμβάνει ένα γραμμικό στρώμα προβολής που μετασχηματίζει τις αναπαραστάσεις του αποκωδικοποιητή στον χώρο του λεξιλογίου. Το στρώμα αυτό αποτελείται από έναν πίνακα βαρών διαστάσεων $d_{\text{model}} \times |V|$, όπου $|V|$ είναι το μέγεθος του λεξιλογίου. Η εφαρμογή της συνάρτησης softmax (2.7) στις προβολές αυτές παράγει κατανομή πιθανοτήτων για κάθε δυνατό επόμενο token. Η διαδικασία αυτή επαναλαμβάνεται αυτοπαλίνδρομα μέχρι την παραγωγή του τελικού token της ακολουθίας εξόδου, επιτρέποντας στο μοντέλο την αυτοπαλίνδρομη παραγωγή κειμένου υψηλής ποιότητας.

2.5 ΜΕΓΑΛΑ ΓΛΩΣΣΙΚΑ ΜΟΝΤΕΛΑ

Η εξέλιξη των μετασχηματιστών οδήγησε στη δημιουργία μιας νέας κατηγορίας μοντέλων που επαναπροσδιόρισε το τοπίο της τεχνητής νοημοσύνης: τα Μεγάλα Γλωσσικά Μοντέλα (Large Language Models, LLMs). Σε αντίθεση με την κλασική αρχιτεκτονική κωδικοποιητή-αποκωδικοποιητή των αρχικών μετασχηματιστών, τα σύγχρονα LLMs υιοθετούν αποκλειστικά την αρχιτεκτονική αποκωδικοποιητή, αποτελώντας μια στρατηγική σχεδιαστική επιλογή που αποδείχτηκε υπολογιστικά απο-

δοτική και εμπειρικά επιβεβαιωμένη για εργασίες παραγωγής και κατανόησης φυσικής γλώσσας [43].

2.5.1 Η Αρχιτεκτονική Αποκωδικοποιητή

Τα μοντέλα αποκωδικοποιητή αποτελούν εστιασμένη αρχιτεκτονική εκδοχή της αρχικής δομής των μετασχηματιστών, διατηρώντας αποκλειστικά το τμήμα του αποκωδικοποιητή χωρίς τον κωδικοποιητή και το μηχανισμό διασταυρούμενης προσοχής. Αυτή η αρχιτεκτονική βασίζεται στη μασκαρισμένη αυτοπροσοχή, όπου κάθε λεκτική μονάδα μπορεί να αλληλεπιδρά μόνο με τις προηγούμενες λεκτικές μονάδες στην ακολουθία, εξασφαλίζοντας τη διατήρηση της αιτιακής φύσης κατά τη διάρκεια της εκπαίδευσης και της απόφασης (inference).

Η μάσκα αυτοπροσοχής για μια ακολουθία μήκους n ορίζεται ως ο πίνακας $M \in \mathbb{R}^{n \times n}$:

$$M_{i,j} = \begin{cases} 0 & \text{αν } j \leq i \\ -\infty & \text{αν } j > i \end{cases} \quad (2.22)$$

όπου ο δείκτης i αντιστοιχεί στη θέση ερωτήματος (query) και ο δείκτης j στη θέση κλειδιού (key). Αυτός ο πίνακας μάσκας προστίθεται στα attention scores πριν την εφαρμογή της συνάρτησης softmax, διασφαλίζοντας ότι οι μασκαρισμένες θέσεις με τιμή $-\infty$ παράγουν μηδενικές πιθανότητες προσοχής μετά τη softmax εφαρμογή, καθώς $\text{softmax}(-\infty) = 0$.

Θεωρητική Τεκμηρίωση της Αρχιτεκτονικής Επιλογής

Η επιλογή της αρχιτεκτονικής αποκωδικοποιητή στα LLMs προκύπτει από συγκλίνουσες θεωρητικές και πρακτικές εκτιμήσεις. Θεωρητικά, η γλωσσική μοντελοποίηση αποτελεί εγγενώς αυτοπαλίνδρομη διαδικασία όπου η πρόβλεψη κάθε επόμενης λεκτικής μονάδας εξαρτάται αποκλειστικά από το προηγούμενο ιστορικό, χωρίς την ανάγκη επεξεργασίας διακριτής ακολουθίας εισόδου.

Η αυτοπαλίνδρομη φύση της γλωσσικής μοντελοποίησης εκφράζεται μαθηματικά ως:

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(x_i | x_1, x_2, \dots, x_{i-1}) \quad (2.23)$$

όπου κάθε λεκτική μονάδα x_i προβλέπεται βάσει του προηγούμενου πλαισίου x_1, x_2, \dots, x_{i-1} . Αυτή η διατύπωση αποκαλύπτει την εγγενή αιτιακή φύση της γλωσσικής παραγωγής, όπου κάθε λεκτική μονάδα επηρεάζεται μόνον από τις χρονικά προγενέστερες μονάδες.

Σε αυτό το θεωρητικό πλαίσιο, η παρουσία ξεχωριστού κωδικοποιητή και διασταυρούμενης προσοχής δεν προσφέρει επιπλέον εκφραστική δύναμη, καθώς δεν υπάρχει διακριτή ακολουθία εισόδου που να απαιτεί ανεξάρτητη κωδικοποίηση. Η γλωσσική μοντελοποίηση αποτελεί εγγενώς μονοδιάστατη διαδικασία όπου κάθε νέα λεκτική μονάδα συνθέτει το υφιστάμενο πλαίσιο, δημιουργώντας μια ενιαία, συνεχώς αυξανόμενη ακολουθία [43].

Από πρακτικής άποψης, η αρχιτεκτονική αποκωδικοποιητή παρουσιάζει σημαντικά πλεονεκτήματα σε όρους υπολογιστικής αποδοτικότητας και κλιμάκωσης. Η εξάλειψη του κωδικοποιητή και του μηχανισμού διασταυρούμενης προσοχής μειώνει τον αριθμό των εκπαιδεύσιμων παραμέτρων κατά περίπου το ένα τρίτο σε σχέση με τη συμβατική αρχιτεκτονική κωδικοποιητή-αποκωδικοποιητή, επιτρέποντας την ανάπτυξη μοντέλων μεγαλύτερης κλίμακας με ισοδύναμους υπολογιστικούς πόρους. Επιπλέον, η ομοιογενής αρχιτεκτονική διευκολύνει την παραληλοποίηση των υπολογισμών κατά την εκπαίδευση, καθώς όλα τα στρώματα ακολουθούν την ίδια δομική διάταξη χωρίς εξαρτήσεις μεταξύ διαφορετικών τμημάτων του μοντέλου.

Η αποτελεσματικότητα της αρχιτεκτονικής αποκωδικοποιητή επιβεβαιώθηκε εμπειρικά μέσω των μοντέλων GPT [6, 31, 7], τα οποία επέδειξαν εξαιρετικές επιδόσεις σε ευρύ φάσμα γλωσσικών εργασιών παρά τη φαινομενικά εστιασμένη αρχιτεκτονική τους. Η ικανότητα αυτών των μοντέλων να εκτελούν εργασίες κατανόησης και παραγωγής κειμένου με εξίσου υψηλή απόδοση αποκάλυψε ότι η πολυπλοκότητα της αρχιτεκτονικής κωδικοποιητή-αποκωδικοποιητή δεν αποτελεί προαπαιτούμενο για την επίτευξη γενικής γλωσσικής ικανότητας.

Η εμπειρική επιβεβαίωση των θεωρητικών πλεονεκτημάτων ενισχύθηκε περαιτέρω από τις μελέτες κλιμάκωσης που ανέδειξαν προβλέψιμους νόμους για την απόδοση των νευρωνικών δικτύων σε σχέση με την αύξηση των παραμέτρων [44, 45], επιβεβαιώνοντας ότι η αρχιτεκτονική αποκωδικοποιητή παρουσιάζει ανώτερη αποδοτικότητα στην αξιοποίηση των υπολογιστικών πόρων για την επίτευξη κλιμάκωσης.

2.5.2 Μηχανισμός Πρόβλεψης και Αυτοπαλίνδρομη Παραγωγή

Η κατανόηση του τρόπου με τον οποίο ένα Μεγάλο Γλωσσικό Μοντέλο προβλέπει την επόμενη λεκτική μονάδα και παράγει συνεκτικό κείμενο μέσω αυτοπαλίνδρομης διαδικασίας αποτελεί κεντρικό στοιχείο για την κατανόηση της λειτουργίας των LLMs. Η διαδικασία αυτή περιλαμβάνει τη μετατροπή μιας ακολουθίας εισόδου σε κατανομή πιθανότητας επί του λεξιλογίου και τη διαδοχική εφαρμογή αυτής της διαδικασίας για την παραγωγή εκτεταμένων κειμένων.

Στόχος Εκπαίδευσης και Μαθηματική Διατύπωση

Σύμφωνα με την τυπική πρακτική, η εισαγωγή ενός γλωσσικού μοντέλου αποτελείται από μια ακολουθία λεκτικών μονάδων $\{x_0, x_1, \dots, x_{m-1}\}$. Το γλωσσικό μοντέλο εξάγει μια κατανομή $\text{Pr}(\cdot | x_0, \dots, x_{i-1})$ σε κάθε θέση i , και η λεκτική μονάδα x_i επιλέγεται σύμφωνα με αυτή την κατανομή [46].

Το μοντέλο εκπαίδευται μέσω μεγιστοποίησης της λογαριθμικής πιθανοφάνειας:

$$\mathcal{L} = \sum_{i=1}^m \log \text{Pr}(x_i | x_0, \dots, x_{i-1}) \quad (2.24)$$

Αυτή η στόχευση εκπαίδευσης επιτρέπει στο μοντέλο να μαθαίνει στατιστικές εξαρτήσεις μεταξύ λεκτικών μονάδων και να αναπτύσσει την ικανότητα παραγωγής συνεκτικού κειμένου.

Αρχιτεκτονική Δομή και Στρώματα Μετασχηματιστή

Η κύρια δομή του μοντέλου αποτελείται από μια στοίβα L στρωμάτων μετασχηματιστή. Κάθε στρώμα περιλαμβάνει δύο υπο-στρώματα: το στρώμα αυτοπροσοχής και το Feed-Forward δίκτυο, συνδεδεμένα με υπολειματικές συνδέσεις και κανονικοποίηση στρώματος:

$$\text{output} = \text{LN}(F(\text{input}) + \text{input}) \quad (2.25)$$

όπου $F(\cdot)$ αποτελεί τη βασική συνάρτηση του υπο-στρώματος.

Σημείωση: Η παραπάνω μορφή αντιστοιχεί στη λεγόμενη **Post-Layer Normalization** εκδοχή του μετασχηματιστή, όπου η κανονικοποίηση εφαρμόζεται μετά την υπολειμματική σύνδεση. Σε πολές σύγχρονες υλοποιήσεις (π.χ. GPT-2/3), χρησιμοποιείται η εναλλακτική **Pre-Layer Normalization** μορφή, στην οποία η κανονικοποίηση προηγείται του υποστρώματος:

$$\text{output} = \text{input} + F(\text{LN}(\text{input})).$$

Η παραλλαγή αυτή συμβάλλει στη βελτίωση της σταθερότητας κατά την εκπαίδευση πολύ βαθιών μοντέλων.

Η αυτοπροσοχή εφαρμόζει τον μηχανισμό προσοχής QKV με ενσωματωμένη μάσκα για τη διατήρηση της αιτιακής φύσης:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} + \text{Mask}\right)\mathbf{V} \quad (2.26)$$

όπου \mathbf{Q} , \mathbf{K} και $\mathbf{V} \in \mathbb{R}^{m \times d_k}$ είναι τα ερωτήματα, κλειδιά και αξίες αντίστοιχα, με d_k τη διάσταση του χώρου προβολής για κάθε κεφαλή προσοχής.

Δεδομένης αναπαράστασης $\mathbf{H} \in \mathbb{R}^{m \times d_{\text{model}}}$, η πολυκέφαλη αυτοπροσοχή επεκτείνει αυτόν τον μηχανισμό σε h παράλληλους υπο-χώρους:

$$\mathbf{F}(\mathbf{H}) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)\mathbf{W}^{\text{head}} \quad (2.27)$$

όπου κάθε κεφαλή head_j υπολογίζεται ως:

$$\text{head}_j = \text{Attention}(\mathbf{HW}_j^Q, \mathbf{HW}_j^K, \mathbf{HW}_j^V) \quad (2.28)$$

με $\mathbf{W}_j^Q, \mathbf{W}_j^K, \mathbf{W}_j^V \in \mathbb{R}^{d_{\text{model}} \times d_k}$ τους πίνακες προβολής, όπου $d_k = d_{\text{model}}/h$ και h ο αριθμός των κεφαλών προσοχής.

Μετά την επεξεργασία από τα L στρώματα μετασχηματιστή, η τελική πρόβλεψη πραγματοποιείται μέσω γραμμικού μετασχηματισμού και εφαρμογής softmax:

$$\mathbf{P} = \text{Softmax}(\mathbf{H}^L \mathbf{W}^o) \quad (2.29)$$

όπου $\mathbf{H}^L \in \mathbb{R}^{m \times d_{\text{model}}}$ είναι η έξοδος του τελευταίου στρώματος, $\mathbf{W}^o \in \mathbb{R}^{d_{\text{model}} \times |V|}$ ο πίνακας παραμέτρων εξόδου, και $|V|$ το μέγεθος του λεξιλογίου. Το αποτέλεσμα $\mathbf{P} \in \mathbb{R}^{m \times |V|}$ περιέχει κατανομές πιθανότητας για κάθε θέση στην ακολουθία.

Η παραγωγή κειμένου υλοποιείται μέσω αυτοπαλίνδρομης και επαναληπτικής εφαρμογής της διαδικασίας πρόβλεψης. Δεδομένης αρχικής ακολουθίας (prompt), το μοντέλο εκτελεί τα εξής βήματα μέχρι την επίτευξη κριτηρίου τερματισμού:

1. Υπολογισμός κατανομής πιθανότητας για το επόμενο token βάσει του τρέχοντος πλαισίου 2. Επιλογή token μέσω στρατηγικής δειγματοληψίας από την παραγόμενη κατανομή 3. Επέκταση της ακολουθίας με το νέο token και ενημέρωση του πλαισίου 4. Επανάληψη της διαδικασίας με το επικαιροποιημένο πλαίσιο

Στρατηγικές Δειγματοληψίας και Επίδραση στην Ποιότητα

Η επιλογή στρατηγικής δειγματοληψίας επηρεάζει καθοριστικά τη ποιότητα, συνοχή και ποικιλομορφία του παραγόμενου κειμένου [46]:

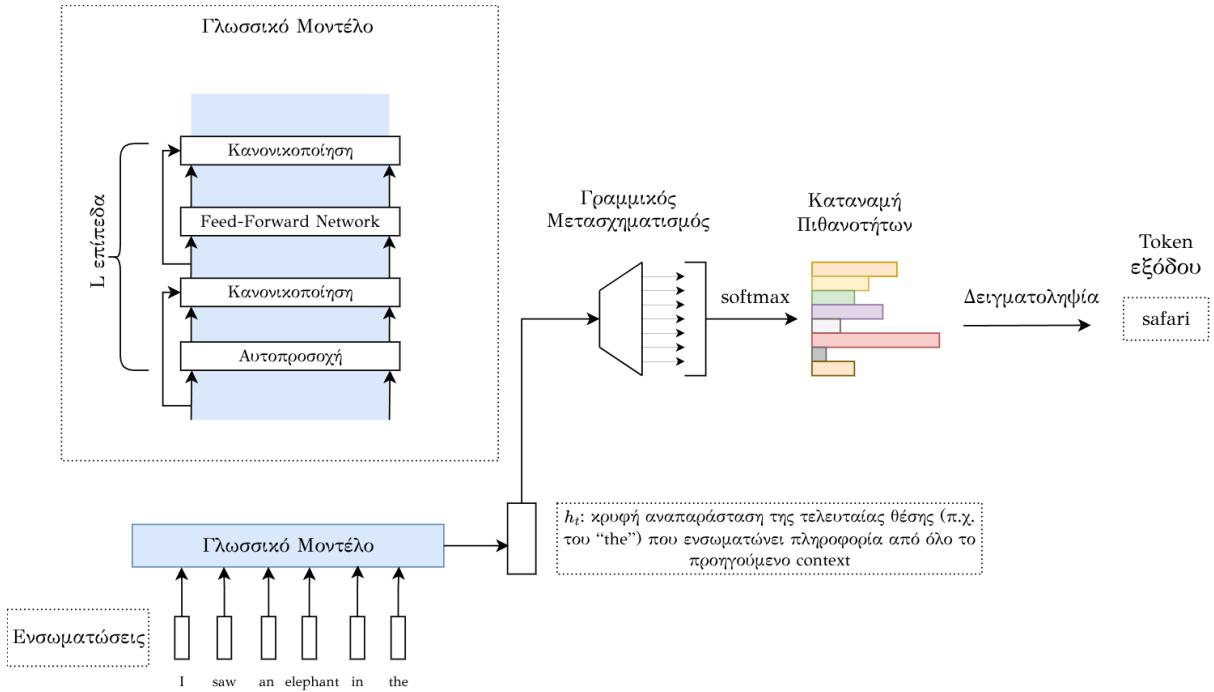
- **Greedy Decoding:** Επιλογή της λεξικής μονάδας με τη μέγιστη πιθανότητα, παρέχοντας ντετερμινιστικά αποτελέσματα με υψηλή τοπική συνοχή αλλά περιορισμένη γλωσσική ποικιλομορφία.
- **Temperature Sampling:** Τροποποίηση της κατανομής πιθανότητας με παράμετρο θερμοκρασίας τ :

$$P_\tau(x_i|\text{context}) = \frac{\exp(\text{logit}_i/\tau)}{\sum_j \exp(\text{logit}_j/\tau)} \quad (2.30)$$

όπου χαμηλότερες τιμές $\tau < 1$ οδηγούν σε πιο συντηρητικές και προβλέψιμες επιλογές με ενισχυμένη τοπική συνοχή, ενώ υψηλότερες τιμές $\tau > 1$ αυξάνουν τη στοχαστικότητα και τη δημιουργικότητα του παραγόμενου κειμένου.

- **Nucleus Sampling (Top-p):** Δειγματοληψία από το ελάχιστο σύνολο λεξικών μονάδων με αθροιστική πιθανότητα που υπερβαίνει το κατώφλι p , εξισορροπώντας αποδοτικά την ποιότητα και ποικιλομορφία του παραγόμενου κειμένου διατηρώντας παράλληλα τη σημασιολογική συνοχή.

Αυτός ο ολοκληρωμένος μηχανισμός επιτρέπει στα LLMs να παράγουν συνεκτικό κείμενο που διατηρεί γλωσσική και σημασιολογική συνοχή σε εκτεταμένες ακολουθίες, αποτελώντας τη βάση για την ευρεία εφαρμογή τους σε εργασίες παραγωγής και επεξεργασίας φυσικής γλώσσας [43].



Σχήμα 2.9: Απεικόνιση της διαδικασίας πρόβλεψης σε ένα Μεγάλο Γλωσσικό Μοντέλο (LLM). Οι ενσωματώσεις εισόδου (token + positional encodings) περνούν από μια στοίβα L στρωμάτων μετασχηματιστή με αυτοπροσοχή, κανονικοποίηση και Feed-Forward δίκτυα, ώστε να παραχθεί η κρυφή αναπαράσταση h_t για τη θέση t . Η αναπαράσταση αυτή ενσωματώνει πληροφορία από όλο το πλαίσιο και τροφοδοτεί τον γραμμικό μετασχηματισμό και τη συνάρτηση softmax, με αποτέλεσμα την κατανομή πιθανοτήτων στο λεξιλόγιο. Τέλος, μέσω δειγματοληψίας, επιλέγεται το επόμενο token εξόδου.

2.5.3 Εκπαίδευση Μεγάλων Γλωσσικών Μοντέλων

Η εκπαίδευση ενός Μεγάλου Γλωσσικού Μοντέλου (LLM) βασίζεται στη μεγιστοποίηση της πιθανοφάνειας σε μεγάλα σύνολα δεδομένων κειμένου. Έστω ένα σύνολο εκπαίδευσης D που αποτελείται από K ακολουθίες. Για κάθε ακολουθία $x = (x_0, x_1, \dots, x_m) \in D$, η λογαριθμική πιθανοφάνεια ορίζεται ως:

$$\mathcal{L}_\theta(x) = \sum_{i=1}^m \log \Pr_\theta(x_i | x_0, \dots, x_{i-1}) \quad (2.31)$$

όπου το θ δηλώνει τις παραμέτρους του μοντέλου. Ο στόχος της εκπαίδευσης είναι η εκτίμηση των παραμέτρων θ μέσω της μεγιστοποίησης της συνολικής πιθανοφάνειας όλων των ακολουθιών εκπαίδευσης:

$$\hat{\theta} = \arg \max_{\theta} \sum_{x \in D} \mathcal{L}_\theta(x) \quad (2.32)$$

Η βελτιστοποίηση του αντικειμενικού σκοπού υλοποιείται με παραλλαγές του αλγορίθμου **επικλινής καθόδου**, οι οποίες υποστηρίζονται από σύγχρονες βιβλιο-

θήκες βαθιάς μάθησης. Ωστόσο, καθώς τα μοντέλα αυξάνονται σε μέγεθος και πολυπλοκότητα, αναδύονται σοβαρές προκλήσεις κλιμάκωσης.

Από πρακτική σκοπιά, η εκπαίδευση LLMs με δεκάδες ή εκατοντάδες δισεκατομμύρια παραμέτρους απαιτεί χιλιάδες ώρες επεξεργασίας σε κατανεμημένα υπερυπολογιστικά clusters [7, 47]. Τέτοια περιβάλλοντα είναι επιρρεπή σε σφάλματα: μία μεμονωμένη πτώση κόμβου ή αστοχία δικτύου μπορεί να οδηγήσει στην αποτυχία ολόκληρης της εκπαίδευτικής διαδικασίας, καθιστώντας απαραίτητη την υλοποίηση μηχανισμών ανοχής σε σφάλματα (*fault tolerance*) και περιοδικής αποθήκευσης κατάστασης (*checkpointing*) [48, 49]. Παράλληλα, η επικοινωνία μεταξύ κόμβων, ιδιαίτερα για τον συγχρονισμό παραμέτρων στο πλαίσιο της παράλληλης εκπαίδευσης δεδομένων ή της παράλληλης εκπαίδευσης μοντέλου, δημιουργεί σημαντικά επικοινωνιακά σημεία συμφόρησης [50, 47]. Η ανισορροπία στην απόδοση των κόμβων (*stragglers*) μπορεί να μειώσει δραστικά την αποδοτικότητα του cluster.

Επιπλέον, η κατανάλωση ενέργειας και το οικονομικό κόστος σε τέτοια συστήματα είναι τεράστια, καθιστώντας την εκπαίδευση απαγορευτική για μικρότερους οργανισμούς [51, 15]. Από αλγορίθμική σκοπιά, η εκπαίδευση βαθιών και πολύ μεγάλων νευρωνικών δικτύων ενέχει κινδύνους αστάθειας στη βελτιστοποίηση, απαιτώντας προσαρμογές στην αρχιτεκτονική (π.χ. κανονικοποίηση στρώματος, υπολειμματικές συνδέσεις, προσαρμοστικούς χρονοπρογραμματιστές ρυθμού μάθησης) για να εξασφαλιστεί η σύγκλιση [52, 42].

Παρά τις προκλήσεις αυτές, η κλιμάκωση των LLMs έχει οδηγήσει σε σταθερές βελτιώσεις απόδοσης, όπως περιγράφουν οι νόμοι κλιμάκωσης [53, 45], προσφέροντας ισχυρό κίνητρο για την ανάπτυξη όλο και μεγαλύτερων μοντέλων.

2.5.4 Προσαρμογή

Η προεκπαίδευση (pretraining) ενός Μεγάλου Γλωσσικού Μοντέλου (LLM) παρέχει γενικευμένες γλωσσικές ικανότητες που αποδεικνύονται χρήσιμες σε ένα ευρύ φάσμα εφαρμογών. Ωστόσο, η απόδοση του μοντέλου σε εξειδικευμένα καθήκοντα ή τομείς συχνά απαιτεί προσαρμογή (fine-tuning) [54, 33]. Η βασική ιδέα συνίσταται στην εκπαίδευση του προεκπαιδευμένου μοντέλου σε μικρότερα, εξειδικευμένα σύνολα δεδομένων, με στόχο τη βελτίωση της γενίκευσης σε συγκεκριμένους τομείς.

Οι κύριες προσεγγίσεις περιλαμβάνουν:

- **Πλήρης προσαρμογή (full fine-tuning):** ενημέρωση όλων των παραμέτρων του μοντέλου. Παρά την υψηλή αποτελεσματικότητα, η μέθοδος αυτή είναι υπολογιστικά δαπανηρή και δύσκολα εφαρμόσιμη σε LLMs δισεκατομμυρίων παραμέτρων.
- **Αποδοτική προσαρμογή παραμέτρων (parameter-efficient tuning):** τεχνικές όπως οι προσαρμογείς (adapters), η προρρυθμισμένη προσαρμογή προθέματος (prefix tuning) και η μέθοδος χαμηλόβαθμης προσαρμογής (LoRA, Low-Rank Adaptation) [55], όπου προσαρμόζεται μόνο ένα μικρό υποσύνολο παραμέτρων, επιτυγχάνοντας σημαντική μείωση των απαιτήσεων σε μνήμη και υπολογιστική ισχύ.

- **Εκπαίδευση με οδηγίες και ενίσχυση από ανθρώπινη ανατροφοδότηση:** *instruction tuning* και *reinforcement learning from human feedback (RLHF)*, τα οποία στοχεύουν στην ευθυγράμμιση του μοντέλου με ανθρώπινες προτιμήσεις και κοινωνικές προσδοκίες [56].

Η διαδικασία fine-tuning έχει καθιερωθεί ως ισχυρό εργαλείο για την προσαρμογή των LLMs σε εξειδικευμένα περιβάλλοντα. Ωστόσο, παραμένει περιορισμένη από τη θεμελιώδη στατικότητα των παραμέτρων του μοντέλου: η γνώση που ενσωματώνεται κατά την προεκπαίδευση και προσαρμογή δεν μπορεί να επικαιροποιηθεί δυναμικά. Η αναγκαιότητα πρόσβασης σε εξωτερική, επίκαιρη και αξιόπιστη γνώση καθιστά αναγκαία την ανάπτυξη συμπληρωματικών προσεγγίσεων. Στη συνέχεια εξετάζεται διεξοδικά η **Επαυξημένη Παραγωγή μέσω Ανάκτησης (Retrieval-Augmented Generation, RAG)**, μία τεχνική που συνδυάζει την υπολογιστική ισχύ των LLMs με την αξιοποίηση εξωτερικών γνωσιακών βάσεων, παρέχοντας ένα δυναμικό και επεκτάσιμο πλαίσιο για την υπέρβαση των περιορισμών της στατικής εκπαίδευσης.

2.6 ΕΠΑΥΞΗΜΕΝΗ ΠΑΡΑΓΩΓΗ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

Παρά την εντυπωσιακή τους ισχύ, τα Μεγάλα Γλωσσικά Μοντέλα (LLMs) παραμένουν εγγενώς περιορισμένα από τη στατικότητα των παραμέτρων τους: η γνώση που κατέχουν αντανακλά αποκλειστικά το σύνολο εκπαίδευσης και τη χρονική στιγμή κατά την οποία αυτό συλλέχθηκε. Αυτό δημιουργεί σοβαρά προβλήματα σε εφαρμογές που απαιτούν πρόσβαση σε επικαιροποιημένη πληροφορία ή σε εξειδικευμένη γνώση που δεν περιλαμβάνεται στο αρχικό corpus.

Ένα επιπλέον ζήτημα είναι το φαινόμενο της **παραισθητικότητας** (*hallucination*), όπου το μοντέλο παράγει συντακτικά ορθό και πειστικό κείμενο το οποίο όμως δεν έχει αντικειμενική αντιστοίχιση με τα δεδομένα ή την πραγματικότητα [13]. Οι παραισθήσεις μπορεί να προκύψουν λόγω έλλειψης σχετικής πληροφορίας, στατιστικής μεροληφίας στο σύνολο εκπαίδευσης, ή λόγω της φύσης των αυτοπαλίνδρομων μηχανισμών πρόβλεψης που στοχεύουν στη γλωσσική συνοχή αλλά όχι απαραίτητα στην πραγματολογική ακρίβεια. Αυτό περιορίζει σοβαρά την αξιοποίησία των LLMs σε ευαίσθητες εφαρμογές (π.χ. ιατρική πληροφόρηση, νομικά κείμενα, επιστημονικές αναφορές).

Η τεχνική της **Επαυξημένης Παραγωγής μέσω Ανάκτησης (Retrieval-Augmented Generation, RAG)** προτείνει μια υβριδική προσέγγιση: συνδυάζει τις δυνατότητες γενίκευσης και κατανόησης των LLMs με τη δυναμική ανάκτηση πληροφοριών από εξωτερικές βάσεις γνώσης. Με αυτό τον τρόπο, το μοντέλο δεν περιορίζεται στις παραμέτρους του, αλλά μπορεί να ενισχύσει την παραγωγή κειμένου με επίκαιρες, σχετικές και αξιόπιστες πηγές [57, 58]. Η RAG έχει αναδειχθεί ως κρίσιμη μεθοδολογία για την ενίσχυση της ακρίβειας, τη μείωση φαινομένων παραισθήσεων και την αύξηση της επεξηγησιμότητας των αποτελεσμάτων.

2.6.1 Αρχιτεκτονική του RAG

Η βασική αρχιτεκτονική της Επαυξημένης Παραγωγής μέσω Ανάκτησης (RAG) συνδυάζει δύο διακριτά υποσυστήματα: τον *Ανακτητή* (Retriever) και τον *Γεννήτορα* (Generator). Ο Ανακτητής έχει ως ρόλο την αναζήτηση σχετικών εγγράφων ή αποσπασμάτων από μια εξωτερική βάση γνώσης C , με βάση το ερώτημα q που παρέχεται στο σύστημα. Στη συνέχεια, ο Γεννήτορας (συνήθως ένα LLM αρχιτεκτονικής αποκωδικοποιητή) λαμβάνει ως είσοδο τον συνδυασμό του ερωτήματος με τα ανακτημένα τεκμήρια, παράγοντας την τελική απάντηση [57, 59].

Τυπικά, η διαδικασία μπορεί να περιγραφεί ως εξής. Για κάθε ερώτημα q , ο Ανακτητής υπολογίζει μια πιθανότητα $p(d|q)$ για κάθε τεκμήριο $d \in C$. Η πιθανότητα παραγωγής μιας απάντησης y δίνεται από τον Γεννήτορα ως:

$$P(y | q) = \sum_{d \in C} P(y | q, d) p(d | q) \quad (2.33)$$

όπου $P(y | q, d)$ είναι η πιθανότητα να παραχθεί η απάντηση y δεδομένου του ερωτήματος q και του τεκμηρίου d , και $p(d|q)$ είναι η κατανομή πιθανοτήτων που αποδίδει ο Ανακτητής στο σύνολο των εγγράφων. Η εξίσωση αυτή αποτυπώνει την αρχή του RAG: η τελική γενεσιοναργία δεν εξαρτάται αποκλειστικά από τις παραμέτρους του LLM, αλλά ενισχύεται από την πληροφορία που ανακτάται δυναμικά.

Ο Ανακτητής μπορεί να υλοποιηθεί είτε με αραιές αναπαραστάσεις (sparse retrievers) όπως το BM25 [16], είτε με πυκνές αναπαραστάσεις (dense retrievers) που βασίζονται σε ενσωματώσεις μέσω νευρωνικών δικτύων [17]. Σε πρακτικές εφαρμογές συχνά χρησιμοποιούνται υβριδικές μέθοδοι που συνδυάζουν τα πλεονεκτήματα και των δύο [60]. Ο Γεννήτορας είναι συνήθως ένα LLM τύπου Transformer αποκωδικοποιητή, εκπαιδευμένο για αυτοπαλίνδρομη παραγωγή κειμένου, με δυνατότητα να ενσωματώνει τα ανακτημένα αποσπάσματα ως επιπρόσθετο context.

2.6.2 Βάσεις Δεδομένων Διανυσμάτων (Vector Databases)

Η επιτυχία των συστημάτων RAG εξαρτάται σε μεγάλο βαθμό από την ικανότητα αποθήκευσης και αποδοτικής αναζήτησης μεγάλου όγκου ενσωματώσεων (embeddings). Κάθε τεκμήριο κειμένου $d \in C$ χαρτογραφείται σε ένα διάνυσμα υψηλής διάστασης μέσω ενός προεκπαιδευμένου μοντέλου ενσωμάτωσης. Η αναζήτηση της σχετικότητας μεταξύ ερωτήματος και τεκμηρίων ανάγεται έτσι σε πρόβλημα υπολογισμού ομοιότητας διανυσμάτων σε χώρους με εκατοντάδες διαστάσεις.

Η πιο διαδεδομένη μετρική σε τέτοιες βάσεις είναι η **ομοιότητα συνημιτόνου** (*cosine similarity*). Δοθέντων δύο διανυσμάτων $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$, η ομοιότητα συνημιτόνου ορίζεται ως:

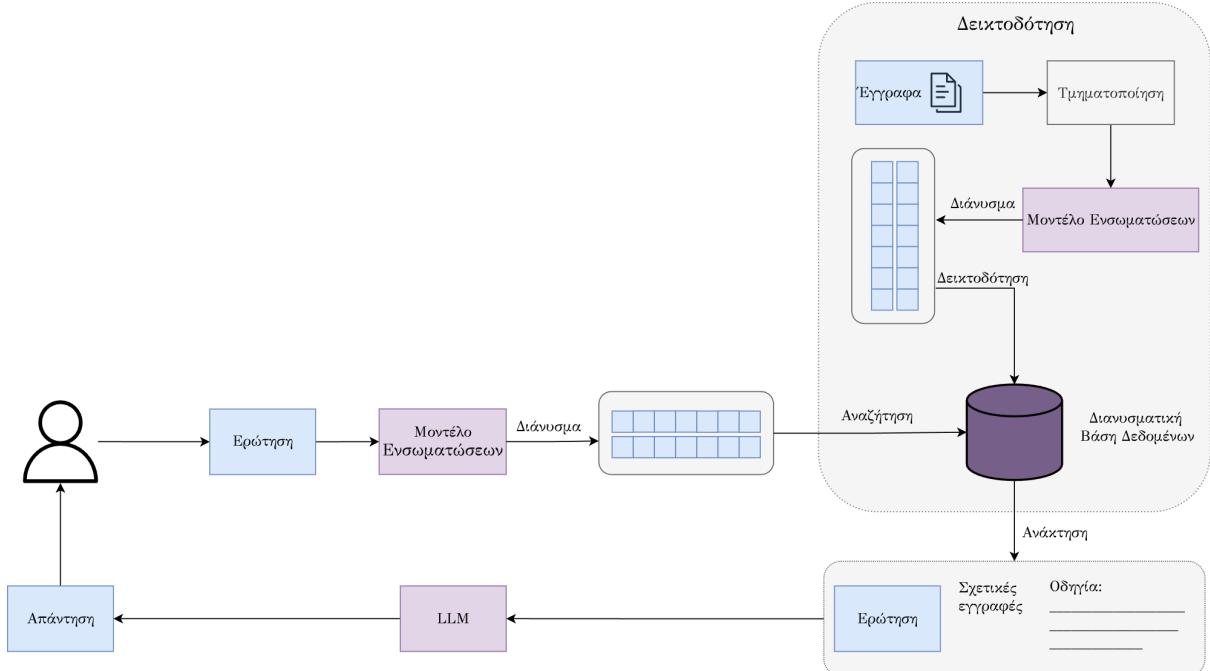
$$\text{sim}_{\cos}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}. \quad (2.34)$$

Η τιμή της sim_{\cos} κυμαίνεται στο $[-1, 1]$, με την τιμή 1 να υποδηλώνει ταυτόσημη κατεύθυνση, 0 ορθογωνιότητα και -1 αντίθετη κατεύθυνση. Στην πράξη χρησιμοποιείται συχνά η **απόσταση συνημιτόνου**:

$$\text{dist}_{\cos}(\mathbf{u}, \mathbf{v}) = 1 - \text{sim}_{\cos}(\mathbf{u}, \mathbf{v}), \quad (2.35)$$

ώστε να εκφράζεται η εγγύτητα ως θετική μετρική. Εφόσον τα embeddings κανονικοποιούνται σε μοναδιαίο μήκος ($\|\mathbf{u}\| = \|\mathbf{v}\| = 1$), η ομοιότητα ισοδυναμεί με το εσωτερικό γινόμενο $\mathbf{u} \cdot \mathbf{v}$, καθιστώντας τον υπολογισμό αποδοτικό και εύκολα υλοποιήσιμο σε κλίμακα [61].

Για την αποδοτική διαχείριση τέτοιων συλλογών χρησιμοποιούνται βάσεις δεδομένων διανυσμάτων (vector databases), όπως FAISS, Milvus και Qdrant. Οι βάσεις αυτές υλοποιούν δομές δεικτοδότησης και αλγορίθμους προσεγγιστικής εύρεσης πλησιέστερου γείτονα *approximate nearest neighbor search (ANN)*, που μειώνουν δραστικά την πολυπλοκότητα αναζήτησης σε σχέση με την εξαντλητική αναζήτηση, επιτρέποντας την ταχεία ανάκτηση ακόμα και σε συλλογές δισεκατομμυρίων τεκμηρίων. Επιπλέον, οι σύγχρονες βάσεις υποστηρίζουν υβριδικούς δείκτες που συνδυάζουν πυκνές και αραιές αναπαραστάσεις, καθιστώντας δυνατή την ταυτόχρονη αξιοποίηση σημασιολογικής και λεκτικής πληροφορίας [60].



Σχήμα 2.10: Αρχιτεκτονική ροή ενός συστήματος Επαυξημένης Παραγωγής μέσω Ανάκτησης (RAG). Η διαδικασία περιλαμβάνει δύο διακριτά στάδια: (i) τη δεικτοδότηση, όπου τα έγγραφα τεμαχίζονται, μετατρέπονται σε διανύσματα μέσω μοντέλου ενσωματώσεων και αποθηκεύονται σε διανυσματική βάση δεδομένων, και (ii) την **online ανάκτηση** και **παραγωγή**, όπου το ερώτημα του χρήστη ενσωματώνεται, αναζητά σχετικές εγγραφές στη βάση και, σε συνδυασμό με τις οδηγίες (prompt), τροφοδοτεί το LLM για την παραγωγή της τελικής απάντησης.

2.6.3 Προκλήσεις και Υπερπαράμετροι στα Συστήματα RAG

Παρότι τα συστήματα RAG βελτιώνουν σημαντικά την ακρίβεια και μειώνουν τις παραισθήσεις, η απόδοσή τους εξαρτάται έντονα από μια σειρά υπερπαραμέτρων και σχεδιαστικών επιλογών. Οι σημαντικότερες προκλήσεις συνοψίζονται παρακάτω:

- **Τμηματοποίηση τεκμηρίων (chunking):** Τα κείμενα διασπώνται σε τμήματα (*chunks*) τα οποία μετατρέπονται σε ενσωμάτωση. Το μέγεθος του chunk (l_{chunk}) επηρεάζει άμεσα την ισορροπία ανάμεσα στη λεπτομέρεια και στη σημασιολογική πληρούτητα. Πολύ μικρά chunks οδηγούν σε απώλεια συμφραζομένων, ενώ πολύ μεγάλα chunks αυξάνουν τον θόρυβο και το κόστος [62].
- **Αριθμός ανακτώμενων τεκμηρίων (top- k):** Ο Ανακτητής επιστρέφει τα k πιο σχετικά τεκμήρια. Η επιλογή του k επηρεάζει την απόδοση: μικρό k μπορεί να οδηγήσει σε απώλεια κρίσιμης πληροφορίας, ενώ πολύ μεγάλο k αυξάνει την καθυστέρηση και εισάγει θόρυβο από άσχετα έγγραφα [63].
- **Ποιότητα Ανακτητή:** Η επιλογή ανάμεσα σε αραιούς, πυκνούς ή υβριδικούς ανακτητές καθορίζει τη σχετικότητα των αποτελεσμάτων. Πυκνοί ανακτητές (π.χ. DPR) αποδίδουν καλύτερα σημασιολογικά, αλλά έχουν υψηλότερο κόστος εκπαίδευσης και συντήρησης [17].
- **Συγχώνευση και επαναβαθμολόγηση (fusion, reranking):** Προηγμένες μέθοδοι όπως το reciprocal rank fusion ή rerankers βασισμένοι σε γλωσσικά μοντέλα (π.χ. MonoT5, ColBERT) βελτιώνουν την ακρίβεια αλλά αυξάνουν το υπολογιστικό φορτίο [64, 65].

Συμπερασματικά, η Επαυξημένη Παραγωγή μέσω Ανάκτησης (RAG) συνιστά μια μεθοδολογική καινοτομία που επιτρέπει στα LLMs να υπερβούν τον περιορισμό της στατικής γνώσης, μέσω της δυναμικής ενσωμάτωσης εξωτερικών τεκμηρίων. Παρά τα σημαντικά της πλεονεκτήματα, η απόδοση της προσέγγισης παραμένει ιδιαίτερα ευαίσθητη σε κρίσιμες υπερπαραμέτρους και σχεδιαστικές επιλογές, όπως η στρατηγική διάσπασης τεκμηρίων, το μέγεθος του top- k κατά την ανάκτηση και η ποιότητα των αλγορίθμων επαναβαθμολόγησης. Οι παράγοντες αυτοί αναδεικνύουν την ανάγκη για συστηματική μελέτη και μεθοδολογίες βελτιστοποίησης που να μπορούν να ανταποκριθούν στις απαιτήσεις εφαρμογών μεγάλης κλίμακας.

Το επόμενο κεφάλαιο επικεντρώνεται στις υφιστάμενες ερευνητικές προκλήσεις που αφορούν την αξιολόγηση, τη βελτιστοποίηση και την κλιμάκωση συστημάτων RAG. Η ανάλυση αυτή αποσκοπεί στην ανάδειξη των περιορισμών των τρεχουσών προσεγγίσεων και στην τεκμηρίωση των ανοιχτών ζητημάτων που χρήζουν περαιτέρω διερεύνησης.

3

Κριτική Ανασκόπηση Μεθοδολογιών Επαυξημένης Παραγωγής μέσω Ανάκτησης

3.1 ΕΙΣΑΓΩΓΗ

Η επαυξημένη παραγωγή μέσω ανάκτησης (Retrieval-Augmented Generation - RAG) αποτελεί μία από τις πλέον σημαντικές καινοτομίες στον τομέα της επεξεργασίας φυσικής γλώσσας, συνδυάζοντας τις δυνατότητες των μεγάλων γλωσσικών μοντέλων με εξωτερικές βάσεις δεδομένων. Η εξέλιξη των μεθοδολογιών RAG διανύει, από την πρώτη τους εμφάνιση το 2020 μέχρι σήμερα, τρεις διακριτές φάσεις ανάπτυξης, οι οποίες αντανακλούν τη συνεχή ωρίμανση του πεδίου και την προοδευτική απάντηση στους περιορισμούς των προγενέστερων προσεγγίσεων. Στο παρόν κεφάλαιο, παρουσιάζεται μια συστηματική ανάλυση των τριών κύριων παραδειγμάτων RAG: του Αρχικού (Naive), του Προηγμένου (Advanced) και του Αρθρωτού (Modular) RAG, εξετάζοντας τα τεχνικά χαρακτηριστικά, τις μεθοδολογικές προσεγγίσεις και τις πρακτικές εφαρμογές τους.

3.2 ΑΡΧΙΚΟ RAG (NAIVE RAG)

3.2.1 Αρχιτεκτονική και Αρχές Λειτουργίας

Το αρχικό παράδειγμα της επαυξημένης παραγωγής μέσω ανάκτησης (Retrieval-Augmented Generation), [10] αποτελεί μια γενικού σκοπού μεθοδολογία (general-purpose fine-tuning recipe) που συνδυάζει προεκπαιδευμένη παραμετρική μνήμη με μη-παραμετρική μνήμη. Η παραμετρική μνήμη υλοποιείται μέσω ενός μοντέλου

«ακολουθία σε ακολουθία» (seq2seq), ενώ η μη-παραμετρική μνήμη αποτελείται από έναν δείκτη πυκνού διανύσματος (dense vector index).

Η διαδικασία ανάκτησης βασίζεται στη χρήση πυκνών κωδικοποιητών περάσματος κειμένου (Dense Passage Retrieval - DPR) [17]. Το ερώτημα του χρήστη μετασχηματίζεται σε διανυσματική αναπαράσταση, επιτρέποντας την εύρεση σημασιολογικά όμοιων τμημάτων κειμένου από μία βάση γνώσης (knowledge base). Η αναζήτηση χρησιμοποιεί τη μετρική ομοιότητας συνημιτόνου (cosine similarity):

$$\text{sim}(q, d_i) = \frac{\mathbf{q} \cdot \mathbf{d}_i}{\|\mathbf{q}\| \|\mathbf{d}_i\|} \quad (3.1)$$

όπου \mathbf{q} είναι η διανυσματική αναπαράσταση του ερωτήματος και \mathbf{d}_i η αντίστοιχη αναπαράσταση του i -οστού εγγράφου. Τα ανακτηθέντα τμήματα συνενώνονται με το αρχικό ερώτημα και τροφοδοτούνται στο γλωσσικό μοντέλο για την παραγωγή της απάντησης.

3.2.2 Περιορισμοί και Προκλήσεις

Η εφαρμογή του αρχικού RAG σε πραγματικά σενάρια αποκάλυψε δομικούς περιορισμούς που επηρεάζουν την αποτελεσματικότητα του συστήματος. Το φαινόμενο της «χαμένης στη μέση» πληροφορίας (lost-in-the-middle phenomenon) τεκμηριώνεται εμπειρικά από τους Liu et al. [5]. Η έρευνα καταδεικνύει την αδυναμία των γλωσσικών μοντέλων να επεξεργαστούν αποτελεσματικά μεγάλα σύνολα ανακτηθέντων εγγράφων, με την απόδοση να υποβαθμίζεται όταν οι σχετικές πληροφορίες τοποθετούνται στα μεσαία τμήματα του συμφραζομένου. Το φαινόμενο αυτό εμφανίζει μια χαρακτηριστική καμπύλη τύπου U (3.1), όπου η ακρίβεια ανάκτησης πληροφορίας είναι υψηλότερη για στοιχεία που βρίσκονται στην αρχή και στο τέλος του συμφραζομένου, ενώ μειώνεται αισθητά για πληροφορίες που βρίσκονται στο μέσο του. Με άλλα λόγια, τα μοντέλα τείνουν να «θυμούνται» καλύτερα τα πρώτα και τα τελευταία tokens, αγνοώντας εν μέρει εκείνα που βρίσκονται στο κεντρικό τμήμα της ακολουθίας. Το φαινόμενο αυτό παρατηρείται ακόμη και σε μοντέλα με εκτεταμένα παράθυρα συμφραζομένων, όπως τα GPT-3.5-Turbo (16K tokens) και Claude-1.3 (100K tokens), υποδηλώνοντας ότι η απλή αύξηση του μήκους του παραθύρου δεν αρκεί για την ομοιόμορφη αξιοποίηση πληροφοριών σε όλο το εύρος του συμφραζομένου.

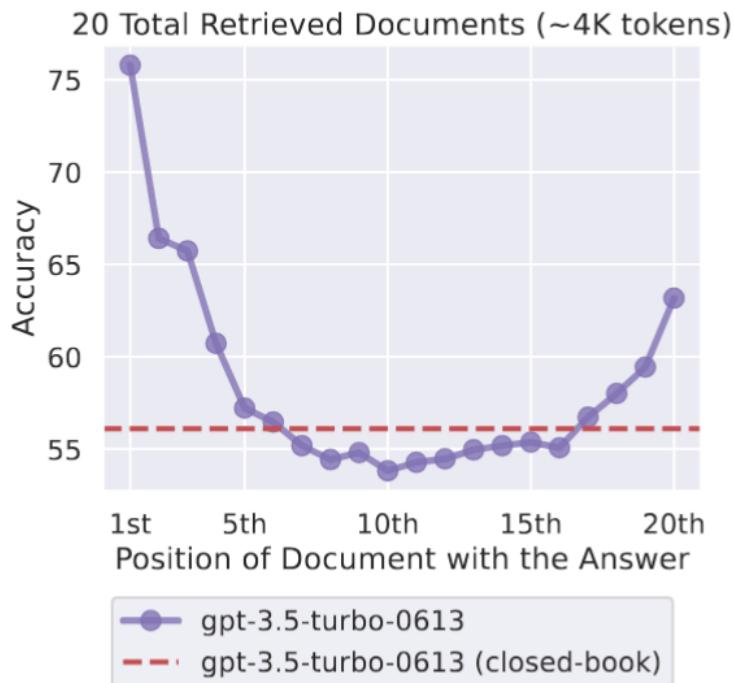
Η απουσία μηχανισμών επαλήθευσης της σχετικότητας των ανακτηθέντων πληροφοριών αποτελεί επιπλέον περιορισμό. Το σύστημα εμπιστεύεται αποκλειστικά τη μετρική ομοιότητας για την επιλογή των εγγράφων, χωρίς περαιτέρω αξιολόγηση της πραγματικής χρησιμότητας ή ακρίβειάς τους. Αυτό οδηγεί σε παραγωγή παραπλανητικών απαντήσεων ή παραισθήσεων (hallucinations), ιδιαίτερα όταν τα ανακτηθέντα έγγραφα περιέχουν θόρυβο ή ασυνεπείς πληροφορίες.

Η στατική φύση της διαδικασίας ανάκτησης συνιστά περαιτέρω περιορισμό. Το σύστημα εκτελεί πάντοτε ανάκτηση εξωτερικών πληροφοριών, ακόμη και όταν το γλωσσικό μοντέλο διαθέτει επαρκή παραμετρική γνώση για να απαντήσει στο ερώτημα. Αυτό προκαλεί περιττή υπολογιστική επιβάρυνση, αύξηση του χρόνου απόκρισης και πιθανή εισαγωγή θορύβου στη διαδικασία παραγωγής απάντησης.

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

Το αρχικό RAG παρουσιάζει αδυναμίες σε περιπτώσεις όπου απαιτείται πολυβηματική συλλογιστική (multi-hop reasoning) ή σύνθεση πληροφοριών από πολλαπλές διαφορετικές πηγές. Η φύση της αρχιτεκτονικής δεν επιτρέπει την επαναληπτική βελτίωση της ανάκτησης βάσει των ενδιάμεσων αποτελεσμάτων παραγωγής. Οι περιορισμοί αυτοί οδήγησαν στην ανάπτυξη προηγμένων παραδειγμάτων RAG που αντιμετωπίζουν συστηματικά τις εντοπισμένες αδυναμίες.

Παρά τους περιορισμούς του, το αρχικό RAG καθιέρωσε την αρχή της ενσωμάτωσης εξωτερικής γνώσης στη διαδικασία παραγωγής κειμένου, θέτοντας τις βάσεις για τις μεταγενέστερες εξελίξεις στον τομέα.



Σχήμα 3.1: Τα γλωσσικά μοντέλα εμφανίζουν μεροληφία θέσης (U-shaped performance): βρίσκουν ευκολότερα τις πληροφορίες όταν βρίσκονται στην αρχή ή στο τέλος του κειμένου, ενώ η απόδοσή τους πέφτει σημαντικά όταν η απάντηση βρίσκεται στη μέση [5].

3.3 ΠΡΟΗΓΜΕΝΟ RAG (ADVANCED RAG)

3.3.1 Βελτιστοποιήσεις στη Διαδικασία Ανάκτησης

Η μετάβαση από το αρχικό στο προηγμένο RAG χαρακτηρίζεται από την εισαγωγή εξελιγμένων τεχνικών βελτιστοποίησης που στοχεύουν στην ενίσχυση της αποτελεσματικότητας του συστήματος σε όλα τα στάδια της διαδικασίας επεξεργασίας. Αυτές οι τεχνικές οργανώνονται σε δύο κύριες κατηγορίες: τις βελτιστοποιήσεις προ-ανάκτησης που επικεντρώνονται στη βελτίωση του ερωτήματος και

3.3. ΠΡΟΗΓΜΕΝΟ RAG (ADVANCED RAG)

της δομής των δεδομένων πριν την πραγματοποίηση της αναζήτησης, και τις βελτιστοποιήσεις κατά την ανάκτηση που αφορούν τη στρατηγική και τη μεθοδολογία εύρεσης των σχετικών εγγράφων.

Στο επίπεδο της προ-ανάκτησης, η επανεγγραφή ερωτημάτων (query rewriting) αποτελεί μία από τις θεμελιώδεις τεχνικές που χρησιμοποιούν εξειδικευμένα γλωσσικά μοντέλα για τη δημιουργία πολλαπλών παραλλαγών του αρχικού ερωτήματος του χρήστη. Η προσέγγιση αυτή, όπως περιγράφεται από τους Xinbei Ma et al. [66], αυξάνει την πιθανότητα ανάκτησης σχετικών εγγράφων μέσω της διεύρυνσης του σημασιολογικού χώρου αναζήτησης, επιτρέποντας στο σύστημα να αντιμετωπίσει φαινόμενα όπως η λεξική ασυμφωνία μεταξύ ερωτήματος και εγγράφων, η διφορούμενη ορολογία, και οι υποκείμενες προθέσεις του χρήστη που δεν εκφράζονται άμεσα στο αρχικό ερώτημα. Η επανεγγραφή μπορεί να περιλαμβάνει την επέκταση του ερωτήματος με συνώνυμα και σχετικούς όρους, την αναδιατύπωσή του σε διαφορετικές γλωσσικές δομές, ή την αποσαφήνιση ασαφών ή ελλιπτικών διατυπώσεων.

Παράλληλα με την επανεγγραφή, η τεχνική HyDE (Hypothetical Document Embeddings) [62] εισάγει μια ριζικά διαφορετική φιλοσοφία στην επέκταση ερωτημάτων. Αυτή να αναδιατυπώνει το ερώτημα, η μέθοδος χρησιμοποιεί ένα γλωσσικό μοντέλο για να παράγει ένα υποθετικό έγγραφο που θα μπορούσε να αποτελεί απάντηση στο ερώτημα του χρήστη. Στη συνέχεια, το σύστημα υπολογίζει την ενσωμάτωση αυτού του υποθετικού εγγράφου και το χρησιμοποιεί ως βάση για την ανάκτηση από τη συλλογή. Η θεμελιώδης διαίσθηση πίσω από την προσέγγιση βασίζεται στην παρατήρηση ότι οι ενσωματώσεις των εγγράφων τείνουν να είναι σημασιολογικά πιο όμοιες μεταξύ τους από ότι οι ενσωματώσεις των ερωτημάτων με τις ενσωματώσεις των εγγράφων. Επομένως, η χρήση ενός υποθετικού εγγράφου ως μεσάζοντα μειώνει το σημασιολογικό χάσμα και βελτιώνει την ανάκληση. Παρότι το υποθετικό έγγραφο μπορεί να περιέχει ανακρίβειες ή παραισθήσεις, η συνολική διαδικασία αποδεικνύεται αποτελεσματική επειδή το σύστημα ανάκτησης αναζητά πραγματικά έγγραφα με παρόμοιο σημασιολογικό περιεχόμενο, όχι ακριβείς αντιστοιχίες.

Μία άλλη τεχνική για σύνθετα ερωτήματα που απαιτούν συλλογισμό πολλαπλών βημάτων είναι η αποσύνθεση ερωτημάτων (query decomposition). Αποτελεί κρίσιμη τεχνική που διασπά ένα ερώτημα πολλαπλού βήματος (multi-hop) σε ακολουθία απλούστερων ερωτημάτων μονού βήματος. Η προσέγγιση αυτή, εμπνευσμένη από το πλαίσιο StrategyQA, επιτρέπει στο σύστημα να ανακτά πληροφορίες σταδιακά, κατασκευάζοντας την τελική απάντηση μέσω αλυσιδωτού συλλογισμού. Για παράδειγμα, ένα ερώτημα όπως «Ποια είναι η πρωτεύουσα της χώρας όπου γεννήθηκε ο εφευρέτης του τηλεφώνου;» αποσυντίθεται στα επιμέρους ερωτήματα: «Ποιος ήταν ο εφευρέτης του τηλεφώνου;», «Πού γεννήθηκε αυτό το άτομο;», και «Ποια είναι η πρωτεύουσα αυτής της χώρας;». Κάθε υπο-ερώτημα απευθύνεται στο σύστημα ανάκτησης ξεχωριστά, και οι απαντήσεις συντίθενται για την παραγωγή της τελικής απόκρισης.

Πέρα από τη βελτιστοποίηση των ερωτημάτων, η υβριδική αναζήτηση (hybrid search) αντιπροσωπεύει μια ουσιώδη εξέλιξη στο επίπεδο της ανάκτησης, συνδυάζοντας τα πλεονεκτήματα πυκνών (dense) και αραιών (sparse) μεθόδων. Οι πυκνές μέθοδοι, που βασίζονται σε διανυσματικές αναπαραστάσεις υψηλής διάστασης παραγόμενες από νευρωνικά δίκτυα, υπερέχουν στη σύλληψη της σημασιολογικής

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

ομοιότητας και μπορούν να αναγνωρίσουν σχετικά έγγραφα ακόμα και όταν δεν υπάρχει λεξική επικάλυψη με το ερώτημα. Οι αραιές μέθοδοι, όπως το κλασικό BM25, διατηρούν ισχυρή απόδοση στην ακριβή λεξική αντιστοίχιση και είναι ιδιαίτερα αποτελεσματικές όταν τα ερωτήματα περιέχουν εξειδικευμένη ορολογία, ονόματα οντοτήτων, ή τεχνικούς όρους που πρέπει να αντιστοιχιστούν με ακρίβεια. Ο συνδυασμός των δύο προσεγγίσεων επιτυγχάνει σημαντική βελτίωση της ανάκλησης σε σχέση με τη χρήση μεμονωμένων μεθόδων.

Η μαθηματική θεμελίωση της υβριδικής βαθμολόγησης μπορεί να επιτευχθεί με διάφορους τρόπους. Μία απλή προσέγγιση είναι η γραμμική συνδυασμός κανονικοποιημένων βαθμολογιών:

$$S_{\text{hybrid}} = (1 - \alpha) \cdot \text{norm}(S_{\text{BM25}}) + \alpha \cdot \text{norm}(S_{\text{vector}}) \quad (3.2)$$

όπου $\alpha \in [0, 1]$ ελέγχει την ισορροπία μεταξύ λεξιλογικής και σημασιολογικής αντιστοίχισης, ενώ η συνάρτηση $\text{norm}(\cdot)$ εξασφαλίζει την κανονικοποίηση των βαθμολογιών σε συγκρίσιμο εύρος τιμών. Ωστόσο, η κανονικοποίηση βαθμολογιών από διαφορετικές μεθόδους μπορεί να εισάγει προκαταλήψεις (biases) και να είναι υπολογιστικά απαιτητική.

Μια καλύτερη εναλλακτική προσέγγιση είναι ο αλγόριθμος Reciprocal Rank Fusion (RRF) [67], ο οποίος αποφεύγει εντελώς την ανάγκη κανονικοποίησης, αφού βασίζεται αποκλειστικά στις σχετικές θέσεις (ranks) των εγγράφων σε κάθε λίστα αποτελεσμάτων. Η μέθοδος ορίζεται ως:

$$\text{RRF}(d) = \sum_{r \in R} \frac{1}{k + \text{rank}_r(d)} \quad (3.3)$$

όπου $\text{rank}_r(d)$ είναι η θέση του εγγράφου d στη λίστα αποτελεσμάτων r και k είναι μια σταθερά που συνήθως λαμβάνει την τιμή 60. Η παράμετρος k εξυπηρετεί διπλό σκοπό: αποτρέπει την κυριαρχία των πρώτων θέσεων και εξασφαλίζει ότι η συνεισφορά κάθε λίστας μειώνεται σταδιακά με τη θέση. Το RRF έχει αποδειχθεί αποτελεσματικό σε ένα ευρύ φάσμα σεναρίων και είναι ιδιαίτερα ανθεκτικό στις διακυμάνσεις της ποιότητας των επιμέρους μεθόδων ανάκτησης.

Παράλληλα με τον συνδυασμό μεθόδων, η εισαγωγή προσαρμοστικών στρατηγικών ανάκτησης επιτρέπει στο σύστημα να προσαρμόζει δυναμικά παραμέτρους όπως ο αριθμός των ανακτηθέντων εγγράφων ανάλογα με την πολυπλοκότητα του ερωτήματος, τη διαθεσιμότητα σχετικών πληροφοριών στη βάση γνώσης, και την εμπιστοσύνη του συστήματος στην ποιότητα των αρχικών αποτελεσμάτων. Αυτή η δυναμική προσαρμογή αποτρέπει τόσο την υποφόρτωση με ανεπαρκείς πληροφορίες όσο και την υπερφόρτωση με θόρυβο και μη σχετικό περιεχόμενο.

Αλγόριθμος 3.1 Συγχώνευση Κατατάξεων με Reciprocal Rank Fusion (RRF)

Require: Λίστες κατάταξης από M διαφορετικούς ανακτητές: $\mathcal{R} = \{R_1, \dots, R_M\}$, παράμετρος $k > 0$ (συνήθως $k = 60$), προαιρετικά επιθυμητό top- K .

Ensure: Μία ενιαία, συγχωνευμένη λίστα εγγράφων S .

```

1:  $U \leftarrow \bigcup_{m=1}^M \{d \mid d \in R_m\}$  # Όλα τα έγγραφα που εμφανίστηκαν
2: Για κάθε  $d \in U$ : score[ $d$ ]  $\leftarrow 0$ 
3: for  $m = 1$  έως  $M$  do
4:   for  $r = 1$  έως  $|R_m|$  do
5:      $d \leftarrow R_m[r]$  # Το έγγραφο στη θέση  $r$  του ανακτητή  $m$ 
6:     score[ $d$ ]  $\leftarrow \text{score}[d] + \frac{1}{k+r}$ 
7:   end for
8: end for
9:  $S \leftarrow$  ταξινόμησε τα στοιχεία του  $U$  κατά φθίνουσα score[ $d$ ]
10: if δόθηκε  $K$  then
11:    $S \leftarrow$  κράτα τα πρώτα  $K$  στοιχεία του  $S$ 
12: end if
13: return  $S$ 
```

3.3.2 Μηχανισμοί Επεξεργασίας και Βελτίωσης Ανακτηθέντων Εγγράφων

Μετά την αρχική ανάκτηση, τα συστήματα προηγμένου RAG εφαρμόζουν μια σειρά τεχνικών μετα-επεξεργασίας που στοχεύουν στη βελτίωση της ποιότητας και της συνάφειας των εγγράφων που τελικά τροφοδοτούνται στο γλωσσικό μοντέλο. Αυτές οι τεχνικές αντιμετωπίζουν τρία θεμελιώδη προβλήματα: την ανακριβή αρχική κατάταξη που μπορεί να τοποθετεί λιγότερο σχετικά έγγραφα σε προεξέχουσες θέσεις, την παρουσία πλεονάζουσας ή μη σχετικής πληροφορίας που εισάγει θόρυβο, και την υπερβολική μακροσκέλεια του πλαισίου που επιβαρύνει το γλωσσικό μοντέλο και αυξάνει το υπολογιστικό κόστος.

Η επανακατάταξη (reranking) αποτελεί την πρώτη γραμμή άμυνας κατά του προβλήματος της ανακριβούς αρχικής κατάταξης. Οι διασταυρούμενοι κωδικοποιητές (cross-encoders), όπως περιγράφονται από τους Nogueira et al. [68], επεξεργάζονται το ερώτημα και κάθε υποφήφιο έγγραφο από κοινού, επιτρέποντας στο μοντέλο να αξιολογήσει τη σχετικότητά τους μέσω πλήρους διασταυρωμένης προσοχής (cross-attention) μεταξύ όλων των tokens. Αυτή η προσέγγιση αντιτίθεται στους παραδοσιακούς bi-encoders που κωδικοποιούν το ερώτημα και το έγγραφο ανεξάρτητα και στη συνέχεια υπολογίζουν την ομοιότητα των ενσωματώσεών τους. Η βαθύτερη σημασιολογική κατανόηση που προσφέρουν οι διασταυρωμένοι κωδικοποιητές οδηγεί σε σημαντικά ανώτερη απόδοση. Συγκεκριμένα, η εφαρμογή επανακατάταξης διασταυρωμένου κωδικοποιητή βασισμένου σε BERT στο σύνολο δεδομένων MS MARCO επέτυχε Mean Reciprocal Rank στη δεκάδα (MRR@10) περίπου 36.5 τοις εκατό για το BERTlarge μοντέλο, υπερβαίνοντας σημαντικά την απόδοση προηγούμενων μεθόδων. Το υπολογιστικό κόστος των διασταυρωμένων κωδικοποιητών είναι σημαντικά υψηλότερο από τους bi-encoders, καθώς απαιτεί-

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

ται ξεχωριστή επεξεργασία για κάθε ζεύγος ερωτήματος-εγγράφου, αλλά αυτό το κόστος είναι αποδεκτό για την επανακατάταξη ενός περιορισμένου συνόλου υποψηφίων που έχουν ήδη προεπιλεγεί από ταχύτερες μεθόδους πρώτου σταδίου.

Πέρα από την επανακατάταξη, η δυναμική επιλογή του αριθμού των εγγράφων που θα διατηρηθούν αποτελεί κρίσιμη απόφαση που επηρεάζει τόσο την ποιότητα όσο και την αποδοτικότητα του συστήματος. Οι στατικές προσεγγίσεις που διατηρούν σταθερό αριθμό εγγράφων (π.χ., πάντα τα πρώτα k) δεν λαμβάνουν υπόψη τη σημαντική διακύμανση στην ποιότητα και τη διαθεσιμότητα σχετικών πληροφοριών μεταξύ διαφορετικών ερωτημάτων. Ένα ερώτημα για το οποίο υπάρχουν πολλά άρτια σχετικά έγγραφα μπορεί να επωφεληθεί από τη διατήρηση περισσότερων αποτελεσμάτων, ενώ ένα ερώτημα με λίγα οριακά σχετικά έγγραφα θα έπρεπε να περιοριστεί σε μικρότερο σύνολο για την αποφυγή θορύβου. Οι δυναμικοί μηχανισμοί φιλτραρίσματος χρησιμοποιούν κατώφλια βαθμολογίας σχετικότητας, αναλύουν τη διασπορά των βαθμολογιών, ή εφαρμόζουν μοντέλα αξιολόγησης σχετικότητας για να καθορίσουν προσαρμοστικά το βέλτιστο σύνολο εγγράφων για κάθε ερώτημα.

Μια συχνά παραβλεπόμενη αλλά ουσιώδης τεχνική είναι η επαύξηση αποσπασμάτων (passage augmentation), η οποία αντιμετωπίζει το πρόβλημα της αποκοπής σημασιολογικού πλαισίου που προκύπτει από τη διαδικασία της τμηματοποίησης (chunking). Όταν ένα μεγάλο έγγραφο διαιρείται σε μικρότερα αποσπάσματα για την αποδοτικότερη ανάκτηση, συχνά χάνονται σημαντικές πληροφορίες που βρίσκονται ακριβώς πριν ή μετά τα όρια του αποσπάσματος. Η τεχνική prev-next augmentation εμπλουτίζει κάθε ανακτηθέν απόσπασμα προσθέτοντας το αμέσως προηγούμενο και επόμενο τμήμα από το πηγαίο έγγραφο, διατηρώντας έτσι τη σημασιολογική συνοχή και παρέχοντας επαρκές πλαίσιο για την κατανόηση. Αυτή η απλή αλλά αποτελεσματική στρατηγική μπορεί να βελτιώσει σημαντικά την ποιότητα των παραγόμενων απαντήσεων χωρίς να απαιτεί περίπλοκες αλλαγές στην αρχιτεκτονική του συστήματος.

Η συμπίεση πλαισίου (context compression) αντιπροσωπεύει μια διαφορετική φιλοσοφία που αντί να επιλέγει ποια έγγραφα να διατηρήσει, εστιάζει στην εξάλειψη μη ουσιωδών πληροφοριών εντός των επιλεγμένων εγγράφων. Η υπερφόρτωση με πληροφορίες μπορεί να επηρεάσει αρνητικά την απόδοση των γλωσσικών μοντέλων μέσω διαφόρων μηχανισμών: η παρουσία μεγάλου όγκου μη σχετικών πληροφοριών μπορεί να αποσπάσει την προσοχή του μοντέλου από τις πραγματικά σημαντικές πληροφορίες, το φαινόμενο "lost in the middle" οδηγεί τα μοντέλα να εστιάζουν δυσανάλογα στην αρχή και το τέλος μακρών πλαισίων παραβλέποντας κρίσιμες πληροφορίες στη μέση, και το αυξημένο πλήθος tokens οδηγεί σε υψηλότερο υπολογιστικό και οικονομικό κόστος. Το πλαίσιο LLMLingua [69] και οι επεκτάσεις του χρησιμοποιούν μικρά γλωσσικά μοντέλα, όπως τα GPT-2 Small ή LLaMA-7B, για να εντοπίσουν και να αφαιρέσουν μη σημαντικά tokens από το πλαίσιο. Η μέθοδος επιτυγχάνει compression ratios που κυμαίνονται από 2x έως 10x χωρίς σημαντική απώλεια πληροφορίας, μετασχηματίζοντας το πλαίσιο σε μια μορφή που μπορεί να φαίνεται δυσανάγνωστη για ανθρώπους αλλά παραμένει κατανοητή από τα μεγάλα γλωσσικά μοντέλα. Εναλλακτικές προσεγγίσεις όπως το RECOMP εκπαιδεύουν εξειδικευμένα μοντέλα συμπύκνωσης πληροφοριών (information condensers) μέσω αντικρουόμενης μάθησης (contrastive learning), όπου

το μοντέλο μαθαίνει να διατηρεί τις ουσιώδεις πληροφορίες ενώ απορρίπτει τον θόρυβο συγκρίνοντας θετικά παραδείγματα σχετικών αποσπασμάτων με αρνητικά παραδείγματα μη σχετικών.

3.3.3 Προσαρμοστικές Στρατηγικές Ανάκτησης

Οι προσαρμοστικές στρατηγικές ανάκτησης αντιπροσωπεύουν μια ριζική αλλαγή παραδείγματος στη φιλοσοφία του RAG, μετακινώντας την ευθύνη της απόφασης για το πότε και πώς να ανακτηθούν πληροφορίες από το στατικό σχεδιασμό του συστήματος στη δυναμική κρίση του γλωσσικού μοντέλου. Αντί να ακολουθούν μια σταθερή διαδικασία ανάκτησης για κάθε ερώτημα, αυτά τα συστήματα αξιολογούν συνεχώς την ανάγκη για εξωτερικές πληροφορίες και προσαρμόζουν τη συμπεριφορά τους ανάλογα.

Το πλαίσιο Self-RAG [70] υλοποιεί αυτήν την ιδέα μέσω της εισαγωγής ειδικών token στοχασμού (reflection tokens) που λειτουργούν ως μετα-γνωστικοί μηχανισμοί επιτρέποντας στο μοντέλο να αναστοχαστεί πάνω στην ποιότητα και την αναγκαιότητα της ανάκτησης. Τα tokens στοχασμού οργανώνονται σε δύο κατηγορίες: τα "tokens ανάκτησης" που καθορίζουν πότε το σύστημα πρέπει να ενεργοποιήσει την ανάκτηση, και τα "tokens κριτικής" που αξιολογούν τη σχετικότητα των ανακτηθέντων αποσπασμάτων και την ποιότητα των παραγόμενων απαντήσεων. Κατά τη διάρκεια της παραγωγής, το μοντέλο μπορεί να αποφασίσει αυτόνομα να ενεργοποιήσει την ανάκτηση όταν εντοπίζει κενά στη γνώση του ή χαμηλή εμπιστοσύνη στις παραγόμενες πληροφορίες. Μετά την ανάκτηση, το μοντέλο αξιολογεί κριτικά κάθε ανακτηθέν απόσπασμα για τη σχετικότητά του με το ερώτημα και χρησιμοποιεί αυτήν την αξιολόγηση για να καθοδηγήσει τη διαδικασία παραγωγής. Η προσέγγιση αυτή επιτρέπει στο σύστημα να εκτελεί αναζήτηση δέσμης σε επίπεδο αποσπασμάτων (fragment-level beam search) πάνω από πολλαπλά αποσπάσματα, επιλέγοντας τη συνεκτικότερη ακολουθία παραγωγής. Οι βαθμολογίες κριτικής μπορούν να ενημερώνουν τις βαθμολογίες των υπο-ακολουθιών, με τη δυνατότητα προσαρμογής των βαρών κατά το στάδιο απόφασης για την εξατομίκευση της συμπεριφοράς του μοντέλου.

Το πλαίσιο FLARE (Active Retrieval Augmented Generation) [71] υιοθετεί μια διαφορετική στρατηγική βασιζόμενη στην παρακολούθηση της εμπιστοσύνης του μοντέλου κατά την παραγωγή. Το σύστημα υπολογίζει συνεχώς την πιθανότητα των επόμενων tokens που παράγει το γλωσσικό μοντέλο, χρησιμοποιώντας αυτές τις πιθανότητες ως δείκτες εμπιστοσύνης. Όταν η εμπιστοσύνη πέφτει κάτω από ένα προκαθορισμένο κατώφλι, υποδεικνύοντας αβεβαιότητα ή έλλειψη γνώσης, το FLARE ενεργοποιεί αυτόματα την ανάκτηση για να αναζητήσει πρόσθετες πληροφορίες που θα υποστηρίξουν την παραγωγή. Αυτή η προληπτική στρατηγική εξασφαλίζει ότι το σύστημα αναζητά βοήθεια ακριβώς όταν τη χρειάζεται, αποφεύγοντας τόσο την περιττή ανάκτηση για απλά ερωτήματα που το μοντέλο μπορεί να απαντήσει από τη γνώση του, όσο και την ανεπαρκή ανάκτηση για σύνθετα ερωτήματα που απαιτούν εξωτερική τεκμηρίωση.

Οι προσαρμοστικές στρατηγικές εντάσσονται στη γενικότερη τάση των αυτόνομων συστημάτων που χρησιμοποιούν γλωσσικά μοντέλα ως πράκτορες με δυνατότητα ενεργού κρίσης και χρήσης εργαλείων. Παρόμοια συστήματα όπως το WebGPT

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

ενσωματώνουν πλαίσια ενισχυτικής μάθησης για να εκπαιδεύσουν το μοντέλο να χρησιμοποιεί αυτόνομα μηχανές αναζήτησης κατά τη διάρκεια της παραγωγής κειμένου, αυτοματοποιώντας την πλοήγηση μέσω ειδικών tokens που διευκολύνουν ενέργειες όπως η υποβολή ερωτημάτων, η περιήγηση αποτελεσμάτων, και η παραπομπή σε πηγές. Αυτή η μετάβαση από στατικούς προ-σχεδιασμένους αγωγούς σε δυναμικά προσαρμοστικά συστήματα αντιπροσωπεύει την εξέλιξη του RAG από ένα παθητικό εργαλείο ανάκτησης σε ένα ενεργό σύστημα λογικής που μπορεί να αξιολογεί τις πληροφοριακές του ανάγκες και να ενεργεί αναλόγως.

3.4 Αρθρωτό RAG (MODULAR RAG)

3.4.1 Ιεραρχική Ανάκτηση και το Παράδειγμα RAPTOR

Η ιεραρχική οργάνωση της πληροφορίας αποτελεί θεμελιώδη καινοτομία στο πλαίσιο του Αρθρωτού RAG, επιτρέποντας την ανάκτηση σε πολλαπλά επίπεδα αφαίρεσης. Το σύστημα RAPTOR (Recursive Abstractive Processing for Tree-Organized Retrieval) [72] εισάγει μία ριζικά διαφορετική προσέγγιση στην οργάνωση και ανάκτηση πληροφοριών από εκτενή έγγραφα.

Η αρχιτεκτονική του RAPTOR βασίζεται στην αναδρομική κατασκευή δενδρικής δομής μέσω τριών βασικών λειτουργιών: της ενσωμάτωσης (embedding), της ομαδοποίησης (clustering) και της αφαιρετικής περίληψης (abstractive summarization). Η διαδικασία ξεκινά από την τμηματοποίηση του εγγράφου σε τμήματα σταθερού μήκους (συνήθως 100 tokens), τα οποία αποτελούν τα φύλλα του δέντρου. Στη συνέχεια, εφαρμόζεται επαναληπτικά η ακόλουθη διαδικασία:

- 1. Διανυσματική Αναπαράσταση:** Κάθε τμήμα κειμένου μετασχηματίζεται σε πυκνή διανυσματική αναπαράσταση μέσω προεκπαιδευμένων ακόλουθων μοντέλων.
- 2. Ομαδοποίηση:** Εφαρμόζεται αλγόριθμος Γκαουσιανού μοντέλου μίξης (Gaussian Mixture Model, GMM) για την ταυτοποίηση ομάδων σημασιολογικά συσχετισμένων τμημάτων.
- 3. Αφαιρετική Περίληψη:** Κάθε ομάδα συνοφίζεται από γλωσσικό μοντέλο, παράγοντας ένα νέο κείμενο που αποτελεί τον γονικό κόμβο της ομάδας.

Η διαδικασία επαναλαμβάνεται αναδρομικά, δημιουργώντας επίπεδα αυξανόμενης αφαίρεσης, μέχρι να καταστεί αδύνατη η περαιτέρω ομαδοποίηση. Το αποτέλεσμα είναι μία ιεραρχική δομή όπου:

- Οι κόμβοι-φύλλα περιέχουν το πρωτότυπο, λεπτομερές περιεχόμενο
- Οι ενδιάμεσοι κόμβοι αποθηκεύουν περιλήψεις μεσαίου επιπέδου
- Η ρίζα αντιπροσωπεύει μία ολιστική, υψηλού επιπέδου περίληψη του εγγράφου

Κατά τη διαδικασία ανάκτησης, το σύστημα υπολογίζει τη σημασιολογική ομοιότητα μεταξύ του ερωτήματος και όλων των κόμβων του δέντρου (collapsed tree retrieval). Αυτή η προσέγγιση επιτρέπει στο σύστημα να ανακτά πληροφορίες σε διαφορετικά επίπεδα λεπτομέρειας ανάλογα με τη φύση του ερωτήματος:

$$\text{sim}(q, n_i) = \cos(\mathbf{E}_q, \mathbf{E}_{n_i}) \quad (3.4)$$

όπου \mathbf{E}_q είναι η ενσωμάτωση του ερωτήματος και \mathbf{E}_{n_i} η ενσωμάτωση του i-οστού κόμβου.

Πειραματικά αποτελέσματα επιδεικνύουν σημαντικές βελτιώσεις σε καθήκοντα που απαιτούν σύνθετη, πολυβηματική συλλογιστική. Στο benchmark QASPER, το RAPTOR σε συνδυασμό με το GPT-4 επιτυγχάνει F1-score 55.7%, υπερβαίνοντας σημαντικά τις παραδοσιακές μεθόδους ανάκτησης όπως το DPR (53.0%) και τη χρήση μόνο των τίτλων και περιλήψεων (22.2%). Στο QuALITY benchmark, η απόλυτη ακρίβεια βελτιώνεται κατά 20% σε σχέση με τις κορυφαίες προηγούμενες μεθόδους.

Η αρχιτεκτονική του RAPTOR αντιμετωπίζει αποτελεσματικά τον θεμελιώδη περιορισμό των παραδοσιακών RAG συστημάτων που ανακτούν μόνο σύντομα, συνεχόμενα τμήματα κειμένου. Παρέχοντας πρόσβαση σε πολλαπλά επίπεδα πληροφορίας - από λεπτομερείς αναφορές μέχρι ολιστικές περιλήψεις - το σύστημα διευκολύνει τη βαθύτερη κατανόηση και σύνθεση πληροφοριών από εκτενή έγγραφα. Ωστόσο, η προσέγγιση παρουσιάζει σημαντικό υπολογιστικό κόστος κατά την κατασκευή του δέντρου και απαιτεί ιδιαίτερη προσοχή στην επιλογή των παραμέτρων ομαδοποίησης και των prompts περίληψης για να διασφαλιστεί η ποιότητα των παραγόμενων συνόψεων.

3.4.2 Πολυβηματική Συλλογιστική και Λογική Ανάκτηση

Η ανάπτυξη συστημάτων πολυβηματικής συλλογιστικής στο πλαίσιο του αρθρωτού RAG αντιμετωπίζει την πρόκληση της σύνθετης ερωταπόκρισης που απαιτεί συνδυασμό πληροφοριών από πολλαπλές πηγές. Το σύστημα HopRAG [73] κατασκευάζει δυναμικά γραφήματα αποσπασμάτων όπου:

- Οι κόμβοι αντιπροσωπεύουν τμήματα κειμένου (text chunks)
- Οι ακμές δημιουργούνται μέσω ψεύδο-ερωτημάτων παραγόμενα από μεγάλα γλωσσικά μοντέλα, που συνδέουν σημασιολογικά συσχετισμένα τμήματα
- Η διάσχιση του γραφήματος ακολουθεί τη λογική ανέκτησε-συλλογίσου-αφαίρεσε (retrieve-reason-prune)

Η διαδικασία πολυβηματικής συλλογιστικής μοντελοποιείται ως πρόβλημα βέλτιστης διαδρομής στο γράφημα:

$$P^* = \arg \max_{P \in \mathcal{P}} \prod_{(v_i, v_j) \in P} w(v_i, v_j) \cdot r(v_j, q) \quad (3.5)$$

όπου P είναι μια διαδρομή στο γράφημα, $w(v_i, v_j)$ το βάρος της ακμής και $r(v_j, q)$ η σχετικότητα του κόμβου v_j με το ερώτημα q .

3.4.3 Μετρικές Ανάκτησης Πληροφοριών

Η αξιολόγηση της απόδοσης των συστημάτων RAG απαιτεί τη χρήση εξειδικευμένων μετρικών που καλύπτουν τόσο την ποιότητα ανάκτησης όσο και την ακρίβεια παραγωγής. Οι κλασσικές μετρικές ανάκτησης πληροφοριών, οι οποίες έχουν αναπτυχθεί και επικυρωθεί στο ευρύτερο πεδίο της Ανάκτησης Πληροφοριών (Information Retrieval), αποτελούν τη θεμελιώδη βάση για την αξιολόγηση του ανακτητή στα σύγχρονα συστήματα RAG.

Η **Ακρίβεια (Precision)** και η **Ανάκληση (Recall)** αποτελούν θεμελιώδεις μετρικές που ποσοτικοποιούν την ικανότητα του συστήματος ανάκτησης να εντοπίζει σχετικά έγγραφα από μια συλλογή. Η ακρίβεια εκφράζει το κλάσμα των ανακτηθέντων εγγράφων που είναι πραγματικά σχετικά με το ερώτημα του χρήστη, ενώ η ανάκληση εκφράζει το κλάσμα των συνολικά σχετικών εγγράφων που επιτυχώς ανακτήθηκαν από το σύστημα. Μαθηματικά, οι μετρικές αυτές ορίζονται ως:

$$\text{Precision}@k = \frac{|\text{Relevant} \cap \text{Retrieved}_k|}{k} \quad (3.6)$$

$$\text{Recall}@k = \frac{|\text{Relevant} \cap \text{Retrieved}_k|}{|\text{Relevant}|} \quad (3.7)$$

όπου Retrieved_k αντιπροσωπεύει το σύνολο των πρώτων k ανακτηθέντων εγγράφων και Relevant το σύνολο όλων των σχετικών εγγράφων στη συλλογή. Ο αρμονικός μέσος των δύο μετρικών δίνει το **F1-Score**, το οποίο προσφέρει μια ισορροπημένη εκτίμηση της απόδοσης, αποφεύγοντας ακραίες τιμές σε μία μόνο διάσταση:

$$\text{F1}@k = 2 \cdot \frac{\text{Precision}@k \cdot \text{Recall}@k}{\text{Precision}@k + \text{Recall}@k} \quad (3.8)$$

Η μετρική **Success@k** (ή **Hit Rate@k**) αξιολογεί το ποσοστό των ερωτημάτων για τα οποία τουλάχιστον ένα σχετικό έγγραφο περιλαμβάνεται στα πρώτα k ανακτηθέντα αποτελέσματα. Σε αντίθεση με τις μετρικές ακρίβειας και ανάκλησης που αποτυπώνουν ποσοτικά τον βαθμό συνάφειας, η Success@k εστιάζει στη δυαδική επιτυχία της ανάκτησης ανά ερώτημα, παρέχοντας μια καθαρή ένδειξη για το αν το σύστημα «βρήκε» έστω ένα σωστό τεκμήριο μέσα στα ανακτηθέντα. Ορίζεται ως:

$$\text{Success}@k = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \mathbb{I}[|\text{Relevant}_i \cap \text{Retrieved}_i, k| > 0] \quad (3.9)$$

όπου $\mathbb{I}[\cdot]$ είναι η ενδεικτική συνάρτηση, η οποία λαμβάνει τιμή 1 αν για το ερώτημα i έχει ανακτηθεί τουλάχιστον ένα σχετικό τεκμήριο στα πρώτα k αποτελέσματα, και 0 διαφορετικά.

Η **Mean Reciprocal Rank (MRR)** [74] επικεντρώνεται στην αξιολόγηση της ικανότητας ενός συστήματος να κατατάσσει το πρώτο σχετικό έγγραφο όσο το δυνατόν ψηλότερα στη λίστα αποτελεσμάτων. Η μετρική αυτή είναι ιδιαίτερα σημαντική σε εφαρμογές όπου ο χρήστης ενδιαφέρεται για την άμεση εύρεση μιας σωστής απάντησης, όπως στα συστήματα ερωτοαπόκρισης. Ορίζεται ως:

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \quad (3.10)$$

όπου rank_i είναι η θέση του πρώτου σχετικού εγγράφου για το ερώτημα i και $|Q|$ το πλήθος των ερωτημάτων στο σύνολο αξιολόγησης.

To **Mean Average Precision (MAP)** [75] επεκτείνει την ιδέα του MRR, λαμβάνοντας υπόψη όλες τις θέσεις των σχετικών εγγράφων στη λίστα αποτελεσμάτων, αντί μόνο της πρώτης. Για κάθε ερώτημα, υπολογίζεται η μέση ακρίβεια (*Average Precision, AP*), δηλαδή ο μέσος όρος της ακρίβειας στις θέσεις όπου εντοπίζονται σχετικά έγγραφα. Η μέση ακρίβεια για το ερώτημα q_i ορίζεται ως:

$$\text{AP}(q_i) = \frac{1}{|\text{Relevant}_i|} \sum_{k=1}^n \text{Precision}@k(q_i) \cdot \text{rel}_i(k) \quad (3.11)$$

όπου $\text{rel}_i(k)$ είναι μια δυαδική μεταβλητή που λαμβάνει τιμή 1 αν το έγγραφο στη θέση k είναι σχετικό, και 0 διαφορετικά. Η τελική τιμή του MAP προκύπτει ως ο μέσος όρος των τιμών AP για όλα τα ερωτήματα:

$$\text{MAP} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \text{AP}(q_i) \quad (3.12)$$

Η μετρική MAP είναι ιδιαίτερα χρήσιμη όταν για κάθε ερώτημα υπάρχουν πολλαπλά σχετικά τεκμήρια, καθώς μετρά τόσο την ικανότητα εντοπισμού όσο και την ορθότητα κατάταξής τους.

To **Normalized Discounted Cumulative Gain (NDCG)** [76] αντιπροσωπεύει μια προηγμένη μετρική που επιτρέπει την αξιολόγηση συστημάτων όπου η σχετικότητα δεν είναι δυαδική αλλά βαθμολογημένη. Η μετρική λαμβάνει υπόψη τόσο τη σχετικότητα κάθε εγγράφου όσο και τη θέση του στη λίστα αποτελεσμάτων, μειώνοντας τη συνεισφορά των εγγράφων σε χαμηλότερες θέσεις μέσω λογαριθμικής απόσβεσης. Ορίζεται ως:

$$\text{DCG}@k = \sum_{i=1}^k \frac{2^{\text{rel}_i} - 1}{\log_2(i + 1)} \quad (3.13)$$

$$\text{NDCG}@k = \frac{\text{DCG}@k}{\text{IDCG}@k} \quad (3.14)$$

όπου rel_i αντιπροσωπεύει τη βαθμολογία σχετικότητας του εγγράφου στη θέση i και $\text{IDCG}@k$ το ιδανικό DCG, που προκύπτει όταν τα σχετικά έγγραφα είναι κατατεταγμένα σε φθίνουσα σειρά σχετικότητας. Η κανονικοποίηση επιτρέπει τη σύγκριση της απόδοσης σε διαφορετικά σύνολα ερωτημάτων.

Συνολικά, ο συνδυασμός των μετρικών **Precision@k**, **Recall@k**, **F1@k**, **MRR**, **MAP** και **NDCG@k**, **Success@k** προσφέρει μια πολυδιάστατη αξιολόγηση της απόδοσης του συστήματος ανάκτησης, καλύπτοντας τόσο τη σχετικότητα όσο και τη θέση των εγγράφων στην κατάταξη — κρίσιμες παραμέτρους για την αξιολόγηση RAG συστημάτων.

3.4.4 Μετρικές Αξιολόγησης Παραγωγής Κειμένου

Το BLEU (Bilingual Evaluation Understudy) [77] αποτελεί μία από τις πιο ευρέως διαδεδομένες μετρικές για την αξιολόγηση αυτόματα παραγόμενου κειμένου, αναπτυχθείσα αρχικά για την αξιολόγηση μηχανικής μετάφρασης. Η θεμελιώδης ιδέα της μετρικής είναι η μέτρηση της επικάλυψης n-grams μεταξύ του υποψηφίου κειμένου που παράχθηκε αυτόματα και ενός ή περισσότερων κειμένων αναφοράς που δημιουργήθηκαν από ανθρώπους. Η μετρική υπολογίζει την ακρίβεια των n-grams διαφόρων μηκών και τα συνδυάζει χρησιμοποιώντας έναν γεωμετρικό μέσο, εφαρμόζοντας ταυτόχρονα μία ποινή συνοπτικότητας (brevity penalty) για την αποφυγή τεχνητής αύξησης της βαθμολογίας μέσω υπερβολικά σύντομων απαντήσεων. Μαθηματικά, το BLEU ορίζεται ως:

$$\text{BLEU} = \text{BP} \cdot \exp \left(\sum_{n=1}^N w_n \log p_n \right) \quad (3.15)$$

όπου p_n αντιπροσωπεύει την ακρίβεια των n-grams μήκους n , υπολογιζόμενη με κομμένη μέτρηση (clipped counting) που περιορίζει την επικάλυψη στο μέγιστο πλήθος εμφανίσεων κάθε n-gram στα κείμενα αναφοράς. Συγκεκριμένα:

$$p_n = \frac{\sum_{\text{n-gram} \in C} \text{Count}_{\text{clip}}(\text{n-gram})}{\sum_{\text{n-gram} \in C} \text{Count}(\text{n-gram})} \quad (3.16)$$

Ο όρος BP (brevity penalty) εισάγει ποινή για κείμενα που είναι συντομότερα από το μήκος αναφοράς, εξασφαλίζοντας ότι η μετρική δεν επιβραβεύει τεχνητά υψηλή ακρίβεια που επιτυγχάνεται μέσω της παράλειψης πληροφορίας.

Το ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [78] αποτελεί οικογένεια μετρικών που αναπτύχθηκαν για την αξιολόγηση αυτόματων συνοπτικών περιλήψεων, εστιάζοντας στην ανάκληση παρά στην ακρίβεια. Η διάκριση αυτή είναι κρίσιμη διότι στην περίληψη κειμένου, η πληρότητα της πληροφορίας είναι συχνά πιο σημαντική από την ακρίβεια κάθε επιμέρους φράσης. Το ROUGE-N μετρά την επικάλυψη n-grams εστιάζοντας στο ποσοστό των n-grams του κειμένου αναφοράς που εμφανίζονται στο παραγόμενο κείμενο:

$$\text{ROUGE-N} = \frac{\sum_{S \in \{\text{RefSum}\}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \{\text{RefSum}\}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_n)} \quad (3.17)$$

Μια εναλλακτική παραλλαγή, το ROUGE-L, χρησιμοποιεί την έννοια της μακρύτερης κοινής υποακολουθίας (Longest Common Subsequence, LCS), η οποία επιτρέπει την αναγνώριση δομικών ομοιοτήτων ακόμα και όταν υπάρχουν παρεμβαλλόμενες λέξεις:

$$\text{ROUGE-L} = \frac{(1 + \beta^2) R_{\text{lcs}} P_{\text{lcs}}}{R_{\text{lcs}} + \beta^2 P_{\text{lcs}}} \quad (3.18)$$

όπου R_{lcs} και P_{lcs} αντιπροσωπεύουν την ανάκληση και την ακρίβεια βάσει της LCS αντίστοιχα, και η παράμετρος β ελέγχει τη σχετική σημασία μεταξύ ανάκλησης και ακρίβειας.

Το BERTScore [79] αντιπροσωπεύει μια σύγχρονη προσέγγιση που υπερβαίνει τους περιορισμούς των λεξικογραφικών μετρικών αξιοποιώντας contextual embeddings από προεκπαιδευμένα γλωσσικά μοντέλα για την αξιολόγηση σημασιολογικής ομοιότητας. Η θεμελιώδης διαφορά έγκειται στο ότι το BERTScore μπορεί να συλλάβει σημασιολογικές αντιστοιχίες μεταξύ φράσεων που δεν μοιράζονται κοινά tokens, αντιμετωπίζοντας έτσι φαινόμενα όπως συνωνυμία και παράφραση. Η μετρική υπολογίζει την ανάκληση, την ακρίβεια και το F1-score βασιζόμενη στη μέγιστη cosine ομοιότητα μεταξύ των embeddings των tokens:

$$R_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} x_i^T \hat{x}_j \quad (3.19)$$

$$P_{BERT} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} x_i^T \hat{x}_j \quad (3.20)$$

$$F_{BERT} = 2 \cdot \frac{P_{BERT} \cdot R_{BERT}}{P_{BERT} + R_{BERT}} \quad (3.21)$$

όπου x και \hat{x} αντιπροσωπεύουν τα embeddings των tokens του κειμένου αναφοράς και του υποψήφιου κειμένου αντίστοιχα.

3.4.5 Εξειδικευμένες Μετρικές για Συστήματα RAG

Πέρα από τις παραδοσιακές μετρικές ανάκτησης και παραγωγής που αναπτύχθηκαν για ανεξάρτητα συστήματα, η ιδιαίτερη φύση των συστημάτων RAG απαιτεί εξειδικευμένες μετρικές που αξιολογούν ολιστικά τη διασύνδεση μεταξύ ανάκτησης και παραγωγής. Το πλαίσιο RAGAS (Retrieval Augmented Generation Assessment) [80] αποτελεί ένα πρωτοποριακό εργαλείο που αναπτύχθηκε ειδικά για την αυτόματη αξιολόγηση συστημάτων RAG χωρίς την απαίτηση απαντήσεων αναφοράς. Η καινοτομία της προσέγγισης έγκειται στη χρήση μεγάλων γλωσσικών μοντέλων ως κριτών (LLM-as-judge paradigm), τα οποία αξιολογούν την ποιότητα των διαφόρων συστατικών του RAG pipeline μέσω προσεκτικά σχεδιασμένων prompts.

Το πλαίσιο RAGAS θεμελιώνεται σε τρεις κεντρικές διαστάσεις ποιότητας που αποτυπώνουν διαφορετικές πτυχές της απόδοσης. Η πρώτη διάσταση αφορά τη σχετικότητα πλαισίου (Context Relevance), η οποία αξιολογεί κατά πόσον το ανακτηθέν πλαίσιο περιέχει αποκλειστικά πληροφορίες που σχετίζονται με το ερώτημα του χρήστη. Η διάσταση αυτή είναι αρκετά διότι η παρουσία μη σχετικής πληροφορίας όχι μόνο αυξάνει το υπολογιστικό κόστος λόγω του μεγαλύτερου πλήθους tokens που πρέπει να επεξεργαστεί το γλωσσικό μοντέλο, αλλά μπορεί επίσης να υποβαθμίσει την ποιότητα της παραγόμενης απάντησης μέσω της εισαγωγής θορύβου. Η δεύτερη διάσταση, η πιστότητα (Faithfulness), διασφαλίζει ότι οι ισχυρισμοί που διατυπώνονται στην παραγόμενη απάντηση μπορούν να εξαχθούν και να τεκμηριωθούν από το ανακτηθέν πλαίσιο. Αυτό αποτελεί θεμελιώδη απαίτηση για την αποφυγή παραισθήσεων (hallucinations), όπου το σύστημα παράγει πληροφορίες που δεν υποστηρίζονται από την παρεχόμενη τεκμηρίωση. Η τρίτη διάσταση, η σχετικότητα απάντησης (Answer Relevance), αξιολογεί κατά πόσον η παραγόμενη απάντηση αντιμετωπίζει άμεσα και πλήρως το τεθέν ερώτημα, αποφεύγοντας ασαφείς, ελλιπείς ή εκτός θέματος απαντήσεις.

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

Η μέτρηση της πιστότητας στο RAGAS πραγματοποιείται μέσω μιας διαδικασίας δύο σταδίων. Στο πρώτο στάδιο, το σύστημα χρησιμοποιεί ένα γλωσσικό μοντέλο για να αποσυνθέσει την παραγόμενη απάντηση σε ατομικούς, επαληθεύσιμους ισχυρισμούς. Αυτή η αποσύνθεση είναι απαραίτητη διότι οι απαντήσεις συχνά περιέχουν σύνθετες προτάσεις με πολλαπλούς ισχυρισμούς, οι οποίοι πρέπει να αξιολογηθούν ξεχωριστά. Στο δεύτερο στάδιο, για κάθε εξαχθέντα ισχυρισμό, το σύστημα επαληθεύει εάν αυτός μπορεί να συναχθεί από το ανακτηθέν πλαίσιο. Η τελική μετρική πιστότητας υπολογίζεται ως το κλάσμα των επαληθευμένων ισχυρισμών προς το σύνολο των ισχυρισμών:

$$\text{Faithfulness} = \frac{|V|}{|S|} \quad (3.22)$$

όπου $|V|$ αντιπροσωπεύει τον αριθμό των ισχυρισμών που επαληθεύονται από το πλαίσιο και $|S|$ τον συνολικό αριθμό ισχυρισμών που εξήγησαν από την απάντηση.

Η σχετικότητα απάντησης αξιολογείται μέσω μιας έμμεσης διαδικασίας που βασίζεται στην αρχή της αμφίδρομης συνέπειας: μια καλή απάντηση θα πρέπει να είναι ικανή να οδηγήσει πίσω στο αρχικό ερώτημα. Για μια δεδομένη απάντηση, το σύστημα παράγει πολλαπλά πιθανά ερωτήματα που θα μπορούσαν λογικά να οδηγήσουν σε αυτήν την απάντηση. Στη συνέχεια, υπολογίζεται η σημασιολογική ομοιότητα μεταξύ του αρχικού ερωτήματος και κάθε παραγόμενου ερωτήματος χρησιμοποιώντας embeddings και cosine similarity. Η μέση ομοιότητα αποτελεί την τελική βαθμολογία:

$$\text{Answer Relevance} = \frac{1}{n} \sum_{i=1}^n \cos(E_q, E_{q_i}) \quad (3.23)$$

όπου E_q αντιπροσωπεύει το embedding του αρχικού ερωτήματος και E_{q_i} τα embeddings των παραγόμενων ερωτημάτων. Η προσέγγιση αυτή επιτρέπει την αξιολόγηση όχι μόνο της άμεσης συνάφειας αλλά και της πληρότητας της απάντησης.

Στο πλαίσιο του AutoRAG framework [81], εισάγεται μια παραλλαγή της μετρικής Context Precision που λαμβάνει υπόψη τη θέση των σχετικών εγγράφων στη λίστα αποτελεσμάτων. Η μετρική αυτή υπολογίζεται ως σταθμισμένο άθροισμα της ακρίβειας σε κάθε θέση, όπου το βάρος καθορίζεται από έναν δείκτη σχετικότητας:

$$\text{Context Precision@K} = \frac{\sum_{k=1}^K (\text{Precision}@k \times v_k)}{\text{true positives@K}} \quad (3.24)$$

όπου $v_k \in \{0, 1\}$ είναι δείκτης σχετικότητας που υποδεικνύει εάν το έγγραφο στη θέση k είναι σχετικό με το ερώτημα.

3.5 Το Πλαισιο AutoRAG: Αυτοματοποιημενη Βελτιστοποιηση Συστηματων RAG

3.5.1 Αρχιτεκτονική και Φιλοσοφία Σχεδιασμού

Το πλαίσιο AutoRAG [81] εισάγει μια ολοκληρωμένη μεθοδολογία για την αυτοματοποιημένη βελτιστοποίηση συστημάτων RAG, μετασχηματίζοντας το πρόβλημα της επιλογής και διαμόρφωσης των επιμέρους συστατικών από μια χειροκίνητη διαδικασία σε ένα συστηματικό πρόβλημα βελτιστοποίησης παρόμοιο με τις μεθοδολογίες AutoML που έχουν αναπτυχθεί στην μηχανική μάθηση. Η κεντρική φιλοσοφική θεώρηση που διέπει το AutoRAG βασίζεται στην αρχή της αρθρωτής αποσύνθεσης, σύμφωνα με την οποία ένα σύνθετο σύστημα RAG μπορεί να αναλυθεί σε ένα σύνολο ανεξάρτητων κόμβων, ο καθένας από τους οποίους εκτελεί μια συγκεκριμένη λειτουργία και μπορεί να υλοποιηθεί με διαφορετικούς τρόπους.

Η αρχιτεκτονική του συστήματος οργανώνεται σε τρία ιεραρχικά επίπεδα αφαιρεσης που επιτρέπουν τη συστηματική εξερεύνηση του χώρου σχεδιασμού. Το πρώτο επίπεδο, το επίπεδο κόμβων, αντιπροσωπεύει τις θεμελιώδεις λειτουργίες που απαιτούνται σε ένα τυπικό RAG pipeline, όπως η επαύξηση ερωτημάτων που εμπλουτίζει ή αναδιατυπώνει το αρχικό ερώτημα του χρήστη, η ανάκτηση που εντοπίζει σχετικά έγγραφα από μια συλλογή, η επανακατάταξη που βελτιώνει τη σειρά των ανακτηθέντων εγγράφων, η επαύξηση αποσπασμάτων που εμπλουτίζει τα ανακτηθέντα αποσπάσματα με πρόσθετο πλαίσιο, η κατασκευή προτροπών που διαμορφώνει το τελικό prompt για το γλωσσικό μοντέλο, και η παραγωγή που δημιουργεί την τελική απάντηση. Το δεύτερο επίπεδο, το επίπεδο αρθρωμάτων, περιέχει συγκεκριμένες εναλλακτικές τεχνικές υλοποιήσεις για κάθε κόμβο. Για παράδειγμα, ο κόμβος ανάκτησης μπορεί να υλοποιηθεί με αλγορίθμους όπως το BM25 που βασίζεται σε λεξικογραφική αντιστοίχιση, το DPR (Dense Passage Retrieval) που χρησιμοποιεί πυκνές διανυσματικές αναπαραστάσεις, το ColBERT που συνδυάζει αποδοτικότητα με βαθιά σημασιολογική κατανόηση, ή υψηλούς προσεγγίσεις που συνδυάζουν πολλαπλές μεθόδους. Το τρίτο επίπεδο, το επίπεδο παραμέτρων, ορίζει τις υπερπαραμέτρους που ελέγχουν τη λεπτή συμπεριφορά κάθε module, όπως ο αριθμός των ανακτώμενων εγγράφων k , οι διαστάσεις των embeddings, οι παράμετροι των reranking μοντέλων, ή οι θερμοκρασίες παραγωγής.

Η θεμελιώδης πρόκληση που αντιμετωπίζει το AutoRAG είναι η αποδοτική εξερεύνηση του τεράστιου χώρου συνδυαστικών επιλογών. Εάν κάθε κόμβος έχει m_i διαθέσιμα modules και υπάρχουν n κόμβοι στο pipeline, τότε ο συνολικός αριθμός πιθανών configurations είναι $\prod_{i=1}^n m_i$, ο οποίος μπορεί εύκολα να φτάσει σε εκατοντάδες χιλιάδες ή εκατομμύρια συνδυασμούς. Η εξαντλητική αξιολόγηση όλων των συνδυασμών είναι υπολογιστικά ανέφικτη, ειδικά όταν η αξιολόγηση κάθε configuration απαιτεί κλήσεις σε ακριβά LLM APIs. Το AutoRAG αντιμετωπίζει αυτήν την πρόκληση μέσω μιας έξυπνης άπληστης στρατηγικής που εκμεταλλεύεται τη σειριακή δομή του RAG pipeline.

Η βασική ιδέα της greedy προσέγγισης είναι η σταδιακή κατασκευή του βέλτιστου pipeline μέσω τοπικά βέλτιστων επιλογών. Σε κάθε στάδιο, το σύστημα αξιολογεί όλα τα διαθέσιμα modules για τον τρέχοντα κόμβο, διατηρώντας σταθερές τις

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

επιλογές που έχουν γίνει στους προηγούμενους κόμβους, και επιλέγει το module που μεγιστοποιεί την απόδοση σύμφωνα με προκαθορισμένες μετρικές. Αυτή η επιλογή στερεώνεται και το σύστημα προχωρά στον επόμενο κόμβο. Η προσέγγιση αυτή μειώνει την υπολογιστική πολυπλοκότητα από $O(\prod_{i=1}^n m_i)$ σε $O(\sum_{i=1}^n m_i)$, μια δραματική βελτίωση που καθιστά τη βελτιστοποίηση πρακτικά εφικτή.

Ωστόσο, η άπληστη προσέγγιση παρουσιάζει μια σημαντική πρόκληση: πώς αξιολογούμε modules για κόμβους όπου η απόδοση δεν είναι άμεσα μετρήσιμη; Για παράδειγμα, στον κόμβο επαύξησης ερωτημάτων, η έξοδος είναι ένα ή περισσότερα αναδιατυπωμένα ερωτήματα, για τα οποία δεν υπάρχει προφανής μετρική ποιότητας. Παρομοίως, ο κόμβος κατασκευής προτροπών παράγει ένα διαμορφωμένο prompt, το οποίο είναι δύσκολο να αξιολογηθεί ανεξάρτητα από την τελική απάντηση. Το AutoRAG επιλύει αυτό το πρόβλημα μέσω της αξιολόγησης τελικής εξόδου (downstream evaluation). Για κόμβους με δύσκολη άμεση αξιολόγηση, το σύστημα μετρά την απόδοσή τους έμμεσα, μέσω της επίδρασής τους στους επόμενους κόμβους του αγωγού. Συγκεκριμένα, για την αξιολόγηση εναλλακτικών (modules) στον κόμβο επαύξησης ερωτημάτων, το σύστημα διατηρεί σταθερό ένα προεπιλεγμένο module για τον επόμενο κόμβο ανάκτησης, αξιολογεί την ποιότητα ανάκτησης για κάθε εναλλακτική επαύξηση ερωτήματος, και επιλέγει την επαύξηση που οδηγεί στην καλύτερη ανάκτηση.

Η στρατηγική αυτή μειώνει σημαντικά τον αριθμό των απαιτούμενων πειραμάτων. Έστω ότι ο κόμβος επαύξησης ερωτημάτων έχει m εναλλακτικά modules και ο κόμβος ανάκτησης έχει n εναλλακτικά. Η εξαντλητική αξιολόγηση όλων των συνδυασμών θα απαιτούσε $m \times n$ πειράματα. Η downstream evaluation προσέγγιση απαιτεί μόνο m πειράματα για την αξιολόγηση του κόμβου επαύξησης (με σταθερό module ανάκτησης) και επιπλέον n πειράματα για την αξιολόγηση του κόμβου ανάκτησης (με το επιλεγμένο module επαύξησης), συνολικά $m + n$ πειράματα. Αυτή η μείωση από πολλαπλασιαστική σε αθροιστική πολυπλοκότητα είναι κρίσιμη για την πρακτική εφαρμοσιμότητα του πλαισίου.

3.5.2 AutoRAG-HP: Online Βελτιστοποίηση Υπερπαραμέτρων

Η επέκταση AutoRAG-HP [63] που παρουσιάστηκε στο συνέδριο Empirical Methods in Natural Language Processing (EMNLP) 2024 αντιμετωπίζει το συμπληρωματικό πρόβλημα της βελτιστοποίησης των υπερπαραμέτρων εντός κάθε επιλεγμένου module. Ενώ το AutoRAG επιλέγει ποια modules να χρησιμοποιήσει, το AutoRAG-HP βελτιστοποιεί πώς να διαμορφώσει αυτά τα modules για μέγιστη απόδοση. Η καινοτομία της προσέγγισης έγκειται στη διαδικτυακή (online) φύση της βελτιστοποίησης, όπου το σύστημα μαθαίνει προσαρμοστικά από την ανατροφοδότηση που λαμβάνει κατά την εκτέλεση, σε αντίθεση με τις παραδοσιακές offline μεθόδους όπως το Grid Search ή το Random Search που απαιτούν πλήρη αξιολόγηση του χώρου παραμέτρων εκ των προτέρων.

Το πρόβλημα μοντελοποιείται ως ένα πρόβλημα «ληστή με πολλά χέρια» (Multi-Armed Bandit, MAB), μια κλασική διατύπωση στη θεωρία online learning που προέρχεται από την αναλογία με έναν παίκτη που επιλέγει μεταξύ πολλαπλών μοχλών κουλοχέρη (slot machine) σε ένα καζίνο, προσπαθώντας να μεγιστοποιήσει τη συνολική του απόδοση. Στο πλαίσιο του AutoRAG-HP, κάθε συνδυασμός υπερπαρα-

3.5. ΤΟ ΠΛΑΙΣΙΟ AUTORAG: ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΩΝ RAG

μέτρων αντιστοιχεί σε έναν μοχλό (arm), και η απόδοση του συστήματος με αυτόν τον συνδυασμό αποτελεί την ανταμοιβή (reward). Ο στόχος είναι η μεγιστοποίηση της αναμενόμενης αθροιστικής ανταμοιβής σε ένα χρονικό ορίζοντα:

$$\max_{\theta \in \Theta} \mathbb{E}[R(\theta)] = \max_{\theta} \sum_{t=1}^T r_t(\theta_t) \quad (3.25)$$

όπου θ αντιπροσωπεύει το διάνυσμα υπερπαραμέτρων που επιλέγεται σε κάθε χρονική στιγμή. Θ τον χώρο όλων των δυνατών συνδυασμών υπερπαραμέτρων, r_t την ανταμοιβή που παρατηρείται στο χρόνο t , και T τον συνολικό αριθμό επαναλήψεων. Η κεντρική πρόκληση στο MAB problem είναι η εξισορρόπηση μεταξύ εξερεύνησης (exploration), δηλαδή της δοκιμής νέων συνδυασμών για την απόκτηση πληροφορίας, και εκμετάλλευσης (exploitation), δηλαδή της χρήσης των συνδυασμών που φαίνεται να έχουν καλή απόδοση βάσει των μέχρι τώρα παρατηρήσεων.

Όταν ο χώρος υπερπαραμέτρων είναι μεγάλος, η άμεση εφαρμογή ενός MAB αλγορίθμου μπορεί να είναι αναποτελεσματική διότι ο αριθμός των χεριών (arms) γίνεται πολύ μεγάλος. Το AutoRAG-HP εισάγει μια ιεραρχική δομή που οργανώνει τον χώρο αναζήτησης σε δύο επίπεδα. Στο υψηλό επίπεδο, ένα MAB επιλέγει ποιο δομοστοιχείο (module) θα βελτιστοποιηθεί σε κάθε επανάληψη. Στο χαμηλό επίπεδο, για κάθε δομοστοιχείο υπάρχει ένα ξεχωριστό MAB που επιλέγει συγκεκριμένες τιμές για τις υπερπαραμέτρους του. Αυτή η ιεραρχική οργάνωση επιτρέπει την αποδοτική εξερεύνηση μεγάλων χώρων με πολλές διαστάσεις, διατηρώντας παράλληλα έναν εύλογο αριθμό arms σε κάθε MAB.

Η στρατηγική επιλογής βασίζεται στον αλγόριθμο «άνω όριο εμπιστοσύνης» (Upper Confidence Bound, UCB), ο οποίος επιλέγει arms βάσει ενός κριτηρίου που συνδυάζει την παρατηρημένη μέση απόδοση με ένα εύρος εμπιστοσύνης που αντανακλά την αβεβαιότητα. Για κάθε arm i στο χρόνο t , το κριτήριο επιλογής είναι:

$$UCB_i(t) = \bar{X}_i(t) + \sqrt{\frac{2 \ln t}{n_i(t)}} \quad (3.26)$$

όπου $\bar{X}_i(t)$ είναι η μέση παρατηρημένη ανταμοιβή του arm i μέχρι το χρόνο t , και $n_i(t)$ ο αριθμός φορών που το arm έχει επιλεγεί. Ο πρώτος όρος ενθαρρύνει την εκμετάλλευση των arms με υψηλή παρατηρημένη απόδοση, ενώ ο δεύτερος όρος, το όριο εμπιστοσύνης (confidence bound), ενθαρρύνει την εξερεύνηση των arms που έχουν επιλεγεί λίγες φορές και συνεπώς έχουν μεγάλη αβεβαιότητα. Το εύρος εμπιστοσύνης μειώνεται με την αύξηση του $n_i(t)$, αντανακλώντας την αυξανόμενη βεβαιότητα καθώς συλλέγουμε περισσότερες παρατηρήσεις.

3.5.3 Πειραματικά Αποτελέσματα και Αξιολόγηση Απόδοσης

Η εμπειρική αξιολόγηση του AutoRAG framework διενεργήθηκε χρησιμοποιώντας το σύνολο δεδομένων ARAGOG (Advanced RAG Output Grading) [82], ένα εξειδικευμένο benchmark που σχεδιάστηκε για την αξιολόγηση συστημάτων RAG στο πεδίο της Τεχνητής Νοημοσύνης και των Μεγάλων Γλωσσικών Μοντέλων. Το σύνολο δεδομένων περιλαμβάνει 423 επιστημονικά άρθρα προερχόμενα από το αρχείο arXiv, τα οποία καλύπτουν ένα ευρύ φάσμα θεμάτων από βαθιά μάθηση και

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

επεξεργασία φυσικής γλώσσας μέχρι ενισχυτική μάθηση και υπολογιστική όραση. Επιπλέον, δημιουργήθηκαν 107 ζεύγη ερωτήσεων-απαντήσεων με τη βοήθεια του GPT-4, σχεδιασμένα να είναι τεχνικά απαιτητικά και αντιπροσωπευτικά των πραγματικών ερωτημάτων που θα υποβάλλονταν σε ένα τέτοιο σύστημα. Κάθε ζεύγος επικυρώθηκε από ανθρώπινους εμπειρογνόμονες για τη διασφάλιση της ποιότητας, της τεχνικής ακρίβειας και της συνάφειας με το περιεχόμενο της συλλογής εγγράφων.

Τα πειραματικά αποτελέσματα στο επίπεδο ανάκτησης αποκαλύπτουν σημαντικές διαφορές στην αποτελεσματικότητα διαφόρων προσεγγίσεων. Η υβριδική μέθοδος Hybrid DBSF (Dense-BM25 Score Fusion), η οποία συνδυάζει αραιές λεξικογραφικές αντιστοιχίσεις με πυκνές σημασιολογικές αναπαραστάσεις, επιτυγχάνει Context Precision@10 ίσο με 0.696 [81], υπερτερώντας σημαντικά του κλασικού BM25 που επιτυγχάνει 0.649. Αυτό αντιπροσωπεύει σχετική βελτίωση 7.2 τοις εκατό, επιδεικνύοντας τα οφέλη του συνδυασμού συμπληρωματικών σημάτων ανάκτησης. Ακόμη πιο εντυπωσιακή είναι η υπεροχή έναντι της αμιγούς διανυσματικής βάσης δεδομένων (VectorDB) που επιτυγχάνει μόνο 0.522, παρουσιάζοντας βελτίωση 33.3 τοις εκατό. Αυτά τα αποτελέσματα υπογραμμίζουν ότι για τεχνικά datasets με εξειδικευμένη ορολογία, η καθαρά σημασιολογική αναζήτηση μπορεί να αστοχεί στην ακριβή λεξική αντιστοίχιση που είναι απαραίτητη για την ανάκτηση σχετικών αποσπασμάτων.

Στο στάδιο της επανακατάταξης, η εισαγωγή του Flag Embedding LLM Reranker οδηγεί σε σημαντική βελτίωση της ποιότητας των τελικά επιλεγμένων εγγράφων. Η μετρική Context Precision@5, η οποία αξιολογεί την ακρίβεια των πέντε κορυφαίων ανακτηθέντων εγγράφων μετά την επανακατάταξη, αυξάνεται από 0.770 σε 0.838, μια απόλυτη βελτίωση 0.068 ή σχετική βελτίωση 8.8 τοις εκατό. Η βελτίωση αυτή είναι ιδιαίτερα σημαντική διότι τα πέντε κορυφαία έγγραφα είναι συνήθως αυτά που τελικά συμπεριλαμβάνονται στο πλαίσιο που τροφοδοτείται στο γλωσσικό μοντέλο, και επομένως η ποιότητά τους επηρεάζει άμεσα την τελική απάντηση.

Η αποδοτικότητα της βελτιστοποίησης υπερπαραμέτρων μέσω του Hierarchical MAB είναι εξίσου εντυπωσιακή. Η μετρική Recall@5, η οποία μετρά πόσο συχνά μία από τις πέντε καλύτερες διαμορφώσεις βρίσκεται μέσα στα πέντε πρώτα χέρια που επιλέγονται από τον αλγόριθμο, φτάνει περίπου το 0.8. Αυτό επιτυγχάνεται χρησιμοποιώντας μόνο περίπου το 20 τοις εκατό των κλήσεων της «προγραμματιστικής διεπαφής μεγάλου γλωσσικού μοντέλου» (LLM API) που απαιτεί η εξαντλητική μέθοδος της αναζήτησης πλέγματος (grid search). Η οικονομία αυτή είναι κρίσιμη στην πράξη, όπου το κόστος των API calls μπορεί να είναι σημαντικό και η χρονική καθυστέρηση της βελτιστοποίησης να αποτελεί εμπόδιο στην ταχεία ανάπτυξη συστημάτων.

Στο επίπεδο παραγωγής κειμένου, το βελτιστοποιημένο σύστημα επιτυγχάνει υψηλή ποιότητα όπως αποτιμάται από πολλαπλές μετρικές. Το G-Eval score, μια μετρική βασισμένη σε γλωσσικό μοντέλο που αξιολογεί διαστάσεις όπως η συνάφεια, η συνοχή, η ευφράδεια και η πληρότητα, φτάνει περίπου το 3.85 σε κλίμακα 1 έως 5. Επιπλέον, ο βαθμός σημασιολογικής ομοιότητας (semantic similarity score), ο οποίος μετρά τη σημασιολογική εγγύτητα μεταξύ της παραγόμενης και της αναφορικής απάντησης χρησιμοποιώντας ενσωματώσεις, επιτυγχάνει 0.919, υποδεικνύοντας ότι το σύστημα παράγει απαντήσεις που είναι σημασιολογικά πολύ κοντά στις

3.6. ΕΝΣΩΜΑΤΩΣΗ ΣΕ ΠΑΡΑΓΩΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

επιθυμητές απαντήσεις παρότι μπορεί να διαφέρουν λεξικά.

Το πλαίσιο AutoRAG αντιπροσωπεύει μια ουσιαστική μεθοδολογική εξέλιξη στην κατεύθυνση της αυτοματοποίησης και βελτιστοποίησης των συστημάτων RAG. Η modular αρχιτεκτονική του, η συστηματική μεθοδολογία αξιολόγησης μέσω καθορισμένων μετρικών, και η αποδοτική greedy στρατηγική βελτιστοποίησης, συνδυάζονται για να δημιουργήσουν ένα πλαίσιο που γεφυρώνει αποτελεσματικά το χάσμα μεταξύ ερευνητικών καινοτομιών και πρακτικών εφαρμογών παραγωγής. Η δυνατότητα του συστήματος να προσαρμόζεται σε διαφορετικά domains και use cases μέσω της αυτοματοποιημένης επιλογής και διαμόρφωσης των κατάλληλων δομοστοιχείων καθιστά την ταχεία ανάπτυξη και βελτιστοποίηση RAG συστημάτων όχι μόνο εφικτή αλλά και συστηματική, μειώνοντας την εξάρτηση από επί τούτου (ad-hoc) πειραματισμό και εμπειρική γνώση.

3.6 ΕΝΣΩΜΑΤΩΣΗ ΣΕ ΠΑΡΑΓΩΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

3.6.1 Προκλήσεις ένταξης του RAG στην παραγωγή

Η μετάβαση των ερευνητικών συστημάτων RAG σε παραγωγικό περιβάλλον αποκαλύπτει σημαντικές προκλήσεις που υπερβαίνουν τις θεωρητικές επιδόσεις των συστατικών μερών. Η εμπειρία από την ανάπτυξη παραγωγικών (production-ready) RAG εφαρμογών καταδεικνύει ότι η αξιοπιστία, η κλιμάκωση και η διαχείριση πόρων αποτελούν κρίσιμους παράγοντες επιτυχίας που συχνά υποτιμώνται στη φάση του πρωτοτύπου.

Η υπολογιστική πολυπλοκότητα των RAG συστημάτων εκδηλώνεται σε πολλαπλά επίπεδα της αρχιτεκτονικής. Η συνολική καθυστέρηση (latency) αποτελεί συνάρτηση όλων των επιμέρους σταδίων: της διανυσματικής αναζήτησης στη βάση γνώσης, της επανακατάταξης των ανακτηθέντων εγγράφων μέσω διασταυρωμένης κωδικοποίησης, της σύνθεσης της τελικής οδηγίας (prompt) και της παραγωγής της απάντησης από το γλωσσικό μοντέλο. Έρευνες στον τομέα της αξιολόγησης RAG συστημάτων υπογραμμίζουν τη σημασία της συνεχούς παρακολούθησης των μετρικών καθυστέρησης για τη διασφάλιση ικανοποιητικής εμπειρίας χρήστη. Συγκεκριμένα, πλαίσια αξιολόγησης όπως αυτά που προτείνονται από την κοινότητα των παραγωγικών συστημάτων τονίζουν την ανάγκη για παρακολούθηση σε πραγματικό χρόνο (real-time monitoring) του χρόνου απόκρισης σε όλα τα στάδια του pipeline.

Η κλιμάκωση των RAG συστημάτων εγείρει πρόσθετες προκλήσεις σχετικά με τη διαχείριση μνήμης και την υπολογιστική υποδομή. Οι διανυσματικές βάσεις δεδομένων απαιτούν σημαντικούς πόρους μνήμης για την αποθήκευση και την αποτελεσματική αναζήτηση εκατομμυρίων ενσωματώσεων, ενώ παράλληλα τα γλωσσικά μοντέλα επιβαρύνουν τη μνήμη της κάρτας γραφικών (GPU) κατά τη φάση της παραγωγής. Η χρήση τεχνικών όπως ο κβαντισμός των διανυσμάτων (quantization) και η εφαρμογή στρατηγικών προσωρινής αποθήκευσης (caching) μπορεί να μειώσει σημαντικά τις απαιτήσεις μνήμης, ωστόσο η ισορροπία μεταξύ απόδοσης και κατανάλωσης πόρων παραμένει κρίσιμη σχεδιαστική απόφαση.

Η αξιοπιστία και η ανθεκτικότητα σε σφάλματα (fault tolerance) αποτελούν

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

επίσης θεμελιώδεις απαιτήσεις για παραγωγικά συστήματα. Η αποτυχία οποιουδήποτε δομοστοιχείου του RAG pipeline, είτε της βάσης διανυσμάτων, είτε του δομοστοιχείου αναδιάταξης, είτε του LLM - δεν πρέπει να οδηγεί σε ολική αποτυχία του συστήματος. Η υλοποίηση μηχανισμών ομαλής υποβάθμισης (graceful degradation), όπου το σύστημα μπορεί να επιστρέψει σε απλούστερες λειτουργίες (π.χ. άμεση παραγωγή χωρίς ανάκτηση) σε περίπτωση αποτυχίας κάποιου δομοστοιχείου, διασφαλίζει τη συνέχεια λειτουργίας. Επιπλέον, η λεπτομερής καταγραφή (logging) όλων των σταδίων επεξεργασίας και η παρακολούθηση μετρικών απόδοσης σε πραγματικό χρόνο καθίστανται απαραίτητες για την έγκαιρη ανίχνευση και αντιμετώπιση προβλημάτων.

3.6.2 Η Συνεισφορά του AutoRAG σε περιβάλλοντα παραγωγής

Το πλαίσιο AutoRAG [81] προσφέρει συγκεκριμένες λύσεις που διευκολύνουν τη μετάβαση από την ερευνητική φάση στην παραγωγική λειτουργία. Η δυνατότητα αυτόματης ανακάλυψης του βέλτιστου pipeline επιτρέπει τη συστηματική εξερεύνηση εκατοντάδων πιθανών συνδυασμών δομοστοιχείων και παραμέτρων, μειώνοντας δραστικά τον χρόνο ανάπτυξης από εβδομάδες χειροκίνητης ρύθμισης σε λίγες ώρες αυτοματοποιημένης βελτιστοποίησης. Αυτή η συστηματική προσέγγιση διασφαλίζει επίσης την πλήρη καταγραφή και αναπαραγωγιμότητα της διαδικασίας βελτιστοποίησης, στοιχείο κρίσιμο για την τεκμηρίωση και τη μελλοντική συντήρηση του συστήματος.

Ιδιαίτερα σημαντική είναι η δυνατότητα online προσαρμογής που προσφέρει η επέκταση AutoRAG-HP [63] μέσω των Multi-Armed Bandit (MAB) αλγορίθμων. Οι MAB αλγόριθμοι αποτελούν μία κατηγορία μεθόδων ενισχυτικής μάθησης (reinforcement learning) που αντιμετωπίζουν το πρόβλημα της ισορροπίας μεταξύ εξερεύνησης (exploration) νέων επιλογών και εκμετάλλευσης (exploitation) των γνωστών βέλτιστων επιλογών.

Η διαδικασία περιλαμβάνει τη συλλογή σημάτων ανταμοιβής από τις αλληλεπιδράσεις των χρηστών, την ενημέρωση των κατανομών πιθανότητας των MAB agents, και τη σταδιακή ανάπτυξη βελτιωμένων ρυθμίσεων μέσω A/B testing σε περιορισμένο ποσοστό της κυκλοφορίας. Το A/B testing αποτελεί πειραματική μεθοδολογία όπου δύο ή περισσότερες εκδοχές του συστήματος αναπτύσσονται ταυτόχρονα σε διαφορετικά υποσύνολα χρηστών, επιτρέποντας τη συγκριτική αξιολόγηση της απόδοσής τους υπό ρεαλιστικές συνθήκες. Όταν επιβεβαιώνεται βελτίωση που υπερβαίνει προκαθορισμένο κατώφλι, η νέα ρύθμιση προωθείται στο πλήρες παραγωγικό περιβάλλον. Αυτός ο μηχανισμός συνεχούς βελτίωσης διασφαλίζει ότι το σύστημα προσαρμόζεται δυναμικά στις μεταβαλλόμενες ανάγκες και προτιμήσεις των χρηστών.

Η αρθρωτή αρχιτεκτονική του AutoRAG εναρμονίζεται φυσικά με τις σύγχρονες πρακτικές DevOps και cloud deployment. Ο όρος DevOps αναφέρεται σε ένα σύνολο πρακτικών που ενοποιούν την ανάπτυξη λογισμικού (Development) με τις λειτουργίες πληροφορικής (Operations), επιτρέποντας τη ταχύτερη και αξιόπιστη ανάπτυξη εφαρμογών. Κάθε δομοστοιχείο του pipeline μπορεί να αντιμετωπιστεί ως ανεξάρτητη μικρουπηρεσία (microservice), μία αυτόνομη υπηρεσία που εκτελεί συγκεκριμένη λειτουργία και επικοινωνεί με άλλες υπηρεσίες μέσω καθορι-

3.7. ΕΡΕΥΝΗΤΙΚΟ ΚΕΝΟ ΚΑΙ ΣΥΝΕΙΣΦΟΡΑ ΤΗΣ ΠΑΡΟΥΣΑΣ ΕΡΓΑΣΙΑΣ

σμένων διεπαφών. Αυτή η προσέγγιση επιτρέπει την ανεξάρτητη κλιμάκωση, την απομόνωση αποτυχιών (fault isolation) και τη σταδιακή αναβάθμιση χωρίς διακοπή λειτουργίας (rolling updates). Η δυναμική κατανομή υπολογιστικού φορτίου ανάλογα με τις ανάγκες κάθε δομοστοιχείου, σε συνδυασμό με την υποστήριξη για containerization μέσω τεχνολογιών όπως το Kubernetes, διευκολύνει την αποδοτική διαχείριση πόρων και τη διατήρηση υψηλής διαθεσιμότητας του συστήματος. Το Kubernetes αποτελεί πλατφόρμα ορχήστρωσης (orchestration) για την αυτοματοποιημένη ανάπτυξη, κλιμάκωση και διαχείριση containerized εφαρμογών, όπου containers είναι ελαφριές, φορητές μονάδες λογισμικού που περιλαμβάνουν όλες τις απαραίτητες εξαρτήσεις για την εκτέλεση μιας εφαρμογής.

3.7 ΕΡΕΥΝΗΤΙΚΟ ΚΕΝΟ ΚΑΙ ΣΥΝΕΙΣΦΟΡΑ ΤΗΣ ΠΑΡΟΥΣΑΣ ΕΡΓΑΣΙΑΣ

3.7.1 Μεθοδολογικά και Πρακτικά Εμπόδια στην Εφαρμογή Υφιστάμενων Προσεγγίσεων

Παρά τη σημαντική πρόοδο που έχει επιτευχθεί τα τελευταία χρόνια στην ανάπτυξη και αξιολόγηση συστημάτων RAG, τα περισσότερα ερευνητικά αποτελέσματα παρουσιάζουν περιορισμένη δυνατότητα αναπαραγωγής και μεταφοράς σε εξειδικευμένα πεδία.

Πρώτον, η αναπαραγωγή των αποτελεσμάτων (replicability) παραμένει δύσκολη στην πράξη. Πολλές δημοσιευμένες εργασίες στηρίζονται σε σύνθετες πειραματικές ρυθμίσεις που είτε δεν τεκμηριώνονται επαρκώς είτε βασίζονται σε ιδιωτικά μοντέλα και μη διαθέσιμα σύνολα δεδομένων. Ως αποτέλεσμα, η επαλήθευση ή η περαιτέρω αξιολόγηση των αποτελεσμάτων καθίσταται ιδιαίτερα απαιτητική ακόμη και για ερευνητές με παρόμοια τεχνική υποδομή. Η έλλειψη τυποποιημένων αγωγών και ανοιχτών εργαλείων δυσχεραίνει επίσης τη σύγκριση μεθόδων υπό χοινά πειραματικά πρωτόκολλα.

Δεύτερον, τα περισσότερα πειραματικά δεδομένα και benchmarks παραμένουν γενικού σκοπού, όπως τα Natural Questions και MS MARCO, γεγονός που περιορίζει την προσαρμοστικότητα σε εξειδικευμένα domains. Σε τομείς όπως ο προγραμματισμός, η βιοπληροφορική ή η ιατρική, τα ερωτήματα περιλαμβάνουν εξειδικευμένη ορολογία, κώδικα ή αριθμητικές σχέσεις που δεν αποτυπώνονται σε γενικά σύνολα δεδομένων. Ως εκ τούτου, η μεταφορά των συμπερασμάτων από τα υπάρχοντα benchmarks σε πραγματικά, τεχνικά σενάρια παραμένει επισφαλής.

Τρίτον, παρότι οι μεγάλες πλατφόρμες νέφους (cloud) (π.χ. Amazon Bedrock, Google Vertex AI, Azure AI) παρέχουν υποδομές για πειραματισμό με RAG pipelines, η φύση των περισσότερων δεδομένων και η απουσία αντικειμενικών κριτηρίων καθιστούν δύσκολη την ποσοτική αξιολόγηση πέραν της χρόνης ενός LLM ως κριτή (LLM-as-judge). Αν και αυτή η προσέγγιση έχει επικρατήσει στην πρόσφατη βιβλιογραφία ως πρακτική λύση, εισάγει σημαντικούς κινδύνους μεροληφίας, καθώς η αξιολόγηση βασίζεται σε ένα μοντέλο του ίδιου τύπου με αυτό που παράγει τις απαντήσεις. Η εξάρτηση από τέτοιου είδους υποκειμενικά κριτήρια υπονομεύει τη διαφάνεια και την αξιοπιστία της έρευνας.

Ένα επιπλέον εμπόδιο αφορά την έλλειψη έγκυρων συνόλων δεδομένων με

ΚΕΦΑΛΑΙΟ 3. ΚΡΙΤΙΚΗ ΑΝΑΣΚΟΠΗΣΗ ΜΕΘΟΔΟΛΟΓΙΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΠΑΡΑΓΩΓΗΣ ΜΕΣΩ ΑΝΑΚΤΗΣΗΣ

ground truth, τόσο σε ερευνητικά όσο και σε παραγωγικά περιβάλλοντα. Ακόμη και οι οργανισμοί που αναπτύσσουν εμπορικά συστήματα RAG διαθέτουν συνήθως μόνο ακατέργαστα δεδομένα (τεκμηρίωση, άρθρα, συζητήσεις), χωρίς επισημασμένες ερωτήσεις και απαντήσεις που να επιτρέπουν αντικειμενική αξιολόγηση. Ως αποτέλεσμα, η διαδικασία αξιολόγησης βασίζεται σε συνθετικά δεδομένα που παράγονται από LLMs ή σε αυτόματη κρίση από τα ίδια τα μοντέλα (LLM-as-judge). Αυτή η εξάρτηση από συνθετικά κριτήρια εισάγει αβεβαιότητα και μεροληφία, υπονομεύοντας τη συγκρισιμότητα των αποτελεσμάτων μεταξύ διαφορετικών συστημάτων.

Η παρούσα εργασία επιχειρεί να απαντήσει σε αυτές τις προκλήσεις προτείνοντας ένα ανοιχτό, επαναχρησιμοποιήσιμο και πλήρως παραμετροποιήσιμο πλαίσιο πειραματισμού, σχεδιασμένο ώστε να ενισχύει την αναπαραγωγιμότητα, να επιτρέπει τη σύγκριση μεθόδων σε πραγματικά τεχνικά δεδομένα και να υποστηρίζει αντικειμενικές διαδικασίες αξιολόγησης πέραν της στοχαστικής κρίσης των LLMs.

Πέραν της ερευνητικής του αξίας, το προτεινόμενο πλαίσιο στοχεύει επίσης στη διευκόλυνση της πρακτικής ανάπτυξης και ενσωμάτωσης RAG pipelines, παρέχοντας κώδικα και εργαλεία που μπορούν να χρησιμοποιηθούν άμεσα σε διαφορετικά περιβάλλοντα και υποδομές. Σε αντίθεση με τις έτοιμες, συχνά κλειστές λύσεις των μεγάλων παρόχων νέφους (cloud providers), το πλαίσιο αυτό προάγει τη διαφάνεια, την ευελιξία και την επεκτασιμότητα, επιτρέποντας στους ερευνητές και μηχανικούς να δοκιμάζουν, προσαρμόζουν και αξιολογούν τα συστήματα ανάκτησης και παραγωγής με ελάχιστο κόπο ανάπτυξης.

4

Γλοποίηση

4.1 ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ ΕΠΙΛΟΓΕΣ

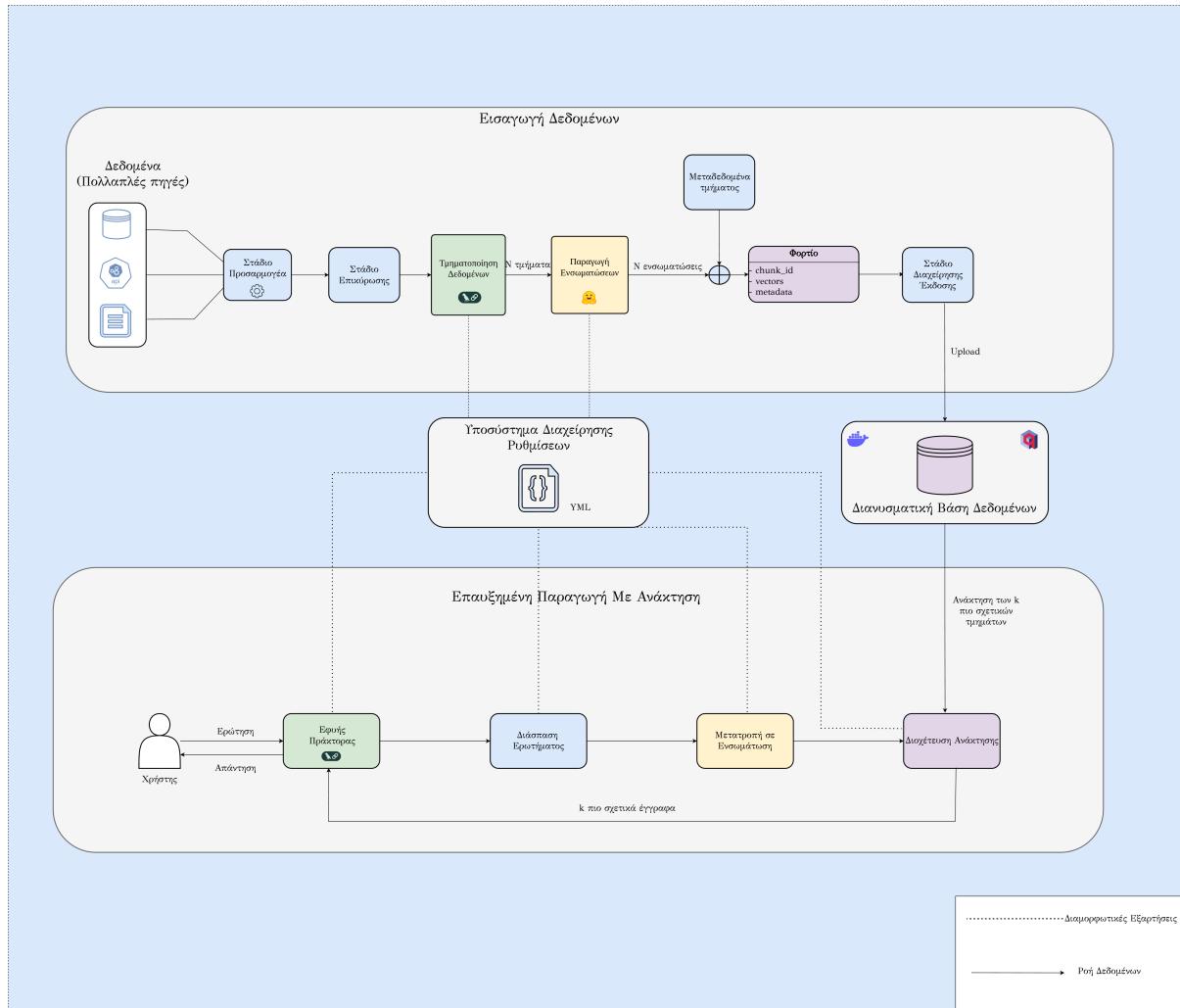
Η αρχιτεκτονική του προτεινόμενου συστήματος βασίζεται σε δύο θεμελιώδεις αγωγούς που συνιστούν τον πυρήνα της λειτουργικότητάς του. Ο πρώτος αγωγός αφορά την εισαγωγή και επεξεργασία δεδομένων (data ingestion pipeline), ενώ ο δεύτερος περιλαμβάνει την ανάκτηση πληροφορίας και τη διαδικασία παραγωγής απαντήσεων (retrieval and generation pipeline). Αυτοί οι δύο αγωγοί συνδέονται μέσω μιας κοινής υποδομής διαχείρισης διανυσματικών αναπαραστάσεων, η οποία υλοποιείται μέσω της διανυσματικής βάσης δεδομένων, διασφαλίζοντας τη συνοχή, την αποδοτικότητα και την κλιμακωσιμότητα του συστήματος.

Ο αγωγός εισαγωγής δεδομένων ακολουθεί μια σταδιακή προσέγγιση μετασχηματισμού των ακατέργαστων εγγράφων σε δομημένες διανυσματικές ενσωματώσεις (embeddings). Κάθε στάδιο του αγωγού υλοποιείται μέσω εξειδικευμένων δομοστοιχείων που επικοινωνούν μέσω τυποποιημένων διεπαφών, επιτρέποντας την απρόσκοπτη αντικατάσταση ή επέκταση της λειτουργικότητας χωρίς επιπτώσεις στον πυρήνα του συστήματος. Αντίστοιχα, ο αγωγός ανάκτησης και παραγωγής ενορχηστρώνεται από έναν ευφυή πράκτορα (intelligent agent) βασισμένο στο πλαίσιο LangGraph. Ο πράκτορας ερμηνεύει τα ερωτήματα των χρηστών, αποφασίζει δυναμικά για την ανάγκη ανάκτησης εξωτερικής πληροφορίας, και συνθέτει συνεκτικές απαντήσεις αξιοποιώντας τις δυνατότητες των μεγάλων γλωσσικών μοντέλων σε συνδυασμό με το ανακτημένο πλαίσιο.

Το σύστημα έχει σχεδιαστεί ακολουθώντας τη φιλοσοφία της διαμόρφωσης ως κώδικα (Configuration as Code), επιτυγχάνοντας τον πλήρη διαχωρισμό μεταξύ διαμόρφωσης και υλοποίησης. Μέσω δηλωτικών αρχείων YAML και μιας εκτελέσιμης τερματικής διεπαφής (CLI), οι χρήστες μπορούν να πειραματιστούν με διαφορετικούς συνδυασμούς παραμέτρων, στρατηγικών τμηματοποίησης, μοντέλων ενσωμάτωσης και τεχνικών ανάκτησης, διασφαλίζοντας την επαναληψιμότητα των αποτελεσμάτων και την προσαρμοστικότητα του συστήματος σε διαφορετικά πεδία.

ΚΕΦΑΛΑΙΟ 4. ΥΛΟΠΟΙΗΣΗ

εφαρμογής.



Σχήμα 4.1: Αρχιτεκτονική του προτεινόμενου συστήματος RAG

4.1.1 Αγωγός Εισαγωγής Δεδομένων

Ο αγωγός εισαγωγής δεδομένων αποτελεί την κρίσιμη φάση κατά την οποία τα ακατέργαστα έγγραφα μετασχηματίζονται σε δομημένες διανυσματικές ενσωματώσεις, έτοιμες για αποδοτική ανάκτηση. Το σύστημα υλοποιεί μια επταστασταδιακή διαδικασία που διασφαλίζει την ποιότητα, την ιχνηλασιμότητα και την αναπαραγωγισμότητα της επεξεργασίας. Η αρχιτεκτονική σχεδιάστηκε με γνώμονα την αρθρωτότητα και την επεκτασιμότητα, επιτρέποντας την απρόσκοπτη ενσωμάτωση νέων πηγών δεδομένων και στρατηγικών επεξεργασίας χωρίς την ανάγκη τροποποίησης του κεντρικού πυρήνα του συστήματος.

Φόρτωση Παραμέτρων Διαμόρφωσης: το πρώτο στάδιο αφορά τη φόρτωση και επεξεργασία των παραμέτρων διαμόρφωσης. Κάθε σύνολο δεδομένων συνοδεύεται από ένα δηλωτικό αρχείο διαμόρφωσης που καθορίζει τον τύπο του προσαρμοστή (adapter type), τη στρατηγική τμηματοποίησης (chunking strategy), την προέλευση

των ενσωματώσεων (embedding provider), τις διαστάσεις των διανυσμάτων, καθώς και πλήθος άλλων παραμέτρων που επηρεάζουν τη συμπεριφορά της επεξεργασίας.

Ανάγνωση και Επικύρωση Δεδομένων: το δεύτερο στάδιο περιλαμβάνει την ανάγνωση των ακατέργαστων δεδομένων και την αρχική τους επικύρωση. Το σύστημα διαθέτει μια συλλογή εξειδικευμένων προσαρμοστών (adapters), καθένας από τους οποίους υλοποιεί το αφηρημένο πρότυπο `DatasetAdapter` και είναι υπεύθυνος για τον χειρισμό συγκεκριμένου τύπου δεδομένων. Οι προσαρμοστές ενθυλακώνουν τη λογική που απαιτείται για την ανάγνωση ετερογενών μορφών αρχείων, το φιλτράρισμα και την τυποποίηση των δεδομένων σε μια κοινή αναπαράσταση. Κάθε εγγραφή των ακατέργαστων δεδομένων μετασχηματίζεται αρχικά σε αντικείμενο τύπου `BaseRow`, το οποίο αποτελεί μια τυποποιημένη αναπαράσταση που περιέχει το κείμενο του εγγράφου, μεταδεδομένα προέλευσης και μοναδικούς αναγνωριστικούς κωδικούς. Στη συνέχεια, τα αντικείμενα `BaseRow` μετατρέπονται σε έγγραφα `LangChain` μέσω του προσαρμοστή, εμπλουτίζοντας τα με επιπλέον πληροφορίες πλαισίου και δομημένα μεταδεδομένα που διευκολύνουν την μεταγενέστερη ανάκτηση και ερμηνεία των αποτελεσμάτων.

Έχοντας ολοκληρώσει τη μετατροπή, τα έγγραφα υποβάλλονται σε αυστηρή διαδικασία επικύρωσης που υλοποιείται από την κλάση `DocumentValidator`. Η επικύρωση περιλαμβάνει τρεις κατηγορίες ελέγχων: επιβεβαίωση ότι το μήκος κάθε εγγράφου βρίσκεται εντός προκαθορισμένων ορίων, απομάκρυνση περιεχομένου HTML που ενδέχεται να επηρεάσει την ποιότητα των ενσωματώσεων, και ανίχνευση διπλότυπων εγγράφων βασιζόμενη σε κρυπτογραφικές συναρτήσεις κατακερματισμού τύπου SHA-256.

Τμηματοποίηση Εγγράφων: το τρίτο στάδιο αφορά την τμηματοποίηση (chunking) των επικυρωμένων εγγράφων σε μικρότερες σημασιολογικά συνεκτικές μονάδες κειμένου. Η τμηματοποίηση αποτελεί κρίσιμο σημείο στην αρχιτεκτονική, καθώς η επιλογή της κατάλληλης στρατηγικής επηρεάζει άμεσα την ποιότητα της ανάκτησης και την αποδοτικότητα του συστήματος. Η κλάση `ChunkingStrategyFactory` υποστηρίζει πολλαπλές στρατηγικές τμηματοποίησης, καθεμία βελτιστοποιημένη για συγκεκριμένους τύπους περιεχομένου. Η αναδρομική στρατηγική (recursive chunking) εφαρμόζει ιεραρχικό διαχωρισμό βασισμένο σε χαρακτήρες, δοκιμάζοντας διαδοχικά διαφορετικούς διαχωριστές μέχρι να επιτευχθεί το επιθυμητό μέγεθος τμήματος. Η σημασιολογική στρατηγική (semantic chunking) λαμβάνει υπόψη τα όρια των προτάσεων και των παραγράφων, διασφαλίζοντας ότι τα τμήματα διατηρούν σημασιολογική ακεραιότητα. Κάθε τμήμα που παράγεται εμπλουτίζεται με εκτεταμένα μεταδεδομένα που περιλαμβάνουν την αναφορά στο πηγαίο έγγραφο, τη θέση εντός του αρχικού κειμένου, το μέγεθος του τμήματος, και ένα ντετερμινιστικό αναγνωριστικό.

Παραγωγή Διανυσματικών Ενσωματώσεων: το τέταρτο στάδιο περιλαμβάνει την παραγωγή διανυσματικών ενσωματώσεων για κάθε τμήμα κειμένου. Το υποσύστημα `EmbeddingPipeline` υλοποιεί μια ευέλικτη αρχιτεκτονική που υποστηρίζει τρεις στρατηγικές ενσωμάτωσης. Η πυκνή στρατηγική (dense embedding) παράγει διανύσματα υψηλής διάστασης που αιχμαλωτίζουν τη σημασιολογική ομοιότητα μεταξύ κειμένων. Η αραιή στρατηγική (sparse embedding) δημιουργεί αραιά διανύσματα βασισμένα σε στατιστικές λέξεων-κλειδιών με προσεγγίσεις όπως BM25 και SPLADE++. Η υβριδική στρατηγική (hybrid embedding) συνδυάζει και τις δύο προ-

ΚΕΦΑΛΑΙΟ 4. ΥΛΟΠΟΙΗΣΗ

σεγγίσεις, επιτρέποντας στο σύστημα να αξιοποιεί τόσο τη σημασιολογική όσο και τη λεξιλογική ομοιότητα. Το υποσύστημα εφαρμόζει ομαδοποίηση των τμημάτων σε δέσμες (batching) για την αποδοτική επεξεργασία και διαθέτει ενσωματωμένη λογική προσωρινής αποθήκευσης (caching) για την αποφυγή επαναϋπολογισμού ενσωματώσεων.

Αποθήκευση, Επαλήθευση και Καταγραφή: το πέμπτο στάδιο αφορά την αποθήκευση των επεξεργασμένων τμημάτων μαζί με τις ενσωματώσεις τους στη διανυσματική βάση δεδομένων. Το δομοστοιχείο `VectorStoreUploader` διαχειρίζεται τη διαδικασία μεταφόρτωσης, διασφαλίζοντας την ταυτοδυναμία (idempotency) μέσω ντετερμινιστικών αναγνωριστικών. Κάθε τμήμα αποθηκεύεται ως ένα σημείο (point) στη βάση δεδομένων που περιέχει το πρωτότυπο κείμενο στο πεδίο δεδομένων (payload), τους διανυσματικούς του εκπροσώπους, και όλα τα σχετικά μεταδεδομένα. Το έκτο στάδιο περιλαμβάνει αυτοματοποιημένους ελέγχους επαλήθευσης (smoke tests) μέσω της κλάσης `SmokeTestRunner`. Τέλος, το έβδομο στάδιο αφορά την καταγραφή της πλήρους προέλευσης (lineage) και του ιστορικού της επεξεργασίας σε μορφή JSON που περιλαμβάνει το πλήρες αρχείο διαμόρφωσης, το αναγνωριστικό έκδοσης κώδικα, εκδόσεις βιβλιοθηκών, χρονοσήμανση σε μορφή ISO 8601, και στατιστικά επεξεργασίας.

```
dataset:
  name: "stackoverflow_sosum_bge_recursive"
  version: "v1.0.0"
  adapter: "pipelines.adapters.stackoverflow.StackOverflowAdapter"
  path: "datasets/sosum/data"

  chunking:
    strategy: "recursive"
    chunk_size: 500
    chunk_overlap: 100
    separators: ["\n\n", "\n", " ", ""]

  embedding:
    strategy: "hybrid"
    dense:
      provider: "hf"
      model: "BAAI/bge-m3"
      dimensions: 1024
      batch_size: 32
    sparse:
      provider: "sparse-splade"
      model: "prithivida/Splade_PP_en_v1"
      batch_size: 32

  qdrant:
    collection: "sosum_stackoverflow_bge_splade_recursive_v2"
    dense_vector_name: "dense"
    sparse_vector_name: "sparse"

  upload:
    batch_size: 50
    wait: true
    versioning: true

  validation:
    min_char_length: 10
    max_char_length: 50000
    remove_duplicates: true
    clean_html: true
```

Σχήμα 4.2: Τυπικό αρχείο YAML για ορισμό του αγωγού εισαγωγής δεδομένων

4.1.2 Αγωγός Ανάκτησης Πληροφορίας

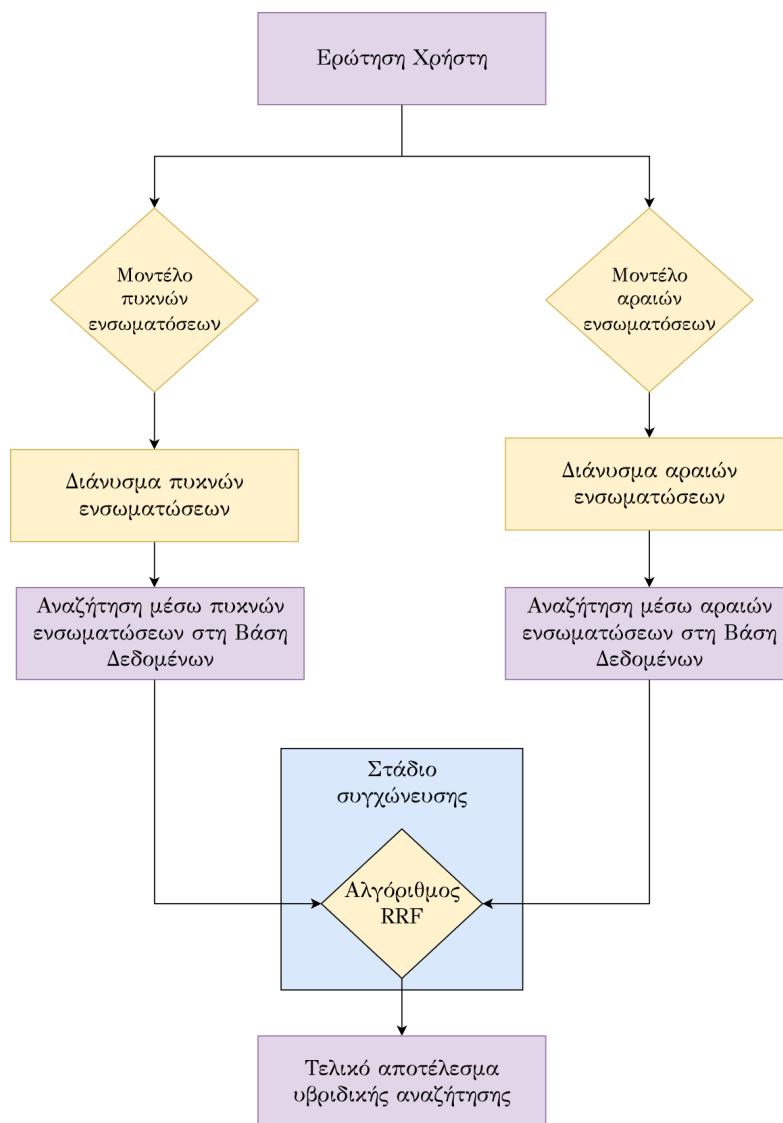
Ο αγωγός ανάκτησης πληροφορίας αποτελεί την κεντρική ενότητα της αρχιτεκτονικής που μεσολαβεί μεταξύ του ερωτήματος του χρήστη και της διανυσματικής βάσης δεδομένων, επιτελώντας την κρίσιμη λειτουργία του ταχέος και ακριβούς εντοπισμού των πιο σχετικών τμημάτων κειμένου. Το σύστημα υλοποιεί μια ευέλικτη αρχιτεκτονική αγωγού επεξεργασίας (retrieval pipeline) που επιτρέπει τη σύνθεση πολλαπλών σταδίων ανάκτησης και βελτίωσης αποτελεσμάτων, διαμορφώσιμη μέσω δηλωτικών αρχείων YAML.

Αρχιτεκτονική και Στάδια: η βασική δομή του αγωγού ακολουθεί μια πεντασταδιακή διαδικασία: επεξεργασία ερωτήματος και παραγωγή διανυσματικών αναπαραστάσεων (query encoding), αρχική ανάκτηση από τη διανυσματική βάση δεδομένων (retrieval), φίλτραρισμα βασισμένο σε κατώφλια βαθμολογίας (filtering), αναδιάταξη για βελτίωση της σειράς των αποτελεσμάτων (reranking), και συναρμολόγηση τελικών εγγράφων (result assembly). Η κεντρική κλάση RetrievalPipeline ενθυλακώνει τη λογική ενορχήστρωσης όλων των σταδίων, υλοποιώντας το σχεδιαστικό πρότυπο της αλυσίδας ευθύνης (Chain of Responsibility pattern), όπου κάθε δομοστοιχείο υλοποιεί την αφηρημένη διεπαφή RetrievalComponent.

Στρατηγικές Ανάκτησης: το σύστημα υποστηρίζει τρεις θεμελιώδεις στρατηγικές ανάκτησης. Η πυκνή στρατηγική (dense retrieval) βασίζεται στον υπολογισμό της ομοιότητας συνημιτόνου μεταξύ πυκνών διανυσματικών ενσωματώσεων υψηλής διάστασης, χρησιμοποιώντας αποδοτικούς αλγορίθμους όπως ο αλγόριθμος εραρχικού πλοηγήσιμου μικρού κόσμου (HNSW) [83]. Η αραιή στρατηγική (sparse retrieval) εφαρμόζει προσέγγιση βασισμένη σε στατιστικές λέξεων-κλειδιών όπως το BM25 [16] και εξελιγμένα νευρωνικά μοντέλα όπως το SPLADE++ [84]. Η υβριδική στρατηγική (hybrid retrieval) συνδυάζει τις δύο προηγούμενες προσεγγίσεις, εκτελώντας παράλληλα πυκνή και αραιή αναζήτηση και συνδυάζοντας τα αποτελέσματα μέσω του αλγορίθμου Reciprocal Rank Fusion.

Φιλτράρισμα και Αναδιάταξη: μετά την αρχική ανάκτηση, η κλάση ScoreFilter εφαρμόζει φιλτράρισμα βασισμένο σε κατώφλια βαθμολογίας για την απομάκρυνση αποτελεσμάτων χαμηλής ποιότητας. Το στάδιο αναδιάταξης (reranking) χρησιμοποιεί εξειδικευμένα μοντέλα νευρωνικών δικτύων μέσω της κλάσης RerankerComponent. Σε αντίθεση με τους αρχικούς μηχανισμούς ανάκτησης που υπολογίζουν ανεξάρτητες αναπαραστάσεις (αρχιτεκτονική διπλού κωδικοποιητή, bi-encoder), τα μοντέλα αναδιάταξης επεξεργάζονται το ερώτημα και το έγγραφο από κοινού (αρχιτεκτονική διασταυρούμενου κωδικοποιητή, cross-encoder), επιτρέποντας τη μοντελοποίηση λεπτών αλληλεπιδράσεων μέσω μηχανισμών προσοχής (attention mechanisms).

ΚΕΦΑΛΑΙΟ 4. ΥΛΟΠΟΙΗΣΗ



Σχήμα 4.3: Υβριδικό σύστημα ανάκτησης από διανυσματική βάση δεδομένων

```

pipelines > configs > retrieval > ! hybrid_optimal.yml
 1  # =====
 2  # Hybrid Retrieval - BGE-M3 + SPLADE (Alpha=0.8, RRF k=20)
 3  # =====
 4  # Dense semantic retrieval (80%) + sparse keyword matching (20%)
 5  # Uses Reciprocal Rank Fusion with k=20
 6  # =====
 7
 8  # === Embedding Configuration ===
 9  embedding:
10    dense:
11      provider: huggingface
12      model: BAAI/bge-m3
13      model_kwargs:
14        device: cpu
15      encode_kwargs:
16        normalize_embeddings: true
17
18    sparse:
19      provider: sparse-splade
20      model: prithivida/Splade_PP_en_v1
21
22  # === Qdrant Configuration ===
23  qdrant:
24    host: localhost
25    port: 6333
26    collection_name: sosum_stackoverflow_bge_splade_recursive_v2
27    dense_vector_name: dense
28    sparse_vector_name: sparse
29
30  fusion:
31    method: rrf          # Reciprocal Rank Fusion
32    alpha: 0.8           # 80% dense, 20% sparse (applies weight to RRF scores)
33    rrf_k: 20            # RRF constant (lower = more weight to top ranks)
34
35  # === Retrieval Pipeline ===
36  retrieval_pipeline:
37    retriever:
38      type: hybrid
39      top_k: 10
40
41    stages:
42      - type: score_filter
43        config:
44          min_score: 0.0

```

Σχήμα 4.4: Τυπικό αρχείο YAML για ορισμό του αγωγού ανάκτησης

4.1.3 Σύστημα Ευφυούς Πράκτορα

Το σύστημα ευφυούς πράκτορα αποτελεί το ανώτερο επίπεδο της αρχιτεκτονικής που ενορχηστρώνει τη συνολική αλληλεπίδραση με τον χρήστη, λαμβάνοντας αποφάσεις σχετικά με την ανάγκη ανάκτησης πληροφορίας και συνθέτοντας συνεχικές απαντήσεις. Το σύστημα υλοποιείται χρησιμοποιώντας το πλαίσιο LangGraph, που επιτρέπει τη μοντελοποίηση σύνθετων ροών εργασίας ως κατευθυνόμενων γραφημάτων με διαχείριση κατάστασης (stateful directed graphs).

Η αρχιτεκτονική βασίζεται στο σχεδιαστικό πρότυπο της μηχανής πεπερασμένων καταστάσεων (Finite State Machine), όπου κάθε κατάσταση αναπαριστά την πρόοδο επεξεργασίας ενός ερωτήματος. Η κατάσταση του συστήματος ενθυλακώνει το αρχικό ερώτημα του χρήστη, τις αποφάσεις δρομολόγησης, το ανακτημένο πλαίσιο, την παραγόμενη απάντηση, και το ιστορικό συνομιλίας. Το γράφημα ροής εργασίας αποτελείται από τέσσερα κύρια στάδια: ανάλυση και ταξινόμηση ερωτήματος όπου το γλωσσικό μοντέλο καθορίζει εάν απαιτείται ανάκτηση, ανάκτηση πληροφορίας που ενεργοποιείται υποθετικά, παραγωγή απάντησης με προσαρμοστική συμπεριφορά (adaptive prompting) ανάλογα με την παρουσία πλαισίου, και δια-

χείριση ιστορικού συνομιλίας με παράθυρο μεταβλητού μεγέθους (sliding window) για πολυστροφικούς διαλόγους.

4.2 ΤΕΧΝΟΛΟΓΙΚΕΣ ΕΠΙΛΟΓΕΣ

4.2.1 Διανυσματική Βάση Δεδομένων

Το σύστημα χρησιμοποιεί τη βάση δεδομένων Qdrant, μια λύση ανοικτού κώδικα εξειδικευμένη στην αποθήκευση και αναζήτηση διανυσματικών αναπαραστάσεων υψηλής διάστασης. Η Qdrant επιλέχθηκε έναντι εναλλακτικών όπως η Pinecone, η Weaviate και η Milvus για τέσσερις κύριους λόγους: εξαιρετική απόδοση σε εφαρμογές πραγματικού χρόνου με χρόνο αναζήτησης της τάξεως των χιλιοστών του δευτερολέπτου, υποστήριξη προηγμένων τεχνικών κβαντοποίησης διανυσμάτων που μειώνουν το memory footprint, πλήρη υποστήριξη για υβριδική αναζήτηση που συνδυάζει πυκνά και αραιά διανύσματα εντός της ίδιας συλλογής, και ευελιξία τοπικής εκτέλεσης χωρίς εξάρτηση από εξωτερικές υπηρεσίες cloud. Για την ανάπτυξη, η Qdrant εκτελείται σε Docker container με persistent volumes που διασφαλίζουν τη διατήρηση των δεδομένων.

4.2.2 Πλαίσιο Ανάπτυξης

Το LangGraph επιλέχθηκε ως το κεντρικό πλαίσιο για την υλοποίηση του πράκτορα, παρέχοντας δηλωτική προσέγγιση (declarative approach) για τον ορισμό γραφημάτων ροής εργασίας. Η Python επιλέχθηκε ως γλώσσα υλοποίησης λόγω του πλούσιου οικοσυστήματός της σε βιβλιοθήκες μηχανικής μάθησης και επεξεργασίας φυσικής γλώσσας, ενώ η Pydantic v2 χρησιμοποιείται για την επικύρωση δεδομένων και τη διασφάλιση συνέπειας μέσω type hints και runtime validation.

4.2.3 Υποδομή Γλωσσικού Μοντέλου

Για τη διαδικασία παραγωγής κειμένου, το σύστημα χρησιμοποιεί την πλατφόρμα Ollama σε συνδυασμό με το γλωσσικό μοντέλο Llama3.1 8B. Το Ollama αποτελεί εργαλείο αγορικού κώδικα που επιτρέπει την τοπική εκτέλεση μεγάλων γλωσσικών μοντέλων με βελτιστοποιημένο κβαντισμό παραμέτρων, διασφαλίζοντας την αυτονομία του συστήματος και την προστασία των δεδομένων. Το Llama3.1 διαθέτει 8 δισεκατομμύρια παραμέτρους και παράθυρο πλαισίου 128,000 tokens και υλοποιεί αυτοπαλίνδρομη παραγωγή κειμένου βασιζόμενο στα ανακτημένα έγγραφα από τη βάση δεδομένων, παράγοντας συνεκτικές απαντήσεις με ελαχιστοποίηση του φαινομένου των παραισθήσεων.

5

Πειράματα και Αποτελέσματα

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστεί μια σειρά πειραμάτων που επιδιώκουν την αξιολόγηση και τη βελτιστοποίηση του συστήματος ανάκτησης και παραγωγής απαντήσεων, πάνω σε ένα πραγματικό σύνολο δεδομένων. Ο σκοπός είναι η δοκιμή μερικών από των υποστηριζόμενων τεχνικών ανάκτησης του συστήματος και η μελέτη της επίδρασης τους πάνω σε ένα πραγματικό σύστημα επαυξημένης παραγωγής με ανάκτηση. Η διαδικασία αξιολόγησης περιλαμβάνει τόσο αντικειμενικές μετρικές, όπως αυτές που παρουσιάστηκαν εκτενώς στην ενότητα 3.4.3, αλλά και μια σειρά από εξειδικευμένες μετρικές που αξιολογούν την επίδοση του συστήματος από την μεριά του τελικού χρήστη.

5.1 ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΟΜΕΑΣ ΕΦΑΡΜΟΓΗΣ

Ένα από τα χαρακτηριστικά της παρούσας εργασίας αφορά την επιλογή του συνόλου δεδομένων για την αξιολόγηση του προτεινόμενου συστήματος. Αντί της υιοθέτησης κάποιου από τα κλασικά συνθετικά μετροπρογράμματα (benchmarks) που χρησιμοποιούνται συνήθως για την αξιολόγηση συστημάτων RAG, όπως τα MS MARCO [85], Natural Questions [86], ή HotpotQA [87], επιλέχθηκε η αξιολόγηση σε πραγματικά δεδομένα από τον τομέα της τεχνολογίας λογισμικού. Η επιλογή αυτή υπαγορεύθηκε από τη στόχευση να αξιολογηθεί η ικανότητα του συστήματος να λειτουργεί αποτελεσματικά σε πραγματικές συνθήκες εφαρμογής, όπου τα δεδομένα παρουσιάζουν υψηλό βαθμό τεχνικότητας, εξειδίκευσης και δομικής πολυπλοκότητας.

5.1.1 Προέλευση και Χαρακτηριστικά Δεδομένων

Το σύνολο δεδομένων προέρχεται από την πρόσφατη εργασία του Θεμιστοκλή Διαμαντόπουλου και του Ανδρέα Συμεωνίδη [88], η οποία δημιούργησε έναν συστη-

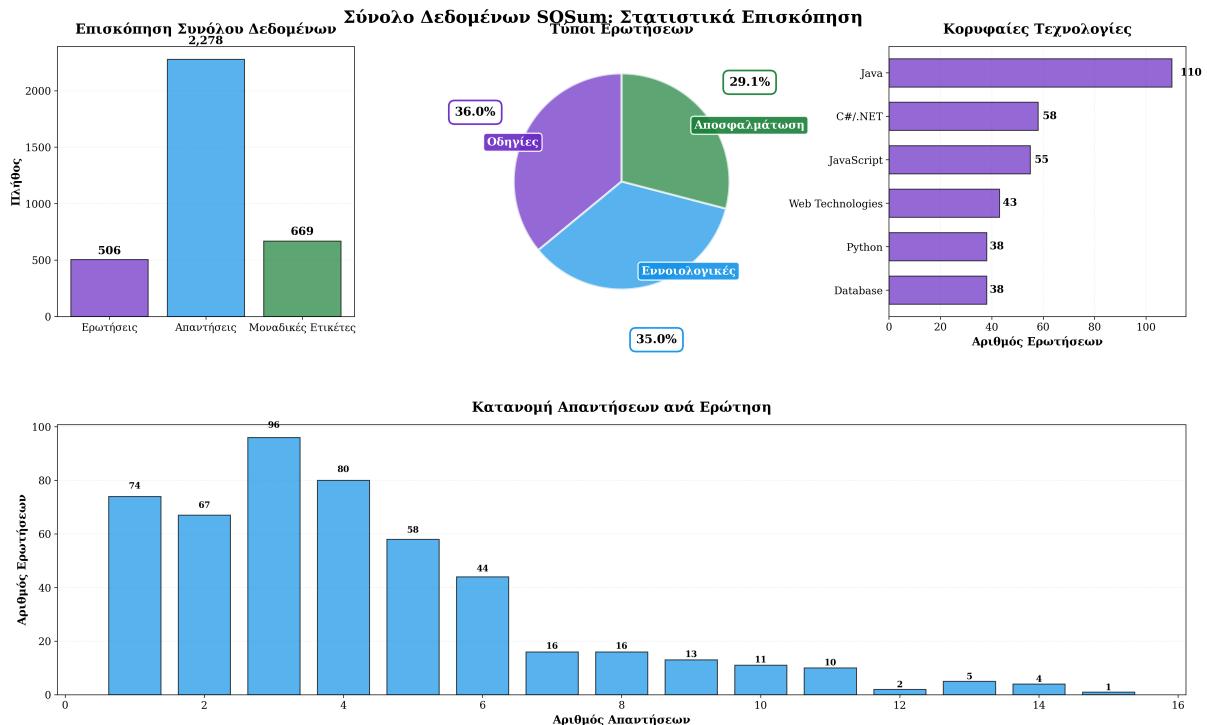
ματικό κατάλογο (directory) με τα papers και τα αντίστοιχα datasets που παρουσιάστηκαν στο data track του συνεδρίου Mining Software Repositories (MSR) κατά τα τελευταία δώδεκα έτη. Το MSR conference αποτελεί ένα από τα κορυφαία διεθνή συνέδρια στον τομέα της εμπειρικής τεχνολογίας λογισμικού, με έμφαση στην εξόρυξη και ανάλυση δεδομένων από αποθετήρια κώδικα, συστήματα παρακολούθησης σφαλμάτων (bug trackers), και άλλες πηγές που σχετίζονται με την ανάπτυξη λογισμικού. Ο κατάλογος περιλαμβάνει μεταδεδομένα και πληροφορίες αναφοράς για όλα τα papers των data tracks, καθώς και χαρακτηρισμό των datasets σύμφωνα με την πηγή δεδομένων και την συμμόρφωσή τους με τις αρχές FAIR (Findable, Accessible, Interoperable, Reusable). Τα datasets που συλλέχθηκαν κατηγοριοποιούνται σε διάφορους τύπους, συμπεριλαμβανομένων:

- **Αποθετήρια κώδικα (Code Repositories):** Δεδομένα από πλατφόρμες όπως GitHub, GitLab, ή Bitbucket, περιλαμβάνοντα ιστορικό commits, pull requests, και code reviews.
- **Συστήματα παρακολούθησης ζητημάτων (Issue Tracking):** Αναφορές σφαλμάτων, αιτήματα χαρακτηριστικών, και τεχνικές συζητήσεις από συστήματα όπως Jira, Bugzilla, ή GitHub Issues.
- **Τεχνική τεκμηρίωσης:** README files, documentation pages, API references, και technical specifications.
- **Συζητήσεις κοινότητας:** Δεδομένα από Stack Overflow, mailing lists, και άλλες πλατφόρμες επικοινωνίας προγραμματιστών.
- **Άλλοι τύποι:** Dockerfiles, Jupyter notebooks, κείμενα αδειών χρήσης, και άλλα εξειδικευμένα τεχνικά έγγραφα.

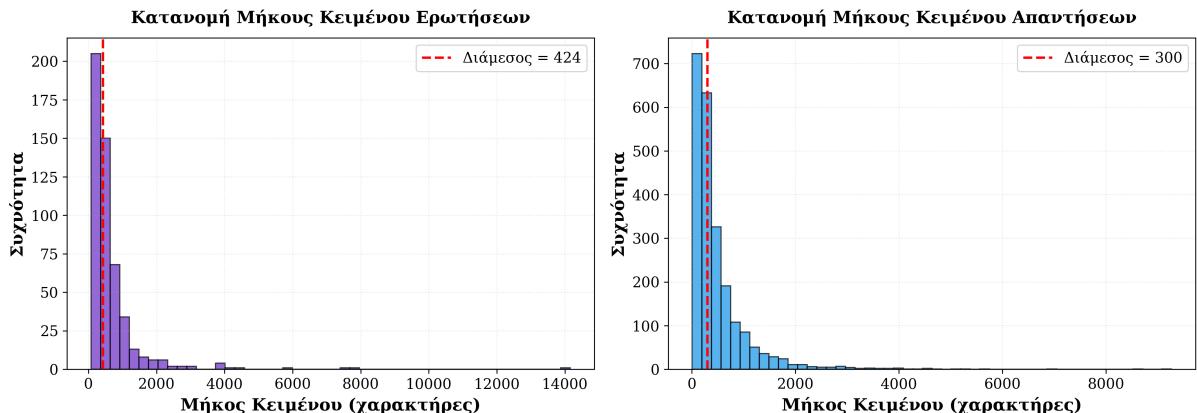
5.1.2 Επιλογή συνόλου δεδομένων

Από το ευρύτερο directory datasets του MSR conference [88], επιλέχθηκε το σύνολο δεδομένων SOSum: A dataset of extractive summaries of Stack Overflow posts and associated labeling tools[89], με FAIR score: 27.08% για την αξιολόγηση του προτεινόμενου συστήματος. Το SOSum αποτελεί ένα dataset που περιλαμβάνει 2,278 δημοφιλείς αναρτήσεις-απαντήσεις από το Stack Overflow με χειροκίνητα επισημειωμένες περιληπτικές προτάσεις. Οι ερωτήσεις στο SOSum καλύπτουν 669 διαφορετικά tags με διάμεσο αριθμό προβολών 253K και διάμεσο post score 17, αντανακλώντας υψηλής ποιότητας τεχνικό περιεχόμενο.

5.1. ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΟΜΕΑΣ ΕΦΑΡΜΟΓΗΣ



Σχήμα 5.1: Το σύνολο δεδομένων SOSum περιληπτικά



Σχήμα 5.2: Κατανομή μήκους κειμένου ερωτήσεων και απαντήσεων στο σύνολο δεδομένων

Κριτήρια Επιλογής

Η επιλογή του **SOSum** dataset βασίστηκε σε τρεις κύριους λόγους που το καθιστούν κατάλληλο για την αξιολόγηση συστημάτων RAG στον τομέα της τεχνολογίας λογισμικού. Πρώτον, περιλαμβάνει πυκνό σύνολο τεχνικών ερωτήσεων και απαντήσεων που αντικατοπτρίζουν ρεαλιστικές πληροφοριακές ανάγκες προγραμματιστών σε ποικιλία γλωσσών, εργαλείων και πλαισίων. Δεύτερον, η δομή του dataset ευνοεί την ανάπτυξη συστημάτων ερώτησης-απάντησης, καθώς κάθε ερώτηση συνοδεύεται από πολλαπλές απαντήσεις διαφορετικής ποιότητας και λεπτομέρειας.

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

ενώ οι χειροκίνητες επισημειώσεις παρέχουν αντικειμενικό μέτρο αξιολόγησης. Τρίτον, παρέχεται σε δομημένη μορφή με πλήρη μεταδεδομένα, διευκολύνοντας την ενσωμάτωσή του στο σύστημα χωρίς ανάγκη εκτεταμένης προεπεξεργασίας ή καθαρισμού δεδομένων.

Περιορισμοί και Προκλήσεις

Παρά τα πλεονεκτήματά του, το **SOSum dataset** παρουσιάζει έναν βασικό περιορισμό που επηρεάζει την αξιολόγηση του συστήματος. Οι απαντήσεις του Stack Overflow κατατάσσονται μέσω προεπιλεγμένης σειράς κατάταξης του Stack Exchange, το οποίο βασίζεται κυρίως στη φημοφορία της κοινότητας και στην αποδοχή από τον ερωτώντα. Ωστόσο, η κατάταξη αυτή δεν αποτελεί πάντα πραγματική σημασιολογική ταξινόμηση, καθώς επηρεάζεται από παράγοντες όπως ο χρόνος δημοσίευσης, η φήμη του χρήστη και η σαφήνεια της διατύπωσης, ανεξάρτητα από την τεχνική πληρότητα της απάντησης. Αυτό δυσχεραίνει την εφαρμογή *rank-aware* μετρικών αξιολόγησης, όπως το **Normalized Discounted Cumulative Gain (NDCG)**, καθώς δεν είναι εγγυημένο ότι η πρώτη απάντηση αντιστοιχεί στην πιο σχετική. Παρά τον θόρυβο που εισάγει αυτή η αβεβαιότητα, η χρήση πραγματικών δεδομένων με τις εγγενείς τους ατέλειες θεωρείται προτιμότερη από τεχνητά benchmarks, καθώς αποτυπώνει πιο ρεαλιστικά τις συνθήκες παραγωγικής χρήσης.

5.2 ΔΙΕΞΑΓΩΓΗ ΠΕΙΡΑΜΑΤΩΝ

Η σειρά πειραμάτων που θα ακολουθήσει επιδιώκει να απαντήσει στα εξής ερευνητικά ερωτήματα. α) Ποιες είναι οι επιλογές ανάκτησης και από αυτές ποιες αποδίδουν καλύτερα στο σύνολο δεδομένων; β) έχοντας ένα σημείο αναφοράς, γίνεται να βελτιωθούν οι επιδώσεις με κατάλληλη επιλογή υπερπαραμέτρων; γ) πόσο βοήθησε το σύστημα ανάκτησης την παραγωγή απαντήσεων και κατά συνέπεια τον τελικό χρήστη; Όλα τα πειράματα εκτελέστηκαν στο ίδιο υπολογιστικό περιβάλλον, ώστε να διασφαλιστούν η συγχρισμότητα και η αναπαραγωγικότητα.

Υπολογιστικό περιβάλλον

Στοιχείο	Περιγραφή
Επεξεργαστής	AMD Ryzen 7 2700X (8 πυρήνες, 16 threads)
Κάρτα Γραφικών	NVIDIA GTX 1080 8GB
Μνήμη	32 GB DDR4 RAM
Αποθήκευση	SSD NVMe 1 TB
Λειτουργικό σύστημα	Manjaro Linux

Πίνακας 5.1: Προδιαγραφές του υπολογιστικού περιβάλλοντος που χρησιμοποιήθηκε για τη διεξαγωγή των πειραμάτων.

5.3. ΠΕΙΡΑΜΑ 1: ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ

*Υποσημείωση: η κάρτα γραφικών χρησιμοποιήθηκε μόνο στο πείραμα 3.

Λογισμικό και ρυθμίσεις

Η υλοποίηση έγινε σε **Python 3.13.7**. Για τη δημιουργία και ανάκτηση των διανυσματικών αναπαραστάσεων χρησιμοποιήθηκαν αποκλειστικά **ανοιχτού κώδικα και τοπικές (local) ενσωματώσεις**, χωρίς κόστος API κλήσεων. Συγκεκριμένα, χρησιμοποιήθηκαν τα εξής μοντέλα:

Κατηγορία	Περιγραφή / Τιμή
Πυκνές ενσωματώσεις	BAAI/bge-m3[90]
Αραιές ενσωματώσεις επέκτασης	prithivida/Splade_PP_en_v1[91]
Αραιές ενσωματώσεις	BM25
<i>top-k</i>	10
Μέγεθος τμημάτων	500 tokens
Επικάλυψη τμημάτων	100 tokens
Στρατηγική τμηματοποίησης	Αναδρομικός διαχωρισμός χαρακτήρων
Διάσταση Πυκνών Ενσωματώσεων	1024
Έκδοση Python	3.13.7

Πίνακας 5.2: Ρυθμίσεις λογισμικού και παραμέτρων του πειραματικού περιβάλλοντος.

Η παράμετρος **top-k** ορίστηκε $k = 10$ για όλα τα πειράματα. Για την εισαγωγή δεδομένων χρησιμοποιήθηκε μέγεθος τμήματος **500 tokens** με αλληλοεπικάλυψη **100 tokens** και με στρατηγική **αναδρομικού διαχωρισμού χαρακτήρων** (recursive character splitting). Οι συγκεκριμένες τιμές βρέθηκαν εμπειρικά να προσφέρουν καλή ισορροπία μεταξύ ακρίβειας ανάκτησης και υπολογιστικής αποδοτικότητας στο σύνολο δεδομένων SOSum. Η σταθερότητα των αποτελεσμάτων διασφαλίστηκε με κοινές ρυθμίσεις σε όλους τους retrievers και σταθερό random seed.

5.3 ΠΕΙΡΑΜΑ 1: ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ

Το πρώτο πείραμα έχει ως απώτερο σκοπό την δημιουργία μίας αναφοράς για τις μεθόδους ανάκτησης μειώνοντας έτσι δραστικά των χώρο των επιθυμητών λύσεων.

5.3.1 Μεθοδολογία Αξιολόγησης

Δημιουργία Συνόλου Αναφοράς σε Επίπεδο Τμημάτων

Για την αξιολόγηση των συστημάτων ανάκτησης πληροφορίας, δημιουργήθηκε σύνολο αναφοράς (ground truth) σε επίπεδο τμημάτων εγγράφων (chunks), όχι σε επίπεδο ολόκληρων εγγράφων. Το σύνολο δεδομένων SOSum περιέχει 506 ερωτήματα, και κάθε έγγραφο (Stack Overflow answer) χωρίζεται σε τμήματα μεγέθους 500 tokens με επικάλυψη 100 tokens. Για τη δημιουργία της χαρτογράφησης (mapping) από το πεδίο των εγγράφων στο πεδίο των τμημάτων, εφαρμόστηκε η ακόλουθη λογική: αν ένα έγγραφο D χαρακτηρίζεται ως σχετικό για ένα ερώτημα Q στο αρχικό dataset, τότε όλα τα τμήματα που προέρχονται από το D θεωρούνται σχετικά για το Q .

Διαδικασία Αξιολόγησης

Η αξιολόγηση πραγματοποιείται αποκλειστικά σε επίπεδο τμημάτων. Κάθε σύστημα ανάκτησης επιστρέφει τα top- k τμήματα για κάθε ερώτημα. Ένα ανακτημένο τμήμα χαρακτηρίζεται ως σχετικό αν προέρχεται από έγγραφο που είναι σχετικό με το ερώτημα. Οι μετρικές απόδοσης (Precision@k, Recall@k, F1-score, MAP, MRR, NDCG@k, Success@k) υπολογίζονται με βάση τον αριθμό των σχετικών τμημάτων στα ανακτημένα αποτελέσματα. Το πείραμα γίνεται σε όλο το σύνολο των δεδομένων χωρίς κάποιον διαχωρισμό.

5.3.2 Στρατηγικές Ανάκτησης

Αξιολογούνται πέντε βασικές στρατηγικές ανάκτησης χωρίς την εφαρμογή αναδιάταξης (reranking):

1. **BM25 Baseline** - Παραδοσιακή λεξικογραφική μέθοδος
2. **SPLADE Baseline** - Νευρωνική αραιή μέθοδος με επέκταση όρων (learned term expansion)
3. **Dense BGE-M3** - Πυκνή σημασιολογική ανάκτηση με διανυσματικές ενσωματώσεις
4. **Hybrid SPLADE + BGE-M3 (Hybrid S)** - Υβριδική προσέγγιση με συνδυασμό αραιών και πυκνών αναπαραστάσεων μέσω RRF
5. **Hybrid BM25 + BGE-M3 (Hybrid B)** - Παραδοσιακή υβριδική προσέγγιση με συνδυασμό BM25 και πυκνής ανάκτησης μέσω RRF

Για τις υβριδικές στρατηγικές, η σταθερά του RRF ορίζεται σε $k = 60$ και οι δύο λίστες αποτελεσμάτων (αραιή και πυκνή) συνδυάζονται με ισοδύναμη συνεισφορά.

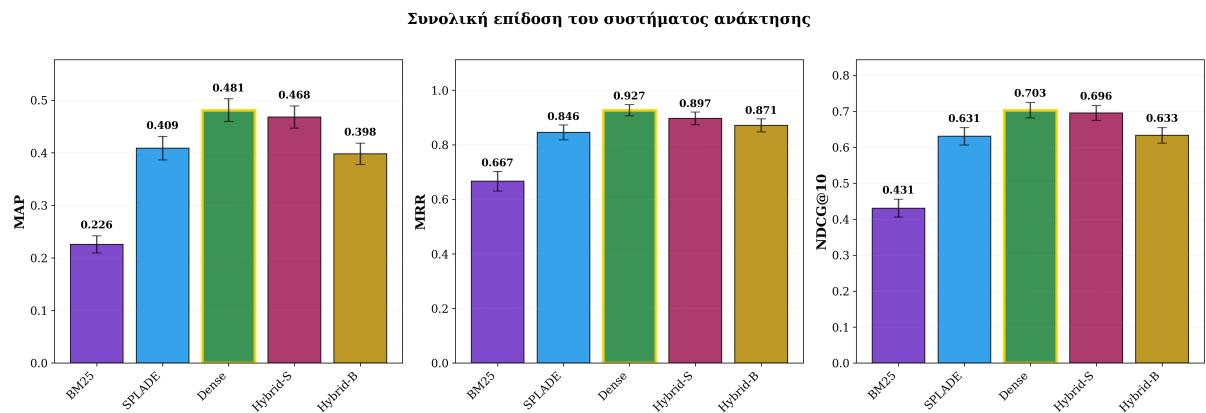
5.3.3 Επιλογή Μετρικών

Η αξιολόγηση εστιάζει κυρίως σε μετρικές βασισμένες σε σύνολα (set-based metrics) όπως Precision@k (3.6), Recall@k (3.7), F1-score (3.8) και MAP (3.12). Αυτή η επιλογή αιτιολογείται από τον περιορισμό του dataset: οι απαντήσεις του Stack Overflow ταξινομούνται με βάση ψηφοφορία της κοινότητας και όχι με χρήση αντικειμενικής αλήθειας (ground truth relevance), εισάγοντας θόρυβο στην κατάταξη.

5.3.4 Αποτελέσματα

Μέθοδος	Precision @5	Recall @5	F1 @5	MAP	MRR	NDCG @5	Latency (ms)
BM25	0.411 ± 0.326	0.205 ± 0.176	0.245 ± 0.179	0.226 ± 0.187	0.667 ± 0.411	0.458 ± 0.333	19.6 ± 131.045
SPLADE	0.585 ± 0.310	0.347 ± 0.232	0.390 ± 0.209	0.409 ± 0.256	0.846 ± 0.317	0.666 ± 0.301	118.3 ± 75.185
Dense	0.649 ± 0.304	0.410 ± 0.249	0.449 ± 0.214	0.481 ± 0.249	0.927 ± 0.240	0.744 ± 0.266	564 ± 769.545
Hybrid-Splade	0.652 ± 0.297	0.396 ± 0.235	0.440 ± 0.201	0.468 ± 0.242	0.897 ± 0.262	0.736 ± 0.272	737.4 ± 905.823
Hybrid-BM25	0.577 ± 0.296	0.349 ± 0.224	0.386 ± 0.186	0.398 ± 0.231	0.871 ± 0.276	0.657 ± 0.282	601.4 ± 786.131

Πίνακας 5.3: Αποτελέσματα αξιολόγησης retrievers: μέσος όρος ± τυπική απόκλιση για Precision@5, Recall@5, F1@5, MAP, MRR, NDCG@5 και Latency.

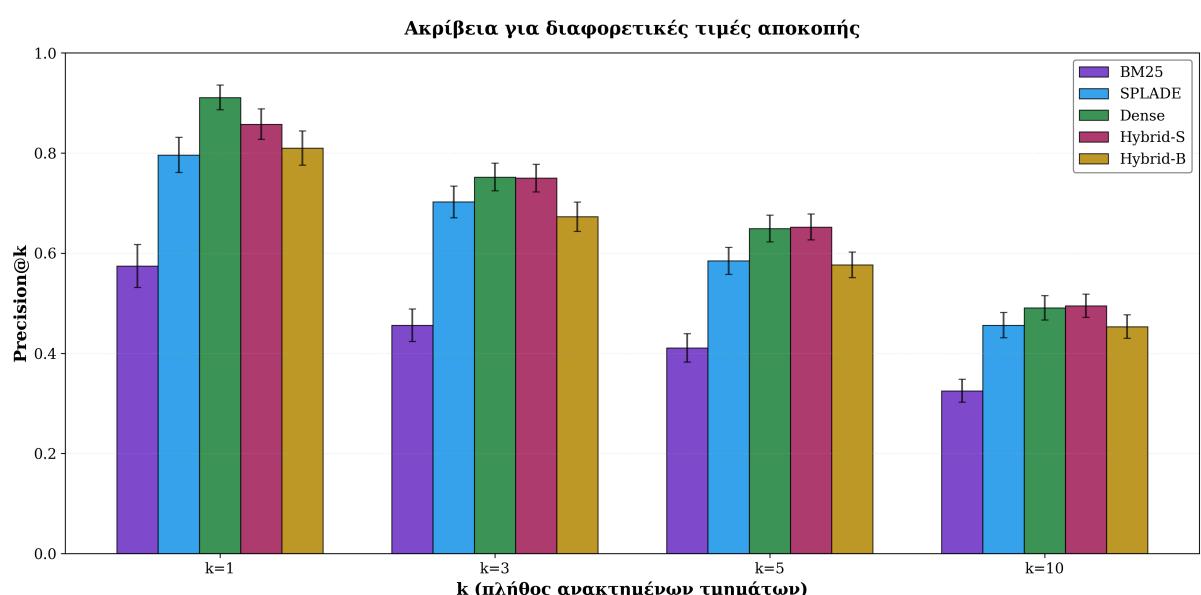


Σχήμα 5.3: Συνολική συνοπτική απόδοση του συστήματος

Το διάγραμμα 5.3 παρουσιάζει τη συγκριτική απόδοση των πέντε στρατηγικών ανάκτησης σε τρεις βασικές μετρικές. Παρατηρούνται τα ακόλουθα: Πρώτον,

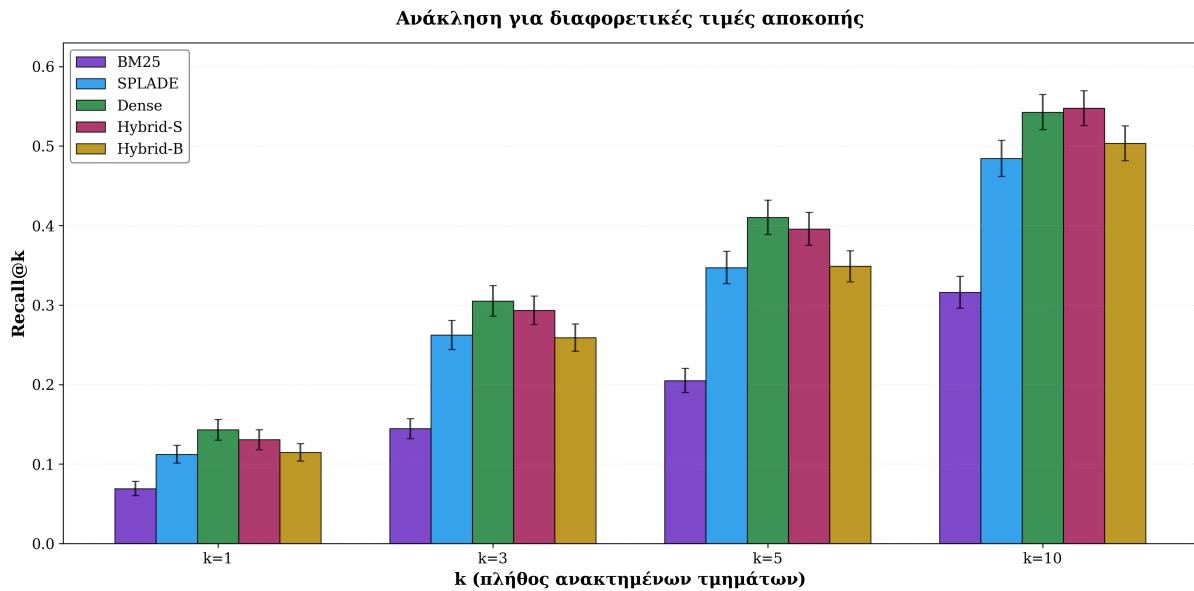
ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

η στρατηγική πυκνής ανάκτησης (Dense BGE-M3) επιδεικνύει την υψηλότερη απόδοση σε όλες τις μετρικές (MAP=0.481, MRR=0.927, NDCC@10=0.703), υπερτερώντας ακόμα και των υβριδικών προσεγγίσεων. Αυτό υποδηλώνει ότι για το συγκεκριμένο σύνολο δεδομένων τεχνικών ερωτημάτων, η σημασιολογική ομοιότητα που αιχμαλωτίζουν οι πυκνές ενσωματώσεις είναι περισσότερο κρίσιμη από την ακριβή λεξιλογική αντιστοίχιση. Δεύτερον, η νευρωνική αραιή μέθοδος SPLADE υπερτερεί σημαντικά της παραδοσιακής BM25 (MAP: 0.409 έναντι 0.226), επιβεβαιώνοντας την αξία της μαθημένης επέκτασης όρων. Η BM25 παρουσιάζει την χειρότερη απόδοση σε όλες τις μετρικές, με απόκλιση που ξεπερνά το 50% από την καλύτερη μέθοδο. Τρίτον, οι υβριδικές στρατηγικές δεν επιτυγχάνουν να ξεπεράσουν την πυκνή ανάκτηση, με τη διαμόρφωση Hybrid-S (SPLADE+Dense) να πλησιάζει αλλά να υστερεί ελαφρώς (MAP: 0.468 έναντι 0.481). Αυτό μπορεί να οφείλεται στη χρήση ίσων βαρών (0.5/0.5) στον αλγόριθμο RRF, που ενδεχομένως δεν είναι βέλτιστη για το συγκεκριμένο τομέα, ή στο ότι η προσθήκη αραιών ενσωματώσεων εισάγει θόρυβο χωρίς να προσθέτει συμπληρωματική πληροφορία.

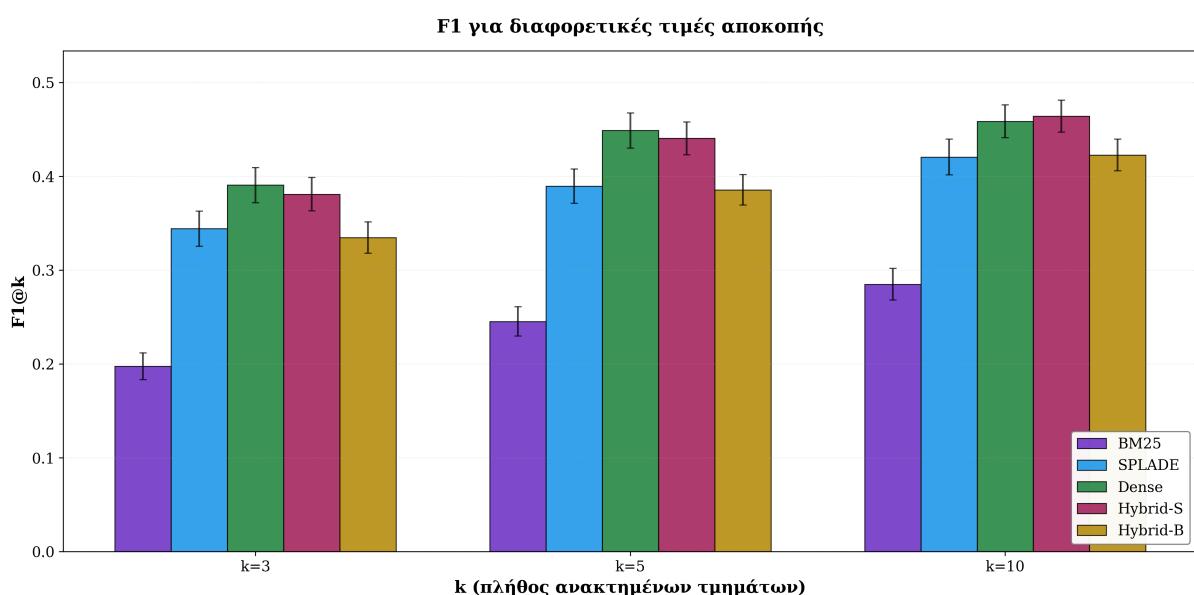


Σχήμα 5.4: Σύγκριση των διαμορφώσεων ως προς την μετρική: ακρίβεια

5.3. ΠΕΙΡΑΜΑ 1: ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ

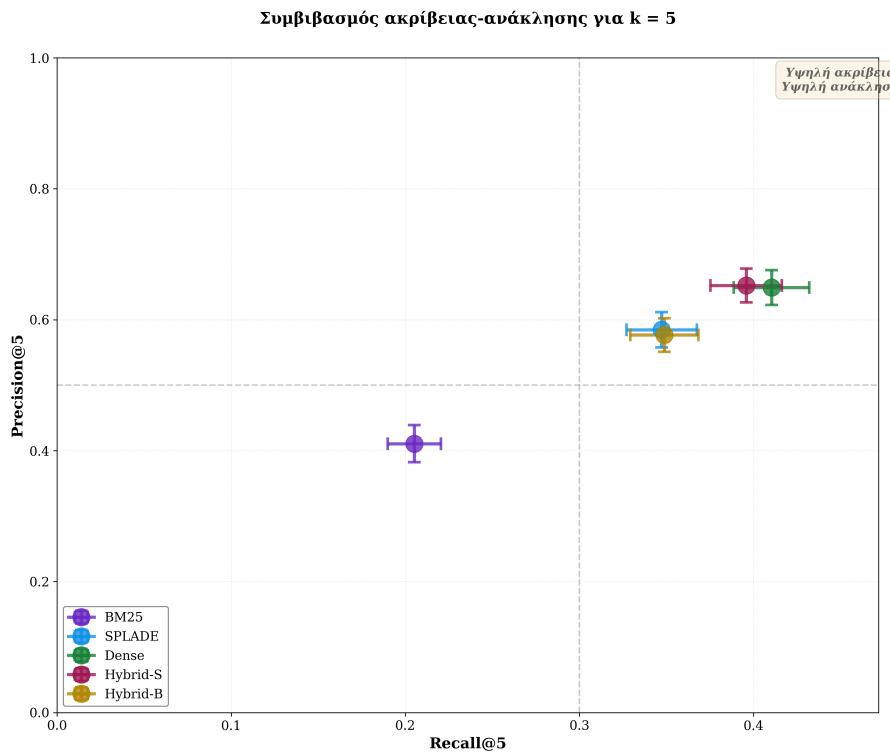


Σχήμα 5.5: Σύγκριση των διαμορφώσεων ως προς την μετρική: ανάκληση

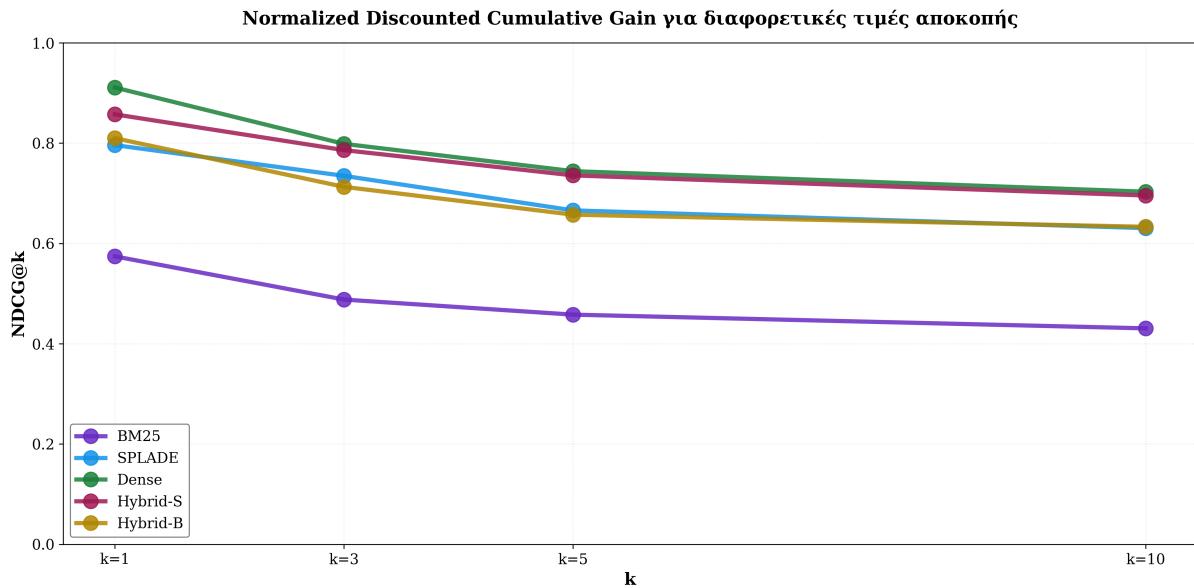


Σχήμα 5.6: Σύγκριση των διαμορφώσεων ως προς την μετρική: F1

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ



Σχήμα 5.7: Συμβιβασμός μεταξύ ανάκλησης και ακρίβειας



Σχήμα 5.8: Σύγκριση των διαμορφώσεων ως προς την μετρική: NDCG

Από το Σχήμα 5.4 (Precision@k) παρατηρείται ότι η πυκνή ανάκτηση (Dense) διατηρεί σταθερά την υψηλότερη ακρίβεια για όλες τις τιμές k, με ιδιαίτερα υψηλή απόδοση για k=1 που υποδηλώνει εξαιρετική ικανότητα τοποθέτησης σχετικών τμημάτων στην πρώτη θέση. Η αναμενόμενη πτώση της ακρίβειας καθώς αυξάνει το k

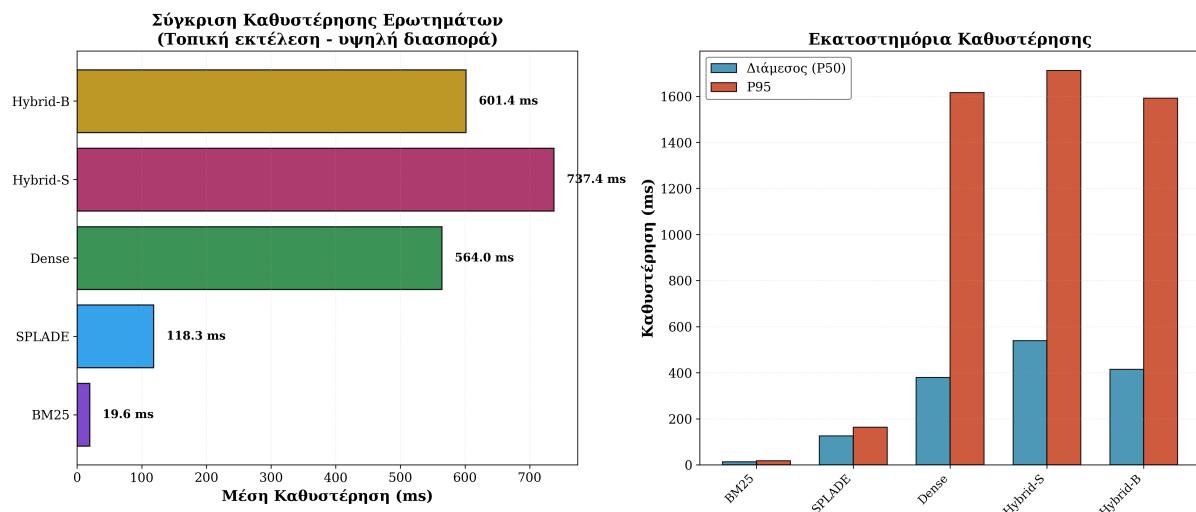
5.3. ΠΕΙΡΑΜΑ 1: ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ

είναι συνεπής με τη θεωρία, καθώς η συμπερίληψη περισσότερων αποτελεσμάτων αυξάνει την πιθανότητα εισαγωγής μη σχετικών τμημάτων.

Αντίστροφα, το Σχήμα 5.5 (Recall@k) επιβεβαιώνει τη θεωρητικά αναμενόμενη αύξηση της ανάκλησης με το k. Σημαντικό είναι ότι η χαμηλή ανάκληση για k=1 δεν αποτελεί αδυναμία του συστήματος, αλλά αντανακλά το γεγονός ότι υπάρχουν πολλαπλά σχετικά τμήματα ανά ερώτημα.

Το Σχήμα 5.7 παρέχει μια συνοπτική απεικόνιση της σχέσης Precision-Recall για k=5, με τη Dense και Hybrid-S να βρίσκονται στην επιθυμητή περιοχή υψηλής απόδοσης. Η οριζόντια διασπορά των σημείων υποδηλώνει ότι οι στρατηγικές διαφοροποιούνται περισσότερο στην ικανότητά τους να ανακτούν ποσοστό των διαθέσιμων σχετικών τμημάτων παρά στην ακρίβεια των επιλογών τους.

Τέλος, το Σχήμα 5.8 (NDCG@k) επιβεβαιώνει τη συνολική υπεροχή της Dense μεθόδου σε rank-aware μετρικές. Η απότομη πτώση όλων των μεθόδων από k=1 σε k=3 υποδηλώνει ότι η ποιότητα των αποτελεσμάτων υποβαθμίζεται γρήγορα πέρα από τις πρώτες θέσεις, ενώ η σχετικά επίπεδη καμπύλη από k=5 έως k=10 υποδηλώνει σταθεροποίηση της κατάταξης.



Σχήμα 5.9: Σύγκριση των διαμορφώσεων ως προς την μετρική: καθυστέρηση

Το Σχήμα 5.9 παρουσιάζει τη σύγκριση των πέντε στρατηγικών ανάκτησης ως προς την υπολογιστική απόδοση, αποκαλύπτοντας σημαντικές διαφορές στην καθυστέρηση εκτέλεσης.

Από το αριστερό διάγραμμα (Μέση Καθυστέρηση) παρατηρείται μια σαφής ιεραρχία απόδοσης. Η BM25 επιδεικνύει εξαιρετική ταχύτητα (19.6 ms), αναμενόμενο αποτέλεσμα δεδομένου ότι βασίζεται σε απλούς υπολογισμούς στατιστικής συχνότητας χωρίς νευρωνικά μοντέλα. Το SPLADE, παρότι αποτελεί νευρωνική προσέγγιση, διατηρεί ανταγωνιστική ταχύτητα (118.3 ms) λόγω της αραιότητας των διανυσμάτων του.

Η πυκνή ανάκτηση (Dense) παρουσιάζει σημαντικά υψηλότερη καθυστέρηση (623.3 ms), που οφείλεται στον υπολογισμό πυκνών διανυσμάτων 1024 διαστάσεων και στην εκτέλεση αναζήτησης πλησιέστερων γειτόνων σε μεγάλο χώρο. Οι υβριδικές στρατηγικές εμφανίζουν ακόμα υψηλότερη καθυστέρηση (Hybrid-S: 737.4 ms,

Hybrid-B: 601.4 ms), αναμενόμενο αφού εκτελούν τόσο αραιή όσο και πυκνή αναζήτηση παράλληλα και στη συνέχεια εφαρμόζουν τον αλγόριθμο RRF για συνδυασμό των αποτελεσμάτων.

Το δεξί διάγραμμα (Εκατοστημόρια Καθυστέρησης) αποκαλύπτει επιπλέον σημαντικές πληροφορίες για την ευστάθεια της απόδοσης. Η διαφορά μεταξύ διάμεσου (P50) και P95 για τις μεθόδους με νευρωνικά μοντέλα είναι εντυπωσιακά μεγάλη, ειδικά για τις υβριδικές προσεγγίσεις όπου το P95 φτάνει περίπου τα 1600 ms. Αυτή η υψηλή διασπορά υποδηλώνει ότι η καθυστέρηση δεν είναι σταθερή και επηρεάζεται από παράγοντες όπως η πολυπλοκότητα του ερωτήματος, το μέγεθος του ευρετηρίου που εξερευνάται, καθώς και την κατανάλωση των επεξεργαστικών πόρων τη δεδομένη χρονική στιγμή. Σημαντικό είναι ότι το διάγραμμα υποδηλώνει σαφή συμβιβασμό (tradeoff) μεταξύ απόδοσης ανάκτησης και υπολογιστικής αποδοτικότητας. Η Dense, που επιδεικνύει την καλύτερη απόδοση σε μετρικές ποιότητας (MAP, MRR, NDCG), απαιτεί περίπου 35 φορές περισσότερο χρόνο από την BM25, ενώ οι υβριδικές στρατηγικές, που προσεγγίζουν την Dense σε ποιότητα, απαιτούν ακόμα μεγαλύτερο υπολογιστικό κόστος. Για παραγωγικές εφαρμογές, αυτή η ανάλυση υποδηλώνει ότι η επιλογή στρατηγικής εξαρτάται από τις απαιτήσεις του συστήματος: για εφαρμογές real-time με αυστηρά όρια καθυστέρησης, η BM25 ή το SPLADE μπορεί να είναι προτιμότερα παρά την κάποια θυσία ποιότητας, ενώ για εφαρμογές όπου η ακρίβεια υπερτερεί της ταχύτητας, η Dense παραμένει η βέλτιστη επιλογή.

5.4 ΠΕΙΡΑΜΑ 2: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΎΒΡΙΔΙΚΗΣ ΑΝΑΚΤΗΣΗΣ

Τα αποτελέσματα του Πειράματος 1 κατέδειξαν ότι οι υβριδικές στρατηγικές, παρά τη θεωρητική τους υπεροχή λόγω του συνδυασμού συμπληρωματικών σημάτων, δεν κατάφεραν να υπερβούν την πυκνή ανάκτηση με τις προεπιλεγμένες παραμέτρους. Αυτή η παρατήρηση εγείρει το ερώτημα εάν η υποδεέστερη απόδοση οφείλεται σε μη βέλτιστη παραμετροποίηση του αλγορίθμου Reciprocal Rank Fusion. Το παρόν πείραμα διερευνά συστηματικά την επίδραση δύο κρίσιμων υπερπαραμέτρων στην απόδοση των υβριδικών μεθόδων: του βάρους σύμφυσης α και της σταθεράς RRF rrf_k . Η μαθηματική περιγραφή του αλγορίθμου (3.1) που χρησιμοποιήθηκε είναι:

$$\text{RRF}_{\text{hybrid}}(d) = \alpha \cdot \frac{1}{k + \text{rank}_{\text{dense}}(d)} + (1 - \alpha) \cdot \frac{1}{k + \text{rank}_{\text{sparse}}(d)} \quad (5.1)$$

5.4.1 Μεθοδολογία Βελτιστοποίησης

Η βελτιστοποίηση διατυπώνεται ως πρόβλημα εύρεσης του βέλτιστου ζεύγους υπερπαραμέτρων που μεγιστοποιεί μια σύνθετη συνάρτηση στόχου (composite objective function). Η παράμετρος α ελέγχει την ισορροπία μεταξύ πυκνών και αραιών αναπαραστάσεων, όπου $\alpha \in [0, 1]$ με $\alpha = 0$ να αντιστοιχεί σε καθαρά αραιή ανάκτηση, $\alpha = 1$ σε καθαρά πυκνή, και ενδιάμεσες τιμές σε υβριδική προσέγγιση. Η παράμετρος rrf_k αποτελεί τη σταθερά κανονικοποίησης του αλγορίθμου Reciprocal Rank

5.4. ΠΕΙΡΑΜΑ 2: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΥΒΡΙΔΙΚΗΣ ΑΝΑΚΤΗΣΗΣ

Fusion που ελέγχει τη βαρύτητα των κορυφαίων αποτελεσμάτων. Το πρόβλημα βελτιστοποίησης διατυπώνεται τυπικά ως:

$$(\alpha^*, rrf_k^*) = \operatorname{argmax}_{(\alpha, rrf_k) \in \Theta} S(\alpha, rrf_k; D_{train}) \quad (5.2)$$

όπου S είναι η συνάρτηση σύνθετης βαθμολογίας που υπολογίζεται στο σύνολο εκπαίδευσης D_{train} , και Θ ο χώρος αναζήτησης των υπερπαραμέτρων. Ο χώρος αναζήτησης ορίζεται ως το καρτεσιανό γινόμενο $\Theta = \{(\alpha_i, k_j) : \alpha_i \in A, k_j \in K\}$, όπου $A = \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ αποτελεί ομοιόμορφη διακριτοποίηση του διαστήματος $[0, 1]$ με βήμα 0.2, και $K = \{20, 40, 60, 80, 100\}$ περιλαμβάνει κοινές τιμές από τη βιβλιογραφία. Αυτός ο ορισμός παράγει $|\Theta| = 30$ συνολικές διαμορφώσεις, επιτρέποντας την εξαντλητική αναζήτηση (exhaustive grid search) χωρίς υπερβολικό υπολογιστικό κόστος. Η επιλογή εξαντλητικής αναζήτησης αντί μεθόδων βασισμένων σε κλίση (gradient-based methods) αιτιολογείται από την πιθανή μη-κυρτότητα της συνάρτησης στόχου και την ανάγκη πλήρους χαρτογράφησης του χώρου για ερμηνευσιμότητα. Η συνάρτηση σύνθετης βαθμολογίας (composite score) σχεδιάστηκε για να εξισορροπεί πολλαπλά κριτήρια απόδοσης, συνδυάζοντας βαθμολογία ποιότητας (quality score) με ποινή καθυστέρησης (latency penalty). Συγκεκριμένα, ορίζεται ως:

$$S(\alpha, rrf_k) = Q(\alpha, rrf_k) - P(t) \quad (5.3)$$

όπου η βαθμολογία ποιότητας Q υπολογίζεται ως σταθμισμένος συνδυασμός τεσσάρων μετρικών:

$$Q = 0.35 \cdot \text{Success@3} + 0.30 \cdot \text{Precision@3} + 0.20 \cdot \text{Recall@10} + 0.15 \cdot \text{Precision@10} \quad (5.4)$$

Οι σταθμίσεις επιλέχθηκαν έτσι ώστε να δίνεται μεγαλύτερη έμφαση στην πρώιμη επιτυχία (early success) και την ακρίβεια στα πρώτα αποτελέσματα, καθώς οι χρήστες συστημάτων RAG εξετάζουν κυρίως τις πρώτες θέσεις. Η ποινή καθυστέρησης ορίζεται ως:

$$P(t) = 0.1 \cdot \frac{\min(\max(0, t - t_{target}), t_{max})}{t_{max}} \quad (5.5)$$

όπου t είναι ο πραγματικός χρόνος απόκρισης σε milliseconds, $t_{target} = 500$ ms ο στόχος καθυστέρησης, και $t_{max} = 1000$ ms το ανώτατο όριο ποινής. Η ποινή είναι μηδενική για $t \leq 500$ ms, αυξάνεται γραμμικά έως 0.1 για χρόνους μέχρι 1500 ms, και παραμένει σταθερή στο 0.1 για μεγαλύτερες καθυστερήσεις, διασφαλίζοντας ότι η ποιότητα παραμένει το κυρίαρχο κριτήριο ενώ λαμβάνεται υπόψη η πρακτική εφαρμογή.

Στρατηγική Διαχωρισμού Δεδομένων. Για την αξιόπιστη εκτίμηση της απόδοσης και την επιλογή υπερπαραμέτρων που γενικεύουν σε νέα δεδομένα, το σύνολο δεδομένων SOSum χωρίζεται σε σύνολο εκπαίδευσης και σύνολο αξιολόγησης με αναλογία 80-20 μέσω στρωματοποιημένης τυχαίας δειγματοληψίας (stratified random split). Η επιλογή αυτής της μεθοδολογίας αιτιολογείται από το γεγονός ότι το σύνολο δεδομένων SOSum αποτελεί ένα αντιπροσωπευτικό υποσύνολο του

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

ευρύτερου προβλήματος των τεχνικών ερωτημάτων στον τομέα της τεχνολογίας λογισμικού. Στόχος του διαχωρισμού είναι η εύρεση διαμόρφωσης υπερπαραμέτρων που επιδεικνύει ισχυρή ικανότητα γενίκευσης στο ευρύτερο πρόβλημα, αντί της απλής προσαρμογής στα συγκεκριμένα χαρακτηριστικά του διαθέσιμου συνόλου εκπαίδευσης.

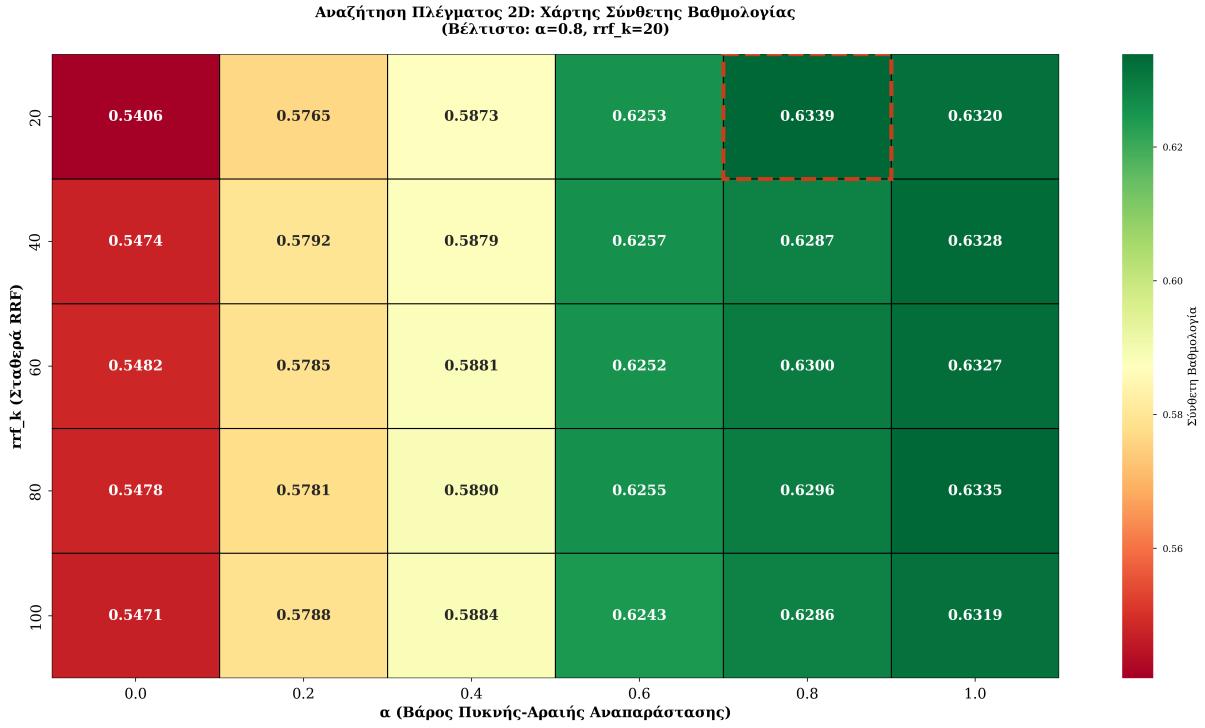
Η στρωματοποίηση (stratification) διασφαλίζει ότι τόσο το σύνολο εκπαίδευσης όσο και το σύνολο αξιολόγησης διατηρούν την αναλογική κατανομή των ερωτημάτων σε σχέση με τρεις κρίσιμες διαστάσεις. Πρώτον, ως προς τον τύπο ερωτήματος (question type) που καθορίζει τη φύση της πληροφορίας που αναζητείται. Δεύτερον, ως προς την κύρια κατηγορία τεχνολογίας (primary tag category) που αντλείται από τις ετικέτες του Stack Overflow και ομαδοποιείται σε έξι συχνότερες κατηγορίες συν μία κατηγορία "Other" για τις υπόλοιπες. Τρίτον, ως προς το πλήθος διαθέσιμων απαντήσεων ανά ερωτήμα, που χωρίζεται σε τέσσερα διακριτά επίπεδα (bins): "none" για ερωτήματα χωρίς απαντήσεις, "low" για 1-3 απαντήσεις, "medium" για 4-6 απαντήσεις, και "high" για 7 ή περισσότερες απαντήσεις. Το πλήθος απαντήσεων αποτελεί σημαντική διάσταση στρωματοποίησης καθώς σχετίζεται με την πολυπλοκότητα του ερωτήματος και τη διαθεσιμότητα σχετικού περιεχομένου στη βάση γνώσης.

Το κλειδί στρωματοποίησης (stratification key) κατασκευάζεται ως η συνένωση των τριών αυτών διαστάσεων, δημιουργώντας διακριτές ομάδες με συγκεκριμένα χαρακτηριστικά. Για την αποφυγή υπερβολικά μικρών ομάδων που δεν μπορούν να χωριστούν ισορροπημένα, διατηρούνται μόνο οι ομάδες με ελάχιστο πλήθος τουλάχιστον 10 ερωτημάτων, επιτρέποντας την ύπαρξη τουλάχιστον 8 ερωτημάτων στο σύνολο εκπαίδευσης και 2 στο σύνολο αξιολόγησης. Αυτή η φίλτραριστική διαδικασία διατηρεί την πλειονότητα των δεδομένων ενώ εξασφαλίζει στατιστικά αξιόπιστες εκτιμήσεις.

Διαδικασία Grid Search. Για κάθε διαμόρφωση (α_i, k_j) του χώρου αναζήτησης Θ , υπολογίζεται η βαθμολογία $S(\alpha_i, k_j; D_{train})$ στο σύνολο εκπαίδευσης. Η βέλτιστη διαμόρφωση επιλέγεται με ιεραρχική στρατηγική που μεγιστοποιεί πρωτίστως την τιμή της συνάρτησης στόχου, με δευτερεύοντας κριτήριο την προτίμηση ισορροπημένων τιμών α κοντά στο 0.5 (καθώς υβριδικές μέθοδοι που αξιοποιούν και τις δύο ενσωματώσεις τείνουν να γενικεύουν καλύτερα) και τριτέυοντας κριτήριο την προτίμηση της standard τιμής $rrf_k = 60$ που αποτελεί κοινή πρακτική στη βιβλιογραφία. Τέλος, η επιλεγμένη διαμόρφωση (α^*, rrf_k^*) αξιολογείται στο σύνολο αξιολόγησης D_{test} για αμερόληπτη εκτίμηση της απόδοσης σε δεδομένα που δεν έχουν χρησιμοποιηθεί κατά τη διαδικασία βέλτιστοποίησης, παρέχοντας ρεαλιστική εκτίμηση της αναμενόμενης συμπεριφοράς σε νέα ερωτήματα. Η τελική αναφερόμενη απόδοση υπολογίζεται αποκλειστικά στο D_{test} και αποτελεί την εκτίμηση της ικανότητας γενίκευσης του συστήματος.

5.4. ΠΕΙΡΑΜΑ 2: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΥΒΡΙΔΙΚΗΣ ΑΝΑΚΤΗΣΗΣ

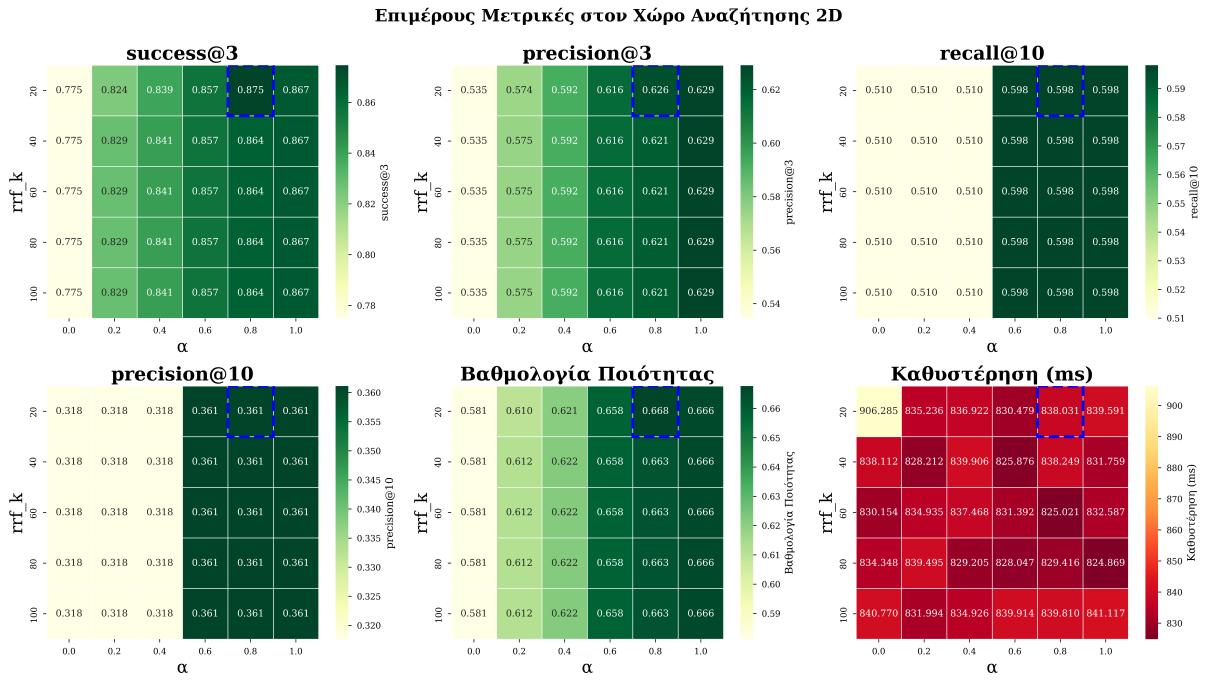
5.4.2 Αποτελέσματα



Σχήμα 5.10: Χάρτης σύνθετης βαθμολογίας: η βέλτιστη διαμόρφωση ($\alpha = 0.8$, $k = 20$) αντιστοιχεί σε υβριδική ανάκτηση με ισχυρή έμφαση στο πυκνό σήμα. Η απόδοση αυξάνεται μονοτονικά με το α , ενώ το k έχει αμελητέα επίδραση.

Το Σχήμα 5.10 παρουσιάζει τον χάρτη σύνθετης βαθμολογίας στον δισδιάστατο χώρο αναζήτησης των υπερπαραμέτρων α και k . Η βέλτιστη διαμόρφωση εντοπίζεται στο $\alpha^* = 0.8$ και $k^* = 20$ με σύνθετη βαθμολογία $S = 0.6339$, υποδεικνύοντας ότι η υβριδική στρατηγική με ισχυρή έμφαση στην πυκνή ανάκτηση (80% dense, 20% sparse) επιτυγχάνει την υψηλότερη απόδοση. Παρατηρείται σαφής διαβάθμιση από χαμηλές βαθμολογίες (σκούρο κόκκινο, 0.54-0.55) για $\alpha \leq 0.2$ προς υψηλές βαθμολογίες (πράσινο, 0.63-0.64) για $\alpha \geq 0.6$, με το κρίσιμο σημείο μετάβασης να βρίσκεται περίπου στο $\alpha = 0.4$. Η παράμετρος k επιδεικνύει αμελητέα επίδραση στην απόδοση, με όλες οι τιμές στο εύρος 20-100 να παράγουν σχεδόν ταυτόσημες βαθμολογίες για το ίδιο α . Αξιοσημείωτο είναι ότι η περαιτέρω αύξηση του α πέρα από το 0.8 (π.χ. στο $\alpha = 1.0$ που αντιστοιχεί σε καθαρά πυκνή ανάκτηση) δεν φαίνεται να προσφέρει επιπλέον βελτίωση, υποδηλώνοντας ότι η μικρή συνεισφορά του αραιού σήματος (20%) παίζει κάποιο ρόλο στη βέλτιστη απόδοση.

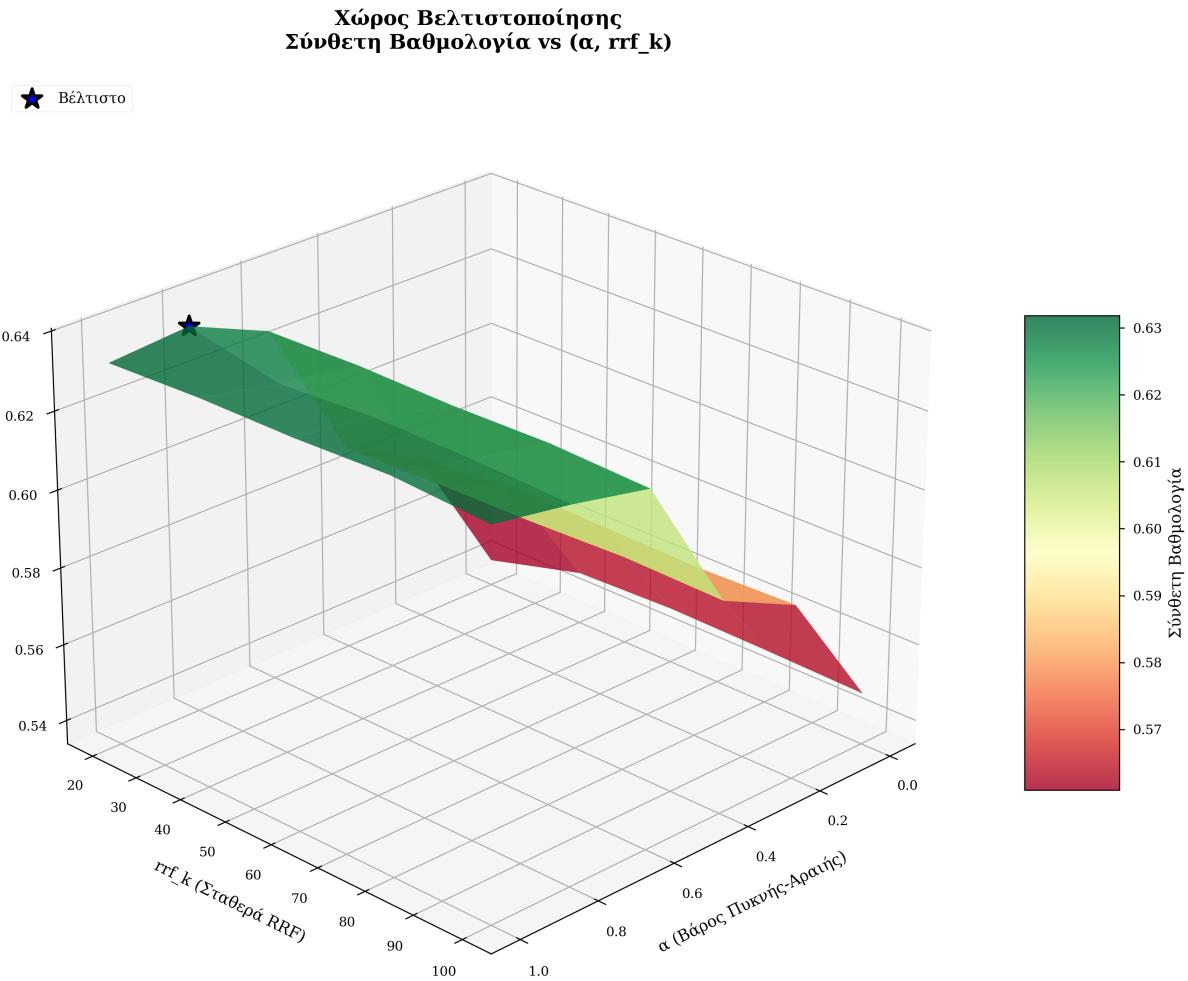
ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ



Σχήμα 5.11: Επιμέρους μετρικές: Success@3 και Precision@3 βελτιώνονται δραματικά με το α , η Recall@10 παραμένει σταθερή (0.51), και η καθυστέρηση εμφανίζει μικρές διακυμάνσεις. Το k δεν επηρεάζει συστηματικά τις μετρικές.

Το Σχήμα 5.11 αποσυνθέτει τη σύνθετη βαθμολογία στις επιμέρους μετρικές που τη συνθέτουν, αποκαλύπτοντας τους υποκείμενους μηχανισμούς της παρατηρούμενης απόδοσης. Η μετρική Success@3 εμφανίζει δραματική βελτίωση από 0.775 για $\alpha = 0$ σε 0.875 για $\alpha = 0.8$, επιβεβαιώνοντας την υπεροχή της πυκνής ανάκτησης στον εντοπισμό σχετικών τμημάτων στις πρώτες θέσεις. Η Precision@3 ακολουθεί παρόμοια τάση, αυξανόμενη από 0.535 σε 0.629, ενώ η Recall@10 παραμένει αξιοσημείωτα σταθερή στο 0.51 ανεξάρτητα από το α , υποδηλώνοντας ότι όλες οι στρατηγικές ανακτούν παρόμοιο ποσοστό των διαθέσιμων σχετικών εγγράφων όταν εξετάζονται τα πρώτα δέκα αποτελέσματα. Η Precision@10 σταθεροποιείται στο 0.361 για $\alpha \geq 0.6$, αλλά υποβαθμίζεται σημαντικά για χαμηλότερες τιμές (0.318). Η βαθμολογία ποιότητας αντανακλά τη σταθμισμένη συνεισφορά αυτών των μετρικών, με την κυριαρχία των early success μετρικών να οδηγεί στην προτίμηση υψηλού α . Τέλος, η καθυστέρηση εμφανίζει περιορισμένη διακύμανση, με αραιές μεθόδους ($\alpha = 0$) να επιτυγχάνουν χρόνους απόκρισης 906 ms έναντι υβριδικών και πυκνών μεθόδων (830-840 ms για $\alpha \geq 0.6$). Η απόλυτη διαφορά (70-80 ms) παραμένει σχετικά μικρή σε σχέση με τις βελτιώσεις ποιότητας. Η παράμετρος k δεν επιδεικνύει συστηματική επίδραση σε καμία από τις επιμέρους μετρικές, επιβεβαιώνοντας ότι ο κύριος παράγοντας απόδοσης είναι η ισορροπία πυκνών-αραιών σημάτων μέσω του α .

5.4. ΠΕΙΡΑΜΑ 2: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΥΒΡΙΔΙΚΗΣ ΑΝΑΚΤΗΣΗΣ



Σχήμα 5.12: Τρισδιάστατη επιφάνεια σύνθετης βαθμολογίας στον χώρο υπερπαραμέτρων. Η επιφάνεια εμφανίζει μονοτονική αύξηση με το α και επιπεδότητα κατά μήκος του k . Το βέλτιστο σημείο ($\alpha = 0.8$, $k = 20$) επισημαίνεται με αστέρι.

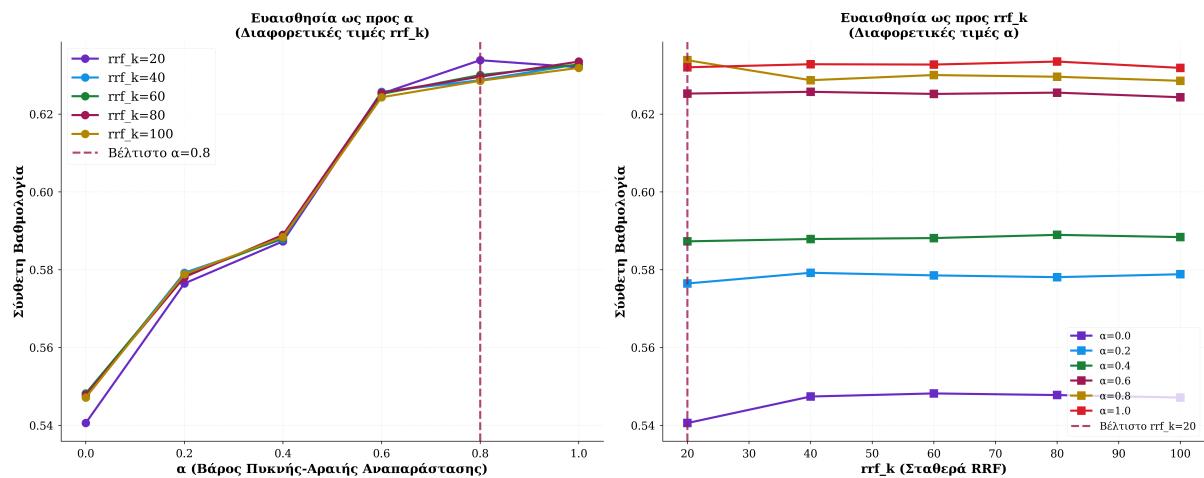
Το Σχήμα 5.12 απεικονίζει την τρισδιάστατη επιφάνεια της σύνθετης βαθμολογίας στον χώρο των υπερπαραμέτρων, προσφέροντας γεωμετρική διαίσθηση της τοπολογίας της συνάρτησης στόχου. Η επιφάνεια εμφανίζει σαφή μονοτονική αύξηση κατά μήκος του άξονα α , με το πράσινο οροπέδιο στην περιοχή $\alpha \geq 0.6$ να αντιστοιχεί στη ζώνη υψηλής απόδοσης. Αντίθετα, κατά μήκος του άξονα k , η επιφάνεια παραμένει σχεδόν επίπεδη, επιβεβαιώνοντας την αμελητέα επίδραση αυτής της παραμέτρου. Η βέλτιστη διαμόρφωση, επισημασμένη με αστέρι στο σημείο ($\alpha = 0.8$, $k = 20$), βρίσκεται στο πράσινο οροπέδιο με βαθμολογία 0.63. Η οπικοποίηση αποκαλύπτει την απουσία τοπικών μεγίστων στο εσωτερικό του χώρου αναζήτησης, με την απόδοση να αυξάνεται σταθερά καθώς το α προσεγγίζει το 0.8, υποδηλώνοντας ότι η σημαντική έμφαση στην πυκνή ενσωμάτωση με μικρή συμπληρωματική συνεισφορά από το αραιό σήμα αποδίδει τα καλύτερα αποτελέσματα.

Το Σχήμα 5.13 παρουσιάζει την ανάλυση ευαισθησίας (sensitivity analysis) των υπερπαραμέτρων, διαχωρίζοντας την επίδραση κάθε παραμέτρου ξεχωριστά. Το

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

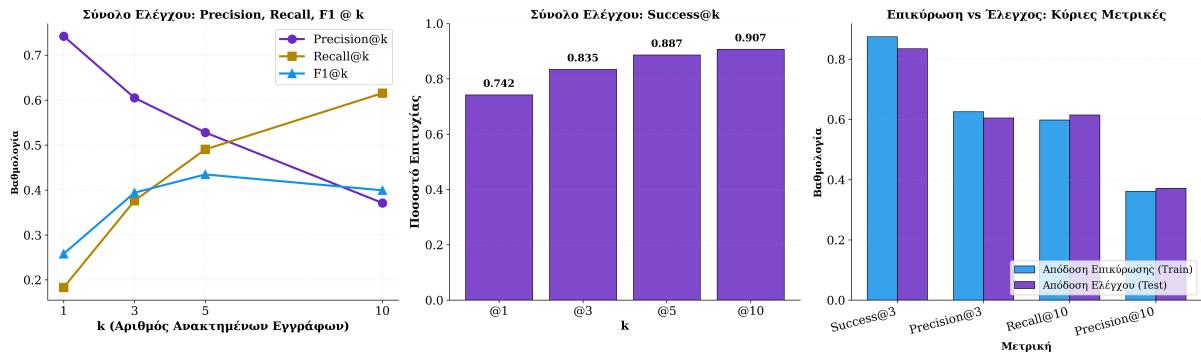
αριστερό διάγραμμα απεικονίζει την ευαισθησία ως προς το α για διαφορετικές σταθερές τιμές k . Παρατηρείται ότι όλες οι καμπύλες ακολουθούν σχεδόν ταυτόσημη πορεία, με απότομη αύξηση της βαθμολογίας από 0.54 στο $\alpha = 0$ έως 0.63 στο $\alpha = 0.8$, και με το κρίσιμο σημείο καμπής να εντοπίζεται περίπου στο $\alpha = 0.4$. Η διακεκομμένη κόκκινη γραμμή στο $\alpha = 0.8$ επισημαίνει το βέλτιστο σημείο. Η σύγκλιση των καμπυλών επιβεβαιώνει ότι η επιλογή του k δεν επηρεάζει ουσιαστικά τη σχέση απόδοσης- α . Αξιοσημείωτο είναι ότι η απόδοση φαίνεται να σταθεροποιείται ή ακόμα και να υποχωρεί ελαφρώς για $\alpha > 0.8$, υποδηλώνοντας ότι η μικρή συνεισφορά της αραιής ενσωμάτωσης (20%) είναι ωφέλιμη.

Το δεξί διάγραμμα εξετάζει την ευαισθησία ως προς το k για διαφορετικές σταθερές τιμές α . Οι καμπύλες εμφανίζονται σχεδόν οριζόντιες για όλες τις τιμές α , με την απόδοση να παραμένει ουσιαστικά σταθερή σε όλο το εύρος $k \in [20, 100]$. Η διαστρωμάτωση των καμπυλών ανάλογα με το α επιβεβαιώνει την κυριαρχία αυτής της παραμέτρου: οι καμπύλες για $\alpha \geq 0.8$ (κόκκινες αποχρώσεις) σταθεροποιούνται στο 0.63, οι ενδιάμεσες τιμές $\alpha = 0.4, 0.6$ (πράσινες αποχρώσεις) στο 0.59, ενώ οι χαμηλές τιμές $\alpha \leq 0.2$ (μπλε αποχρώσεις) στο 0.54. Η κατακόρυφη διακεκομμένη γραμμή στο $k = 20$ επισημαίνει την επιλεγμένη βέλτιστη τιμή, η οποία όμως θα μπορούσε να ήταν οποιαδήποτε άλλη στο εύρος χωρίς ουσιαστική επίπτωση στην απόδοση. Η ανάλυση ευαισθησίας καταδεικνύει ότι το α αποτελεί την κρίσιμη υπερπαραμέτρο του υβριδικού συστήματος, ενώ το k παίζει δευτερεύοντα ρόλο.



Σχήμα 5.13: Ανάλυση ευαισθησίας υπερπαραμέτρων. Αριστερά: η απόδοση αυξάνεται μονοτονικά με το α , με όλες τις καμπύλες (διαφορετικά k) να συγκλίνουν. Δεξιά: η απόδοση παραμένει σταθερή για όλα τα k , με διαστρωμάτωση βάσει του α . Το βέλτιστο $\alpha = 0.8$ επισημαίνεται με διακεκομμένη γραμμή.

5.4. ΠΕΙΡΑΜΑ 2: ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΥΒΡΙΔΙΚΗΣ ΑΝΑΚΤΗΣΗΣ



Σχήμα 5.14: Επικύρωση απόδοσης στο test set: σύγκριση κύριων μετρικών και Success@k για διαφορετικές τιμές k. Το train-test gap είναι ελάχιστο, επιβεβαιώνοντας την ικανότητα γενίκευσης του βέλτιστου μοντέλου.

Το Σχήμα 5.14 παρουσιάζει την τελική επικύρωση της απόδοσης στο test set για τη βέλτιστη διαμόρφωση ($\alpha = 0.8$, $k = 20$). Το αριστερό διάγραμμα δείχνει την εξέλιξη των μετρικών Precision, Recall και F1 καθώς αυξάνεται ο αριθμός των ανακτημένων εγγράφων k. Παρατηρείται η κλασική αντίστροφη σχέση Precision-Recall: η Precision μειώνεται από 0.75 στο k=1 σε 0.37 στο k=10, ενώ η Recall αυξάνεται από 0.18 σε 0.62. Το μέσο διάγραμμα απεικονίζει τη μετρική Success@k, η οποία αυξάνεται από 0.742 στο k=1 σε 0.907 στο k=10, επιβεβαιώνοντας την υψηλή ικανότητα του συστήματος να εντοπίζει τουλάχιστον ένα σχετικό έγγραφο στα k πρώτα αποτελέσματα. Το δεξί διάγραμμα συγκρίνει την απόδοση μεταξύ train και test set για τις τέσσερις κύριες μετρικές. Η απόκλιση της επίδοσης μεταξύ του σύνολου ελέγχου και του συνόλου εκπαίδευσης είναι ελάχιστη για όλες τις μετρικές (Success@3: 0.867 vs 0.835, Precision@3: 0.629 vs 0.605, Recall@10: 0.598 vs 0.615, Precision@10: 0.361 vs 0.370), υποδηλώνοντας ότι η βέλτιστη διαμόρφωση γενικεύει αποτελεσματικά σε νέα δεδομένα χωρίς σημάδια υπερπροσαρμογής.

5.4.3 Συμπεράσματα Πειράματος 2

Η συστηματική βέλτιστοποίηση των υπερπαραμέτρων του αλγορίθμου RRF οδηγεί σε σημαντικό εύρημα: η βέλτιστη διαμόρφωση αντιστοιχεί σε υβριδική ανάκτηση με ισχυρή έμφαση στην πυκνή ενσωμάτωση ($\alpha^* = 0.8$, $k^* = 20$), υποδεικνύοντας ότι η συνεισφορά της αραιής ενσωμάτωσης, αν και μικρή (20%), εξακολουθεί να προσφέρει οριακή βελτίωση έναντι της καθαρά πυκνής ανάκτησης.

Η ανάλυση ευαισθησίας αποκαλύπτει ότι το α αποτελεί την κρίσιμη υπερπαράμετρο (μονοτονική αύξηση από $S \approx 0.54$ σε $S \approx 0.63$), ενώ το k έχει αμελητέα επίδραση. Οι μετρικές πρώτης επιτυχίας βελτιώνονται δραματικά με αυξανόμενο α , ενώ η Recall@10 παραμένει σταθερή (~0.51), υποδηλώνοντας ότι η βελτίωση προέρχεται από καλύτερη κατάταξη των σχετικών εγγράφων στις πρώτες θέσεις. Η επικύρωση στο σύνολο αξιολόγησης επιβεβαιώνει την ικανότητα γενίκευσης με ελάχιστη απόκλιση μεταξύ εκπαίδευσης και ελέγχου.

Η εξαιρετική απόδοση της πυκνής ανάκτησης στο συγκεκριμένο πεδίο μπορεί να ερμηνευτεί μέσω των ιδιαίτερων χαρακτηριστικών του συνόλου δεδομένων. Η στατιστική ανάλυση του SOSum αποκαλύπτει ότι οι ερωτήσεις έχουν διάμεσο μή-

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

κος 424 χαρακτήρες (5.2) (περίπου 106 λεκτικές μονάδες), ενώ οι απαντήσεις 300 χαρακτήρες (περίπου 75 λεκτικές μονάδες). Το συνδυασμένο μήκος ενός τυπικού ζεύγους ερώτησης-απάντησης είναι περίπου 180 λεκτικές μονάδες, δηλαδή σημαντικά μικρότερο από το μέγιστο μήκος ακολουθίας των μοντέλων ενσωμάτωσης. Οι κατανομές εμφανίζουν έντονη ασυμμετρία προς χαμηλές τιμές, με την πλειονότητα των εγγράφων να συγκεντρώνεται κάτω από τους 1000 χαρακτήρες. Αυτό επιτρέπει στα μοντέλα ενσωμάτωσης να αποτυπώνουν ολοκληρωμένα το σημασιολογικό περιεχόμενο κάθε εγγράφου στον χώρο αναπαράστασης, χωρίς απώλεια πληροφορίας λόγω συμπίεσης ή ανάγκη για τμηματοποίηση (chunking).

Αντίθετα, σε σύνολα δεδομένων με εκτενή έγγραφα όπως βιβλία, επιστημονικές δημοσιεύσεις, ή τεχνικές αναφορές, το περιεχόμενο υπερβαίνει το μέγιστο μήκος ακολουθίας και πρέπει να διασπαστεί σε τμήματα των 256-512 λεκτικών μονάδων. Η τμηματοποίηση οδηγεί σε απώλεια του ευρύτερου πλαισίου και της συνοχής, καθώς κάθε τμήμα ενσωματώνεται ανεξάρτητα. Σε τέτοιες περιπτώσεις, οι αραιές μέθοδοι διατηρούν το πλεονέκτημα της ακριβούς λεξιλογικής αντιστοίχισης σε ολόκληρο το έγγραφο, και η υβριδική προσέγγιση αναμένεται να επιδείξει μεγαλύτερη σχετική βελτίωση. Αυτή η υπόθεση συνάδει με τα ευρήματα της εργασίας Blended RAG [60], όπου οι υβριδικές μέθοδοι ανάκτησης απέδωσαν σημαντικά καλύτερα σε σύνολα δεδομένων μεγάλου μήκους (όπως το TREC-COVID), επιβεβαιώνοντας την υπεροχή της λεξιλογικής-σημασιολογικής σύζευξης σε αναζήτηση εκτενών εγγράφων.

Επομένως, το πείραμα αποδεικνύει ότι για το πεδίο των τεχνικών ερωτημάτων Stack Overflow, όπου τα έγγραφα είναι σύντομα και σημασιολογικά πυκνά, η υβριδική προσέγγιση με έμφαση στην πυκνή ανάκτηση (80-20) αποδίδει βέλτιστα. Η διαπίστωση ότι η καθαρά πυκνή ανάκτηση δεν υπερτερεί του βέλτιστου υβριδικού μοντέλου υποδηλώνει κάποια συμπληρωματικότητα μεταξύ λεξιλογικής και σημασιολογικής πληροφορίας, αν και σε μικρότερο βαθμό από τον αναμενόμενο. Τα ευρήματα υπογραμμίζουν την ανάγκη αξιολόγησης ανάλογα με το πεδίο εφαρμογής και προσεκτικής ρύθμισης υπερπαραμέτρων, λαμβάνοντας υπόψη τα χαρακτηριστικά του συνόλου δεδομένων (μήκος εγγράφων, σημασιολογική πυκνότητα, ανάγκη για τμηματοποίηση). Η γενίκευση των συμπερασμάτων σε άλλα πεδία με διαφορετικά χαρακτηριστικά απαιτεί περαιτέρω έρευνα. Αξίζει επίσης να σημειωθεί πώς

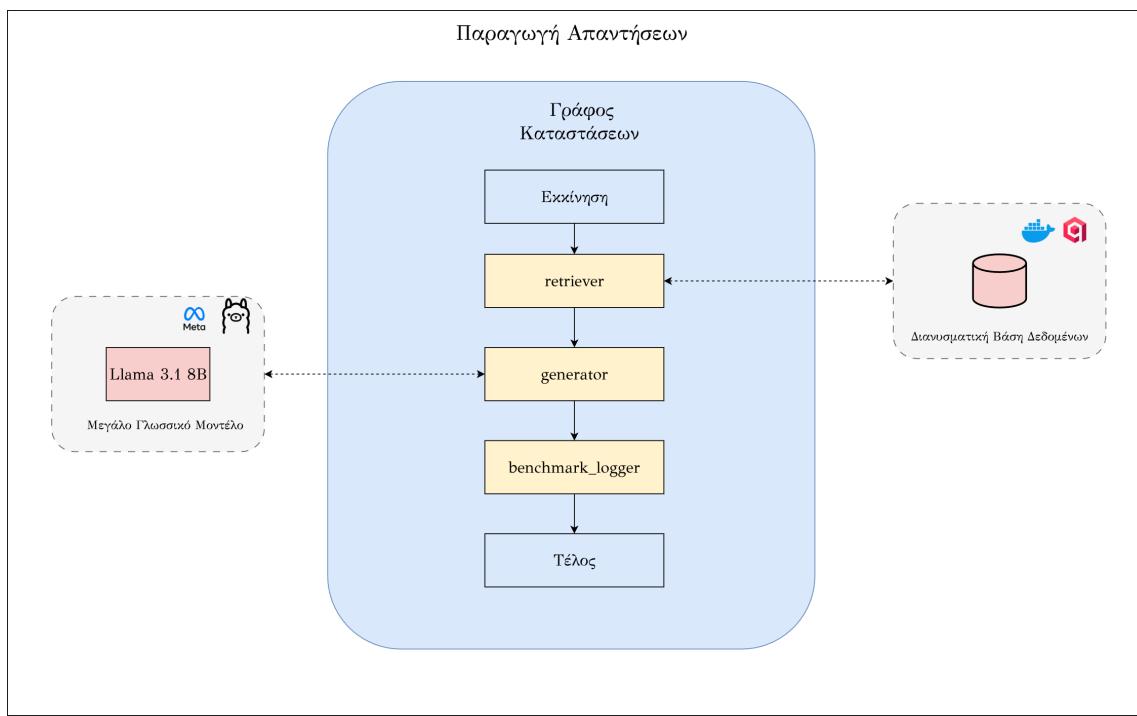
5.5 ΠΕΙΡΑΜΑ 3: ΑΞΙΟΛΟΓΗΣΗ ΠΑΡΑΓΩΓΗΣ ΑΠΑΝΤΗΣΕΩΝ

Η βελτιστοποίηση του υποσυστήματος ανάκτησης, όπως παρουσιάστηκε στο Πείραμα 2 (5.4), δεν διασφαλίζει αναγκαστικά την παραγωγή υψηλής ποιότητας απαντήσεων. Το στάδιο της παραγωγής (generation) συνιστά ανεξάρτητη πηγή σφραλμάτων που μπορεί να επηρεάσει την τελική απόδοση του συστήματος επαυξημένης παραγωγής μέσω ανάκτησης (RAG). Σε αυτό το πείραμα, πραγματοποιείται ολοκληρωμένη αξιολόγηση του συστήματος από άκρη σε άκρη (end-to-end evaluation), εστιάζοντας στην ποιότητα των παραγόμενων απαντήσεων σε τρεις κρίσιμες διαστάσεις: πιστότητα στο ανακτημένο πλαίσιο (faithfulness), συνάφεια με το ερώτημα (relevance), και χρησιμότητα για τον χρήστη (helpfulness).

5.5.1 Μεθοδολογία

Αρχιτεκτονική Αξιολόγησης

Για την αξιολόγηση του συστήματος γεννεσιουργίας, υιοθετήθηκε η μεθοδολογία LLM-as-a-Judge 3.4.5, η οποία αξιοποιεί τις ικανότητες μεγάλων γλωσσικών μοντέλων για την αυτόματη αξιολόγηση της ποιότητας κειμένου. Η προσέγγιση αυτή έχει αποδειχθεί ότι παρουσιάζει υψηλή συσχέτιση με ανθρώπινες κρίσεις και επιτρέπει την κλιμακούμενη αξιολόγηση μεγάλων συνόλων δεδομένων χωρίς την ανάγκη χειροκίνητης προσημείωσης (human annotation). Το σύστημα αξιολόγησης υλοποιήθηκε ως ευφυής πράκτορας με χρήση της βιβλιοθήκης LangGraph, μοντελοποιημένος ως γράφος πεπερασμένων καταστάσεων. Η αρχιτεκτονική του πράκτορα απεικονίζεται στο Σχήμα 5.15 και αποτελείται από τέσσερα διακριτά στάδια επεξεργασίας:



Σχήμα 5.15: Αρχιτεκτονική ευφυούς πράκτορα για την αξιολόγηση του υποσυστήματος γεννεσιουργίας. Το σύστημα επεξεργάζεται ερωτήματα μέσω αλληλουχίας κόμβων: εκκίνηση, ανάκτηση, γεννεσιουργία, και καταγραφή αποτελεσμάτων.

Ο κόμβος εκκίνησης λαμβάνει το ερώτημα του χρήστη και αρχικοποιεί την κατάσταση του γραφήματος. Ο κόμβος ανάκτησης (retriever) χρησιμοποιεί το ερώτημα για να εντοπίσει σχετικά έγγραφα στη διανυσματική βάση δεδομένων, εφαρμοζόντας τη μέθοδο που βρέθηκε στο Πείραμα 2. Ο κόμβος γεννεσιουργίας (generator) λαμβάνει ως είσοδο τόσο το αρχικό ερώτημα όσο και το ανακτημένο πλαίσιο, και παράγει την τελική απάντηση χρησιμοποιώντας το γλωσσικό μοντέλο Llama3.1 8B μέσω της πλατφόρμας Ollama. Τέλος, ο κόμβος καταγραφής (benchmark_logger) αποθηκεύει τόσο την παραγόμενη απάντηση όσο και τα ανακτημένα έγγραφα για την επακόλουθη αξιολόγηση.

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

Σημαντική παρατήρηση: Για το συγκεκριμένο πείραμα, το σύστημα λειτούργησε χωρίς μηχανισμό μνήμης (memory-less mode), ώστε να αποφευχθεί η επίδραση προηγούμενων ερωτημάτων στις παραγόμενες απαντήσεις.

Διαδικασία Αξιολόγησης

Η αξιολόγηση πραγματοποιήθηκε σε δύο φάσεις. Κατά την πρώτη φάση, το σύστημα επεξεργάστηκε το σύνολο των 500 ερωτημάτων από το dataset SOSum, καταγράφοντας για κάθε ερώτημα την παραγόμενη απάντηση και το σύνολο των ανακτημένων εγγράφων. Κατά τη δεύτερη φάση, χρησιμοποιήθηκε το μοντέλο GPT-5 της OpenAI μέσω προγραμματιστικής διεπαφής (API) ως αξιολογητής (judge), το οποίο βαθμολόγησε κάθε απάντηση σε τρεις διαστάσεις:

- Πιστότητα (Faithfulness):** Αξιολογεί κατά πόσον οι ισχυρισμοί που διατυπώνονται στην απάντηση μπορούν να συναχθούν από το ανακτημένο πλαίσιο, ελαχιστοποιώντας το φαινόμενο των παραισθήσεων (hallucinations).
- Συνάφεια (Relevance):** Εκτιμά το βαθμό στον οποίο η απάντηση αντιμετωπίζει άμεσα το ερώτημα που τέθηκε, χωρίς περιττές πληροφορίες ή παρεκβάσεις.
- Χρησιμότητα (Helpfulness):** Αξιολογεί την πρακτική αξία της απάντησης για τον χρήστη, λαμβάνοντας υπόψη παράγοντες όπως η πληρότητα, η σαφήνεια, και η παρουσία χρήσιμων παραδειγμάτων κώδικα.

Κάθε διάσταση βαθμολογήθηκε στην κλίμακα [1, 5], όπου 0 υποδηλώνει πλήρη αποτυχία και 5 υποδηλώνει ιδανική απόδοση. Έπειτα οι επιμέρους βαθμολογίες διαρούνται με το 5 ώστε να έρθουν στην κλίμακα 0 έως ένα. Η συνολική βαθμολογία υπολογίστηκε ως ο αριθμητικός μέσος όρος των τριών διαστάσεων:

$$\text{Συνολική Βαθμολογία} = \frac{\text{Πιστότητα} + \text{Συνάφεια} + \text{Χρησιμότητα}}{3} \quad (5.6)$$

5.5.2 Αποτελέσματα

Περιγραφικά Στατιστικά

Τα περιγραφικά στατιστικά των μετρικών αξιολόγησης παρουσιάζονται στον Πίνακα 5.5. Το σύστημα επιτυγχάνει μέσο όρο συνολικής βαθμολογίας 0.755 ± 0.177 , με σημαντικές διαφοροποιήσεις μεταξύ των επιμέρους διαστάσεων.

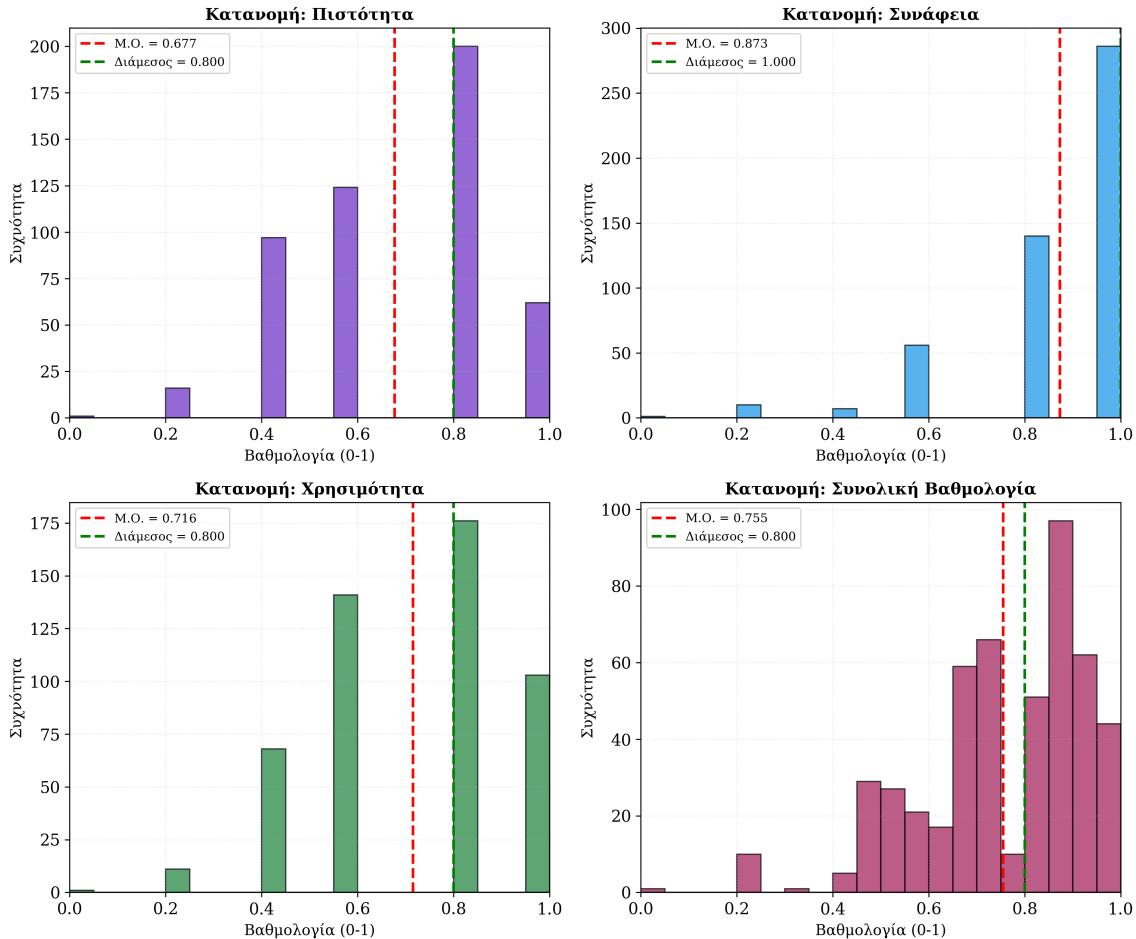
Μετρική	Πιστότητα	Συνάφεια	Χρησιμότητα	Συνολική Βαθμολογία
Μέσος όρος	0.677 ± 0.209	0.873 ± 0.181	0.716 ± 0.208	0.755 ± 0.177

Πίνακας 5.4: Περιγραφικά στατιστικά των μετρικών αξιολόγησης γεννεσιουργίας ($N=500$). Οι τιμές παρουσιάζονται ως μέσος όρος \pm τυπική απόκλιση. Το μοντέλο GPT-5 χρησιμοποιήθηκε ως αξιολογητής μέσω της μεθοδολογίας LLM-as-a-Judge.

5.5. ΠΕΙΡΑΜΑ 3: ΑΞΙΟΛΟΓΗΣΗ ΠΑΡΑΓΩΓΗΣ ΑΠΑΝΤΗΣΕΩΝ

Ανάλυση Κατανομών

Το Σχήμα 5.16 απεικονίζει τις κατανομές συχνότητας των βαθμολογιών για κάθε μετρική. Η ανάλυση των κατανομών αποκαλύπτει σημαντικά χαρακτηριστικά της συμπεριφοράς του συστήματος:



Σχήμα 5.16: Κατανομές συχνότητας των βαθμολογιών αξιολόγησης για τις τέσσερις μετρικές. Οι κόκκινες διακεκομένες γραμμές υποδεικνύουν τον μέσο όρο ενώ οι πράσινες διακεκομένες γραμμές τη διάμεσο κάθε κατανομής. Παρατηρείται έντονη ασυμμετρία στη μετρική της Συνάφειας (skewness προς υψηλές τιμές) και ευρύτερη διασπορά στην Πιστότητα.

Πιστότητα (Faithfulness): Η κατανομή παρουσιάζει μέσο όρο 0.677 με διάμεσο 0.800, υποδεικνύοντας ασυμμετρία προς χαμηλότερες τιμές. Το γεγονός ότι ο μέσος όρος είναι σημαντικά χαμηλότερος της διαμέσου υποδηλώνει την ύπαρξη υποσυνόλου ερωτημάτων όπου το σύστημα παράγει απαντήσεις με σημαντικές παραισθήσεις. Η υψηλή τυπική απόκλιση (0.209) επιβεβαιώνει τη μεταβλητότητα της απόδοσης ανάλογα με τη φύση του ερωτήματος και την ποιότητα του ανακτημένου πλαισίου.

Συνάφεια (Relevance): Η μετρική της Συνάφειας επιτυγχάνει την υψηλότερη βαθμολογία (μέσος όρος 0.873, διάμεσος 1.000), με έντονη συγκέντρωση τιμών στο

ΚΕΦΑΛΑΙΟ 5. ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

ανώτερο άκρο της κλίμακας. Αυτό υποδεικνύει ότι το σύστημα είναι αποτελεσματικό στην παραγωγή απαντήσεων που αντιμετωπίζουν το κεντρικό ερώτημα, χωρίς σημαντικές παρεκβάσεις.

Χρησιμότητα (Helpfulness): Η κατανομή της Χρησιμότητας (μέσος όρος 0.716, διάμεσος 0.800) παρουσιάζει παρόμοια χαρακτηριστικά με την Πιστότητα, με σημαντική διασπορά τιμών. Η χαμηλότερη απόδοση σε σχέση με τη Συνάφεια υποδηλώνει ότι ενώ το σύστημα απαντά στα ερωτήματα, η πρακτική αξία των απαντήσεων μπορεί να περιορίζεται από παράγοντες όπως η έλλειψη συγκεκριμένων παραδειγμάτων κώδικα ή η ασαφής διατύπωση.

Συνολική Βαθμολογία: Η κατανομή της συνολικής βαθμολογίας (μέσος όρος 0.755, διάμεσος 0.800) αντικατοπτρίζει την ισορροπία μεταξύ των τριών διαστάσεων, με σημαντική συγκέντρωση τιμών στο εύρος [0.7, 0.9], υποδεικνύοντας γενικά ικανοποιητική αλλά όχι άριστη απόδοση.

Η χαμηλότερη απόδοση στη μετρική της Πιστότητας (0.677) σε σχέση με τις άλλες δύο διαστάσεις αποτελεί κρίσιμο εύρημα. Αυτό υποδηλώνει ότι το κύριο πρόβλημα του συστήματος δεν έγκειται στην κατανόηση του ερωτήματος ή στη γενική δομή των απαντήσεων, αλλά στην τάση του μοντέλου να διατυπώνει ισχυρισμούς που δεν υποστηρίζονται πλήρως από το ανακτημένο πλαίσιο. Αυτή η παρατήρηση καθιστά επιτακτική την ανάγκη για μηχανισμούς αυτοδιορθωσης και επαλήθευσης των παραγόμενων απαντήσεων, όπως αναλύεται στην επόμενη ενότητα.

5.5.3 Πρόταση Βελτίωσης: Μηχανισμός Αυτοδιορθούμενης Παραγωγής

Με βάση την παρατήρηση ότι η χαμηλή πιστότητα αποτελεί το κύριο περιοριστικό παράγοντα της απόδοσης, προτείνεται η ενσωμάτωση μηχανισμού αυτοδιορθούμενης γεννεσιοναργίας (Self-Correcting RAG). Η προτεινόμενη αρχιτεκτονική βασίζεται στην ιδέα της επαναληπτικής βελτίωσης μέσω ανατροφοδότησης[70].

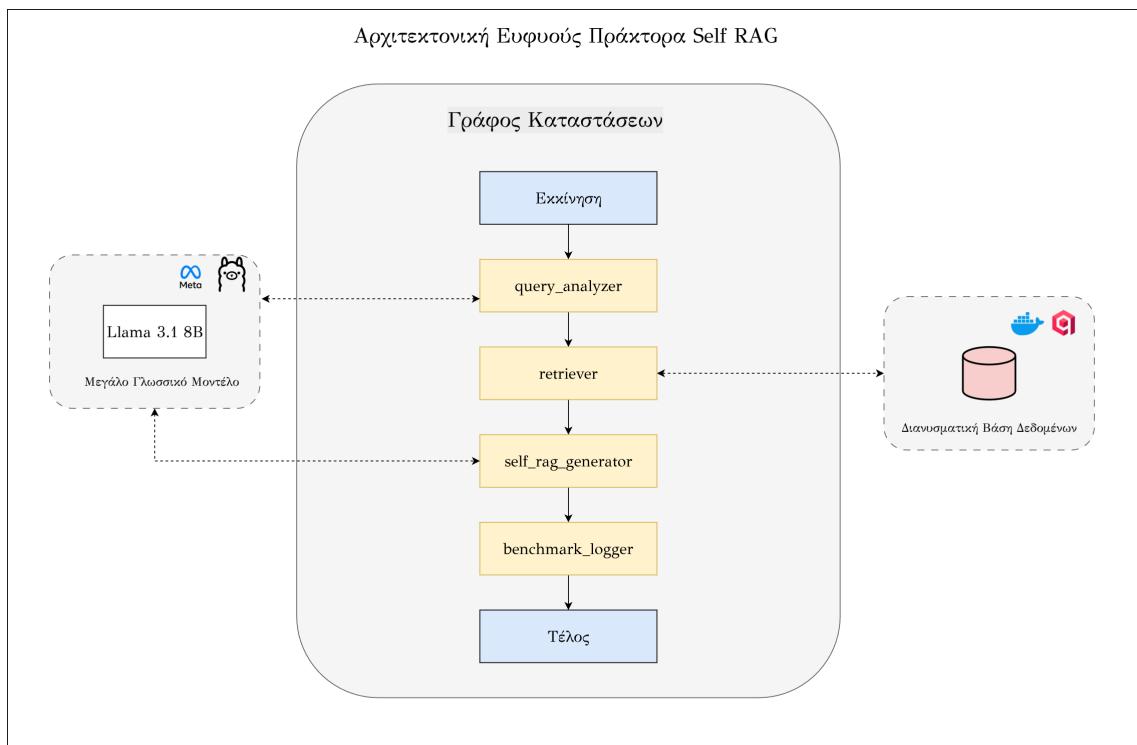
Αρχιτεκτονική Self-RAG

Η προτεινόμενη αρχιτεκτονική απεικονίζεται στο Σχήμα 5.17 και εισάγει δύο νέους κόμβους στο γράφημα καταστάσεων που επεκτείνουν τη λειτουργικότητα του βασικού συστήματος.

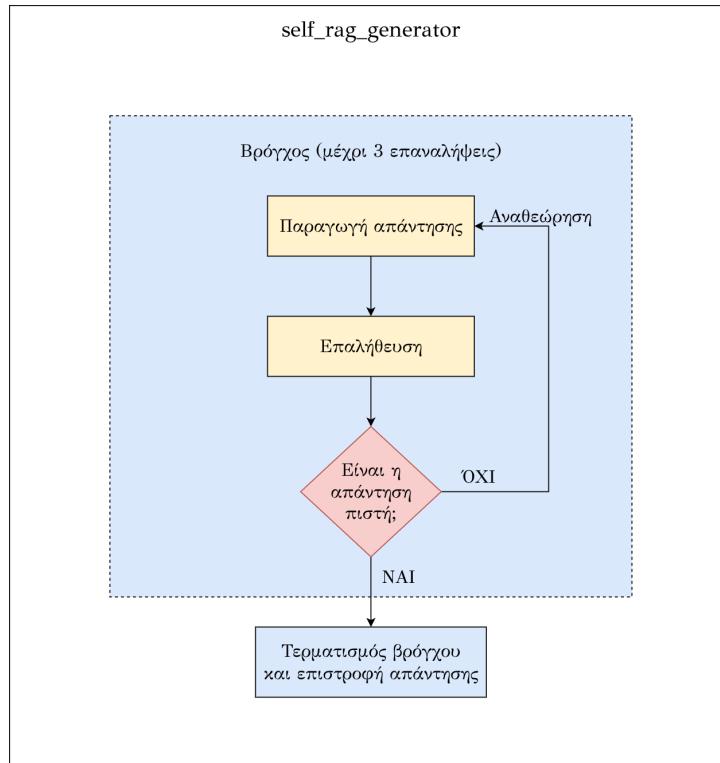
Ο πρώτος από τους νέους κόμβους που εισάγονται στην αρχιτεκτονική είναι ο κόμβος ανάλυσης ερωτήματος (query_analyzer). Ο κόμβος αυτός αναλαμβάνει την αποσύνθεση του αρχικού ερωτήματος του χρήστη σε ένα σύνολο ατομικών απαιτήσεων (atomic requirements) που πρέπει να ικανοποιηθούν από την τελική απάντηση. Η διαδικασία αυτή περιλαμβάνει τον προσδιορισμό των βασικών εννοιών (key concepts) που εμπλέκονται στο ερώτημα και τη διατύπωση των βημάτων συλλογιστικής (reasoning steps) που απαιτούνται για την παραγωγή ολοκληρωμένης απάντησης. Η δομημένη ανάλυση που παράγεται από αυτόν τον κόμβο παρέχει στο σύστημα γεννεσιοναργίας σαφή καθοδήγηση σχετικά με τα επιμέρους στοιχεία που απαιτείται να συμπεριληφθούν, διευκολύνοντας παράλληλα την επακόλουθη επαλήθευση της πληρότητας και της ορθότητας της παραγόμενης απάντησης.

5.5. ΠΕΙΡΑΜΑ 3: ΑΞΙΟΛΟΓΗΣΗ ΠΑΡΑΓΩΓΗΣ ΑΠΑΝΤΗΣΕΩΝ

Ο δεύτερος και κεντρικότερος κόμβος που προστίθεται είναι ο κόμβος αυτοδιορθούμενης γεννεσιουργίας (self_rag_generator), ο οποίος υλοποιεί επαναληπτικό βρόχο γεννεσιουργίας-επαλήθευσης-αναθεώρησης. Η λεπτομερής λειτουργία αυτού του μηχανισμού απεικονίζεται στο Σχήμα 5.18, όπου παρουσιάζεται η αλληλουχία των σταδίων επεξεργασίας και η ροή ελέγχου του συστήματος.



Σχήμα 5.17: Προτεινόμενη αρχιτεκτονική ευφυούς πράκτορα με μηχανισμό Self-RAG. Προστίθενται οι κόμβοι query_analyzer για την αποσύνθεση του ερωτήματος και self_rag_generator για την επαναληπτική βελτίωση της απάντησης μέσω επαλήθευσης.



Σχήμα 5.18: Εσωτερικός βρόγχος του κόμβου self_rag_generator. Το σύστημα παράγει μια αρχική απάντηση, την επαληθεύει έναντι του πλαισίου, και αναθεωρεί επαναληπτικά την απάντηση μέχρι να επιτευχθεί ικανοποιητική πιστότητα ή να εξαντληθεί ο μέγιστος αριθμός επαναλήψεων.

Λεπτομερής Περιγραφή του Μηχανισμού

Ο μηχανισμός αυτοδιορθούμενης παραγωγής υλοποιείται μέσω επαναληπτικού βρόγχου που συνδυάζει τρία διακριτά στάδια: την παραγωγή απάντησης, την επαλήθευση πιστότητας, και την αναθεώρηση με βάση δομημένη ανατροφοδότηση. Το σύστημα χρησιμοποιεί δύο διαφορετικά πρότυπα οδηγιών (prompt templates) που διαφοροποιούνται ως προς τη λειτουργία τους αλλά διατηρούν κοινή φιλοσοφία. Το πρώτο πρότυπο, που εφαρμόζεται κατά την αρχική γεννεσιούργια, λαμβάνει ως είσοδο το ερώτημα του χρήστη, το ανακτημένο πλαίσιο από τη βάση γνώσης Stack Overflow και την ανάλυση του ερωτήματος που παράγεται από τον κόμβο query_analyzer. Το δεύτερο πρότυπο, που χρησιμοποιείται κατά τις φάσεις αναθεώρησης, ενσωματώνει επιπρόσθετα την προηγούμενη εκδοχή της απάντησης και δομημένη ανατροφοδότηση που προκύπτει από τη διαδικασία επαλήθευσης. Η ανατροφοδότηση αυτή αποτελείται από τέσσερα στοιχεία: τα συγκεκριμένα ζητήματα που εντοπίστηκαν στην απάντηση, την κατηγοριοποίηση της σοβαρότητάς τους σε τρία επίπεδα (minor, moderate, major), τη συνιστώμενη ενέργεια διόρθωσης, και την αιτιολόγηση της κρίσης του επαληθευτή. Ο επαναληπτικός βρόγχος συνεχίζεται μέχρι να ικανοποιηθεί ένα από τα τρία κριτήρια τερματισμού που εφαρμόζονται με συγκεκριμένη προτεραιότητα. Το πρωταρχικό κριτήριο εξετάζει αν η τρέχουσα απάντηση έχει επαληθευτεί ως πιστή στο πλαίσιο, οπότε ο βρόγχος τερματίζεται επιτυχώς. Το δευτερεύον κριτήριο λειτουργεί ως μηχανισμός ασφαλείας και τερμα-

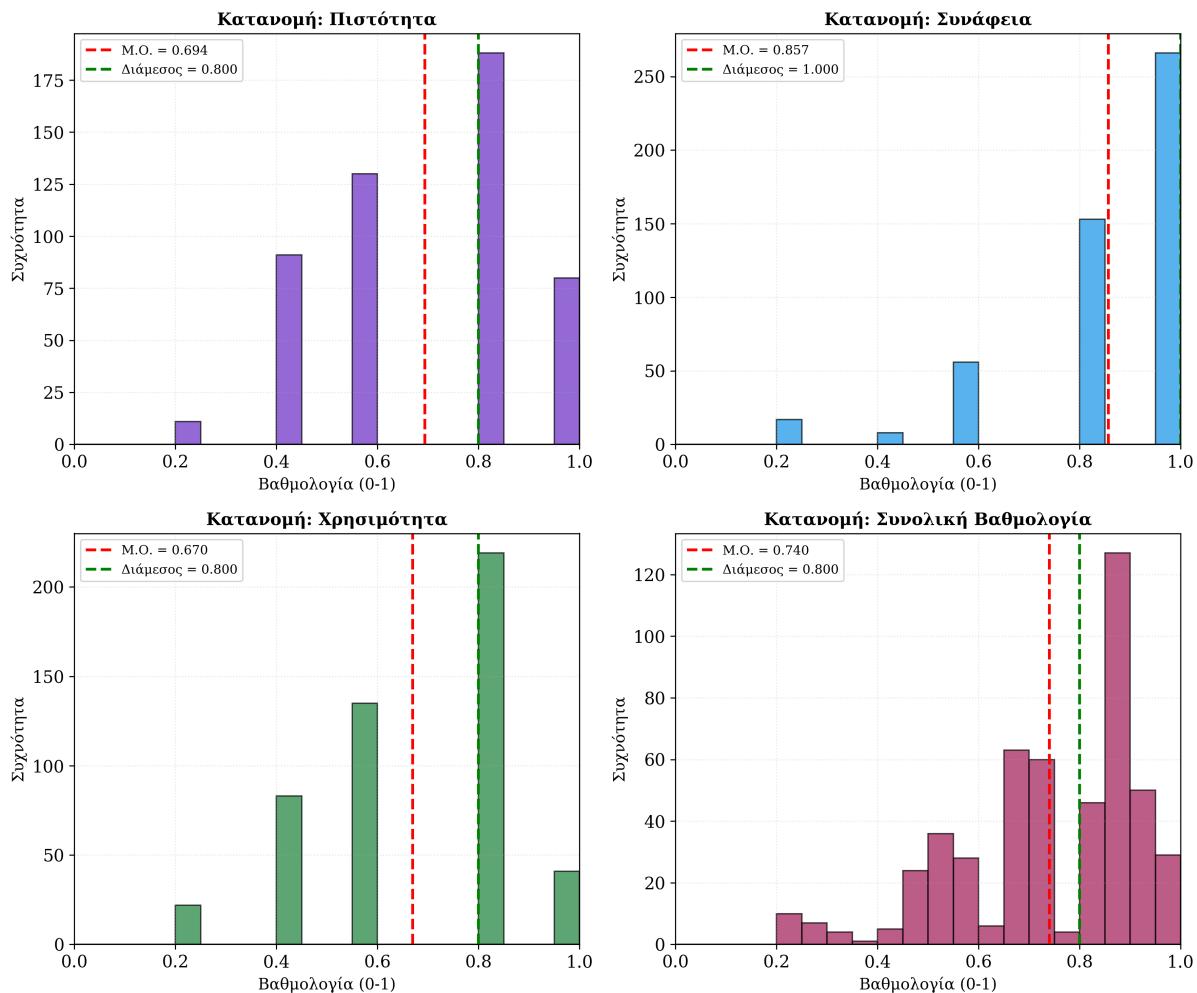
5.5. ΠΕΙΡΑΜΑ 3: ΑΞΙΟΛΟΓΗΣΗ ΠΑΡΑΓΩΓΗΣ ΑΠΑΝΤΗΣΕΩΝ

τίζει τη διαδικασία αν εξαντληθεί ο μέγιστος αριθμός επαναλήψεων, ο οποίος έχει οριστεί σε τρεις προκειμένου να διασφαλιστεί αποδεκτός χρόνος απόκρισης. Το τριτογενές κριτήριο αποτελεί βελτιστοποίηση και επιτρέπει πρόωρο τερματισμό όταν η βαθμολογία εμπιστοσύνης υπερβαίνει το κατώφλι των 0.8 και η σοβαρότητα των εντοπισμένων ζητημάτων χαρακτηρίζεται ως ελάχιστη, αποτρέποντας έτσι υπερβολικές επαναλήψεις για μικροπροβλήματα που δεν επηρεάζουν ουσιαστικά την ποιότητα της απάντησης. Το σύστημα διατηρεί λεπτομερή μεταδεδομένα για κάθε επανάληψη, καταγράφοντας την παραγόμενη απάντηση, τα αποτελέσματα επαλήθευσης, και την ενέργεια που πραγματοποιήθηκε. Στην τελική έξοδο περιλαμβάνονται δείκτες που υποδεικνύουν αν το σύστημα συνέκλινε σε πιστή απάντηση πριν την εξάντληση των επαναλήψεων και αν εντοπίστηκαν και διορθώθηκαν παραισθήσεις κατά τη διαδικασία.

Αποτελέσματα Βελτιωμένης Αρχιτεκτονικής

Μετρική	Πιστότητα	Συνάφεια	Χρησιμότητα	Συνολική Βαθμολογία
Μέσος όρος	0.694 ± 0.206	0.857 ± 0.193	0.669 ± 0.198	0.740 ± 0.180

Πίνακας 5.5: Περιγραφικά στατιστικά των μετρικών αξιολόγησης γεννεσιοναργίας χρησιμοποιώντας την αρχιτεκτονική self RAG ($N=500$). Οι τιμές παρουσιάζονται ως μέσος όρος \pm τυπική απόκλιση. Το μοντέλο GPT-5 χρησιμοποιήθηκε ως αξιολογητής μέσω της μεθοδολογίας LLM-as-a-Judge.



Σχήμα 5.19: Κατανομές συχνότητας των βαθμολογιών αξιολόγησης για τις τέσσερις μετρικές χρησιμοποιώντας την αρχιτεκτονική self RAG.

Τα αποτελέσματα αποκαλύπτουν θεμελιώδη συμβιβασμό μεταξύ πιστότητας και πληρότητας. Ο μηχανισμός επαλήθευσης υιοθετεί συντηρητική στρατηγική, απομακρύνοντας προτάσεις που δεν υποστηρίζονται πλήρως από το πλαίσιο, ακόμα και όταν αυτές συμβάλλουν στην πρακτική χρησιμότητα. Αυτό συνάδει με τη βιβλιογραφία για την αφαιρετική περίληψη [92], όπου συστήματα βελτιστοποιημένα για πιστότητα παράγουν πιο συντηρητικές και λιγότερο πλούσιες περιλήψεις. Η υπόθεση για αυτή τη συμπεριφορά είναι τριπλή: ο επαληθευτής εμφανίζει υπέρμετρη ευαισθησία, η αναθεώρηση προτιμά απομάκρυνση αντί τροποποίησης, και κρισιμότερο, η περιορισμένη ικανότητα του υποκείμενου μοντέλου αποτελεί σημαντικό παράγοντα.

Το σύστημα χρησιμοποιεί το Llama3.1 8B μέσω Ollama, το οποίο εμφανίζει περιορισμούς σε πολύπλοκες εργασίες μετα-γνωστικής φύσης όπως η αυτοεπαλήθευση. Η ασυμμετρία με τον αξιολογητή GPT-5 αποκαλύπτει σημαντική διαφορά δυνατοτήτων συλλογιστικής. Οι ρυθμίσεις (παράθυρο 8192, παραγωγή 2048 λεκτικές μονάδες) επιτρέπουν την επεξεργασία της εισόδου αναθεώρησης (1250-1750 λεκτικές μονάδες), αλλά η γνωστική πολυπλοκότητα σε συνδυασμό με το μικρό

5.5. ΠΕΙΡΑΜΑ 3: ΑΞΙΟΛΟΓΗΣΗ ΠΑΡΑΓΩΓΗΣ ΑΠΑΝΤΗΣΕΩΝ

μέγεθος μοντέλου οδηγεί σε υποβέλτιστη αξιοποίηση. Μεγαλύτερα μοντέλα επιδεικνύουν ανώτερη απόδοση σε εργασίες αυτοδιόρθωσης [93], καθώς διαθέτουν ισχυρότερες ικανότητες αιτιολόγησης και καλύτερη διατήρηση πληρότητας κατά την αναθεώρηση. Ο παρατηρούμενος συμβιβασμός μπορεί επομένως να οφείλεται μερικώς στους περιορισμούς του μοντέλου και όχι σε εγγενή αδυναμία της προσέγγισης Self-RAG.

Παρά τη βελτίωση στην πιστότητα (+2.5%), το σύστημα δεν επιτυγχάνει συνολική βελτίωση λόγω της υποβάθμισης της χρησιμότητας (-6.6%). Βραχυπρόθεσμα, προτείνεται διαβαθμισμένη επαλήθευση που διακρίνει κρίσιμες από ήπιες αποκλίσεις, στρατηγική τροποποίησης αντί απομάκρυνσης, και ρύθμιση κατωφλίων επαλήθευσης. Μεσοπρόθεσμα, η αναβάθμιση σε ισχυρότερο μοντέλο αναμένεται να βελτιώσει την ισορροπία πιστότητας-χρησιμότητας. Μακροπρόθεσμα, εναλλακτικές αρχιτεκτονικές όπως βαθμολογίες απόδοσης αντί επαλήθευσης, ή υβριδικές προσεγγίσεις με ειδικευμένα μοντέλα επαλήθευσης και γενεσιουργίας, μπορούν να προσφέρουν ευέλικτο έλεγχο του συμβιβασμού μεταξύ πιστότητας και πληρότητας.

6

Συμπεράσματα και μελλοντική εργασία

Η παρούσα διπλωματική εργασία διερεύνησε τη βελτιστοποίηση συστημάτων Επαυξημένης Παραγωγής μέσω Ανάκτησης (Retrieval-Augmented Generation, RAG) στο πεδίο των τεχνικών ερωτημάτων μηχανικής λογισμικού, εστιάζοντας στη βελτίωση τόσο του υποσυστήματος ανάκτησης όσο και του υποσυστήματος παραγωγής. Μέσω τριών συστηματικών πειραμάτων και της πρότασης ενός μηχανισμού αυτοδιορθούμενης γενεσιουργίας, αναδείχθηκαν σημαντικά ευρήματα σχετικά με τις ικανότητες και τους περιορισμούς των σύγχρονων τεχνικών, καθώς και με τους θεμελιώδεις συμβιβασμούς που χαρακτηρίζουν τα συστήματα αυτά.

6.1 ΣΥΝΘΕΣΗ ΠΕΙΡΑΜΑΤΙΚΩΝ ΕΓΡΗΜΑΤΩΝ

Η συγκριτική αξιολόγηση πέντε μεθόδων ανάκτησης (BM25, SPLADE, Dense BGE-M3, Hybrid SPLADE+BGE-M3, Hybrid BM25+BGE-M3) αποκάλυψε σαφή ιεραρχία απόδοσης που διαφοροποιείται ανάλογα με τη μετρική. Οι πυκνές μέθοδοι, και ιδιαίτερα η Dense BGE-M3, εμφάνισαν την υψηλότερη επίδοση στις μετρικές πρώτης επιτυχίας και τις rank-aware μετρικές (MAP, MRR, NDCG@k), επιβεβαιώνοντας ότι η σημασιολογική ομοιότητα υπερέχει έναντι της λεξιλογικής αντιστοιχισης σε σύντομα τεχνικά τεκμήρια. Η SPLADE υπερείχε σαφώς της BM25, αποδεικνύοντας την αξία της μαθημένης επέκτασης όρων. Οι υβριδικές προσεγγίσεις δεν ξεπέρασαν την καθαρά πυκνή ανάκτηση με προεπιλεγμένες παραμέτρους σύμφυσης, γεγονός που υποδηλώνει την ανάγκη προσεκτικής παραμετροποίησης των βιαρών των σημάτων. Παράλληλα, παρατηρήθηκε ότι οι λεξιλογικές μέθοδοι είναι ταχύτερες, ενώ οι πυκνές και υβριδικές εμφανίζουν μεγαλύτερη καθυστέρηση, χωρίς ωστόσο να αναιρούν τα οφέλη ποιότητας.

Το δεύτερο πείραμα επικεντρώθηκε στη βελτιστοποίηση των υπερπαραμέτρων του αλγορίθμου Reciprocal Rank Fusion (RRF). Η εξαντλητική αναζήτηση στον χώρο παραμέτρων (α, rrf_k) κατέδειξε ότι η βέλτιστη διαμόρφωση αντιστοιχεί σε υβριδική

6.2. ΠΕΡΙΟΡΙΣΜΟΙ ΤΗΣ ΠΑΡΟΥΣΑΣ ΕΡΕΥΝΑΣ

ανάκτηση με ισχυρή έμφαση στην πυκνή ενσωμάτωση ($\alpha^* = 0.8$, $rrf_k^* = 20$). Η παρόμετρος α αποδείχθηκε κρίσιμη για την απόδοση, με τη σύνθετη βαθμολογία να αυξάνεται μονοτονικά έως το 0.8 και να σταθεροποιείται πέραν αυτού, ενώ το rrf_k είχε αιμελητέα επίδραση. Οι μετρικές Success@3 και Precision@3 αυξήθηκαν σημαντικά με το α , ενώ η Recall@10 παρέμεινε σταθερή, επιβεβαιώνοντας ότι οι διαφορετικές μέθοδοι ανακτούν παρόμοιο πλήθος σχετικών εγγράφων αλλά διαφέρουν στην κατάταξη. Το αποτέλεσμα ερμηνεύεται από τα χαρακτηριστικά του συνόλου δεδομένων SOSum, όπου τα έγγραφα είναι σύντομα (διάμεσος 180 λεκτικές μονάδες), επιτρέποντας στα μοντέλα ενσωμάτωσης να αποτυπώνουν πλήρως το σημασιολογικό τους περιεχόμενο. Σε μακροσκελή έγγραφα, όπου απαιτείται τμηματοποίηση, η συνεισφορά του αραιού σήματος αναμένεται να είναι εντονότερη.

Το τρίτο πείραμα εστίασε στην αξιολόγηση του συστήματος από άκρη σε άκρη, συνδυάζοντας τη βελτιστοποιημένη ανάκτηση με τη διαδικασία παραγωγής απαντήσεων. Η αξιολόγηση πραγματοποιήθηκε μέσω της προσέγγισης LLM-as-a-Judge (αξιολογητής: GPT-5) σε 500 ερωτήματα, μετρώντας Πιστότητα, Συνάφεια και Χρησιμότητα. Το βασικό σύστημα RAG πέτυχε συνολική βαθμολογία 0.755 ± 0.177 , με υψηλή Συνάφεια (0.873) αλλά χαμηλότερη Πιστότητα (0.677), γεγονός που αναδεικνύει το φαινόμενο των παραισθήσεων του μοντέλου Llama3.1 8B. Η εφαρμογή του προτεινόμενου μηχανισμού αυτοδιορθούμενης παραγωγής (Self-RAG) βελτίωσε την Πιστότητα κατά 2.5 ποσοστιαίς μονάδες (0.694), όμως μείωσε τη Χρησιμότητα κατά 6.6 ποσοστιαίς μονάδες (0.669), οδηγώντας σε ελαφρά μείωση της συνολικής απόδοσης. Το αποτέλεσμα αυτό αντικατοπτρίζει τον θεμελιώδη συμβιβασμό μεταξύ ακριβειας και πληρότητας: η αυστηρή επαλήθευση μειώνει τις παραισθήσεις αλλά περιορίζει το πληροφοριακό εύρος των απαντήσεων. Επιπλέον, το μέγεθος του μοντέλου αποδείχθηκε καθοριστικός παράγοντας, καθώς τα μικρότερα μοντέλα εμφανίζουν περιορισμένες ικανότητες αυτοεπαλήθευσης και μετα-γνωστικής βελτίωσης.

Συνολικά, τα πειράματα ανέδειξαν τρία κρίσιμα συμπεράσματα:

- Η πυκνή ανάκτηση αποτελεί το σημείο αναφοράς για σύντομα τεχνικά έγγραφα, με τα υβριδικά μοντέλα να προσφέρουν οριακές βελτιώσεις όταν ρυθμίζονται κατάλληλα.
- Η βελτιστοποίηση των υπερπαραμέτρων του RRF μπορεί να βελτιώσει σημαντικά την απόδοση, αλλά η σχετική επίδραση εξαρτάται από τα χαρακτηριστικά του πεδίου.
- Η απόδοση του υποσυστήματος παραγωγής απαντήσεων εξαρτάται όχι μόνο από την ποιότητα ανάκτησης αλλά και από το μέγεθος και τις γνωστικές ικανότητες του μοντέλου.

6.2 ΠΕΡΙΟΡΙΣΜΟΙ ΤΗΣ ΠΑΡΟΥΣΑΣ ΈΡΕΥΝΑΣ

Η παρούσα έρευνα υπόκειται σε ορισμένους θεμελιώδεις περιορισμούς που επηρεάζουν την ερμηνεία και τη γενίκευση των αποτελεσμάτων, αντανακλώντας κοινές προκλήσεις στο πεδίο των συστημάτων επαυξημένης παραγωγής μέσω ανάκτησης (RAG).

ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

Στοχαστικότητα και αναπαραγωγιμότητα των γλωσσικών μοντέλων. Η εγγενής στοχαστική φύση των μεγάλων γλωσσικών μοντέλων προκαλεί σημαντική μεταβλητότητα στις απαντήσεις, καθιστώντας δύσκολη την ακριβή αναπαραγωγή των πειραμάτων. Ακόμη και με σταθερές ρυθμίσεις, παρατηρούνται αποκλίσεις στις μετρικές απόδοσης μεταξύ επαναλήψεων, ενώ η χρήση LLMs ως κριτών (*LLM-as-a-judge*) εισάγει επιπλέον αστάθεια και μεροληφία.

Περιορισμένη αξιολόγηση επανακατάταξης. Αν και το σύστημα υποστηρίζει μηχανισμούς επανακατάταξης μέσω διασταυρωμένων κωδικοποιητών (*cross-encoders*), η συστηματική αξιολόγησή τους δεν πραγματοποιήθηκε λόγω υπολογιστικών περιορισμών. Προηγούμενες μελέτες [65, 63] έχουν δείξει ότι μοντέλα όπως τα *MonoT5* και *ColBERT* μπορούν να βελτιώσουν την ακρίβεια κατά 8–15%, γεγονός που υποδεικνύει ότι τα ευρήματα της παρούσας μελέτης ενδέχεται να υποεκτιμούν το πραγματικό δυναμικό βελτίωσης.

Περιορισμοί του συνόλου δεδομένων. Η αξιολόγηση βασίστηκε σε ένα μόνο σύνολο δεδομένων (*SOSum*) με σύντομα τεχνικά κείμενα, γεγονός που περιορίζει τη γενίκευση των συμπερασμάτων σε πεδία με μακροσκελή έγγραφα ή πολυεπίπεδη δομή πληροφορίας. Επιπλέον, η απουσία επισημασμένων σχέσεων συνάφειας (*relevance labels*) εισάγει αβεβαιότητα σε μετρικές κατάταξης όπως το NDCG.

Συμβιβασμός μεταξύ πιστότητας και πληρότητας. Τα πειραματικά αποτελέσματα του μηχανισμού *Self-RAG* ανέδειξαν την ύπαρξη εγγενούς συμβιβασμού μεταξύ ακρίβειας (*faithfulness*) και πληροφοριακού πλούτου. Η αυξημένη πιστότητα συνοδεύεται από απώλεια χρησιμότητας, φαινόμενο που επιβεβαιώνει τον περιορισμό των τρεχουσών τεχνικών στην επίτευξη ισορροπίας μεταξύ αυστηρής επαλήθευσης και παραγωγικής ευελιξίας.

Οι παραπάνω περιορισμοί υπογραμμίζουν τις συστηματικές προκλήσεις που αντιμετωπίζει το πεδίο και αναδεικνύουν την ανάγκη για μελλοντική έρευνα σε τομείς όπως η ποσοτικοποίηση της αβεβαιότητας, η ανάπτυξη σταθερότερων μηχανισμών παραγωγής και η δημιουργία συνόλων δεδομένων με πλήρεις επισημάνσεις συνάφειας.

6.3 ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ ΈΡΕΥΝΑΣ

Η εργασία ανοίγει πολλαπλές κατευθύνσεις για μελλοντική έρευνα και πρακτική ανάπτυξη:

1. **Ενσωμάτωση του Πλαισίου AutoRAG.** Η υιοθέτηση του πλαισίου AutoRAG θα επιτρέψει την αυτοματοποιημένη σύνθεση και βελτιστοποίηση αγωγών (retrievers, re-rankers, generators, evaluators), ενσωματώνοντας τεχνικές meta-optimization και multi-objective αξιολόγηση κόστους, ποιότητας και καθυστέρησης. Η προσέγγιση αυτή θα προσφέρει δυναμική επιλογή βέλτιστων διαμορφώσεων για διαφορετικά περιβάλλοντα χρήσης.

6.3. ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ ΕΡΕΥΝΑΣ

2. **Online Βελτιστοποίηση με αλγόριθμο MAB.** Η ενσωμάτωση στρατηγικών τύπου *Multi-Armed Bandit (MAB)* επιτρέπει την προσαρμοστική επιλογή υπερπαραμέτρων (π.χ. α , rrf_k , top-k) και παραλλαγών prompt σε πραγματικό χρόνο, βελτιώνοντας σταδιακά την απόδοση του συστήματος με βάση ανατροφοδότηση από προηγούμενα ερωτήματα. Η διαδικασία στοχεύει στη μείωση της μεταμέλειας (regret) υπό περιορισμούς καθυστέρησης (latency) και κόστους, σύμφωνα με τη λογική της online βελτιστοποίησης υπερπαραμέτρων.
3. **Επέκταση σε Πολλαπλά Σύνολα Δεδομένων και Πεδία.** Η εφαρμογή της προτεινόμενης μεθοδολογίας σε διαφορετικά πεδία, όπως επιστημονικές δημοσιεύσεις, τεχνική τεκμηρίωση και νομικά κείμενα, θα επιτρέψει τη γενίκευση των συμπερασμάτων και τη διερεύνηση της σχέσης μεταξύ μήκους εγγράφου και βέλτιστης ισορροπίας αραιών/πυκνών σημάτων.
4. **Ανάπτυξη Γραφικού Περιβάλλοντος Πειραματισμού και Deployment Agent.** Η δημιουργία γραφικού περιβάλλοντος (GUI) για τη διαχείριση αγωγών θα επιτρέψει τη διαδραστική σύνθεση, εκτέλεση και σύγκριση πειραμάτων με οπτικοποίηση μετρικών. Επιπλέον, η ενσωμάτωση ενός deployment agent θα επιτρέπει την αυτόματη μεταφορά των πειραματικά βέλτιστων αγωγών σε παραγωγικό περιβάλλον με δυνατότητες παρακολούθησης, επαναφοράς σε προηγούμενη έκδοση (rollback) και A/B testing.
5. **Βελτίωση του Μηχανισμού Self-RAG.** Η περαιτέρω εξέλιξη του Self-RAG μπορεί να περιλαμβάνει διαβαθμισμένη επαλήθευση (διάκριση κρίσιμων και ήπιων αποκλίσεων), στρατηγικές edit-over-delete κατά την αναθεώρηση, καθώς και πιο ευέλικτα κριτήρια τερματισμού του βρόχου επαλήθευσης. Η δοκιμή του μηχανισμού με ισχυρότερα μοντέλα (GPT-5-mini, Claude Haiku 4.5) και η αξιοποίηση μικρότερων εξειδικευμένων μοντέλων για την επαλήθευση αναμένεται να βελτιώσει την ισορροπία μεταξύ πιστότητας και χρησιμότητας.

Οι παραπάνω κατευθύνσεις συνθέτουν ένα ολοκληρωμένο πλάνο εξέλιξης των συστημάτων RAG προς την κατεύθυνση της αυτοματοποίησης, της προσαρμοστικότητας και της πρακτικής αξιοποίησης, με στόχο τη δημιουργία αξιόπιστων και αναπαραγώγιμων υποδομών για την παραγωγή τεκμηριωμένης γνώσης.

Βιβλιογραφία

- [1] MRI Questions. “*Softmax Function Visualization*“. <https://mriquestions.com/softmax.html>, n.d.
- [2] Cerebras Systems. “*Context Is Everything: Why Maximum Sequence Length Matters for AI*“. <https://medium.com/@cerebras/context-is-everything-why-maximum-sequence-length-matters-for-ai-fa1f4c81009f>, 2021.
- [3] Afshin Amidi and Shervine Amidi. “*Super Study Guide: Transformers*“, pages 70–72. 2024. © 2024 Afshin Amidi and Shervine Amidi.
- [4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. “*Attention Is All You Need*“. In “*Advances in Neural Information Processing Systems (NeurIPS)*“, volume 30, 2017.
- [5] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. “*Lost in the Middle: How Language Models Use Long Contexts*“. arXiv, 2023.
- [6] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. “*Improving Language Understanding by Generative Pre-Training*“. OpenAI Technical Report, 2018.
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, et al. “*Language Models Are Few-Shot Learners*“. In “*Advances in Neural Information Processing Systems (NeurIPS)*“, volume 33, pages 1877–1901, 2020.
- [8] OpenAI. “*GPT-4 Technical Report*“. arXiv, 2023.
- [9] Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, et al. “*Large Language Models Encode Clinical Knowledge*“. Nature, 620(7972):172–180, 2023.
- [10] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. “*Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*“. In “*Advances in Neural Information Processing Systems (NeurIPS)*“, volume 33, pages 9459–9474, 2020.

- [11] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, et al. “*Improving Language Models by Retrieving from Trillions of Tokens*“. In “*International Conference on Machine Learning (ICML)*“, pages 2206–2240, 2022.
- [12] Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. “*Atlas: Few-shot learning with retrieval augmented language models*“. arXiv, 2022.
- [13] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. “*Survey of Hallucination in Natural Language Generation*“. ACM Computing Surveys, 55(12):1–38, 2023.
- [14] Stephanie Lin, Jacob Hilton, and Owain Evans. “*TruthfulQA: Measuring How Models Mimic Human Falsehoods*“. In “*Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*“, pages 3214–3252, 2022.
- [15] Or Sharir, Barak Peleg, and Yoav Shoham. “*The Cost of Training NLP Models: A Concise Overview*“. arXiv, 2020.
- [16] Stephen Robertson and Hugo Zaragoza. “*The Probabilistic Relevance Framework: BM25 and Beyond*“. Now Publishers Inc., 2009.
- [17] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. “*Dense Passage Retrieval for Open-Domain Question Answering*“. In “*Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*“, pages 6769–6781, 2020.
- [18] Xueguang Ma, Liang Guo, Ruqing Yang, Yixing Zhang, and Jimmy Lin. “*Fine-Tuning LLaMA for Multi-Stage Text Retrieval*“. arXiv, 2023.
- [19] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. “*Retrieval-Augmented Generation for Large Language Models: A Survey*“. arXiv, 2023.
- [20] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. “*Multilayer Feedforward Networks are Universal Approximators*“. Neural Networks, 2(5):359–366, 1989.
- [21] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. “*Deep Learning*“. MIT Press, 2016.
- [22] Léon Bottou, Frank E. Curtis, and Jorge Nocedal. “*Optimization Methods for Large-Scale Machine Learning*“. SIAM Review, 60(2):223–311, 2018.
- [23] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. “*Learning Representations by Back-Propagating Errors*“. Nature, 323(6088):533–536, 1986.
- [24] Diederik P. Kingma and Jimmy Ba. “*Adam: A Method for Stochastic Optimization*“. arXiv, 2014.

- [25] Xavier Glorot and Yoshua Bengio. “*Understanding the Difficulty of Training Deep Feedforward Neural Networks*“. In “*Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*“, pages 249–256, 2010.
- [26] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. “*Learning Long-Term Dependencies with Gradient Descent is Difficult*“. IEEE Transactions on Neural Networks, 5(2):157–166, 1994.
- [27] Rico Sennrich, Barry Haddow, and Alexandra Birch. “*Neural Machine Translation of Rare Words with Subword Units*“. In “*Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*“, pages 1715–1725, 2015.
- [28] Mike Schuster and Kaisuke Nakajima. “*Japanese and Korean Voice Search*“. In “*2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*“, pages 5149–5152. IEEE, 2012.
- [29] Taku Kudo. “*SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing*“. In “*Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*“, pages 66–71, 2018.
- [30] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. “*Character-Aware Neural Language Models*“. In “*Proceedings of the AAAI Conference on Artificial Intelligence*“, volume 30, 2016.
- [31] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. “*Language Models are Unsupervised Multitask Learners*“. arXiv, 2019. OpenAI Technical Report.
- [32] Sabrina J. Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y. Lee, Benoît Sagot, et al. “*Between Words and Characters: A Brief History of Open-Vocabulary Modeling and Tokenization in NLP*“. arXiv, 2021.
- [33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. “*BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*“. In “*Proceedings of NAACL-HLT*“, 2019.
- [34] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “*Efficient Estimation of Word Representations in Vector Space*“. arXiv, 2013.
- [35] Jeffrey L. Elman. “*Finding Structure in Time*“. Cognitive Science, 14(2):179–211, 1990.
- [36] Sepp Hochreiter and Jürgen Schmidhuber. “*Long Short-Term Memory*“. Neural Computation, 9(8):1735–1780, 1997.

- [37] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. “*Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation*“. In “*Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*“, pages 1724–1734, 2014.
- [38] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank J. Reddi, and Sanjiv Kumar. “*Are Transformers Universal Approximators of Sequence-to-Sequence Functions?*“. In “*International Conference on Learning Representations (ICLR)*“, 2020.
- [39] Afshine Amidi and Shervine Amidi. “*Super Study Guide: Transformers*“, pages 83–91. 2024. © 2024 Afshine Amidi and Shervine Amidi.
- [40] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. “*What Does BERT Look At? An Analysis of BERT’s Attention*“. In “*Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*“, pages 276–286, 2019.
- [41] Jesse Vig and Yonatan Belinkov. “*Analyzing the Structure of Attention in a Transformer Language Model*“. In “*Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*“, pages 63–76, 2019.
- [42] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “*Deep Residual Learning for Image Recognition*“. In “*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*“, pages 770–778, 2016.
- [43] Tong Xiao and Jingbo Zhu. “*Foundations of Large Language Models*“, pages 36–50. NLP Lab, Northeastern University & NiuTrans Research, 2025.
- [44] Joel Hestness, Sharan Narang, Newshea Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. “*Deep Learning Scaling is Predictable, Empirically*“. arXiv, 2017.
- [45] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. “*Training Compute-Optimal Large Language Models*“. arXiv, 2022.
- [46] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. “*A Survey of Large Language Models*“. arXiv, 2023.
- [47] Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Patwary, Vijay Korthikanti, Daniel Vainbrand, Prabhat Kashinkunti, Julie Bernauer, Bryan Catanzaro, et al. “*Efficient Large-Scale Language Model Training on GPU Clusters*“. In “*Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC’21)*“, 2021.

- [48] Samyam Rajbhandari, Olatunji Ruwase, Shaden Li, and Yuxiong He. “*ZeRO: Memory Optimizations Toward Training Trillion Parameter Models*“. In “*Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC’20)*“. IEEE, 2020.
- [49] Mohammad Shoeybi, Mostafa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. “*Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism*“. In “*Proceedings of the International Conference on Machine Learning (ICML)*“, 2019.
- [50] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. “*Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour*“. arXiv, 2017.
- [51] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluís Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. “*Carbon Emissions and Large Neural Network Training*“. arXiv, 2021.
- [52] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. “*Layer Normalization*“. arXiv, 2016.
- [53] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. “*Scaling Laws for Neural Language Models*“. arXiv, 2020.
- [54] Jeremy Howard and Sebastian Ruder. “*Universal Language Model Fine-Tuning for Text Classification*“. In “*Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*“, pages 328–339, 2018.
- [55] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. “*LoRA: Low-Rank Adaptation of Large Language Models*“. In “*International Conference on Learning Representations (ICLR)*“, 2021.
- [56] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. “*Training Language Models to Follow Instructions with Human Feedback*“. In “*Advances in Neural Information Processing Systems (NeurIPS)*“, 2022.
- [57] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Douwe Kiela, and Sebastian Riedel. “*Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*“. In “*Advances in Neural Information Processing Systems (NeurIPS)*“, 2020.
- [58] Kurt Shuster, Spencer Poff, Myle Ott, Emily Dinan, Y-Lan Boureau, and Jason Weston. “*Retrieval-Enhanced Adversarial Training for Neural Response Generation*“. arXiv, 2021.

- [59] Gautier Izacard and Edouard Grave. “*Leveraging Passage Retrieval with Generative Models for Open-Domain Question Answering*“. In “*Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*“, pages 874–880, 2021.
- [60] Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. “*Blended RAG: Improving Retriever-Augmented Generation Accuracy with Semantic Search and Hybrid Query-Based Retrievers*“. In “*Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*“, 2024.
- [61] Amit Singhal. “*Modern Information Retrieval: A Brief Overview*“. IEEE Data Engineering Bulletin, 24(4):35–43, 2001.
- [62] Luyu Gao, Zhuyun Dai, and Jamie Callan. “*Precise Zero-Shot Dense Retrieval without Relevance Labels*“. In “*Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*“, 2022.
- [63] Jia Fu, Xiaoting Qin, Fangkai Yang, Lu Wang, Jue Zhang, Qingwei Lin, Yubo Chen, Dongmei Zhang, Saravan Rajmohan, and Qi Zhang. “*AutoRAG-HP: Automatic Online Hyper-Parameter Tuning for Retrieval-Augmented Generation*“. In “*Findings of the Association for Computational Linguistics: EMNLP 2024*“, pages 3875–3891. Association for Computational Linguistics, 2024.
- [64] Omar Khattab and Matei Zaharia. “*ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT*“. In “*Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*“, 2020.
- [65] Rodrigo Nogueira and Jimmy Lin. “*The MonoT5: Improved Text Ranking with Fine-Tuned T5*“. arXiv, 2020.
- [66] Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. “*Query Rewriting for Retrieval-Augmented Large Language Models*“. In “*Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*“, pages 5657–5669. Association for Computational Linguistics, 2023.
- [67] Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buetzcher. “*Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods*“. In “*Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*“, pages 758–759. ACM, 2009.
- [68] Rodrigo Nogueira and Kyunghyun Cho. “*Passage Re-ranking with BERT*“. arXiv, 2019.
- [69] Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. “*LLMLingua: Compressing Prompts for Accelerated Inference of Large Language Models*“. In “*Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*“, pages 13358–13376. Association for Computational Linguistics, 2023.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [70] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. “*Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection*“. In “*Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*“, 2023.
- [71] Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. “*Active Retrieval Augmented Generation*“. In “*Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*“, pages 7969–7992, 2023.
- [72] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. “*RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval*“. In “*International Conference on Learning Representations (ICLR)*“, 2024.
- [73] Haoran Chen, Zeyu Shi, Yingfan Zhang, and Yang Liu. “*HopRAG: Multi-Hop Reasoning for Logic-Aware Retrieval-Augmented Generation*“. arXiv, 2024.
- [74] Ellen M. Voorhees. “*The TREC-8 Question Answering Track Report*“. In “*Proceedings of TREC*“, volume 8, pages 77–82, 1999.
- [75] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. “*Introduction to Information Retrieval*“, pages 155–161. Cambridge University Press, Cambridge, UK, 2008. ISBN 978-0-521-86571-5.
- [76] Kalervo Järvelin and Jaana Kekäläinen. “*Cumulated Gain-based Evaluation of IR Techniques*“. ACM Transactions on Information Systems (TOIS), 20(4):422–446, 2002.
- [77] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. “*BLEU: A Method for Automatic Evaluation of Machine Translation*“. In “*Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*“, pages 311–318. Association for Computational Linguistics, 2002.
- [78] Chin-Yew Lin. “*ROUGE: A Package for Automatic Evaluation of Summaries*“. In “*Text Summarization Branches Out*“, pages 74–81. Association for Computational Linguistics, 2004.
- [79] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. “*BERTScore: Evaluating Text Generation with BERT*“. arXiv, 2019.
- [80] Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. “*RAGAS: Automated Evaluation of Retrieval Augmented Generation*“. arXiv, 2023.
- [81] Dongkyu Kim, Byoungwook Kim, Donggeon Han, and Matouš Eibich. “*AutoRAG: Automated Framework for Optimization of Retrieval-Augmented Generation Pipeline*“. In “*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*“, 2024.

- [82] Matouš Eibich, Donggeon Han, Dongkyu Kim, and Byoungwook Kim. “ARAGOG: A Benchmark for Advanced RAG Output Grading“. In “Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)“, 2024.
- [83] Yury A. Malkov and Dmitry A. Yashunin. “Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs“. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(4):824–836, 2018.
- [84] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. “SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval“. In “Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval“, pages 2267–2277, 2022.
- [85] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. “MS MARCO: A human generated machine reading comprehension dataset“. In “arXiv preprint arXiv:1611.09268“, 2016.
- [86] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. “Natural Questions: A Benchmark for Question Answering Research“. Transactions of the Association for Computational Linguistics, 7:452–466, 2019.
- [87] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. “HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering“. In “Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing“, pages 2369–2380, 2018.
- [88] Themistoklis Diamantopoulos and Andreas L. Symeonidis. “A Directory of Datasets for Mining Software Repositories“. Data, 10(3):28, 2025.
- [89] Bonan Kou, Shengmai Zhang, and Tianyi Zhang. “SOSum: A Dataset of Stack Overflow Post Summaries“. In “2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR)“, pages 709–713. IEEE, 2022.
- [90] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. “BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation“. arXiv, 2024.
- [91] Prithiviraj Damodaran. “Splade_PP_en_v1: Independent Implementation of SPLADE++ Model (a.k.a. splade-cocondenser* and family) for the Industry setting“, 2024. Computer software.
- [92] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. “On Faithfulness and Factuality in Abstractive Summarization“. In “Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics“, pages 1906–1919. Association for Computational Linguistics, 2020.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [93] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. “*Self-Refine: Iterative Refinement with Self-Feedback*”. In “*Advances in Neural Information Processing Systems*”, volume 36, pages 46534–46594, 2023.