ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Πειραματισμός με το περιβάλλον OpenAI / ChatGPT και ανάπτυξη πρωτότυπου εξειδικευμένου bot

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Σωτηρόπουλου Ανδρέα

Επιβλέπων:

Αλεξόπουλος Χαράλαμπος (Επίκουρος Καθηγητής)

Μέλη εξεταστικής επιτροπής:

Χαραλαμπίδης Ιωάννης (Καθηγητής)

Λουκής Ευριπίδης (Καθηγητής)

Σάμος, Μάρτιος 2025

Περιεχόμενα

Περίληψη

Η παρούσα εργασία αφορά την ανάλυση των μοντέλων Generative AI (Large Language Models - LLMs) και την χρήση τους σε chatbots. Ειδικότερα αναλύουμε την περίπτωση του ChatGPT και της OpenAI.

Υλοποιούμε έναν πρότυπο ψηφιακό βοηθό για την υποστήριξη των Οργανισμών Τοπικής Αυτοδιοίκησης (ΟΤΑ) χρησιμοποιώντας τη μέθοδο Retrieval-Augmented Generation (RAG) για την βελτίωση της ποιότητας των απαντήσεων που παράγουν τα LLMs και την δυνατότητα εμφάνισης των πηγών από τις οποίες προέρχονται οι απαντήσεις, έτσι ώστε να δίνεται η δυνατότητα επιβεβαίωσης της ορθότητας των απαντήσεων.

Το περιεχόμενο αφορά την εθνική νομοθεσία γύρω από τους ΟΤΑ και δίνεται έμφαση στις δημόσιες προμήθειες και τους δήμους.

1. Εισαγωγή

Οι αυτόματοι ψηφιακοί βοηθοί, κοινώς γνωστοί ως chatbots, είναι λογισμικά, τα οποία ανήκουν στον τομέα της τεχνητής νοημοσύνης και σκοπός τους είναι η διεκπαιρέωση συζητήσεων με ανθρώπους σε φυσική γλώσσα (“CHATBOT | English meaning - Cambridge Dictionary”). Τα chatbots μπορούν να βρουν εφαρμογές σε διάφορους τομείς, όπως η παροχή υπηρεσιών, η ενημέρωση, η ανάκτηση πληροφορίας κ.α., τόσο στον δημόσιο όσο και στον ιδιωτικό τομέα.

Από το 1966 που δημιουργήθηκε το πρώτο chatbot (Weizenbaum #36) μέχρι και σήμερα έχουν γίνει πολλές προσπάθειες για την υλοποίηση ενός πιστικού και χρήσιμου chatbot με την πιο σημαντική να είναι αυτή του ChatGPT το 2022. Το ChatGPT είναι αποτέλεσμα της μεγάλης ανάπτυξης στη Μηχανική Μάθηση (Machine Learning - ML) που ξεκίνησε στα μέσα του 2010 και βασίζεται στην αρχιτεκτονική Transformers. Μετά την δημοσίευσή του το 2022 υπήρξε μεγάλο ενδιαφέρον για τα LLM και τους ψηφιακούς βοηθούς γενικότερα. Μέχρι και σήμερα συνεχίζονται οι έρευνες για την εφαρμογή της τεχνολογίας πίσω από το ChatGPT για την υλοποίηση εξειδικευμένων ψηφιακών βοηθών και άλλων εργαλείων.

Σε αυτή την εργασία θα αναλύσουμε τα γλωσσικά μοντέλα που απαρτίζουν το ChatGPT, τις αδυναμίες τους και τους τρόπους με τους οποίους μπορούμε να υλοποιήσουμε έναν ψηφιακό βοηθό για την υποστήριξη των Οργανισμών Τοπικής Αυτοδιοίκησης (ΟΤΑ).

2. Τεχνητή νοημοσύνη και βασικές έννοιες

Η τεχνητή νοημοσύνη (ΤΝ), ή Artificial Intelligence (AI), είναι «η επιστήμη και η μηχανική της δημιουργίας έξυπνων μηχανών» όπως ορίστηκε από τον John McCarthy το 1955. Σκοπός είναι η ανάπτυξη συστημάτων ικανών να εκτελούν εργασίες που απαιτούν νοημοσύνη όταν εκτελούνται από ανθρώπους. Αυτές οι εργασίες μπορεί να περιλαμβάνουν την αναγνώριση προτύπων, τη λήψη αποφάσεων, τη μάθηση, την κατανόηση της φυσικής γλώσσας, τη ρομποτική κίνηση και άλλες πολύπλοκες γνωστικές λειτουργίες.

Παρακάτω εξηγούμε κάποιους βασικούς όρους στο χώρο της ΤΝ δίνοντας έμφαση στους κλάδους που έχουν αναπτυχθεί περισσότερο τα τελευταία χρόνια:

2.1 Επεξεργασία Φυσικής Γλώσσας

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP) είναι ένας υποτομέας της τεχνητής νοημοσύνης που επικεντρώνεται στην αλληλεπίδραση μεταξύ υπολογιστών και ανθρώπινων γλωσσών. Στόχος του είναι η ανάπτυξη αλγορίθμων που επιτρέπουν στους υπολογιστές να κατανοούν, να επεξεργάζονται και να παράγουν ανθρώπινη γλώσσα. Η NLP περιλαμβάνει εργασίες όπως η ανάλυση και η κατανόηση κειμένων, η αναγνώριση ομιλίας, η αυτόματη μετάφραση, η εξόρυξη πληροφοριών από κείμενα και η σύνθεση φυσικής γλώσσας.

Η επεξεργασία φυσικής γλώσσας είναι πολυδιάστατη και περιλαμβάνει τόσο συντακτική ανάλυση (δηλαδή την ανάλυση της δομής μιας πρότασης) όσο και σημασιολογική ανάλυση (την κατανόηση του νοήματος των λέξεων και των προτάσεων). Στις πιο σύγχρονες εφαρμογές, η NLP χρησιμοποιεί τεχνικές βαθιάς μάθησης και νευρωνικών δικτύων, όπως τα μοντέλα Transformers και τα γλωσσικά μοντέλα όπως το GPT, για τη βελτίωση της κατανόησης και της παραγωγής γλώσσας από μηχανές.

2.2 Μηχανική Μάθηση

Η μηχανική μάθηση (machine learning - ML) είναι ένας υποτομέας της ΤΝ, ο οποίος αφορά την ανάπτυξη αλγορίθμων και στατιστικών μοντέλων που ονομάζονται νευρωνικά δίκτυα (Neural Networks - NN) και επιτρέπουν στους υπολογιστές να εκτελούν εργασίες χωρίς ρητές οδηγίες. Τα NN βασίζονται σε πρότυπα και συμπεράσματα που αντλούνται από δεδομένα για την εκτέλεση διαδικασιών, όπως η αναγνώριση εικόνων, η πρόβλεψη τάσεων ή η παροχή συστάσεων. (Mitchell #1)

Βασικά στοιχεία και τύποι ML:

- **Επιβλεπόμενη Μάθηση (Supervised Learning):** Εκπαίδευση μοντέλων με δεδομένα που περιέχουν ετικέτες, όπου η απόδοση του μοντέλου αξιολογείται βάσει της ικανότητάς του να προβλέπει τις σωστές ετικέτες για νέα δεδομένα.

- **Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning):** Μάθηση από δεδομένα που δεν έχουν ετικέτες, με στόχο την αυτόματη εύρεση δομών ή προτύπων στα δεδομένα, όπως ομαδοποίηση (clustering) ή μείωση διαστάσεων (dimensionality reduction).
- **Ενισχυτική Μάθηση (Reinforcement Learning):** Μάθηση μέσω αλληλεπίδρασης με ένα περιβάλλον και βελτίωση της απόδοσης βάσει των ανταμοιβών ή ποινών που λαμβάνονται από τις ενέργειες που εκτελούνται.

Η Μηχανική Μάθηση αποτελεί τον πυρήνα πολλών εφαρμογών, από την ανάλυση μεγάλων δεδομένων (big data analytics) έως την αυτόνομη οδήγηση και τα συστήματα σύστασης.

2.3 Βαθιά Μάθηση

Η βαθιά μάθηση είναι ένας υποτομέας της ML που χρησιμοποιεί νευρωνικά δίκτυα με πολύπλοκες, πολυεπίπεδες δομές. Το κύριο χαρακτηριστικό της βαθιάς μάθησης είναι η χρήση "βαθιών" νευρωνικών δικτύων, δηλαδή δικτύων με πολλές στρώσεις (layers) από τεχνητούς νευρώνες, που επιτρέπουν την αναγνώριση σύνθετων μοτίβων και σχέσεων στα δεδομένα. (LeCun et al. #1)

2.4 Μετασχηματιστές

Οι Μετασχηματιστές είναι ένα πρωτοποριακό μοντέλο βαθιάς μάθησης που εισήχθη το 2017 στο πεδίο της επεξεργασίας φυσικής γλώσσας και της μηχανικής μάθησης. Το βασικό χαρακτηριστικό των μετασχηματιστών είναι ο μηχανισμός προσοχής (attention mechanism), ο οποίος επιτρέπει στο μοντέλο να εντοπίζει και να εστιάζει σε σημαντικά μέρη της εισόδου ανεξαρτήτως της απόστασης μεταξύ των λέξεων ή των στοιχείων μέσα στην ακολουθία. Αυτό επιτρέπει στους μετασχηματιστές να είναι εξαιρετικά ευέλικτοι και αποτελεσματικοί σε εργασίες μεγάλης κλίμακας που περιλαμβάνουν μεγάλα κείμενα ή ακολουθιακά δεδομένα. (Vaswani et al. #1)

Οι Μετασχηματιστές αποτέλεσαν τη βάση για πολλές προχωρημένες αρχιτεκτονικές γλωσσικών μοντέλων, όπως το BERT (Bidirectional Encoder Representations from Transformers) και το GPT (Generative Pre-trained Transformer), τα οποία έχουν επιτύχει κορυφαίες επιδόσεις σε διάφορες εργασίες επεξεργασίας φυσικής γλώσσας.

2.5 Παραγωγική Τεχνητή Νοημοσύνη

Η Παραγωγική Τεχνητή Νοημοσύνη αναφέρεται σε μια κατηγορία αλγορίθμων μηχανικής μάθησης που έχουν τη δυνατότητα να δημιουργούν νέο περιεχόμενο, όπως κείμενο, εικόνες, μουσική ή βίντεο με βάση τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση.

Οι γενεσιουργοί αλγόριθμοι δεν περιορίζονται μόνο στην ανάλυση ή την κατανόηση των δεδομένων, αλλά παράγουν πρωτότυπο υλικό με βάση τα μοτίβα και τις πληροφορίες που έχουν μάθει από τα δεδομένα εισόδου.

2.5.1 Generative AI - Τάσεις και πρότυπα

Η GenAI έχει αναδειχθεί σε μια από τις πιο σημαντικές και καινοτόμες τεχνολογίες του 21ου αιώνα. Αυτή η τεχνολογία επιτρέπει τη δημιουργία περιεχομένου, όπως κείμενο, εικόνες, μουσική και βίντεο, με τρόπο που είναι εντυπωσιακά παρόμοιος με αυτόν που δημιουργούν οι άνθρωποι.

Τα γενετικά μοντέλα έχουν σημαντικές εφαρμογές σε ποικίλους τομείς, όπως η ψυχαγωγία, η ιατρική, η εκπαίδευση, η διαφήμιση κ.α.:

- **Ενσωμάτωση στα Εργαλεία Δημιουργίας Περιεχομένου**: Όλο και περισσότερες πλατφόρμες και εργαλεία δημιουργίας περιεχομένου ενσωματώνουν γενετικά μοντέλα. Από τη δημιουργία κειμένου για διαφημίσεις και ιστοσελίδες έως την παραγωγή προσαρμοσμένων εικόνων και βίντεο, η GenAI προσφέρει νέες δυνατότητες και αυξάνει την παραγωγικότητα.

- **Αύξηση της Ακρίβειας και της Ρεαλιστικότητας**: Τα νέα γενετικά μοντέλα, όπως τα GPT-4o και DALL-E 3, είναι σε θέση να παράγουν περιεχόμενο με εξαιρετική ακρίβεια και ρεαλιστικότητα. Αυτά τα μοντέλα έχουν εκπαιδευτεί σε τεράστια σύνολα δεδομένων (dataset) και είναι ικανά να κατανοούν και να μιμούνται τη φυσική γλώσσα και τις εικόνες με τρόπους που ήταν αδιανοητοι στο παρελθόν.

- **Εξατομίκευση**: Μια από τις σημαντικότερες τάσεις είναι η χρήση γενετικών μοντέλων για την εξατομίκευση του περιεχομένου. Οι επιχειρήσεις χρησιμοποιούν την GenAI για να δημιουργούν προσαρμοσμένο περιεχόμενο που ανταποκρίνεται στις ανάγκες και τις προτιμήσεις των μεμονωμένων πελατών, βελτιώνοντας έτσι την εμπειρία του χρήστη και αυξάνοντας την αφοσίωση των πελατών.

- **Χρήση στη Φροντίδα Υγείας**: Στην ιατρική, η GenAI χρησιμοποιείται για τη δημιουργία προσομοιώσεων και την ανάλυση μεγάλων ποσοτήτων δεδομένων, με στόχο τη βελτίωση της διάγνωσης και της θεραπείας, αλλά και της δημιουργίας νέων φαρμάκων.

- **Ανάπτυξη Εφαρμογών Αυτοματοποίησης**: Τα γενετικά μοντέλα χρησιμοποιούνται επίσης για την αυτοματοποίηση σύνθετων διαδικασιών. Για παράδειγμα, στην ανάπτυξη λογισμικού, μπορούν να γράφουν κώδικα ή να εντοπίζουν σφάλματα, ενώ στην εξυπηρέτηση πελατών, χρησιμοποιούνται για την παροχή αυτοματοποιημένων απαντήσεων σε συχνές ερωτήσεις.

2.6 Μεγάλα Γλωσσικά Μοντέλα

Τα Μεγάλα Γλωσσικά Μοντέλα (Large Language Models - LLM) είναι προηγμένα μοντέλα τεχνητής νοημοσύνης που εκπαιδεύονται σε τεράστια σύνολα δεδομένων κειμένου (από όπου παίρνουν και το όνομά τους) για την κατανόηση και την παραγωγή φυσικής γλώσσας. Αυτά τα μοντέλα βασίζονται σε αρχιτεκτονικές νευρωνικών δικτύων, όπως οι Transformers, και έχουν τη δυνατότητα να παράγουν ανθρώπινη γλώσσα με εξαιρετική ακρίβεια και συνοχή. (“What is a large language model (LLM)?”, n.d.)

2.6.1 Χαρακτηριστικά και Εφαρμογές των LLM:

- **Κατανόηση Κειμένου:** Τα LLM μπορούν να κατανοούν πολύπλοκα κείμενα και να απαντούν σε ερωτήσεις με βάση το περιεχόμενο των κειμένων αυτών. Αυτό τα καθιστά ιδανικά για εφαρμογές όπως οι βοηθοί τεχνητής νοημοσύνης και τα chatbots.

- **Παραγωγή Κειμένου:** Μπορούν να δημιουργούν πρωτότυπο περιεχόμενο, όπως άρθρα, ποιήματα ή ακόμα και κώδικα προγραμματισμού. Τα μοντέλα GPT (Generative Pre-trained Transformer), όπως το GPT-4o, είναι παραδείγματα τέτοιων μοντέλων.

- **Μετάφραση και Σύνοψη:** Τα LLM μπορούν να χρησιμοποιηθούν για τη μετάφραση κειμένων από μια γλώσσα σε άλλη και για τη δημιουργία συνόψεων μεγάλων κειμένων, καθιστώντας τα χρήσιμα σε πολυγλωσσικά περιβάλλοντα και σε εφαρμογές ανάλυσης δεδομένων.

- **Συνεργασία με Ανθρώπους:** Αυτά τα μοντέλα χρησιμοποιούνται σε περιβάλλοντα όπου απαιτείται συνεργασία ανθρώπου και μηχανής, βελτιώνοντας τη διαδικασία λήψης αποφάσεων και ενισχύοντας την ανθρώπινη δημιουργικότητα.

2.7 Generative Pre-trained Transformer (GPT)

Το GPT (Generative Pre-trained Transformer - Παραγωγικός Προεκπαιδευμένος Μετασχηματιστής) είναι μια αρχιτεκτονική μοντέλου μηχανικής μάθησης που αναπτύχθηκε από την OpenAI και ανήκει στην κατηγορία των μετασχηματιστών (Transformers). Το GPT σχεδιάστηκε για να παράγει κείμενο που μοιάζει με αυτό που θα έγραφε ένας άνθρωπος, με την ικανότητα να ολοκληρώνει προτάσεις, να απαντά σε ερωτήσεις, να δημιουργεί περιεχόμενο με φυσική γλώσσα και να εκτελεί πολλές άλλες γλωσσικές εργασίες.

Η διαδικασία εκπαίδευσης του GPT περιλαμβάνει δύο κύρια στάδια (Radford and Narasimhan #3):

1. **Προεκπαίδευση (Pre-training):**
   Στο στάδιο αυτό, το μοντέλο εκπαιδεύεται σε μεγάλα σύνολα δεδομένων από κείμενο, τα οποία περιλαμβάνουν άρθρα, βιβλία, και άλλες πηγές πληροφοριών. Το μοντέλο μαθαίνει να προβλέπει την επόμενη λέξη σε μια πρόταση με βάση τις προηγούμενες λέξεις. Αυτό το στάδιο επιτρέπει στο μοντέλο να αποκτήσει μια γενική κατανόηση της γλώσσας και των δομών της.
2. **Εκπαίδευση με Επιτήρηση (Fine-tuning):**
   Αφού ολοκληρωθεί η προεκπαίδευση, το μοντέλο μπορεί να προσαρμοστεί σε συγκεκριμένες εργασίες ή τομείς μέσω εκπαίδευσης με επιτήρηση, χρησιμοποιώντας πιο περιορισμένα και στοχευμένα σύνολα δεδομένων. Αυτό το στάδιο βοηθά το μοντέλο να βελτιώσει τις επιδόσεις του σε συγκεκριμένες εφαρμογές, όπως η αυτόματη απάντηση σε ερωτήσεις ή η δημιουργία περιεχομένου.

Η αρχιτεκτονική του GPT στηρίζεται στη μηχανή αυτο-προσοχής (self-attention), που επιτρέπει στο μοντέλο να εξετάζει όλες τις λέξεις μιας πρότασης ταυτόχρονα και να αναγνωρίζει τη

σημασία τους σε σχέση με άλλες λέξεις. Αυτό το χαρακτηριστικό καθιστά το GPT εξαιρετικά ικανό στη διαχείριση πολύπλοκων γλωσσικών εργασιών, όπως η σύνθεση κειμένου, η ανάλυση συναισθημάτων και η αυτόματη μετάφραση.

Η OpenAI, δηλαδή η εταιρία πίσω από το GPT, συνεχώς αναπτύσει το μοντέλο εκπαιδεύοντας το με περισσότερα και πιο επικαιροποιημένα δεδομένα. Το ChatGPT αρχικά βασίστηκε στην έκδοση GPT-3.5 και τώρα βρίσκεται στην έκδοση GPT-4o. Τα μοντέλα αυτά είναι διαθέσιμα προς χρήση από τρίτους μέσω της αντίστοιχης πλατφόρμας της εταιρίας.

Επίσης να αναφέρουμε ότι αντίστοιχα μοντέλα διατίθενται και από άλλες εταιρείες με τα πιο σημαντικά να είναι το Llama της Meta, Το Claude της Anthropic και το Gemini της Google.

2.8 Ηθικά Ζητήματα

Με την αυξανόμενη χρήση της τεχνητής νοημοσύνης, προκύπτουν σημαντικά ζητήματα σχετικά με την ηθική χρήση της τεχνολογίας. Τα ζητήματα αυτά περιλαμβάνουν (Floridi and Chiriatti 2020, #):

- **Πνευματικά δικαιώματα και ιδιοκτησία**: Υπάρχουν ερωτήματα σχετικά με το ποιος έχει τα πνευματικά δικαιώματα για το περιεχόμενο που παράγεται από γενετικά μοντέλα και πώς μπορεί να χρησιμοποιηθεί νόμιμα. Επίσης δεν είναι ξεκάθαρο αν έχει δοθεί ή και αν χρειάζεται να δοθεί συγκατάθεση για τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση ενός μοντέλου από τους κατόχους των πνευματικών δικαιωμάτων.
- **Ευθύνη**: Είναι σημαντικό να καθοριστεί ποιός είναι υπεύθυνος για τις αποφάσεις που λαμβάνονται από τα συστήματα που βασίζονται στην ΤΝ.
- **Καταπολέμηση της παραπληροφόρησης**: Η ικανότητα των γενετικών μοντέλων να δημιουργούν πειστικό ψευδές περιεχόμενο θέτει σοβαρά ζητήματα σχετικά με την παραπληροφόρηση και τη διάδοση ψευδών ειδήσεων.
- **Ηθική χρήση και προκαταλήψεις (bias)**: Τα γενετικά μοντέλα ενσωματώνουν προκαταλήψεις που υπάρχουν στα δεδομένα με τα οποία έχουν εκπαιδευτεί. Η διαχείριση αυτών των προκαταλήψεων είναι κρίσιμη για την ηθική χρήση της τεχνολογίας.

Η ανάπτυξη και εφαρμογή κανονιστικών πλαισίων και προτύπων είναι κρίσιμη για την αντιμετώπιση αυτών των ζητημάτων και για τη διασφάλιση ότι η ΤΝ χρησιμοποιείται υπεύθυνα και με τρόπο που προάγει το κοινό καλό.

Τα πρότυπα και οι κανονισμοί γύρω από τη χρήση των τεχνολογιών NLP, ML, και LLM είναι κρίσιμα για την προώθηση υπεύθυνης και ασφαλούς ανάπτυξης της τεχνητής νοημοσύνης. Καθώς αυτές οι τεχνολογίες συνεχίζουν να εξελίσσονται, η διαχείριση των ηθικών και νομικών ζητημάτων που τις συνοδεύουν γίνεται όλο και πιο σημαντική.

3. OpenAI και ChatGPT

Η OpenAI είναι μια εταιρεία τεχνητής νοημοσύνης που εστιάζει στην ανάπτυξη προηγμένων μοντέλων μηχανικής μάθησης και AI. Ένα από τα, πιο γνωστά, προϊόντα της είναι το ChatGPT, ένα chatbot το οποίο βασίζεται στα προηγμένα γλωσσικά μοντέλα της εταιρείας. Το ChatGPT μπορεί να απαντά σε ερωτήσεις, να συμμετέχει σε συνομιλίες, να γράφει κείμενα και να παρέχει πληροφορίες σε πολλούς διαφορετικούς τομείς.

Στην ενότητα αυτή θα αναλύσουμε την ανάπτυξη του ChatGPT, πως χρησιμοποιείται, αλλά και πως μπορούμε να υλοποιήσουμε τα δικά μας εργαλεία χρησιμοποιώντας τα LLMs της OpenAI.

3.1 Η εκπαίδευση του ChatGPT

Σύμφωνα με την OpenAI (“Introducing ChatGPT” 2022) το ChatGPT αποτελεί λεπτομερή προσαρμογή ενός μοντέλου της σειράς GPT-3.5, με την εκπαίδευσή του να έχει ολοκληρωθεί στις αρχές του 2022.

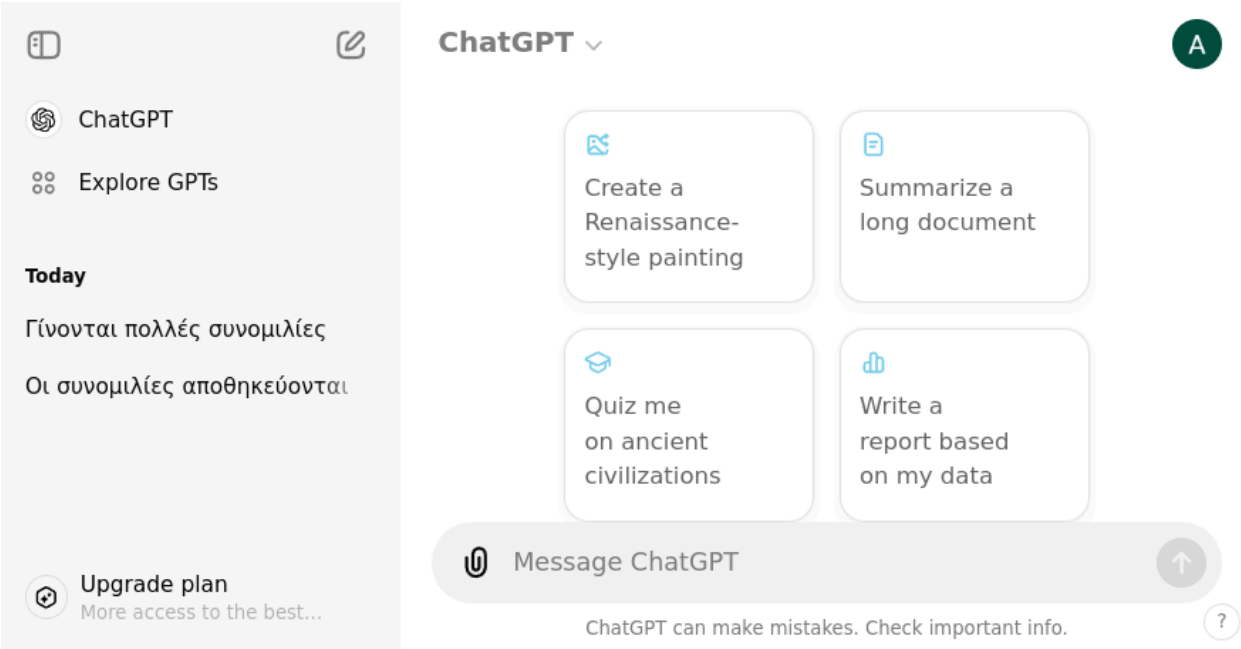
Η εκπαίδευση έγινε με τη χρήση της Ενισχυτικής Μάθησης με Ανθρώπινη Ανατροφοδότηση (Reinforcement Learning from Human Feedback - RLHF), ακολουθώντας τις ίδιες μεθόδους που χρησιμοποιήθηκαν και για το InstructGPT, αλλά με κάποιες διαφορές στον τρόπο συλλογής δεδομένων. Αρχικά, η εκπαίδευση πραγματοποιήθηκε με επιτηρούμενη λεπτομερή προσαρμογή (supervised fine-tuning), όπου οι εκπαιδευτές ανέλαβαν να παίξουν τον ρόλο τόσο του χρήστη όσο και του βοηθού AI σε συνομιλίες. Στους εκπαιδευτές δόθηκε πρόσβαση σε προτάσεις που δημιουργήθηκαν από το ίδιο το μοντέλο, προκειμένου να τους βοηθήσουν στη σύνθεση των απαντήσεών τους. Αυτό το νέο σύνολο διαλόγων συνδυάστηκε με δεδομένα από το InstructGPT, τα οποία μετατράπηκαν σε μορφή διαλόγου. (“Introducing ChatGPT” 2022)

Για τη δημιουργία ενός μοντέλου ανταμοιβής, ήταν απαραίτητη η συλλογή δεδομένων σύγκρισης. Αυτά τα δεδομένα αποτελούνταν από διάφορες απαντήσεις του μοντέλου, οι οποίες ταξινομήθηκαν με βάση την ποιότητά τους. Η συλλογή αυτών των δεδομένων έγινε μέσω συνομιλιών που είχαν οι εκπαιδευτές AI με το chatbot, από τις οποίες επιλέχθηκε τυχαία ένα μήνυμα γραμμένο από το μοντέλο. Στη συνέχεια, δειγματίστηκαν εναλλακτικές ολοκληρώσεις, και οι εκπαιδευτές κλήθηκαν να τις κατατάξουν. Χρησιμοποιώντας αυτά τα δεδομένα ανταμοιβής, το μοντέλο βελτιστοποιήθηκε μέσω της μεθόδου Proximal Policy Optimization (PPO), μια διαδικασία που επαναλήφθηκε αρκετές φορές. (“Introducing ChatGPT” 2022)

Πλέον η OpenAI προσφέρει τη δυνατότητα χρήσης διαφορετικών μοντέλων της πιο ανεπτυγμένων από το αρχικό. Μάλιστα το GPT-3.5 δεν προσφέρεται πλέον και έχει αντικατασταθεί από τα GPT-4o και GPT-4o mini. Η εταιρία συνεχώς εξελίσει και εκπαιδεύει νέα μοντέλα με περισσότερα και πιο πρόσφατα δεδομένα, μικρότερο κόστος χρήσης και περισσότερες δυνατότητες.

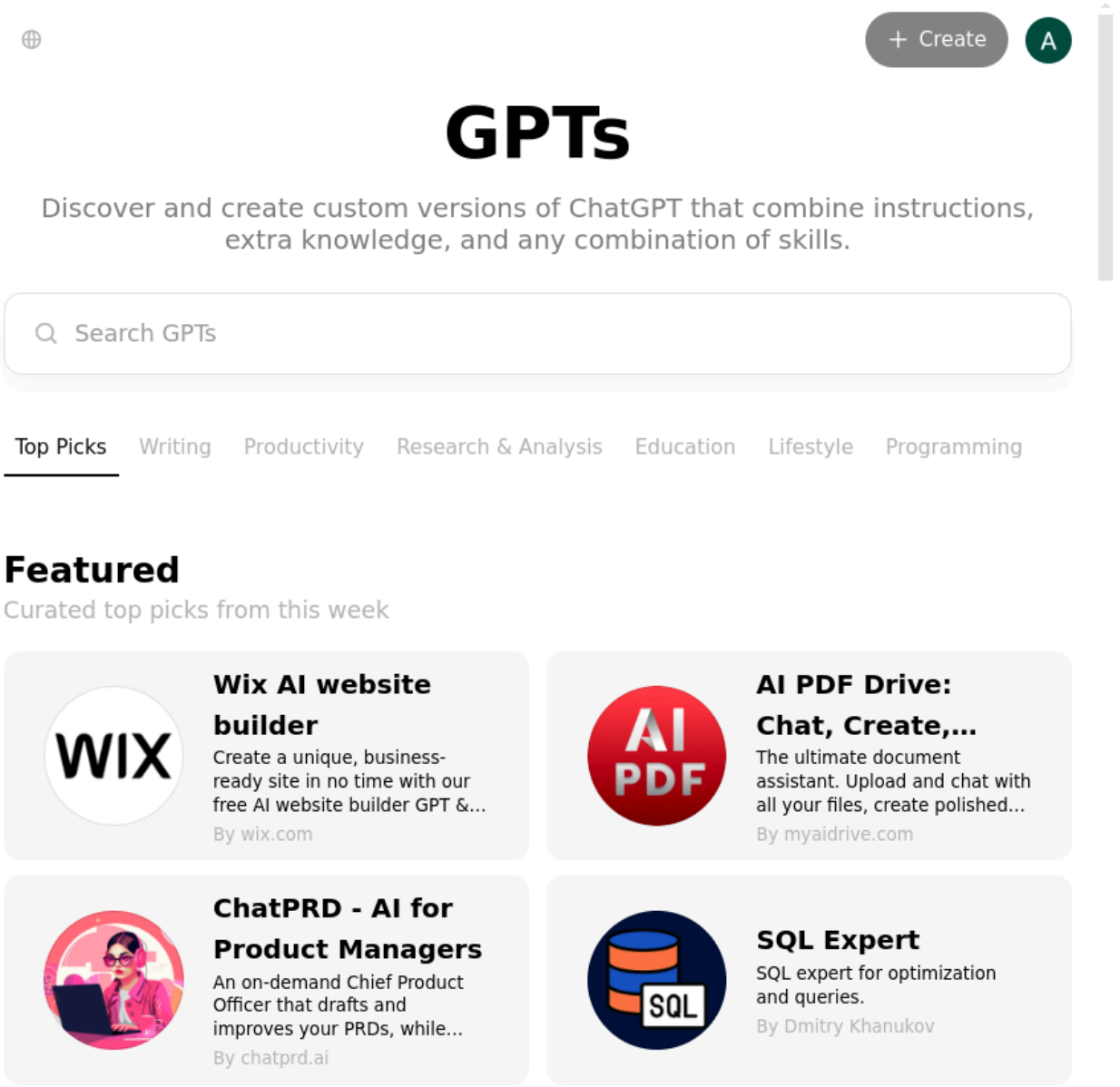
3.2 ChatGPT στην πράξη

Το ChatGPT (τη χρονική στιγμή σύνταξης της εργασίας) είναι διαθέσιμο δωρεάν στη διεύθυνση https://chatgpt.com. Αυτή η έκδοση χρησιμοποιεί το μοντέλο “GPT 4o mini”. Οι χρήστες μπορούν, επίσης, να δημιουργήσουν έναν δωρεάν λογαριασμό στην υπηρεσία, έτσι ώστε να μπορούν να αποθηκεύσουν τις συνομιλίες τους, να έχουν περιορισμένη πρόσβαση, δηλαδή χρήση για συγκεκριμένο αριθμό μηνυμάτων ανα ημέρα, στο μοντέλο “GPT 4o”:



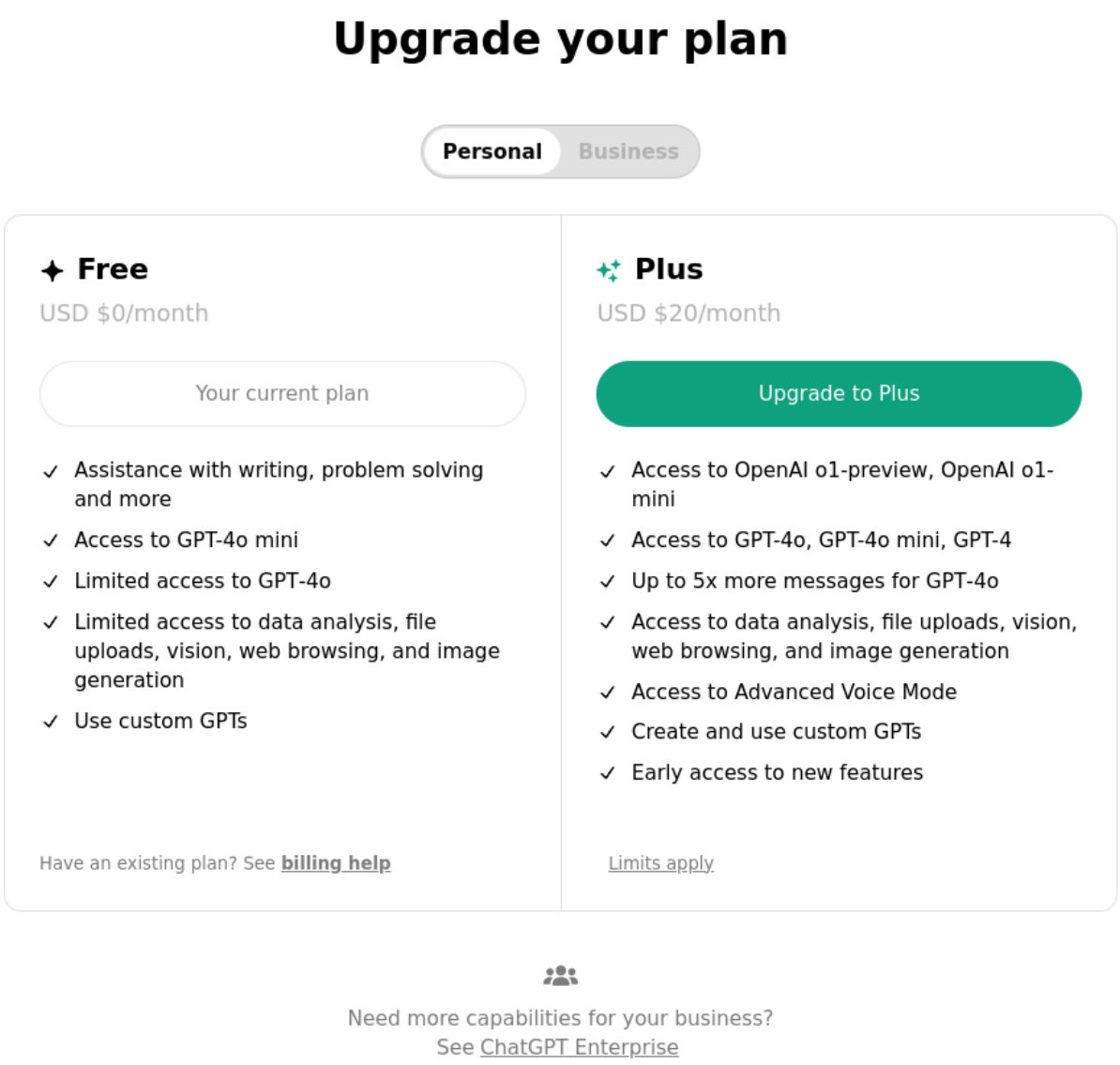
Αρχική σελίδα του ChatGPT με συνδεδεμένο λογαριασμό χρήστη

Ακόμα προσφέρεται η λειτουργία *Explore GPTs* η οποία επιτρέπει τη χρήση προσαρμοσμένων και εξειδικευμένων εφαρμογών των μοντέλων GPT. Αυτού του είδους οι εφαρμογές μπορούν να βοηθήσουν σε εργασίες όπως ο προγραμματισμός, η δημιουργία κειμένων, η δημιουργία εικόνων κλπ.:



Λίστα με προσαρμοσμένες εφαρμογές του ChatGPT

Τέλος, οι χρήστες μπορούν να πληρώσουν για την πλήρη έκδοση της υπηρεσίας και να αποκτήσουν πρόσβαση στα πιο πρόσφατα και ικανά μοντέλα της OpenAI όπως το *OpenAI o1-preview* / *OpenAI o1-mini* / *GPT-4o* κλπ.



Λίστα αναβάθμισης στην επί-πληρωμή έκδοση του ChatGPT

3.3 Ανάπτυξη εφαρμογών μέσω του OpenAI API

Μέσω του API της η OpenAI προσφέρει και ένα σύνολο εργαλείων που επιτρέπουν στους προγραμματιστές να ενσωματώσουν τα μοντέλα τεχνητής νοημοσύνης της στις εφαρμογές τους. Με τη χρήση των εργαλείων αυτών, οι επιχειρήσεις και οι προγραμματιστές μπορούν να δημιουργήσουν εξατομικευμένες εμπειρίες και να αυτοματοποιήσουν διαδικασίες που απαιτούν κατανόηση και παραγωγή γλώσσας, χωρίς να χρειάζεται να δημιουργήσουν και να συντηρήσουν τα δικά τους μοντέλα ΤΝ.

Κάποια από τα εργαλεία που προσφέρονται είναι τα εξής (“API platform”):

- Chat Completions API - Για πρόσβαση στα μοντέλα κειμένου όπως το GPT-4o.

- Realtime API - Για την υλοποίηση εφαρμογών χαμηλής καθυστέρησης όπως το speech-to-speech.
- Assistants API - Για την υλοποίηση ψηφιακών βοηθών σε υπάρχουσες εφαρμογές.
- Function calling - Όπου δίνεται η δυνατότητα αλληλεπίδρασης του μοντέλου με κώδικα.
- Vision - Για την υποστήριξη εικόνων.
- JSON Mode - Ώστε η έξοδος του μοντέλου να είναι σε δομημένη μορφή JSON.
- Fine-tuning - Για την περαιτέρω εκπαίδευση των μοντέλων ώστε να αποδίδουν καλύτερα σε συγκεκριμένες διαδικασίες.

Χρησιμοποιώντας το API της OpenAI θα υλοποιήσουμε παρακάτω το δικό μας chatbot για για την υποστήριξη των Οργανισμών Τοπικής Αυτοδιοίκησης (ΟΤΑ).

3.4 Περιορισμοί των Generative LLMs

Παρά τις πολλές δυνατότητες των LLM, υπάρχουν αρκετοί περιορισμοί τους οποίους θα πρέπει να λάβουμε υπόψη μας κατά την υλοποίηση μιας εφαρμογής.

3.4.1 Η γνώση τους δεν είναι πραγματικού χρόνου

Τα μοντέλα τύπου GPT εκπαιδεύονται σε δεδομένα που συλλέγονται μέχρι ένα συγκεκριμένο χρονικό διάστημα. Από μόνα τους δεν διαθέτουν πρόσβαση στο διαδίκτυο ή σε ενημερωμένες πηγές πληροφόρησης μετά το πέρας της εκπαίδευσής τους. Για παράδειγμα, το GPT-4o mini έχει εκπαιδευτεί με δεδομένα ως τον Οκτώβριο του 2023 (“GPT-4o mini: advancing cost-efficient intelligence”).

Αυτό σημαίνει ότι τα μοντέλα δεν γνωρίζουν για γεγονότα ή εξελίξεις που συνέβησαν μετά από αυτή την ημερομηνία, όπως:

- Τρέχοντα γεγονότα
- Πρόσφατες επιστημονικές ανακαλύψεις
- Τεχνολογικές καινοτομίες ή νέες κυκλοφορίες προϊόντων

Σημειώνεται ότι η OpenAI εκπαιδεύει συνεχώς ενημερωμένες εκδόσεις των μοντέλων με καινούργια πληροφορία. Αυτά το μοντέλα διατίθενται στο κοινό ανα τακτά χρονικά διαστήματα, αλλά σε καμία περίπτωση η γνώση τους δεν είναι σε πραγματικό χρόνο.

3.4.2 Παραγωγή λανθασμένων πληροφοριών - Παραισθήσεις (Hallucinations)

Από τα παραπάνω μπορούμε να συμπεράνουμε πως για κάποιες διαδικασίες το μοντέλο θα αδυνατεί να απαντήσει σωστά, καθώς δεν θα έχει εκπαιδευτεί με τα κατάλληλα δεδομένα. Με άλλα λόγια, τα μοντέλα ενίοτε παράγουν "παραισθήσεις", δηλαδή απαντήσεις που φαίνονται λογικές αλλά είναι εντελώς λανθασμένες ή φανταστικές. Αυτή η συμπεριφορά μπορεί να προκύψει σε ποικίλες περιπτώσεις, όπως:

- Σε εξειδικευμένα τεχνικά ή επιστημονικά θέματα
- Σε ερωτήματα που δεν έχουν απάντηση στο εκπαιδευτικό σύνολο του μοντέλου

3.4.3 Έλλειψη λογικής και κριτικής σκέψης

Τα μοντέλα δεν είναι πάντα σε θέση να εκτελούν πολύπλοκες εργασίες που απαιτούν συνδυασμό λογικής και ανάλυσης. Για παράδειγμα, μπορεί να αποτύχουν σε προβλήματα που απαιτούν πολλαπλά βήματα λογικής, όπως μαθηματικά προβλήματα ή αλληλουχία γεγονότων που πρέπει να αναλυθούν προσεκτικά.

3.4.4 Περιορισμοί σε μακροχρόνιες συζητήσεις (Context Window)

Τα μοντέλα GPT λειτουργούν εντός ενός πλαισίου συνοχής (context window), το οποίο καθορίζει πόση πληροφορία μπορούν να διατηρούν από την προηγούμενη συνομιλία. Για παράδειγμα, το GPT-4o mini μπορεί να διαχειριστεί μέχρι 128K tokens* εισόδου και μέχρι 16K tokens εξόδου.

Αυτό σημαίνει ότι σε πολύ μακροχρόνιες συνομιλίες, το μοντέλο μπορεί να χάσει κάποιες προηγούμενες πληροφορίες και να μην διατηρήσει τη συνοχή της συζήτησης. Επίσης, το context window μας περιορίζει στο μέγεθος της πληροφορίας που μπορούμε να εισάγουμε στο LLM, αλλά και στο μέγεθος της πληροφορίας που μπορεί αυτό να εξάγει.

* Το κείμενο προκειμένου να επεξεργαστεί από το LLM κωδικοποιείται σε μια μορφή ειδική για το κάθε μοντέλο. Οι μονάδες αυτής της κωδικοποίησης ονομάζονται tokens και μπορούν να αντιστοιχούν σε μία λέξη ή σε μερικούς χαρακτήρες.

3.4.5 Ευαισθησία σε διατύπωση (Prompt Sensitivity)

Τα αποτελέσματα που παράγουν τα μοντέλα μπορεί να εξαρτώνται σε μεγάλο βαθμό από τον τρόπο που διατυπώνεται η ερώτηση ή το αίτημα. Μια μικρή αλλαγή στη διατύπωση μπορεί να οδηγήσει σε διαφορετικές απαντήσεις. Αυτό καθιστά τη χρήση των μοντέλων πιο περίπλοκη σε ορισμένες περιπτώσεις, καθώς οι χρήστες μπορεί να χρειαστεί να πειραματιστούν με τη διατύπωση των ερωτήσεων για να λάβουν το επιθυμητό αποτέλεσμα.

3.4.6 Περιορισμοί κλιμάκωσης και κόστους

Τα μοντέλα GPT, ειδικά τα πιο προηγμένα όπως το GPT-4o, είναι ακριβά τόσο για την ανάπτυξη και την εκπαίδευσή τους όσο και για την χρήση τους (Griffith). Η κλιμάκωση των εφαρμογών με αυτά τα μοντέλα μπορεί να αυξήσει το κόστος χρήσης, ειδικά σε περιπτώσεις που απαιτείται συνεχής ή σε πραγματικό χρόνο αλληλεπίδραση με τους χρήστες.

4. Retrieval-Augmented Generation (RAG)

Για την επίλυση κάποιων από τα παραπάνω προβλήματα έχουν παρουσιαστεί διάφορες τεχνικές. Μια από τις πιο διαδεδομένες είναι η διαδικασία Retrieval-Augmented Generation (RAG). Το RAG βελτιώνει τις απαντήσεις των μοντέλων Generative AI, χρησιμοποιώντας τεχνικές ανάκτησης πληροφοριών. Σκοπός της είναι να ξεπεράσει τους περιορισμούς των παραδοσιακών γλωσσικών γεννητικών μοντέλων, τα οποία βασίζονται αποκλειστικά στη γνώση που περιλαμβάνεται στα δεδομένα εκπαίδευσής τους και δεν μπορούν να έχουν πρόσβαση σε πληροφορίες που δεν υπάρχουν σε αυτά τα δεδομένα. Η βασική ιδέα πίσω από τα RAGs είναι η βελτίωση της ακρίβειας και της συνάφειας των απαντήσεων που παρέχονται από τα LLMs, αξιοποιώντας υπάρχουσες βάσεις δεδομένων ή κείμενα για την υποστήριξη της γενετικής διαδικασίας (Izacard and Grave, 2020).

Με το RAG, όταν τίθεται μια ερώτηση ή μια αίτηση για πληροφορίες, το σύστημα πρώτα ανακτά τις σχετικές πληροφορίες από μια βάση δεδομένων ή από το διαδίκτυο. Αυτές οι πληροφορίες λειτουργούν ως πρόσθετα δεδομένα κατά τη διαδικασία παραγωγής της απάντησης από το LLM. Αυτή η προσέγγιση συνδυάζει την ακρίβεια της ανάκτησης πληροφοριών με τη δημιουργικότητα και την ευελιξία του Generative AI (Lewis et al.).

4.1 Πλεονεκτήματα

Η χρήση των RAGs προσφέρει σημαντικά πλεονεκτήματα στην βελτίωση των απαντήσεων που παράγονται από μοντέλα όπως το GPT:

- **Πρόσβαση σε ενημερωμένες πληροφορίες:** Επειδή παρέχονται στο LLM εξωτερικές πηγές δεδομένων, μπορεί να παρέχει απαντήσεις βασισμένες σε πρόσφατες πληροφορίες, γεγονότα ή γεγονότα που δεν περιλαμβάνονται στο εκπαιδευτικό της σύνολο.

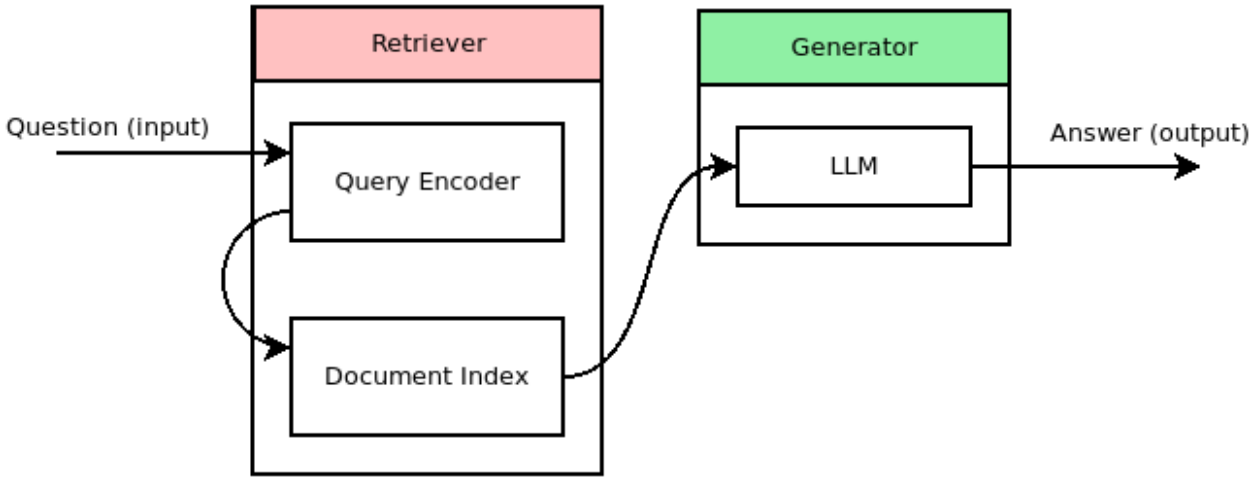
- **Αύξηση ακρίβειας:** Η ανάκτηση πληροφοριών από συγκεκριμένες πηγές (π.χ. επιστημονικές βάσεις δεδομένων) βελτιώνει την ακρίβεια των απαντήσεων, ειδικά σε εξειδικευμένους τομείς όπως η επιστήμη, η ιατρική ή η τεχνολογία και δίνει τη δυνατότητα στον χρήστη να επιβεβαιώσει την ορθότητα της απάντησης.

- **Ευελιξία σε διαφορετικά δεδομένα:** Το RAG μπορεί να συνδεθεί με διάφορες βάσεις δεδομένων και αποθετήρια εγγράφων, επιτρέποντας την προσαρμογή των απαντήσεων ανάλογα με τον τομέα ή το θέμα που ενδιαφέρει τον χρήστη.

- **Μικρότερο κόστος**: Η χρήση RAG έχει μικρότερο κόστος από την επανεκπαίδευση ή το fine-tuning ενός LLM και καθιστά πιο εύκολη την προσθήκη και την αφαίρεση των δεδομένων.

4.2 Αρχιτεκτονική

Το RAG αποτελείται από δύο βασικά μέρη: τον ανακτητή (retriever) και τη γεννήτρια (generator). Ο τρόπος υλοποίησης τους μπορεί να διαφέρει ανάλογα την εφαρμογή. Εσωτερικά η υλοποίησή τους μπορεί να αποτελείται από διαφορετικά υποσυστήματα. Αρχικά θα δούμε την γενική περίπτωση:



Διάγραμμα με τα βασικά μέρη ενός RAG συστήματος

4.2.2 Ανακτητής (Retriever):

Ο ανακτητής έχει ως ρόλο να εντοπίζει σχετικά έγγραφα ή κομμάτια πληροφορίας από μια εξωτερική βάση δεδομένων ή ένα σώμα κειμένων, βασιζόμενος σε μία είσοδο, συνήθως ένα ερώτημα ενός χρήστη. Αυτές οι βάσεις δεδομένων μπορούν να υλοποιηθούν με διαφορετικούς τρόπους ανάλογα τις απαιτήσεις και τα δεδομένα που πρέπει να ανακτηθούν.

Συνήθεις μέθοδοι ανάκτησης περιλαμβάνουν τη Dense Passage Retrieval (DPR) (Karpukhin et al. 2020) και το BM25. Αυτές οι μέθοδοι, ταιριάζουν το ερώτημα με σχετικά έγγραφα υπολογίζοντας ενσωματώσεις (embeddings) ή ομοιότητες. Συνήθης είναι και ο συνδυασμός των δύο, υβριδική (hybrid) προσέγγιση, καθώς σε κάποιες περιπτώσεις, πχ. όταν το αρχικό κείμενο περιέχει ορολογία που δεν περιλαμβάνεται στα δεδομένα του μοντέλου ενσωμάτωσης (embedding model), αυξάνεται σημαντικά η απόδοση του όλου συστήματος.

4.2.3 Γεννήτρια (Generator):

Μετά την ανάκτηση των σχετικών εγγράφων, η γεννήτρια η οποία συνήθως βασίζεται σε LLMs όπως το GPT λαμβάνει αυτά τα έγγραφα ως επιπρόσθετη είσοδο, μαζί με το ερώτημα του χρήστη, και τις κατάλληλες οδηγίες (prompt) για να παράγει μια απάντηση.

Το μοντέλο συνδυάζει τις ανακτημένες πληροφορίες με τη δική του γνώση για να δημιουργήσει μια πιο ενημερωμένη και σχετική απάντηση, η οποία εμφανίζεται στον χρήστη.

4.3 Ευρετηρίαση με embeddings

Ένα μοντέλο ενσωματωσης (embedding model) είναι ένα είδος μοντέλου μηχανικής μάθησης που μετατρέπει λέξεις, φράσεις ή ακόμα και μεγαλύτερα κομμάτια κειμένου σε αριθμητικά διανύσματα, δηλαδή σε αναπαραστάσεις σε έναν πολύ-διάστατο χώρο. Αυτές οι αναπαραστάσεις έχουν ως στόχο να συλλάβουν τη σημασιολογική σχέση μεταξύ των λέξεων ή των κειμένων, έτσι ώστε λέξεις ή φράσεις με παρόμοιο νόημα να βρίσκονται κοντά μεταξύ τους σε αυτόν τον χώρο. Τα μοντέλα ενσωμάτωσης μαθαίνουν να δημιουργούν αριθμητικές αναπαραστάσεις μέσω εκπαίδευσης σε μεγάλα σώματα κειμένων. Κατά την εκπαίδευση, το μοντέλο προσπαθεί να κατανοήσει ποιες λέξεις συχνά εμφανίζονται μαζί και να μάθει τις σημασιολογικές τους σχέσεις. (Mikolov et al. 2013)

4.3.1 Επιλογή μοντέλου

Είναι πολύ σημαντικό να επιλεχθεί το κατάλληλο μοντέλο ενσωμάτωσης έτσι ώστε να έχουμε τα καλύτερα δυνατά αποτελέσματα. Η επιλογή του μοντέλου εξαρτάται από πολλές παραμέτρους όπως:

- Η γλώσσα του κειμένου αναζήτησης
- Η ταχύτητα εκτέλεσης του μοντέλου
- Το μέγεθός του κειμένου
- Το περιεχόμενο αναζήτησης

Κάποια από τα πιο γνωστά μοντέλα που έχουν χρησιμοποιηθεί είναι τα εξής:

- Word2Vec: Είναι ένα από τα πρώτα και πιο γνωστά μοντέλα ενσωματώσεων. Το Word2Vec εκπαιδεύεται με τέτοιο τρόπο ώστε να προβλέπει είτε τις γύρω λέξεις μιας δεδομένης λέξης (συνάρτηση συνεχούς γειτονίας, CBOW) είτε μια λέξη με βάση το περιβάλλον της (skip-gram).
- GloVe: Είναι ένα μοντέλο που μαθαίνει ενσωματώσεις με βάση τη συχνότητα συνύπαρξης λέξεων στο κείμενο. Το GloVe δημιουργεί αναπαραστάσεις που συλλαμβάνουν τις στατιστικές ιδιότητες του κειμένου.
- BERT και άλλες μοντέρνες αναπαραστάσεις: Πιο πρόσφατα μοντέλα όπως το BERT δημιουργούν ενσωματώσεις λέξεων με βάση το πλαίσιο στο οποίο εμφανίζονται. Αυτό σημαίνει ότι η ενσωμάτωση μιας λέξης αλλάζει ανάλογα με τη φράση ή το περιβάλλον της στο κείμενο.

Για την σύγκριση και την επιλογή του κατάλληλου μοντέλου ένα καλό σημείο εκκίνησης είναι ο πίνακας σύγκρισης MTEB https://huggingface.co/spaces/mteb/leaderboard στον οποίο μπορούμε να βρούμε συγκρίσεις και πληροφορίες για έναν μεγάλο αριθμό μοντέλων ενσωμάτωσης. Τα μοντέλα του πίνακα μπορούν να είναι είτε μοντέλα διαθέσιμα για εκτέλεση και περαιτέρω εκπαίδευση, είτε υπηρεσίες επί πληρωμή όπως τα αντίστοιχα μοντέλα της OpenAI. Η σύγκριση γίνεται μεταξύ διαφόρων διαδικασιών όπως είναι η εύρεση παράλληλων προτάσεων μεταξύ δύο γλωσσών, η κατηγοριοποίηση και ομαδοποίηση κειμένου, η αναταξινόμηση αποτελεσμάτων, η ανάκτηση και η περίληψη κειμένου.

Πολλές φορές, αν το περιεχόμενο που επεξεργάζεται η εφαρμογή είναι συγκεκριμένο, μπορούμε να πραγματοποιήσουμε περαιτέρω εκπαίδευση πάνω στα δεδομένα, έτσι ώστε να επιτύχουμε πολύ μεγαλύτερη ακρίβεια από ένα μοντέλο γενικής χρήσης. Η διαδικασία της εκπαίδευσης απαιτεί τη δημιουργία δεδομένων για την εκπαίδευση του μοντέλου και είναι χρονοβόρα και απαιτητική διαδικασία.

4.3.2 Vector databases

Οι βάσεις δεδομένων διανυσμάτων (vector databases) είναι συστήματα τα οποία αναλαμβάνουν την αποθήκευση, τη διαχείριση και την αναζήτηση των embeddings που παράγονται από τα μοντέλα που αναφέραμε στην προηγούμενη ενότητα. Στις βάσεις δεδομένων αυτές, τα δεδομένα αποθηκεύονται ως διανύσματα. Κάθε διάνυσμα αντιπροσωπεύει ένα στοιχείο (όπως ένα έγγραφο, μια εικόνα κλπ.) σε μια υψηλή διάσταση. Οι αλγόριθμοι μπορούν να εκτελέσουν αναζήτηση στα διανύσματα για να βρουν παρόμοια στοιχεία.

Η αναζήτηση χρησιμοποιώντας αυτού του είδους τα συστήματα δεν είναι απαραίτητη, καθώς θα μπορούσε να χρησιμοποιηθεί και ένας απλός αλγόριθμος του Πλησιέστερου Γείτονα (K Nearest Neighbours - KNN), όμως όσο αυξάνεται ο όγκος των δεδομένων τόσο πιο πολύς χρόνος απαιτείται για την εκτέλεση του αλγορίθμου αυτού. Για αυτό το λόγο, όταν ο αριθμός των διανυσμάτων ξεπεράσει, για παράδειγμα, το ένα εκατομμύριο, είναι απαραίτητη η χρήση αλγορίθμων Προσεγγιστικού Πλησιέστερου Γείτονα (Approximate Nearest Neighbour) όπως είναι ο αλγόριθμος Hierarchical Navigable Small World (HNSW), ο οποίος χρησιμοποιείται σε πολλά συστήματα βάσεων δεδομένων διανυσμάτων.

Στο ακολουθο site παρουσιάζονται συγκρίσεις και benchmarks πάνω σε διάφορους αλγορίθμους ANN και συστήματα ή βιβλιοθήκες vector database: https://ann-benchmarks.com/

4.4 Υβριδική αναζήτηση

Εκτός από την αναζήτηση με βάση τα μοντέλα ενσωμάτωσης μπορούμε, επίσης, να χρησιμοποιήσουμε μεθόδους Full-Text αναζήτησης βασισμένες σε αλγορίθμους τύπου Bag of Words (BoW) όπως είναι ο BM25 ή ο TF-IDF.

Αυτού του είδους οι αλγόριθμοι έχουν το πλεονέκτημα ότι βασίζονται μόνο στις λέξεις κλειδιά και αποδίδουν καλύτερα από τα μοντέλα ενσωμάτωσης όταν η αναζήτηση έχει να κάνει με συγκεκριμένη ορολογία και περιεχόμενο, το οποίο μπορεί να μην περιέχεται στα δεδομένα εκπαίδευσης του μοντέλου.

Η χρήση των αλγορίθμων BoW παράλληλα με τα μοντέλα ενσωμάτωσης συνδυάζει τα πλεονεκτήματα των δύο μεθόδων και μπορεί να βοηθήσει στη βελτίωση της απόδοσης του συστήματος ανάκτησης πληροφορίας.

4.5 Τμηματοποίηση κειμένου (text chunking)

Η διάσπαση του κειμένου σε μικρότερα αποσπάσματα αποτελεί ένα σημαντικό βήμα στην προεπεξεργασία των δεδομένων στα συστήματα RAG. Συχνά, όταν χρησιμοποιούνται μεγάλα έγγραφα ως πηγές για την ανάκτηση και την αύξηση της γενετικής ΤΝ, αντιμετωπίζεται το πρόβλημα του περιορισμένου μεγέθους εισόδου. Τόσο τα LLM όσο και τα Embeddings Models έχουν περιορισμούς στο μέγεθος του κειμένου που μπορούν να επεξεργαστούν.

Η πιο συνηθισμένη λύση είναι να χωρίσουμε το έγγραφο σε “τμήματα”, να δημιουργήσουμε μια αναπαράσταση για κάθε τμήμα και να αποθηκεύσουμε όλα τα τμήματα στη βάση δεδομένων διανυσμάτων. Υπάρχουν διάφορες μέθοδοι για να επιτευχθεί αυτή η διάσπαση. Σε αυτή την ενότητα θα παρουσιάσουμε μερικές από αυτές καθώς και τα πλεονεκτήματά και τα μειονεκτήματά τους (Kamradt 2024).

Συνοπτικά, οι βασικοί λόγοι για την διάσπαση του κειμένου σε αποσπάσματα είναι:

- **Το περιορισμένο μέγεθος των διανυσμάτων**: τα διανύσματα έχουν προκαθορισμένο μέγεθος. Αν το έγγραφο είναι πολύ μεγάλο, δεν μπορούμε να το αναπαραστήσουμε με ένα μόνο διάνυσμα.
- **Ο περιορισμός στο μέγεθος εισόδου ενός LLM:** Δεν μπορούμε να εισάγουμε ολόκληρα έγγραφα σε ένα LLM τόσο για λόγους κόστους όσο και για λόγους περιορισμού του συνολικού μεγέθους του κειμένου εισόδου στο μοντέλο.
- **Πιο αποδοτική αναζήτηση**: Κάθε τμήμα αναπαριστά μια συγκεκριμένη πτυχή του κειμένου, έτσι κατά την αναζήτηση μπορούμε να ανακτήσουμε μόνο τα σχετικά αποσπάσματα.

Υπάρχουν διάφορες προσεγγίσεις για το Splitting chunks:

- **Διαίρεση σε παραγράφους ή προτάσεις**: Μπορούμε να χωρίσουμε το κείμενο σε παραγράφους ή ακόμα και σε μικρότερες μονάδες, όπως προτάσεις.
- **Διαίρεση με βάση το περιεχόμενο**: Μπορούμε να επιλέξουμε να χωρίσουμε το κείμενο σε τμήματα που αντιστοιχούν σε θεματικές ενότητες ή σε σημαντικές φράσεις.
- **Διαίρεση με βάση το μέγεθος**: Μπορούμε να ορίσουμε ένα μέγιστο μέγεθος για κάθε τμήμα.

4.5.1 Με βάση τον αριθμό των χαρακτήρων

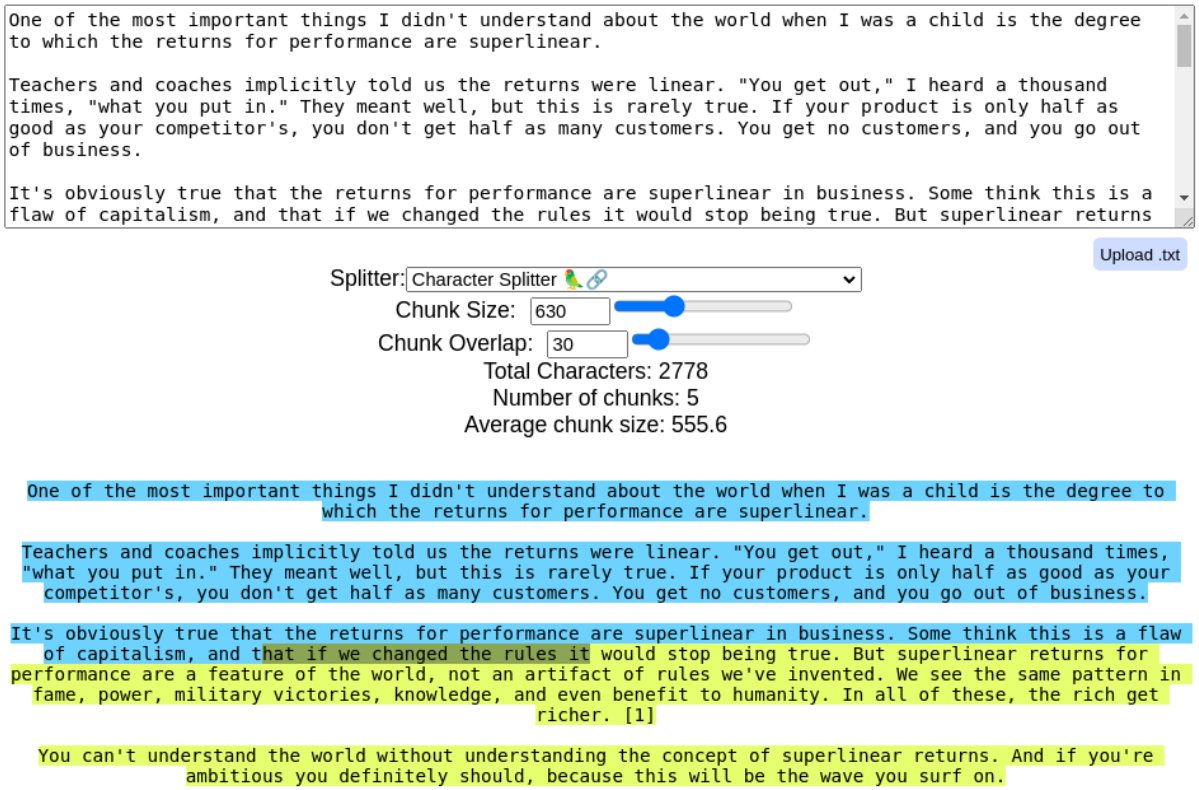
Η διάσπαση με βάση τον αριθμό των χαρακτήρων αποτελεί την πιο απλή μέθοδο τμηματοποίησης κειμένου, κατά την οποία το κείμενο διασπάται σε τμήματα σταθερού μεγέθους με βάση τον αριθμό χαρακτήρων. Η προσέγγιση αυτή είναι απλή, όμως δεν συνιστάται για εφαρμογές που απαιτούν υψηλή ακρίβεια, καθώς αγνοεί τη δομή του κειμένου.

- **Πλεονεκτήματα**: Απλότητα και ευκολία υλοποίησης.
- **Μειονεκτήματα**: Ακαμψία στη διαχείριση της δομής και έλλειψη προσαρμοστικότητας.

Για να ελαχιστοποιησουμε την απώλεια πληροφορίας μεταξύ των αποσπασμάτων μπορούμε να χρησιμοποιήσουμε επικάλυψη μεταξύ των διαδοχικών τμημάτων για τη διατήρηση των

συμφραζομένων. Η βασική μεθοδολογία διάσπασης μπορεί να γίνει χρησιμοποιώντας προτάσεις αντί για χαρακτήρες. Τα παραγόμενα τμήματα, ή έγγραφα (documents), εμπλουτίζονται με μεταδεδομένα για περαιτέρω ανάλυση.

Παρακάτω ακολουθεί ένα οπτικό παράδειγμα αυτού του τρόπου διάσπασης του κειμένου:



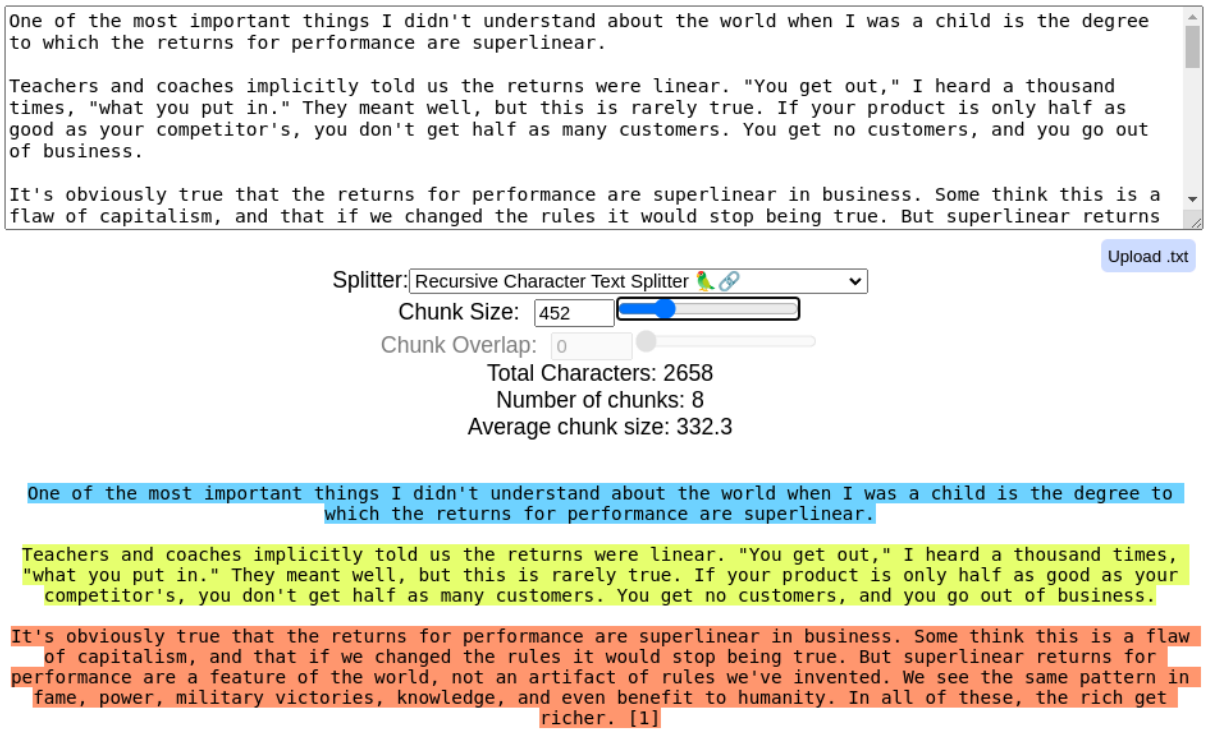
Παράδειγμα από το εργαλείο ChunkViz.com

4.5.2 Με γλωσσικούς χαρακτήρες διαχωρισμού

Η μέθοδος "Recursive Character Text Splitter" (Αναδρομικός Διαχωριστής Κειμένου Χαρακτήρων) αντιμετωπίζει το πρόβλημα της δομής των κειμένων με έναν πιο εξελιγμένο τρόπο, σε σχέση με την απλή διαίρεση βάσει σταθερού αριθμού χαρακτήρων. Χρησιμοποιεί μια σειρά από διαχωριστές (όπως διπλά διαστήματα, νέες γραμμές, απλά διαστήματα ή μεμονωμένους χαρακτήρες) για να διασπάσει τα κείμενα.

Για παράδειγμα, επιλέγοντας ένα μέγιστο αριθμό χαρακτήρων ανα τμήμα, το κείμενο διασπάται σε μικρά κομμάτια που λαμβάνουν υπόψη την τελεία ως πιθανό σημείο τέλους παραγράφου. Αν το κομμάτι παραμένει μεγάλο, τότε η μέθοδος χρησιμοποιεί τον επόμενο διαχωριστή, επαναλαμβάνοντας τη διαδικασία.

Αυτή η μέθοδος είναι ιδανική για τη δημιουργία "κομματιών" κειμένου που διατηρούν τη συνοχή και τη δομή του περιεχομένου, καθιστώντας την κατάλληλη για εφαρμογές όπως η ανάλυση ή η τεχνητή νοημοσύνη.



Παράδειγμα από το εργαλείο ChunkViz.com

4.5.3 Σημασιολογική τμηματοποίηση

Οι προηγούμενες μέθοδοι που αναφέραμε αγνοούν τη σημασιολογική συνοχή του περιεχομένου. Για να αντιμετωπιστεί αυτό το ζήτημα, μπορεί να γίνει μια μέθοδος κατάτμησης βασισμένη σε ενσωματώσεις (embeddings) που αναπαριστούν τη σημασιολογική σημασία των προτάσεων. Με την ανάλυση των αποστάσεων μεταξύ διαδοχικών ενσωματώσεων, εντοπίζονται τα σημεία όπου η σημασιολογική απόσταση είναι μεγαλύτερη και καθορίζονται τα όρια των τμημάτων. Η βασική υπόθεση είναι ότι τα σημασιολογικά συγγενή τμήματα πρέπει να παραμένουν ενωμένα.

Περισσότερες μέθοδοι και πληροφορίες σχετικά με την τμηματοποίηση μπορούν να βρεθούν και στον ακόλουθο σύνδεσμο:
https://github.com/FullStackRetrieval-com/RetrievalTutorials/blob/main/tutorials/LevelsOfTextSplitting/5_Levels_Of_Text_Splitting.ipynb

4.6 Επαναχρησιμοποιήσιμα εργαλεία και βιβλιοθήκες

Για την δημιουργία ενός RAG μπορούμε να χρησιμοποιήσουμε διάφορες ολοκληρωμένες βιβλιοθήκες, οι οποίες μας παρέχουν όλα τα μέρη ενός τέτοιου συστήματος ή και να δημιουργήσουμε το δικό μας χρησιμοποιώντας δικό μας κώδικα και τις βιβλιοθήκες που

χρειαζόμαστε. Η επιλογή του τρόπου υλοποίησης εξαρτάται από τις ανάγκες παραμεροποίησης της εκάστοτε εφαρμογής. Παρακάτω αναφέρονται κάποιες από τις πιο γνωστές βιβλιοθήκες για την εύκολη υλοποίηση συστημάτων ΤΝ:

4.6.1 Ολοκληρωμένες βιβλιοθήκες για γενικές εφαρμογές ΤΝ

- https://www.langchain.com/
- https://www.llamaindex.ai/
- https://haystack.deepset.ai/
- https://embedchain.ai/

4.6.2 RAG με συσχέτιση της σημασιολογίας των εγγράφων με γράφους

- https://microsoft.github.io/graphrag/

5 Αυτόματοι Ψηφιακοί Βοηθοί στον Δημόσιο Τομέα

Τα τελευταία χρόνια οι αυτόματοι ψηφιακοί βοηθοί χρησιμοποιούνται όλο και περισσότερο στον δημόσιο τομέα. Σύμφωνα με τους Ανδρουτσοπούλου, Καρακαπιλίδης, Λουκής και Χαραλαμπίδης (Androutsopoulou et al. 2019), τα chatbots βελτιώνουν την εξυπηρέτηση των πολιτών, προσφέροντας άμεσες και αυτοματοποιημένες απαντήσεις σε ερωτήματα και αιτήματα. Αυτή η αυτοματοποίηση απελευθερώνει τους υπαλλήλους από επαναλαμβανόμενες εργασίες, επιτρέποντάς τους να επικεντρωθούν σε πιο σύνθετα ζητήματα.

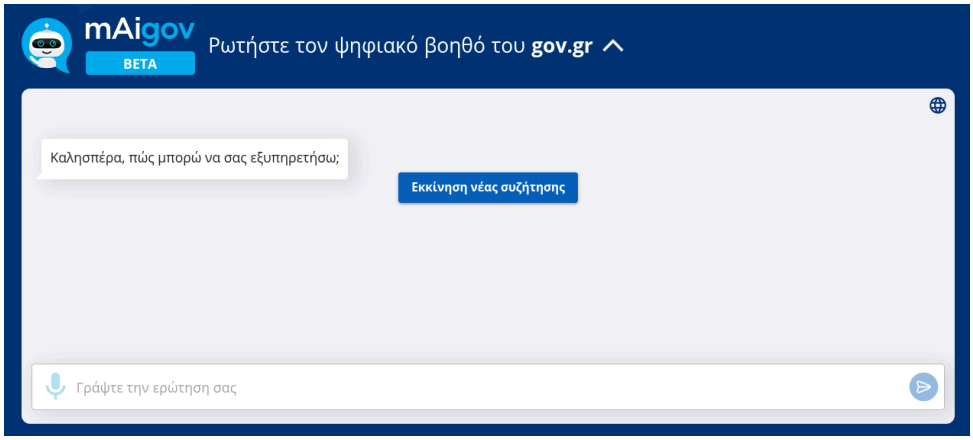
5.1 Πώς τα Chatbots Βελτιώνουν τη Δημόσια Διοίκηση:

- **Εξοικονόμηση χρόνου και πόρων:** Η αυτοματοποίηση διαδικασιών, όπως η υποβολή αιτήσεων ή η έκδοση πιστοποιητικών, μειώνει σημαντικά το χρόνο και το κόστος για τους πολίτες και τη διοίκηση.
- **Ενισχυμένη προσβασιμότητα:** Τα chatbots καθιστούν τις δημόσιες υπηρεσίες πιο προσβάσιμες σε όλους, ανεξαρτήτως τοποθεσίας ή φυσικών περιορισμών. Άτομα με αναπηρίες ή κάτοικοι απομακρυσμένων περιοχών μπορούν να εξυπηρετηθούν εύκολα μέσω internet.
- **Αποτελεσματική διαχείριση κρίσεων:** Σε έκτακτες καταστάσεις, όπως φυσικές καταστροφές ή πανδημίες, τα chatbots μπορούν να παρέχουν έγκαιρη και αξιόπιστη ενημέρωση στο κοινό.
- **Συνεχής βελτίωση υπηρεσιών:** Μέσω της συλλογής δεδομένων και ανατροφοδότησης, τα chatbots βοηθούν στην κατανόηση των αναγκών των πολιτών και τη βελτίωση των υπηρεσιών.

5.2 Παραδείγματα εφαρμογής chatbots στην Ελλάδα:

5.2.1 mAIgov

Το mAIgov είναι ένα σύστημα ΤΝ για την πληροφόρηση των πολιτών. Στόχος της εφαρμογής είναι η βελτίωση της ποιότητας, της ταχύτητας και της προσβασιμότητας των δημόσιων υπηρεσιών που παρέχονται στους πολίτες (“Πολιτική χρήσης mAIgov”, n.d.).

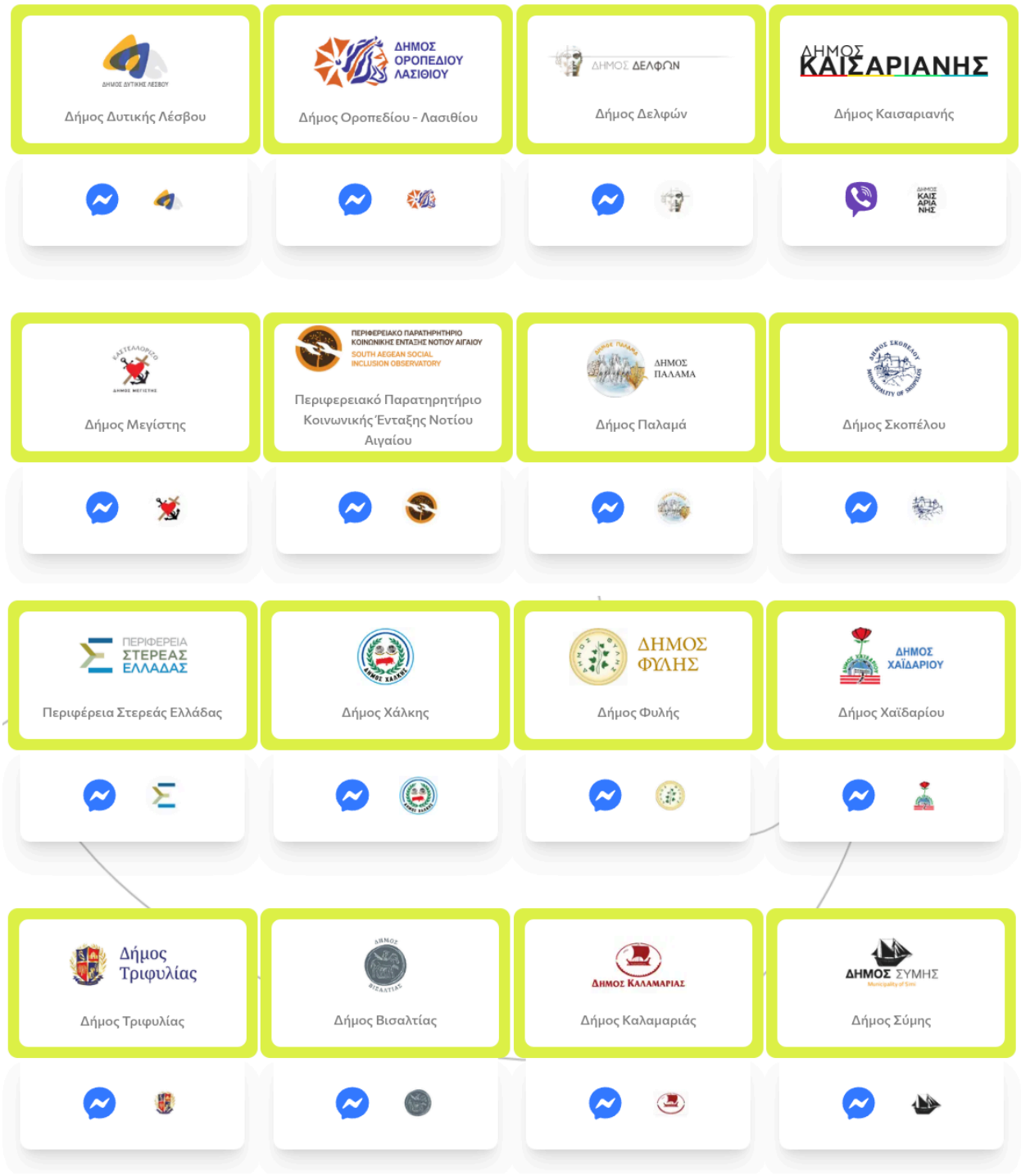


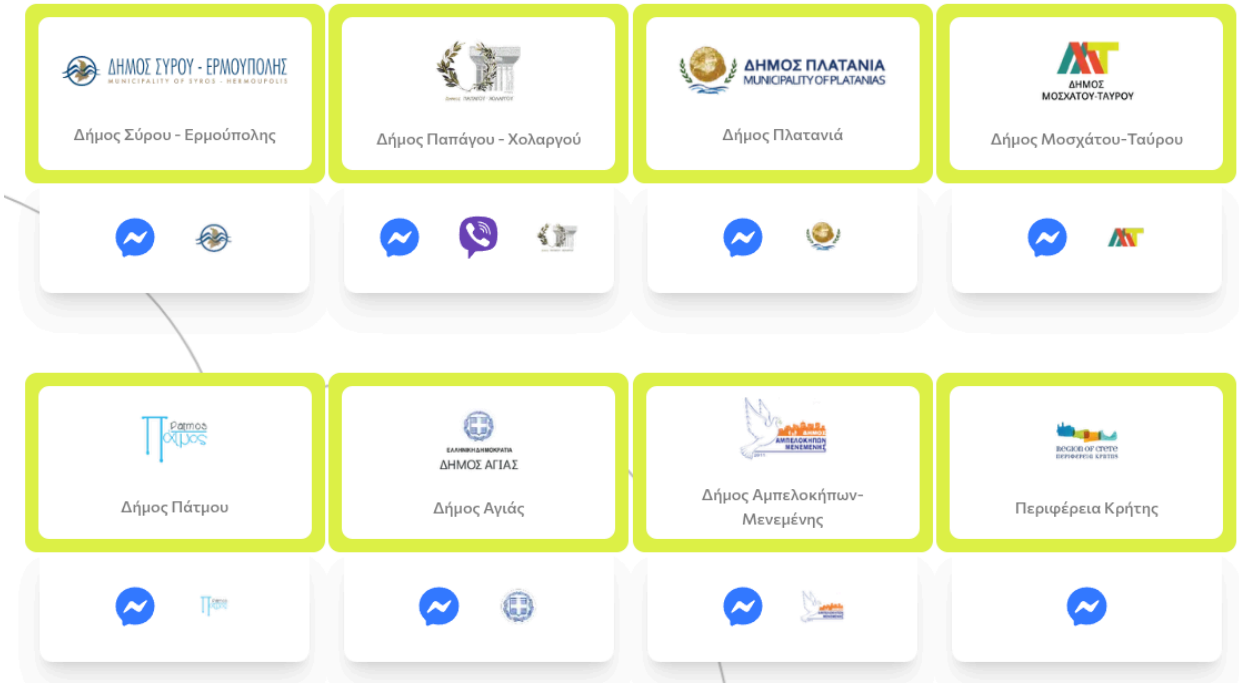
Ψηφιακός βοηθός mAIgov

Ο ψηφιακός βοηθός βασίζεται σε τεχνολογίες Microsoft Azure OpenAI και έχει εκπαιδευτεί κατάλληλα σε ανοικτά δεδομένα που είναι διαθέσιμα μέσω του gov.gr και άλλων ιστοσελίδων δημοσίων φορέων. Ο ψηφιακός βοηθός βρίσκεται σε δοκιμαστική (beta) έκδοση και εκπαιδεύεται διαρκώς για να βελτιωθεί η λειτουργία του.
Υπεύθυνο για τη λειτουργία του ψηφιακού βοηθού είναι το Υπουργείο Ψηφιακής Διακυβέρνησης.

5.2.2 Botakis

Το Botakis είναι μια ελληνική πλατφόρμα που αναπτύχθηκε, με στόχο τη δημιουργία έξυπνων ψηφιακών βοηθών για την ενίσχυση της επικοινωνίας μεταξύ πολιτών και δημόσιων ή ιδιωτικών οργανισμών. Ο ψηφιακός βοηθός αυτός έχει υλοποιηθεί σε περισσότερους από 50 φορείς του ιδιωτικού και δημόσιου τομέα (“Πελάτες & Συνεργάτες”, n.d.).

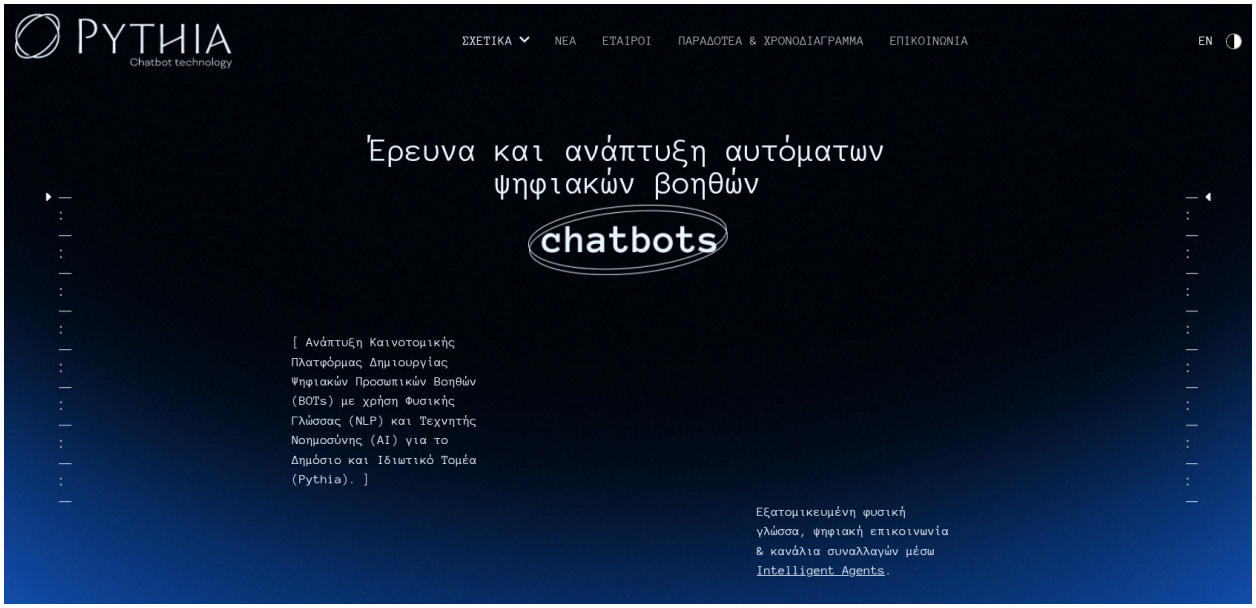

Δήμος Δυτικής Λέσβου
Δήμος Οροπεδίου - Λασιθίου
Δήμος Δελφών
Δήμος Καισαριανής
Δήμος Μεγίστης
Περιφερειακό Παρατηρητήριο Κοινωνικής Ένταξης Νοτίου Αιγαίου
Δήμος Παλαμά
Δήμος Σκοπέλου
Περιφέρεια Στερεάς Ελλάδας
Δήμος Χάλκης
Δήμος Φυλής
Δήμος Χαϊδαρίου
Δήμος Τριφυλίας
Δήμος Βισαλτίας
Δήμος Καλαμαριάς
Δήμος Σύμης



Δημόσιοι φορείς που χρησιμοποιούν τον ψηφιακό βοηθό Botakis

5.3 Το Ερευνητικό έργο ΠΥΘΙΑ (Pythia)

Το έργο ΠΥΘΙΑ αποτελεί ένα καινοτόμο εγχείρημα που συνδυάζει προηγμένες τεχνολογίες Τεχνητής Νοημοσύνης (AI) και Επεξεργασίας Φυσικής Γλώσσας (NLP) για τη δημιουργία εξειδικευμένων chatbots, τα οποία ανταποκρίνονται στις ιδιαίτερες απαιτήσεις και προκλήσεις της ελληνικής γλώσσας. Το έργο συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση και εθνικούς πόρους της Ελλάδας μέσω του Επιχειρησιακού Προγράμματος «Ανταγωνιστικότητα, Επιχειρηματικότητα και Καινοτομία», με κύριο στόχο τη σημαντική βελτίωση της λειτουργικότητας και της ποιότητας επικοινωνίας των chatbots (Bouras et al., n.d.).



https://thepythiaproject.com/

Οι παραδοσιακοί τρόποι εξυπηρέτησης πελατών, όπως τα φυσικά καταστήματα και τα τηλεφωνικά κέντρα, αν και εξακολουθούν να είναι σημαντικοί, συχνά αντιμετωπίζουν περιορισμούς σε ό,τι αφορά την ταχύτητα απόκρισης, την κάλυψη των αναγκών των πελατών 24/7 και την παροχή εξατομικευμένων υπηρεσιών.

Σε αυτό το πλαίσιο, το έργο Pythia προτείνει μια καινοτόμο προσέγγιση, αξιοποιώντας την τεχνολογία των chatbots για να αναβαθμίσει την εμπειρία των πελατών. Στόχος του έργου είναι η ανάπτυξη μιας νέας γενιάς ηλεκτρονικών καναλιών επικοινωνίας που λειτουργούν σε φυσική γλώσσα και παρέχουν εξατομικευμένες υπηρεσίες, προσβάσιμες από όλους ανεξαρτήτως ψηφιακών δεξιοτήτων.

5.3.1 Το Πρόβλημα που Επιχειρεί να Λύσει το Pythia

Οι παραδοσιακοί τρόποι εξυπηρέτησης πελατών αντιμετωπίζουν τα ακόλουθα προβλήματα:

- Χρόνος αναμονής: Οι πελάτες συχνά αναγκάζονται να περιμένουν αρκετή ώρα για να εξυπηρετηθούν, είτε σε ένα κατάστημα είτε σε ένα τηλεφωνικό κέντρο.
- Περιορισμένες ώρες λειτουργίας: Τα καταστήματα και τα τηλεφωνικά κέντρα λειτουργούν συνήθως κατά συγκεκριμένες ώρες, ενώ οι ανάγκες των πελατών μπορεί να προκύπτουν ανά πάσα στιγμή.
- Δυσκολία στην εύρεση πληροφοριών: Οι ιστοσελίδες των επιχειρήσεων συχνά είναι πολύπλοκες και δυσνόητες, καθιστώντας δύσκολη την εύρεση των απαιτούμενων πληροφοριών.
- Έλλειψη εξατομίκευσης: Οι παραδοσιακές μέθοδοι εξυπηρέτησης δεν επιτρέπουν την παροχή εξατομικευμένων υπηρεσιών, καθώς δεν λαμβάνουν υπόψη τις συγκεκριμένες ανάγκες και προτιμήσεις κάθε πελάτη.

5.3.2 Η Τεχνολογική Υποδομή του Pythia

Για την επίτευξη των στόχων του, το έργο Pythia αξιοποιεί μια σειρά από προηγμένες τεχνολογίες, όπως:

- Τεχνολογία chatbot: Τα chatbots είναι προγράμματα λογισμικού που μπορούν να συνομιλούν με ανθρώπους σε φυσική γλώσσα.
- Επεξεργασία φυσικής γλώσσας (NLP): Η NLP επιτρέπει στα συστήματα να κατανοούν και να επεξεργάζονται την ανθρώπινη γλώσσα, επιτρέποντας στα chatbots να διεξάγουν πιο φυσικές και εμπεριστατωμένες συνομιλίες.
- Μηχανική μάθηση: Η μηχανική μάθηση επιτρέπει στα συστήματα να μαθαίνουν από τα δεδομένα και να βελτιώνουν συνεχώς τις επιδόσεις τους.
- Ανάλυση δεδομένων: Η ανάλυση δεδομένων επιτρέπει στα συστήματα να εξάγουν χρήσιμες πληροφορίες από τα δεδομένα των πελατών, οι οποίες μπορούν να χρησιμοποιηθούν για την παροχή εξατομικευμένων υπηρεσιών.

5.3.3 Οι Εφαρμογές του Pythia

Το έργο Pythia έχει πολλές πιθανές εφαρμογές σε διάφορους τομείς, όπως:

- Επιχειρηματικές συναλλαγές: Τα chatbots μπορούν να χρησιμοποιηθούν για την απάντηση σε ερωτήσεις σχετικά με προϊόντα και υπηρεσίες, την υποστήριξη πελατών και την ολοκλήρωση συναλλαγών.
- Τραπεζικές συναλλαγές: Τα chatbots μπορούν να χρησιμοποιηθούν για την παροχή πληροφοριών σχετικά με λογαριασμούς, την εκτέλεση συναλλαγών και την επίλυση προβλημάτων.
- Τουριστικές υπηρεσίες: Τα chatbots μπορούν να χρησιμοποιηθούν για την παροχή πληροφοριών σχετικά με προορισμούς, την κράτηση ξενοδοχείων και την οργάνωση ταξιδιών.
- Δημόσιες υπηρεσίες: Τα chatbots μπορούν να χρησιμοποιηθούν για την παροχή πληροφοριών σχετικά με υπηρεσίες του δημόσιου τομέα, την υποβολή αιτήσεων και την επίλυση προβλημάτων.

Εταιρίες και ιδρύματα που συμμετείχαν στο projects και εφάρμοσαν ψηφιακούς βοηθούς στου παραπάνω τομείς

5.3.4 Τα Οφέλη του Pythia

Το έργο Pythia αναμένεται να προσφέρει τα ακόλουθα οφέλη (“ΣΧΕΤΙΚΑ ΜΕ ΤΟ ΕΡΓΟ”, n.d.):

- Βελτίωση της εμπειρίας των πελατών: Οι πελάτες θα μπορούν να λάβουν γρήγορες και ακριβείς απαντήσεις στις ερωτήσεις τους, 24 ώρες το 24ωρο, 7 ημέρες την εβδομάδα.
- Αύξηση της ικανοποίησης των πελατών: Η παροχή εξατομικευμένων υπηρεσιών θα αυξήσει την ικανοποίηση των πελατών και θα ενισχύσει την εμπιστοσύνη τους στην επιχείρηση.
- Μείωση του κόστους λειτουργίας: Η αυτοματοποίηση επαναλαμβανόμενων εργασιών θα μειώσει το κόστος λειτουργίας των επιχειρήσεων.
- Ενίσχυση της ανταγωνιστικότητας: Οι επιχειρήσεις που θα υιοθετήσουν το έργο Pythia θα αποκτήσουν ένα σημαντικό ανταγωνιστικό πλεονέκτημα.

5.3.5 Χρήσιμα συμπεράσματα του έργου ΠΥΘΙΑ για το πλαίσιο της παρούσας εργασίας

Με βάση παρουσιάσεις του έργου, η χρήση των chatbots στην ελληνική γλώσσα είναι κυρίως περιορισμένη σε δευτερεύουσες εργασίες (Αλεξόπουλος, n.d.):

- Η αποτελεσματικότητα τους αμφισβητείται από πολλούς
- Η πολυπλοκότητα της ελληνικής γλώσσας και των πολυάριθμων διαλέκτων της αποτελεί σημαντική πρόκληση για τα chatbots

- Η έλλειψη τυποποίησης της ελληνικής γλώσσας δυσκολεύει τη δημιουργία αλγορίθμων σύνταξής της
- Υπάρχει περιορισμένη διαθεσιμότητα δεδομένων εκπαίδευσης με αποτέλεσμα «φτωχά» μοντέλα
- Ο μικρός πληθυσμός της Ελλάδας και η μη υπομονετική προσέγγιση των πολιτών σε νέες τεχνολογίες δεν επιτρέπει την ταχεία εξάπλωση τους
- Η οικονομική κρίση των προηγουμένων ετών και η συντηρητική νοοτροπία των εταιρειών αποτρέπει τις επενδύσεις σε μη δοκιμασμένες λύσεις
- Πρόσφατα όμως παρατηρούμε μία μικρή αλλά σταθερή υιοθέτηση τους ως βασικό μέσο επικοινωνίας από τις πολύ μεγάλες επιχειρήσεις (τράπεζες, πάροχοι τηλεπικοινωνιών κτλ).

Επίσης οι βασικές κατηγορίες των Chatbot που αποτυπώνονται από ερευνητικό περιεχόμενο του έργου είναι οι εξής:

- Menu based: Τα συστήματα που βασίζονται σε μενού για να δώσουν απαντήσεις
- Keyword based: Χρησιμοποιεί προκαθορισμένες λέξεις-κλειδιά για να ενεργοποιήσει συγκεκριμένες απαντήσεις.
- Linguistic based: Χρησιμοποιεί τεχνικές επεξεργασίας φυσικής γλώσσας (NLP) για την κατανόηση και την απόκριση.
- Voice based: Τα συστήματα που βασίζονται σε φωνή.
- Machine learning: Χρησιμοποιεί αλγόριθμους AI και ML για την κατανόηση και την απόκριση στις ερωτήσεις
- Hybrid model: Χρησιμοποιεί έναν συνδυασμό generative & retrieval μοντέλων για να παρέχει απαντήσεις.

Σημειώνεται ότι με βάση τις τεχνολογικές εξελίξεις, αξιοποιούνται νέες τάσεις Generative AI ως μέρος της πιλοτικής υλοποίησης για τον ψηφιακό διαδικασιών και προμηθειών οργανισμών ΟΤΑ.

Παρακάτω ακολουθούν πίνακες με chatbot ιδιωτικών και δημόσιων φορέων στην Ελλάδα:

Chatbots στον Δημόσιο Τομέα

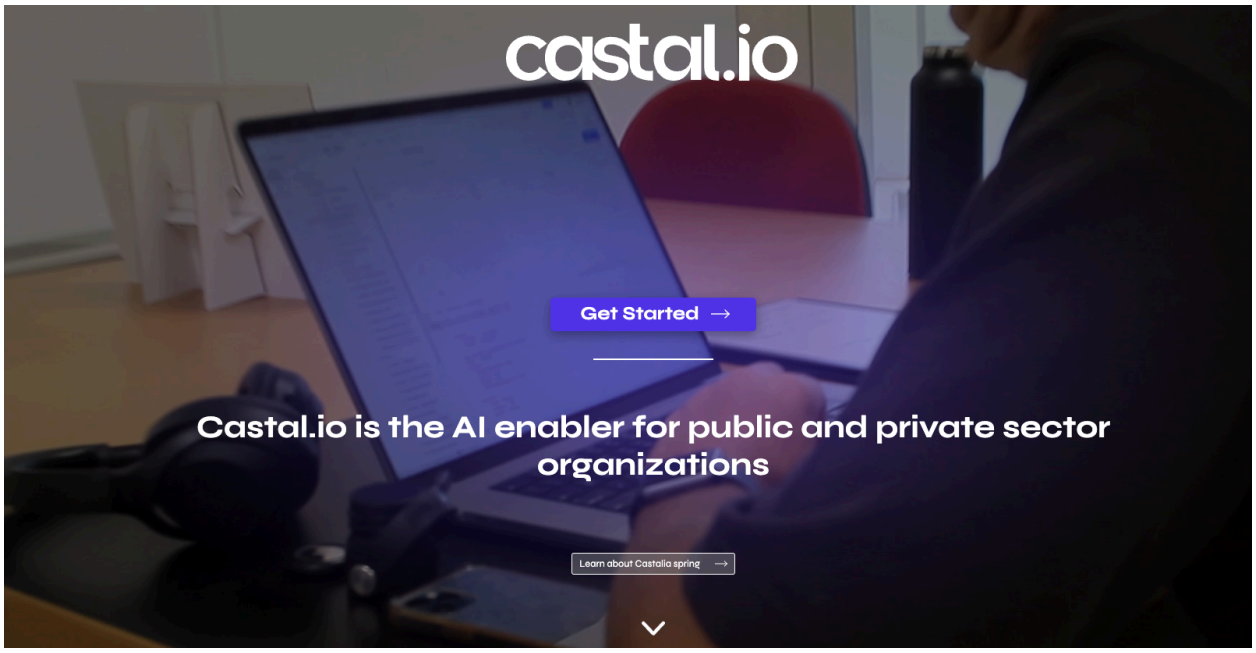
Εταιρεία / Οργανισμός	Κατηγορία λύσης	Application Domain	Κλάδος οικονομίας
ReBrain Greece	Κείμενο	Public	Κρατικές οντότητες
ΟΑΣΑ	Κείμενο	Public	Κοινής ωφέλειας
ΔΕΔΔΗΕ	Κείμενο	Public	Κοινής ωφέλειας
ΔΕΗ	Κείμενο	Public	Κοινής ωφέλειας
ΕΥΔΑΠ	Κείμενο	Public	Κοινής ωφέλειας
ΕΟΠΥΥ	Κείμενο	Public	Κρατικές οντότητες
Δημόσια Υπηρεσία Απασχόλησης	Κείμενο	Public	Κρατικές οντότητες
Περιφέρεια Αττικής	Κείμενο	Public	Κρατικές οντότητες
Περιφέρεια Στερεάς Ελλάδας	Κείμενο	Public	Κρατικές οντότητες
Δήμος Παπαγου-Χολαργού	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Καλαμαριάς	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Πάτμου	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Μοσχάτου-Ταύρου	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Φυλής	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Καστελλόριζου	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Δυτικής Λέσβου	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Πλατανιά	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Αγιάς	Κείμενο	Public	Τοπική αυτοδιοίκηση
Δήμος Βισαλτίας	Κείμενο	Public	Τοπική αυτοδιοίκηση
Τμήμα Ηλεκτρολόγων Μηχανικών - Πανεπιστήμιο Πελοποννήσου	Κείμενο	Public	Εκπαίδευση

Chatbots στον Ιδιωτικό Τομέα

Εταιρεία / Οργανισμός	Κατηγορία λύσης	Application Domain	Κλάδος οικονομίας
2103288000 - Πειραιώς	Φωνή	Private	Χρηματοπιστωτικός
Winbank - Πειραιώς	Κείμενο	Private	Χρηματοπιστωτικός
Vodafone.gr - tobi	Κείμενο	Private	Τηλεπικοινωνίες
13888 - Cosmote	Φωνή	Private	Τηλεπικοινωνίες
Alpha Bank	Φωνή	Private	Χρηματοπιστωτικός
Eurobank	Φωνή	Private	Χρηματοπιστωτικός
Εθνική Τράπεζα	Φωνή	Private	Χρηματοπιστωτικός
Τράπεζα Αττικής	Κείμενο	Private	Χρηματοπιστωτικός
Ελληνική Αναπτυξιακή Τράπεζα	Κείμενο	Private	Χρηματοπιστωτικός
leroymerlin	Κείμενο	Private	Εμπορικός
ikea	Κείμενο	Private	Εμπορικός
eco-mat	Κείμενο	Private	Εμπορικός
pennie.gr	Κείμενο	Private	Εμπορικός
ledison.gr	Κείμενο	Private	Εμπορικός
xtr.gr	Κείμενο	Private	Εμπορικός
acs	Κείμενο	Private	Εμπορικός
coca-cola	Κείμενο	Private	Εμπορικός
goldmall	Κείμενο	Private	Εμπορικός
Market4you	Κείμενο	Private	Εμπορικός

Chatbots στον δημόσια και τον ιδιωτικό τομέα

5.4 Η εφαρμογή castal.io



Η πλατφόρμα castal.io είναι μια προηγμένη πλατφόρμα τεχνητής νοημοσύνης (AI) που αναπτύχθηκε από την Crowdpolicy και παρέχει εξειδικευμένους εσωτερικούς ψηφιακούς βοηθούς. Στόχος της είναι η παροχή αξιόπιστων απαντήσεων σε ζητήματα που αφορούν:

- Εσωτερικές διαδικασίες,
- Νομοθεσία,
- Επιχειρηματικές γνώσεις.

Η πλατφόρμα απευθύνεται σε στελέχη τόσο του δημόσιου τομέα όσο και επιχειρήσεων, προσφέροντας ένα ευρύ φάσμα υπηρεσιών τεχνητής νοημοσύνης για τη βελτίωση της παραγωγικότητας και της λήψης αποφάσεων.

Το castal.io ενσωματώνει προηγμένες AI τεχνολογίες και αλγορίθμους, παρέχοντας υπηρεσίες όπως:

- Σύνθεση κειμένων,
- Περίληψη και ανάλυση εγγράφων,
- Ταξινόμηση πληροφοριών,
- Αυτοματοποίηση διαδικασιών.

Οι λειτουργίες αυτές υλοποιούνται μέσω μιας ενιαίας πλατφόρμας AI, η οποία συνδυάζει προεκπαιδευμένα μοντέλα μεγάλων γλωσσικών μοντέλων (LLMs) και τεχνολογίες από κορυφαίους οργανισμούς.

Η πλατφόρμα castal.io προσφέρει πολλαπλά επίπεδα ενοποίησης με εσωτερικά συστήματα οργανισμών, επιτρέποντας τη δυναμική αξιοποίηση υφιστάμενων δεδομένων. Συγκεκριμένα, η ενσωμάτωση πραγματοποιείται μέσω:

1. Ανάκτησης πληροφοριών από αναφορές, αρχεία και μεγάλες βάσεις δεδομένων SQL.
2. Δημιουργίας απαντήσεων με AI, βασισμένων σε έγγραφα και βάσεις γνώσεων.
3. Χρήσης LLMs, όπως το GPT, για την υποστήριξη καθημερινών εργασιών και την ενίσχυση της αποδοτικότητας.

5.4.1 Το chatbot Καλλικράτης

Ο Καλλικράτης αποτελεί έναν εσωτερικό ψηφιακό βοηθό της πλατφόρμας castal.io, σχεδιασμένο για τη βελτίωση και απλοποίηση των διαδικασιών σε μεγάλους οργανισμούς. Υποστηρίζει τη λειτουργία διοικητικών υπηρεσιών σε πολλαπλές γλώσσες, καθιστώντας τον ένα πολύτιμο εργαλείο για τα στελέχη κάθε οργανισμού.

Ο Καλλικράτης παρέχει απαντήσεις και υποστήριξη σε θέματα όπως:

- Αυτοματοποίηση εσωτερικών διαδικασιών στον δημόσιο και ιδιωτικό τομέα.
- Διαχείριση προμηθειών, προσλήψεων, συμβάσεων και εσωτερικών κανονισμών.

Ήδη, πολλοί οργανισμοί και επιχειρήσεις έχουν υιοθετήσει την πλατφόρμα castal.io και τον ψηφιακό βοηθό Καλλικράτη, αξιοποιώντας τις δυνατότητες της AI για τη βελτίωση της αποδοτικότητας και της λήψης αποφάσεων.

6 Υλοποίηση προσαρμοσμένου chatbot

Στην παρούσα ενότητα παρουσιάζεται η διαδικασία υλοποίησης ενός προσαρμοσμένου chatbot με χρήση της τεχνικής Retrieval-Augmented Generation (RAG) και του GPT 4o μοντέλου. Το chatbot αυτό έχει σχεδιαστεί για εσωτερική χρήση από δήμους, παρέχοντας άμεση και αξιόπιστη πρόσβαση σε πληροφορίες που σχετίζονται με Φύλλα Εφημερίδας της Κυβερνήσεως (ΦΕΚ), νόμους, κανονισμούς και άλλες διοικητικές διαδικασίες.

Η ανάπτυξη του συστήματος επικεντρώνεται στην αποτελεσματική ανάκτηση και κατανόηση εγγράφων, διασφαλίζοντας ότι οι χρήστες μπορούν να λαμβάνουν ακριβείς πληροφορίες. Μέσω της ενσωμάτωσης της RAG αρχιτεκτονικής, το chatbot συνδυάζει τη δύναμη των μεγάλων γλωσσικών μοντέλων με εξειδικευμένες βάσεις δεδομένων, εξασφαλίζοντας έτσι απαντήσεις που βασίζονται σε έγκυρες και τεκμηριωμένες πηγές.

Στα επόμενα υποκεφάλαια, θα αναλυθούν οι τεχνικές προδιαγραφές, η αρχιτεκτονική της λύσης, καθώς και οι μέθοδοι ενσωμάτωσης και εκπαίδευσης του chatbot, προκειμένου να επιτευχθεί η βέλτιστη απόδοση και χρηστικότητα στον δημόσιο τομέα.

6.1 Πηγές Περιεχομένου Νομοθεσίας

Ο ψηφιακός βοηθός αντλεί περιεχόμενο από μια συλλογή νομικών και διοικητικών κειμένων που περιλαμβάνει νομοθετήματα, υπουργικές αποφάσεις και εγκυκλίους που καλύπτουν ένα ευρύ φάσμα θεμάτων της ελληνικής δημόσιας διοίκησης, του περιβαλλοντικού δικαίου, των δημόσιων συμβάσεων, της πολεοδομίας, της τοπικής αυτοδιοίκησης και άλλων κρίσιμων τομέων.

Συγκεκριμένα, περιλαμβάνονται:

- **Νόμοι** που έχουν δημοσιευθεί στην Εφημερίδα της Κυβερνήσεως (ΦΕΚ) και θεσπίζουν διατάξεις για τη δημόσια διοίκηση, το πολεοδομικό και περιβαλλοντικό δίκαιο, τις εκλογικές διαδικασίες, τη διαχείριση φυσικών πόρων, καθώς και ζητήματα οικονομικής και αναπτυξιακής πολιτικής.
- **Υπουργικές Αποφάσεις (Υ.Α.)** που εξειδικεύουν την εφαρμογή των νόμων, παρέχοντας οδηγίες και κατευθύνσεις για την υλοποίησή τους.
- **Εγκύκλιοι** που εκδίδονται από αρμόδιους φορείς και αποσαφηνίζουν ή καθοδηγούν την εφαρμογή της νομοθεσίας στην πράξη.

Οι διατάξεις αυτές συνθέτουν το νομοθετικό και κανονιστικό πλαίσιο που διέπει τη λειτουργία του κράτους, των δήμων και των περιφερειών, καθώς και τη σχέση τους με τους πολίτες και τις επιχειρήσεις.

Η παρούσα συλλογή αποσκοπεί στην παροχή μιας ολοκληρωμένης βάσης αναφοράς για νομικούς, στελέχη δημόσιας διοίκησης, φορείς τοπικής αυτοδιοίκησης και κάθε ενδιαφερόμενο πολίτη ή επαγγελματία που επιθυμεί να κατανοήσει ή να εφαρμόσει τις σχετικές διατάξεις.

Η πλήρης λίστα με τα αρχεία που περιέχονται ακολουθεί παρακάτω:

- N. 4610 2019, (ΦΕΚ 70Α7.5.2019).pdf
- Ν.4014 2011 (ΦΕΚ 209 Α 21-9-2011).pdf
- Ν.4062 2012 (ΦΕΚ 70 Α 30-3-2012).pdf
- Ν.4164 2013 (ΦΕΚ – 156 Α 9-7-2013).pdf
- Ν.4495 2017 (ΦΕΚ 167 Α 3-11-2017).pdf
- Ν.4585 2018 (ΦΕΚ 216 Α 24-12-2018).pdf
- Ν.4685 2020 (ΦΕΚ 92 Α 7-5-2020).pdf
- Ν.4986 2022 (ΦΕΚ 204 Α 28-10-2022).pdf
- fek_a_143_2014.pdf
- fek_a_167_2013.pdf
- fek_a_184_2020.pdf
- fek_a_222_2012.pdf
- fek_a_240_2016.pdf
- fek_a_36_2021.pdf
- ΑΙΤΙΟΛΟΓΙΚΗ ΕΚΘΕΣΗ Ν.4042-2012.pdf
- ΕΑΑΔΗΣΥ-Ν.4412-Υπερκείμενο.html
- ΕΓΚΥΚΛΙΟΣ 108588- ΟΡΓΑΝΩΣΗ Τ.Α.pdf
- ΕΓΚΥΚΛΙΟΣ 17088-ΕΚΛΟΓΕΣ.pdf
- ΕΓΚΥΚΛΙΟΣ 30971-ΟΡΓΑΝΩΣΗ Ο.ΤΑ.pdf
- ΕΓΚΥΚΛΙΟΣ 31709-ΕΚΛΟΓΕΣ.pdf
- ΕΓΚΥΚΛΙΟΣ 33156-ΕΚΛΟΓΕΣ.pdf
- ΕΓΚΥΚΛΙΟΣ 39878- ΕΚΛΟΓΕΣ.pdf
- ΕΓΚΥΚΛΙΟΣ 4835 -ΟΡΓΑΝΩΣΗ ΔΙΟΙΚΗΣΗΣ.pdf
- ΕΓΚΥΚΛΙΟΣ 8050 - ΟΡΓΑΝΩΣΗ Τ.Α.pdf
- ΕΓΚΥΚΛΙΟΣ 8182- ΟΡΓΑΝΩΣΗ Τ.Α.pdf
- Ν 4964 2022 (ΦΕΚ 150Α` 30.7.2022).pdf
- Ν 5037 2023 (ΦΕΚ 58Α 28 3 2023).pdf
- Ν. 16501986 (ΦΕΚ 160Α`16.10.1986) ΣΥΓΚΕΝΤΡΩΤΙΚΟ ΑΡΧΕΙΟ.docx
- Ν. 2947 2001, (ΦΕΚ 228 Α9.10.2001).pdf
- Ν. 3010 2002, (ΦΕΚ 91Α 25.4.2002).pdf
- Ν. 3164 2003, (ΦΕΚ 176Α 2.7.2003).pdf
- Ν. 3536 2007, (ΦΕΚ 42Α 23.2.2007).pdf
- Ν. 4014 2011, (ΦΕΚ 209Α 21.9.2011).pdf
- Ν. 40142011 (ΦΕΚ 209Α` 21.9.2011) ΣΥΓΚΕΝΤΡΩΤΙΚΟ ΑΡΧΕΙΟ.docx
- Ν. 40142011 (ΦΕΚ 209Α` 21.9.2011).pdf
- Ν. 4030 2011 (ΦΕΚ 249Α` 25.11.2011).pdf
- Ν. 4042 2012 (ΦΕΚ 24Α` 13.2.2012).pdf
- Ν. 4042 2012, (ΦΕΚ 24Α 13.2.2012).pdf
- Ν. 4111 2013 (ΦΕΚ 18Α 25.1.2013).pdf
- Ν. 4146 2013 (ΦΕΚ 90Α` 18.4.2013).pdf
- Ν. 4281 2014 (ΦΕΚ 160Α` 8.8.2014).pdf
- Ν. 4315 2014, (ΦΕΚ 269Α 29.12.2014).pdf
- Ν. 4409 2016, (ΦΕΚ 136Α28.7.2016).pdf

- Ν. 4492 2017, (ΦΕΚ 156Α18.10.2017).pdf
- Ν. 4519 2018 (ΦΕΚ 25Α` 20.2.2018).pdf
- Ν. 4635 2019 (ΦΕΚ 167Α` 30.10.2019).pdf
- Ν. 4685 2020 (ΦΕΚ 92Α` 7.5.2020).pdf
- Ν. 4691 2020 (ΦΕΚ 108 Α 9-6-2020).pdf
- Ν. 4819 2021 (ΦΕΚ 129 Α` 23.7.2021).pdf
- Ν. 48192021 (ΦΕΚ 129Α` 23.7.2021) ΣΥΓΚΕΝΤΡΩΤΙΚΟ ΑΡΧΕΙΟ.docx
- Ν. 48192021 (ΦΕΚ 129Α` 23.7.2021).pdf
- Ν. 4843 2021 (ΦΕΚ 193Α' 20.10.2021).pdf
- Ν. 4936 2022 (ΦΕΚ 105Α` 27.5.2022).pdf
- Ν. 4964 2022 (ΦΕΚ 150Α` 30.7.2022).pdf
- Ν. 49642022 (ΦΕΚ 150Α` 30.7.2022).pdf
- Ν. 5037 2023 (ΦΕΚ 58Α` 28.3.2023).pdf
- Ν. 5069 2023 (ΦΕΚ 193Α` 28.11.2023).pdf
- Ν.3584-2007 _ΚΩΔΙΚΑΣ ΔΗΜΟΤΙΚΩΝ ΥΠΑΛΛΗΛΩΝ_.pdf
- Ν.3852-2010 _ΚΑΛΛΙΚΡΑΤΗΣ_.pdf
- Ν.3879-2010.pdf
- Ν.3889-2010.pdf
- Ν.3905-2010.pdf
- Ν.4122 2013 (ΦΕΚ 42 Α 19-2-2013).pdf
- Ν.4180 2013 (ΦΕΚ 182 Α 5-9-2013).pdf
- Ν.4258 2014 (ΦΕΚ 94 Α 14-04-2014).pdf
- Ν.4270-2014 _ΚΩΔΙΚΑΣ ΔΗΜΟΣΙΟΥ ΛΟΓΙΣΤΙΚΟΥ_.pdf
- Ν.4277 2014 (ΦΕΚ 156 Α 01-08-2014).pdf
- Ν.4280 2014 (ΦΕΚ 159 Α 08-08-2014).pdf
- Ν.4315 2014 (ΦΕΚ 269 Α 24-12-2014).pdf
- Ν.4412- ΔΗΜΟΣΙΕΣ ΣΥΜΒΑΣΕΙΣ 1.pdf
- Ν.4412- ΠΑΡΑΡΤΗΜΑ XVII-ΔΗΜΟΣΙΕΣ ΣΥΜΒΑΣΕΙΣ .pdf
- Ν.4412-ΚΩΔΙΚΟΠΟΙΗΜΕΝΟΣ- ΟΠΩΣ ΤΡΟΠΟΙΗΘΗΚΕ ΚΑΙ ΙΣΧΥΕΙ.pdf
- Ν.4412-ΠΑΡΑΡΤΗΜΑ- Ι-ΧΧ-ΔΗΜΟΣΙΕΣ ΣΥΜΒΑΣΕΙΣ .pdf
- Ν.4412-ΠΑΡΑΤΗΜΑ Ι-ΧΙV ΔΗΜΟΣΙΕΣ ΣΥΜΒΑΣΕΙΣ .pdf
- Ν.4555-2018 _ΚΛΕΙΣΘΕΝΗΣ.pdf
- Ν.4804-2021 _ΕΚΛΟΓΗ ΔΗΜΟΤΙΚΩΝ ΚΑΙ ΠΕΡΙΦΕΡΕΙΑΚΩΝ ΑΡΧΩΝ_.pdf
- Ν.4819 2021 (ΦΕΚ 129 Α23-7-2021).pdf
- Ν.4821 2021 (ΦΕΚ 134 Α31-7-2021).pdf
- Ν.4830-2021 -ΖΩΑ ΣΥΝΤΡΟΦΙΑΣ.pdf
- Ν.4903-2022.pdf
- Ν.4912-2022.pdf
- Ν.4914-2022.pdf
- Ν.4955-2022.pdf
- Ν.4965-2022.pdf
- Ν.5002-2022.pdf
- Ν.5016-2023.pdf
- Ν.5039-2023.pdf

- Ν.5043-2023 _ΡΥΘΜΙΣΕΙΣ ΖΩΩΝ ΣΥΝΤΡΟΦΙΑΣ - ΠΡΟΣΩΠΙΚΟΥ ΔΗΜΟΣΙΟΥ ΤΟΜΕΑ ΚΑΙ ΤΟΠΙΚΗΣ ΑΥΤΟΔΙΟΙΚΗΣΗΣ_.pdf
- Ν.5043-2023.pdf
- Ν.5069 2023 (ΦΕΚ 193 Α 28-11-2023).pdf
- Ν.5079-2023.pdf
- ΝΟΜΟΣ 3937 2011 ΦΕΚ 60 31 03 2011 ΣΥΓΚΕΝΤΡΩΤΙΚΟ ΑΡΧΕΙΟ.docx
- ΝΟΜΟΣ 3937 2011 ΦΕΚ 60 31 03 2011.pdf
- ΝΟΜΟΣ 4014_2011_ΦΕΚ 209-Α-2011.pdf
- ΝΟΜΟΣ ΥΠ' ΑΡΙΘ 4722_1999.docx
- Νόμος 1650_1986 ΦΕΚ Α΄160_16.docx
- Υ.Α. 19582012 (ΦΕΚ 21Β` 13.1.2012).pdf
- Υ.Α. ΥΠΕΝ ΔΙΠΑ 53510 3616 2023 (ΦΕΚ 3327Β` 19.5.2023).pdf
- Υ.Α. ΥΠΕΝ ΔΙΠΑ 64712 4464 2022 (ΦΕΚ 3636 Β` 11.7.2022).pdf
- Υ.Α. ΥΠΕΝΔΙΠΑ17185 1069 2022 (ΦΕΚ 841Β` 24.2.2022).pdf
- Υ.Α. ΥΠΕΝΔΙΠΑ1718510692022 (ΦΕΚ 841Β` 24.2.2022) ΣΥΓΚΕΝΤΡΩΤΙΚΟ ΑΡΧΕΙΟ.docx
- Υ.Α. οικ.56882018 (ΦΕΚ 988Β` 21.3.2018).pdf
- Υ.Α.170255_2014_ΦΕΚ 135-Β-2014.pdf
- Υ.Α.17185_2022_ΦΕΚ 841-Β-2022.pdf

6.2 Τεχνικά Χαρακτηριστικά της Υλοποίησης

Η υλοποίηση αξιοποιεί προηγμένα μοντέλα μηχανικής μάθησης και βαθιάς μάθησης, εστιάζοντας στη σημασιολογική κατανόηση και παραγωγή φυσικού λόγου.

- **Embeddings:** Χρησιμοποιείται το `text-embedding-3-large` της OpenAI, το οποίο δημιουργεί αριθμητικές αναπαραστάσεις (vectors) για τα κείμενα. Η τεχνική αυτή επιτρέπει την αποδοτική σύγκριση και ανάκτηση σχετικών εγγράφων, καθώς διευκολύνει τη σημασιολογική αναζήτηση αντί της παραδοσιακής αναζήτησης με λέξεις-κλειδιά.
- **Μεγάλο Γλωσσικό Μοντέλο (LLM):** Για τη δημιουργία απαντήσεων, χρησιμοποιείται το GPT-4o, ένα προηγμένο γλωσσικό μοντέλο με υψηλή ικανότητα κατανόησης και παραγωγής κειμένου. Το μοντέλο αυτό αξιοποιεί τόσο τα δεδομένα από την ερώτηση του χρήστη όσο και τις πληροφορίες που ανακτώνται μέσω των embeddings, παρέχοντας τεκμηριωμένες και συνεκτικές απαντήσεις.

Για τη διαχείριση των αιτημάτων του chatbot και την επικοινωνία με τους χρήστες, υλοποιήθηκε ένα HTTP API χρησιμοποιώντας τις παρακάτω βιβλιοθήκες:

- **Uvicorn:** Χρησιμοποιείται ως ασύγχρονος web server, επιτρέποντας τη γρήγορη και αποδοτική διαχείριση των αιτήσεων.
- **Starlette:** Χρησιμοποιείται ως lightweight web framework, προσφέροντας ευέλικτο API σχεδιασμό και διαχείριση των αιτημάτων μέσω JSON.

Δεδομένου ότι οι πληροφορίες που διαχειρίζεται το chatbot προέρχονται από πολλαπλά είδη εγγράφων, απαιτείται ένας αποδοτικός μηχανισμός εξαγωγής κειμένου (text extraction). Η χρήση των παρακάτω βιβλιοθηκών εξασφαλίζει ότι το chatbot έχει πρόσβαση σε δομημένα δεδομένα, ανεξάρτητα από τη μορφή του αρχικού εγγράφου.

- **`docx2txt`**: Επιτρέπει την ανάκτηση περιεχομένου από έγγραφα Word (.docx), διατηρώντας τη δομή των παραγράφων.
- **`html2text`**: Χρησιμοποιείται για την μετατροπή HTML εγγράφων σε απλό κείμενο, αφαιρώντας τυχόν περιττές διαμορφώσεις.
- **`pymupdf4llm`**: Επιτρέπει την αποτελεσματική εξαγωγή κειμένου από PDF αρχεία, ακόμα και από εγγράφους με περίπλοκη μορφοποίηση ή ενσωματωμένες εικόνες.

Η αποτελεσματική ανάκτηση πληροφοριών από μεγάλα κείμενα απαιτεί μια συνδυαστική προσέγγιση κατακερματισμού κειμένων (chunking) και σημασιολογικής αναζήτησης.

- **Διαχωρισμός κειμένων σε αποσπάσματα (chunking):** Χρησιμοποιείται η βιβλιοθήκη **`chonkie`**, η οποία διαιρεί τα μεγάλα κείμενα σε μικρότερα τμήματα. Αυτό επιτρέπει την ταχύτερη και ακριβέστερη ανάκτηση σχετικών πληροφοριών, διασφαλίζοντας ότι κάθε απόσπασμα περιέχει επαρκές περιεχόμενο για νοηματική κατανόηση.
- **Σημασιολογική αναζήτηση:** Υλοποιείται μέσω cosine similarity από τη SciPy, επιτρέποντας τη σύγκριση των embeddings μεταξύ ερωτημάτων και αποθηκευμένων εγγράφων. Η μέθοδος αυτή διασφαλίζει ότι το chatbot ανακτά τις πιο σχετικές πληροφορίες, ανεξάρτητα από τη διατύπωση του χρήστη.

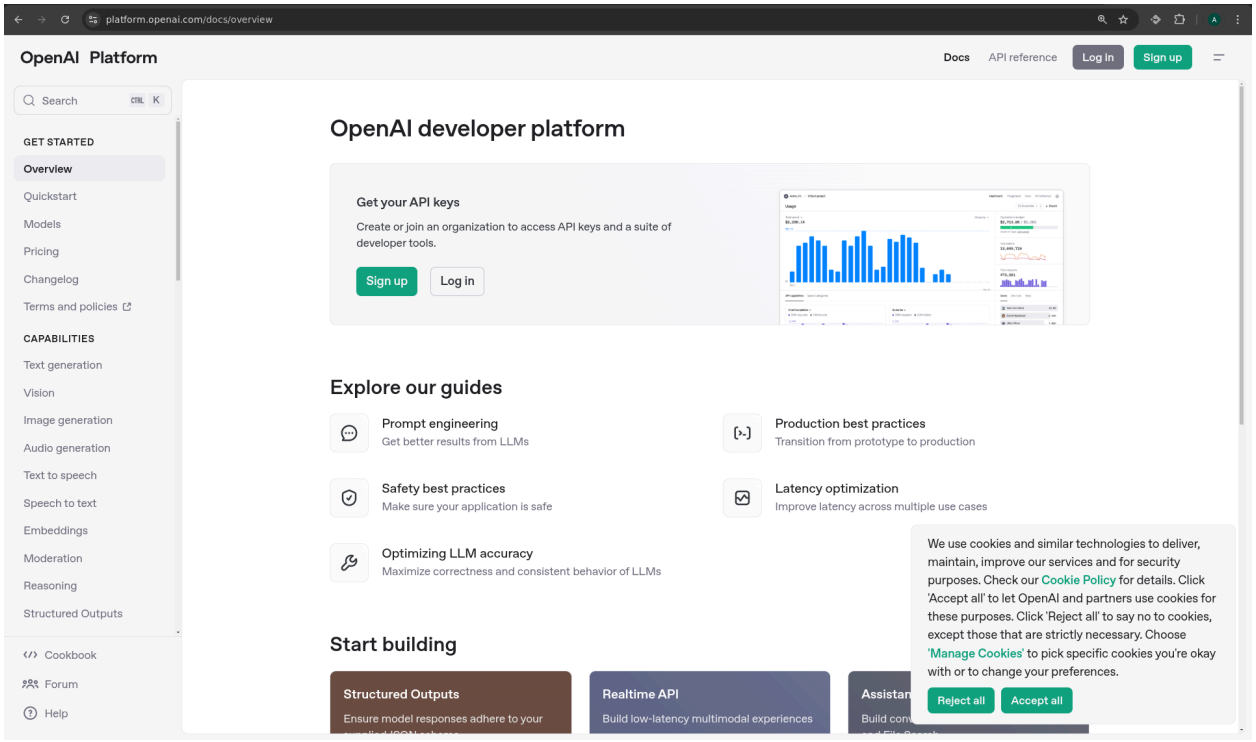
Η επικοινωνία του chatbot με άλλες εφαρμογές και πληροφοριακά συστήματα πραγματοποιείται μέσω ενός **RESTful JSON API**, το οποίο επιτρέπει:

- Αποστολή ερωτημάτων και λήψη απαντήσεων σε δομημένη μορφή.
- Εύκολη επεκτασιμότητα και ενσωμάτωση με web εφαρμογές ή mobile interfaces.

6.3 Πρόσβαση στην πλατφόρμα της OpenAI

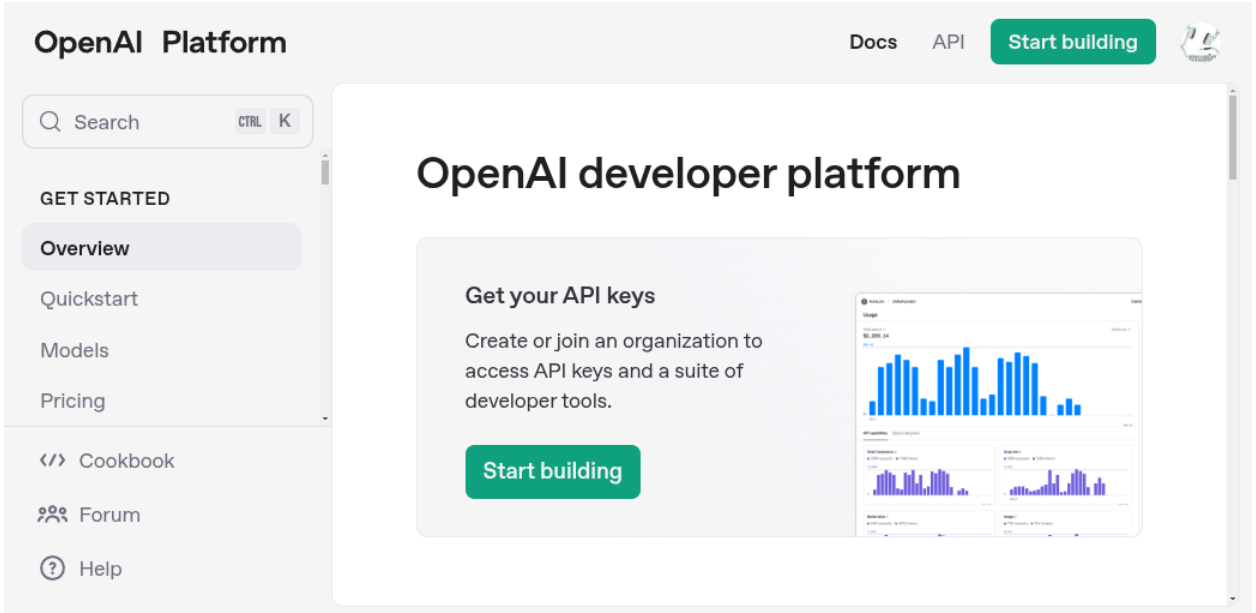
Για την χρήση των μοντέλων της OpenAI πρέπει να αποκτήσουμε πρόσβαση στην πλατφόρμα της εταιρίας και να δημιουργήσουμε ένα κλειδί πρόσβασης στο API.

Αρχικά πρέπει να μεταβούμε στη σελίδα https://platform.openai.com και να πατήσουμε πάνω δεξιά **Log in** ή **Sign up** για να συνδεθούμε.



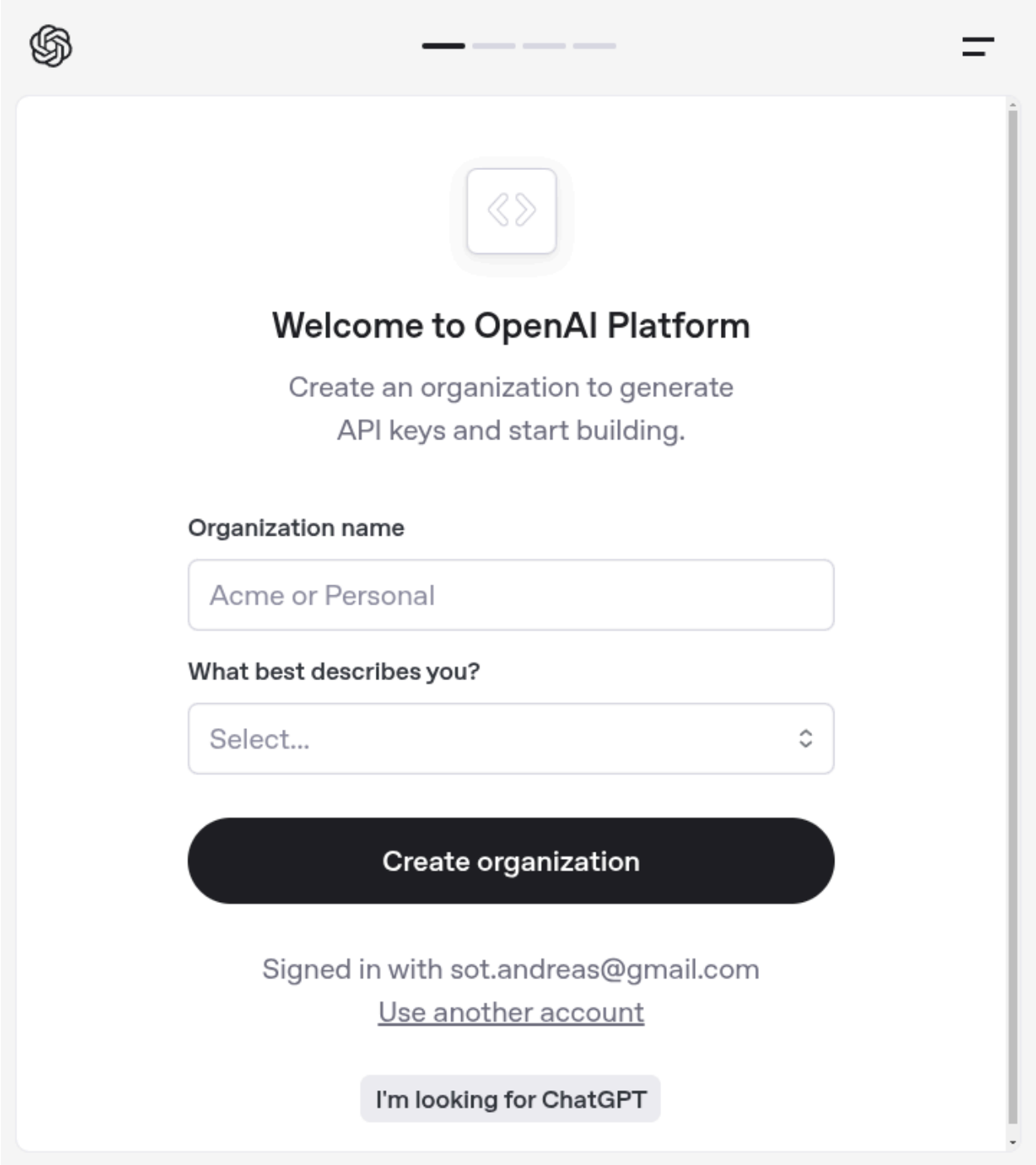
Αρχική σελίδα platform.openai.com

Αφού συνδεθούμε για πρώτη φορά μπορούμε να πατήσουμε στο κουμπί **Start building** για να δημιουργήσουμε τον πρώτο μας οργανισμό στην πλατφόρμα, να δημιουργήσουμε ένα κλειδί πρόσβασης και να προσθέσουμε χρήματα.



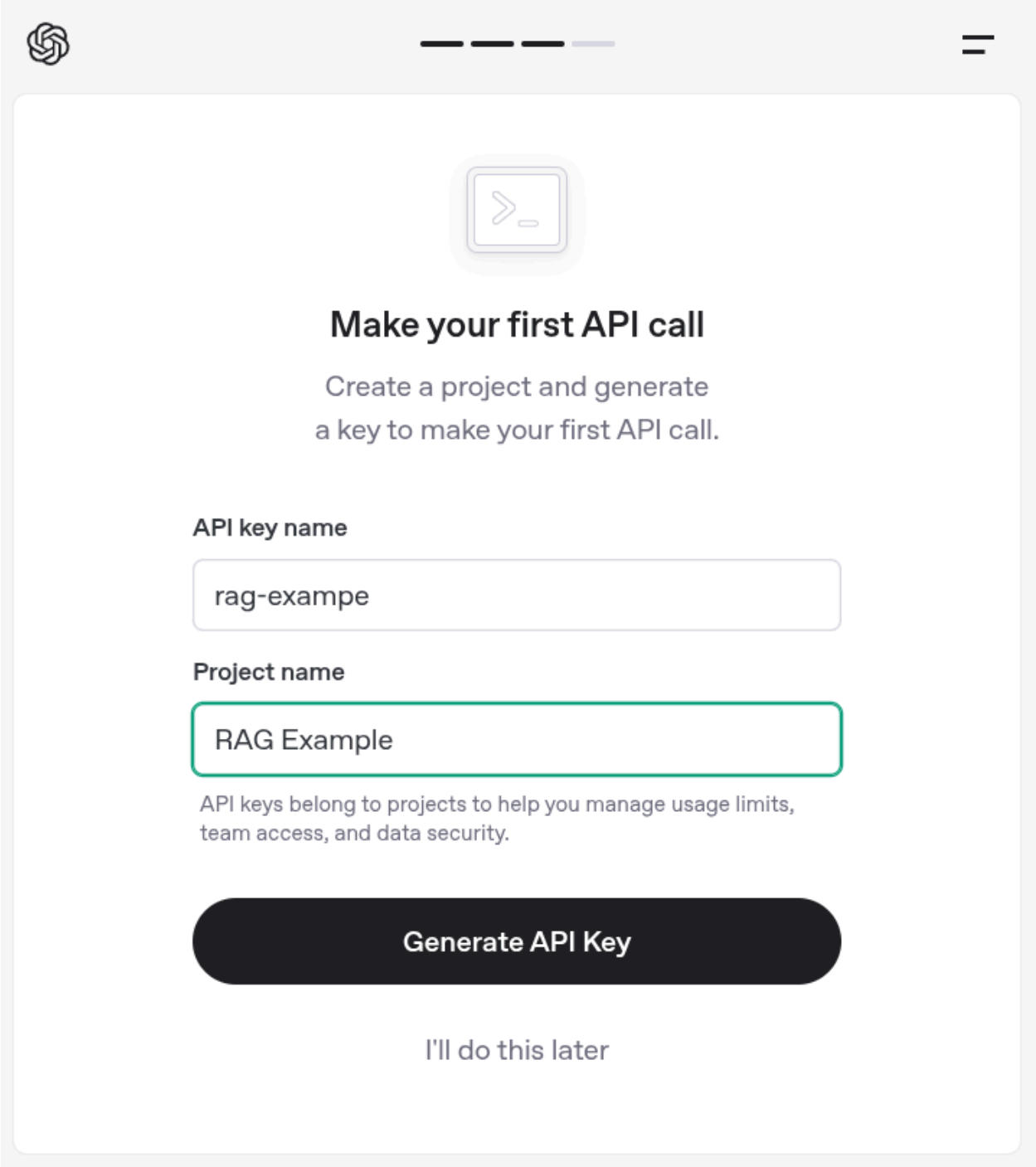
Αρχική σελίδα μετά την πρώτη σύνδεση

Ύστερα, στην επόμενη οθόνη ζητείται να δώσουμε ένα όνομα στον οργανισμό που θα δημιουργήσουμε.

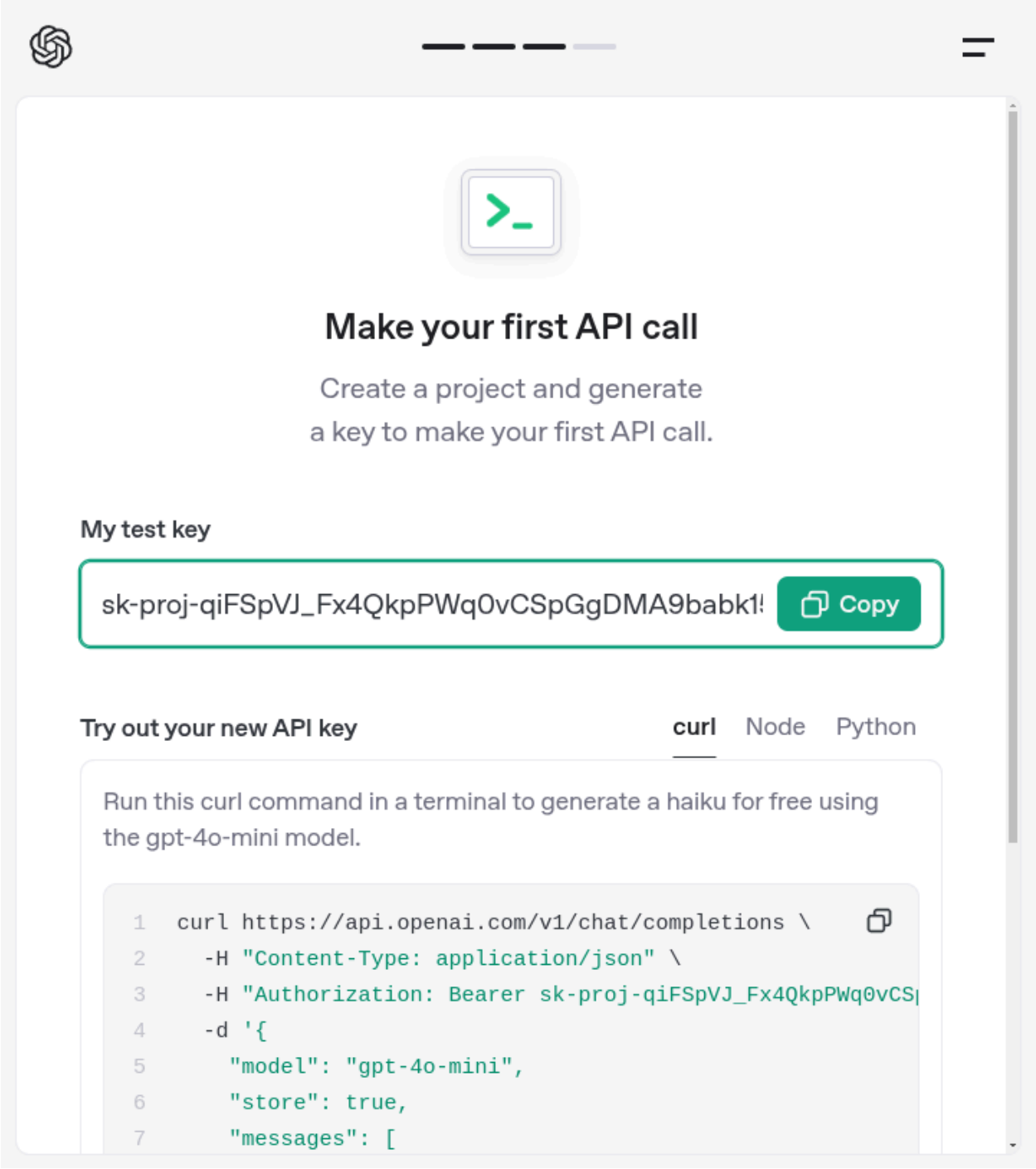


Σελίδα δημιουργίας οργανισμού

Στο επόμενο βήμα δημιουργούμε το κλειδί πρόσβασης (API key) με το οποίο θα μπορέσουμε να πραγματοποιήσουμε κλήσεις προς το API της OpenAI και να χρησιμοποιήσουμε τα μοντέλα. Στην παρακάτω οθόνη δίνουμε ένα όνομα στο κλειδί και επιλέγουμε το project του οργανισμού μας στο οποίο θα χρησιμοποιηθεί το κλειδί.
Πατώντας το κουμπί **Generate API Key** δημιουργείται το κλειδί και εμφανίζεται στην επόμενη οθόνη όπου μπορούμε να το αντιγράψουμε.

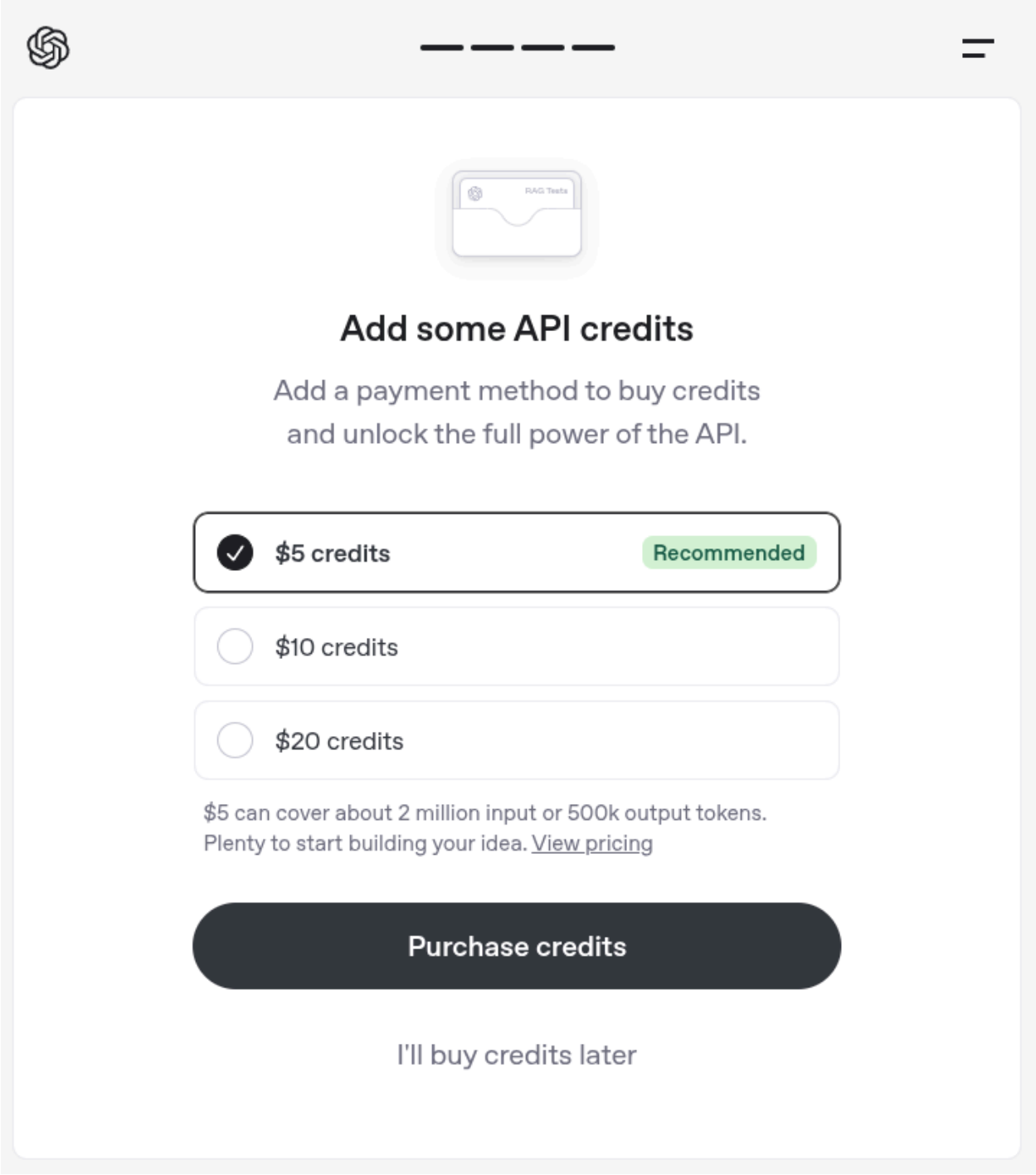


Βήμα δημιουργία κλειδιού πρόσβασης



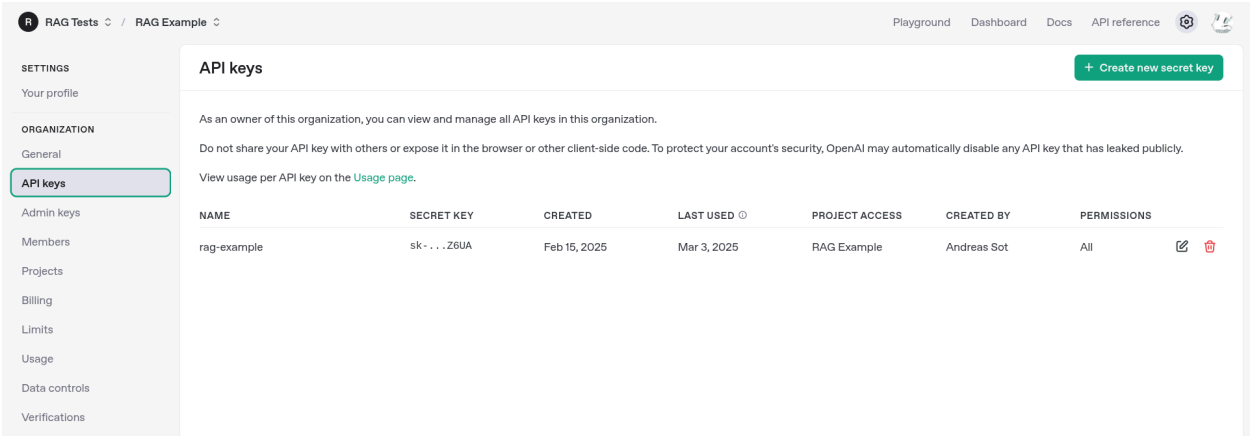
Το κλειδί που δημιουργήσαμε

Τέλος, μπορούμε να αγοράσουμε credits προκειμένου να αποκτήσουμε πρόσβαση σε όλα τα μοντέλα που παρέχονται από το API.



Τελευταία οθόνη για την αγορά credits

Μέσα από το dashboard μπορούμε οποιαδήποτε στιγμή να αγοράσουμε περισσότερα credits ή να δημιουργήσουμε νέα κλειδιά.



Dashboard της πλατφόρμας

6.4 Παραδείγματα χρήσης της βιβλιοθήκης openai και του API

Για την επικοινωνία με το API χρησιμοποιήσαμε την βιβλίοθηκη ***openai*** για την γλώσσα Python. Το API είναι προσβάσιμο και μέσο ενός απλών HTTP κλήσεων.

Αρχικά δημιουργούμε το περιβάλλον της Python (virtual environment):

```
$ python -m venv venv
$ . venv/bin/activate
```

Στη συνέχεια εγκαθιστούμε την βιβλιοθήκη της OpenAI ως εξής:

```
$ pip install openai
```

Τώρα μπορούμε να τρέξουμε κώδικα Python και να χρησιμοποιήσουμε τα μοντέλα της OpenAI. Για την δημιουργία embeddings μπορούμε να τρέξουμε τα παρακάτω:

```
from openai import OpenAI

client = OpenAI(api_key="enter-api-key-here")
embedding = client.embeddings.create(
            input="Εδώ εισάγουμε το κείμενο για το οποίο θέλουμε embeddings",
            model="text-embedding-3-large").data[0].embedding
```

Στην παράμετρο model μπορούμε να επιλέξουμε το μοντέλο που επιθυμούμε. Οι βασικές επιλογές μας είναι τα text-embedding-3-large και text-embedding-3-small.

Στο επόμενο παράδειγμα θα χρησιμοποιήσουμε ένα από τα μοντέλα LLM για την παραγωγή κειμένου:

```
from openai import OpenAI

client = OpenAI(api_key="enter-api-key-here")
response = client.chat.completions.create(
    messages=[
        {
            "role": "system",
            "content": "You answer questions based on the context given."
        },
        {
            "role": "user",
            "content": "Τι ξέρεις για το πανεπιστήμιο αιγαίου"
        }
    ],
    model="gpt-4o-mini",
    temperature=0
)
print(response.choices[0].message.content)
```

Η παράμετρος messages δέχεται μια λίστα με τα μηνύματα της συζήτησης. Στο τελευταίο μήνυμα βρίσκεται το ερώτημα του χρήστη προς το LLM. Τα υπόλοιπα μηνύματα χρησιμοποιούνται ως context για την παραγωγή της τελικής απάντησης.

Περισσότερες πληροφορίες για την χρήση της βιβλιοθήκης βρίσκονται στον ακόλουθο σύνδεσμο: https://github.com/openai/openai-python

Χρησιμοποιώντας αυτές τις δύο βασικές κλήσεις μπορούμε να υλοποιήσουμε το chatbot μας όπως περιγράψαμε στις προηγούμενες ενότητες.

6.5 Επεξεργασία εγγράφων, chunks και embeddings

Για την υλοποίηση ενός αποδοτικού μηχανισμού αναζήτησης στα έγγραφα, απαιτείται η μετατροπή τους σε απλό κείμενο, ο τεμαχισμός τους σε μικρότερα τμήματα (chunking) και, τέλος, η δημιουργία σημασιολογικών αναπαραστάσεων (embeddings) για κάθε επιμέρους τμήμα. Αυτά τα βήματα επιτρέπουν την αποτελεσματική επεξεργασία και ανάκτηση σχετικών πληροφοριών μέσω της χρήσης τεχνικών βαθιάς μάθησης και σημασιολογικής αναζήτησης.

6.5.1 Μετατροπή Εγγράφων σε Απλό Κείμενο

Δεδομένου ότι τα έγγραφα μπορεί να είναι αποθηκευμένα σε διαφορετικές μορφές αρχείων, είναι απαραίτητη η χρήση εξειδικευμένων εργαλείων για την εξαγωγή κειμένου από κάθε τύπο αρχείου. Στην παρούσα υλοποίηση χρησιμοποιούνται οι εξής βιβλιοθήκες:

- **pymupdf4llm**: Εξάγει κείμενο από αρχεία **PDF** και το μετατρέπει σε μορφή Markdown.
- **docx2txt**: Χρησιμοποιείται για την εξαγωγή κειμένου από αρχεία **Microsoft Word (.docx)**.
- **html2text**: Μετατρέπει αρχεία **HTML** σε απλό κείμενο, αφαιρώντας τυχόν HTML tags και διατηρώντας μόνο το περιεχόμενο.

Τα έγγραφα προς επεξεργασία είναι αποθηκευμένα σε έναν καθορισμένο φάκελο (`/data`). Ο παρακάτω κώδικας διαβάζει κάθε αρχείο, αναγνωρίζει τον τύπο του και εφαρμόζει την αντίστοιχη βιβλιοθήκη για την εξαγωγή του κειμένου:

```
import docx2txt
import html2text
import pymupdf4llm

def read_docs():
    for path in Path(config.INPUT_FILES).iterdir():
        match path.suffix.lower():
            case ".pdf":
                text = pymupdf4llm.to_markdown(path)
            case ".docx":
                text = docx2txt.process(path)
            case ".html":
                with open(path, "r") as fp:
                    text = html2text.html2text(fp.read())
            case _:
                print(f"Skipping unexpected file {path}")

        yield path, text
```

Με την εκτέλεση της συνάρτησης `read_docs()`, όλα τα έγγραφα μετατρέπονται σε απλό κείμενο, προετοιμάζοντας το επόμενο στάδιο της επεξεργασίας τους.

6.5.2 Τεμαχισμός Κειμένου (Chunking)

Δεδομένου ότι τα έγγραφα μπορεί να είναι μεγάλα, ο διαχωρισμός τους σε μικρότερα, νοηματικά αυτόνομα τμήματα (chunks) είναι απαραίτητος. Ο τεμαχισμός (chunking) επιτρέπει:

- **Βελτίωση της ακρίβειας των αναζητήσεων**, καθώς μικρότερα αποσπάσματα κειμένου μπορούν να ταιριάξουν καλύτερα με το ερώτημα του χρήστη.
- **Μείωση της υπολογιστικής επιβάρυνσης**, καθώς τα embeddings υπολογίζονται σε μικρότερα τμήματα κειμένου.
- **Διατήρηση της σημασιολογικής συνοχής** μέσω κατάλληλων κανόνων κατακερματισμού.

Για την υλοποίηση του chunking χρησιμοποιείται η βιβλιοθήκη **chonkie**, η οποία επιτρέπει την προσαρμογή του τρόπου κατακερματισμού μέσω καθορισμένων κανόνων (`RecursiveRules`). Ο ακόλουθος κώδικας αρχικοποιεί έναν **RecursiveChunker** με κατάλληλες ρυθμίσεις:

```
import tiktoken
from chonkie import RecursiveChunker, RecursiveRules

chunker = RecursiveChunker(
    tokenizer=tiktoken.encoding_for_model("text-embedding-3-large"),
    chunk_size=1024,  # Ορισμός μεγέθους chunk (1024 tokens)
    rules=RecursiveRules(),  # Χρήση προκαθορισμένων κανόνων τεμαχισμού
    min_characters_per_chunk=12  # Ελάχιστο όριο χαρακτήρων για κάθε chunk
)
```

Με τη χρήση της `RecursiveChunker`, κάθε έγγραφο διαχωρίζεται σε τμήματα μεγέθους έως **1024 tokens**, λαμβάνοντας υπόψη τη σημασιολογική συνοχή του κειμένου.

6.5.3 Δημιουργία Embeddings

Αφού τα έγγραφα έχουν μετατραπεί σε απλό κείμενο και τεμαχιστεί κατάλληλα, το επόμενο βήμα είναι η δημιουργία των σημασιολογικών αναπαραστάσεων (embeddings). Για τον σκοπό αυτό, χρησιμοποιείται το μοντέλο **`text-embedding-3-large`** της OpenAI, το οποίο μετατρέπει κάθε chunk κειμένου σε έναν πολυδιάστατο αριθμητικό πίνακα (vector), επιτρέποντας τη σημασιολογική αναζήτηση μέσω μετρικών ομοιότητας, όπως το cosine similarity.

Ο ακόλουθος κώδικας δείχνει πώς παράγονται τα embeddings για κάθε chunk κειμένου:

```
from openai import OpenAI
client = OpenAI(api_key=config.OPENAI_API_KEY)

def run():
    docs = []
    for path, text in read_docs():
        texts = []
        embds = []
        for chunk in chunker.chunk(text):
            t = chunk.text
            response = client.embeddings.create(
                input=t, model="text-embedding-3-large")
            texts.append(t)
            embds.append(response.data[0].embedding)
```

```
        docs.append(Document(path, texts, embds))
```

6.5.4 Αποθήκευση των Δεδομένων

Τα παραγόμενα embeddings, μαζί με τα αντίστοιχα chunks κειμένου, αποθηκεύονται σε ένα αρχείο ή μια βάση δεδομένων, ώστε να χρησιμοποιηθούν στη διαδικασία της σημασιολογικής αναζήτησης.

Η αποθήκευση αυτή επιτρέπει τη γρήγορη ανάκτηση σχετικών πληροφοριών κάθε φορά που πραγματοποιείται ένα ερώτημα από τον χρήστη, χωρίς να χρειάζεται να επανυπολογίζονται τα embeddings.

6.5.5 Λίγα λόγια για τα μοντέλα embeddings

Τα **text-embedding-3-large** και **text-embedding-3-small** είναι δύο μοντέλα της OpenAI σχεδιασμένα για τη μετατροπή κειμένου σε σημασιολογικές αναπαραστάσεις (embeddings), οι οποίες επιτρέπουν τη σύγκριση και την ανάκτηση πληροφοριών βάσει νοήματος (“New embedding models and API updates” 2024).

Σύγκριση text-embedding-3-large και text-embedding-3-small:

	text-embedding-3-large	**text-embedding-3-small**
Μέγεθος Διανύσματος (Vector Size)	3072	1536
Ακρίβεια και Ικανότητα Κατανόησης	Υψηλή, κατάλληλο για μεγάλα, σύνθετα κείμενα	Καλή, βελτιωμένη σε σχέση με προηγούμενα μοντέλα, αλλά λιγότερο ισχυρό από το "large"
Υπολογιστικό Κόστος	Υψηλότερο λόγω του μεγαλύτερου μεγέθους του διανύσματος	Χαμηλότερο, πιο αποδοτικό για εφαρμογές με περιορισμένους πόρους
Ταχύτητα Υπολογισμού	Πιο αργό λόγω της αυξημένης πολυπλοκότητας	Γρηγορότερο, κατάλληλο για real-time αναζητήσεις
Μνήμη και Χώρος Αποθήκευσης	Απαιτεί περισσότερο αποθηκευτικό χώρο και RAM	Πιο ελαφρύ, καταλαμβάνει λιγότερη μνήμη
Κόστος	$0.13 / 1M tokens	$0.02 / 1M tokens

6.6 Μηχανισμός αναζήτησης

Για να απαντήσει σε ερωτήματα του χρήστη, το chatbot χρησιμοποιεί μία διαδικασία δύο σταδίων:

1. **Ανάκτηση συναφών τμημάτων κειμένου (retrieval)** από την αποθηκευμένη βάση των embeddings.
2. **Παραγωγή απάντησης (generation)** μέσω του γλωσσικού μοντέλου GPT-4o-mini, το οποίο αξιοποιεί τα ανακτημένα τμήματα κειμένου ως πλαίσιο αναφοράς (context).

Αυτή η προσέγγιση, όπως παρουσιάστηκε και σε προηγούμενες ενότητες, είναι γνωστή ως **Retrieval-Augmented Generation (RAG)** και επιτρέπει στο chatbot να παρέχει ακριβείς και τεκμηριωμένες απαντήσεις, αντλώντας πληροφορίες από τα αποθηκευμένα έγγραφα.

Ο συνδυασμός σημασιολογικής αναζήτησης και παραγωγής κειμένου μέσω GPT προσφέρει ένα ισχυρό σύστημα ανάκτησης και κατανόησης εγγράφων.

- **Το chatbot δεν απαντά αυθαίρετα**, αλλά βασίζεται σε πραγματικά αποθηκευμένα δεδομένα.
- **Η σημασιολογική αναζήτηση** επιτρέπει την εύρεση σχετικών πληροφοριών ακόμα και όταν οι λέξεις στο ερώτημα διαφέρουν από τις λέξεις στο κείμενο.
- **Η χρήση του GPT-4o-mini** διαμορφώνει συνεκτικές και κατανοητές απαντήσεις για τον χρήστη, αξιοποιώντας τα παρεχόμενα δεδομένα.

Αυτή η προσέγγιση καθιστά το chatbot κατάλληλο για τη διαχείριση μεγάλων συλλογών νομικών και διοικητικών κειμένων (π.χ. ΦΕΚ, νόμοι, δημοτικές αποφάσεις), βελτιώνοντας την πρόσβαση σε πολύπλοκες πληροφορίες με έναν φυσικό και κατανοητό τρόπο.

6.6.1 Ανάκτηση Συναφών Τμημάτων Κειμένου

Η ανάκτηση των πιο σχετικών τμημάτων κειμένου πραγματοποιείται μέσω αναζήτησης στα embeddings που δημιουργήσαμε στην προηγούμενη ενότητα. Τα embeddings είναι αποθηκευμένα σε ένα αρχείο και φορτώνονται στη μνήμη για να χρησιμοποιηθούν κατά την αναζήτηση.

```
def load(path: str):
    global chunks

    with open(path, 'rb') as f:
        documents = pickle.load(f)

    chunks = []
    for doc in documents:
        for i, chunk in enumerate(doc.chunks):
```

```
            chunks.append(Chunk(doc.path, chunk,
np.array(doc.embeddings[i])))
```

Με αυτόν τον τρόπο, κάθε τεμαχισμένο τμήμα κειμένου (**chunk**) αναπαρίσταται ως ένα αντικείμενο της κλάσης `Chunk`, το οποίο περιέχει:

- Την **πηγή** του εγγράφου (`path`).
- Το **κειμενικό περιεχόμενο** (`text`).
- Το **διάνυσμα embedding** (`embedding`).

Η αναζήτηση πραγματοποιείται μετρώντας την συνεφαπτομένη ομοιότητα (cosine similarity) μεταξύ του embedding του ερωτήματος και των embeddings των αποθηκευμένων chunks.

```
def search(query, n=3, embedded=False):
    global chunks

    if not embedded:
        embedding = client.embeddings.create(
            input=query,
            model="text-embedding-3-large"
        ).data[0].embedding
    else:
        embedding = query

    chunks.sort(key=lambda x: cosine(x.embedding, embedding),
reverse=False)
    return chunks[:n]
```

Η διαδικασία περιλαμβάνει τα εξής βήματα:

1. **Μετατροπή του ερωτήματος** σε embedding μέσω του **text-embedding-3-large** (εκτός αν ήδη δίνεται ως embedding).
2. **Υπολογισμός της cosine similarity** μεταξύ του ερωτήματος και των embeddings των αποθηκευμένων τμημάτων κειμένου.
3. **Ταξινόμηση των chunks** βάσει ομοιότητας (όσο μικρότερη η τιμή, τόσο πιο συναφές το chunk).
4. **Επιστροφή των n πιο συναφών chunks**.

Αυτή η μέθοδος επιτρέπει στο σύστημα να βρίσκει τις πιο σχετικές πληροφορίες ανεξάρτητα από το αν χρησιμοποιούνται ακριβείς λέξεις-κλειδιά, καθώς βασίζεται στη σημασιολογική εγγύτητα.

6.6.2 Παραγωγή Απάντησης μέσω GPT-4o-mini

Αφού ανακτηθούν τα πιο σχετικά τμήματα κειμένου, το σύστημα τα χρησιμοποιεί ως πλαίσιο αναφοράς (context) και διαμορφώνει μία προτροπή (prompt) προς το γλωσσικό μοντέλο GPT-4o-mini.

```
def chat(query: str, n=3):
    results = search(query, n=n)

    content = "\n\n".join([r.text for r in results])
    q = f"""Use context below to answer the subsequent question. If the
answer cannot be found, write "Δεν γνωρίζω". Answer in the same language as
the input.
Context:
\"\"\"
{content}
\"\"\"

Question: {query}
    """

    response = client.chat.completions.create(
        messages=[
            {
                "role": "system",
                "content": "You answer questions based on the context
given."
            },
            {
                "role": "user",
                "content": q
            }
        ],
        model="gpt-4o-mini",
        temperature=0
    )
    return response.choices[0].message.content
```

Τα βήματα που πραγματοποιούνται είναι τα εξής:

1. **Σύνθεση του prompt**:
    - Εισάγεται το συναφές περιεχόμενο (`Context`), που προέκυψε από την αναζήτηση.
    - Διατυπώνεται η ερώτηση του χρήστη (`Question`).
    - Προστίθεται η οδηγία: *Αν δεν υπάρχει απάντηση στο παρεχόμενο περιεχόμενο, απάντησε με "Δεν γνωρίζω".*
2. **Χρήση του GPT-4o-mini** για την παραγωγή της απάντησης.
3. **Επιστροφή της τελικής απάντησης** στον χρήστη.

Το χαμηλό temperature (0) διασφαλίζει ότι οι απαντήσεις είναι πιο συνεπείς και λιγότερο δημιουργικές, εστιάζοντας αποκλειστικά στο παρεχόμενο πλαίσιο πληροφοριών.

6.7 Επίδειξη λειτουργιας

Το chatbot παρέχει ένα HTTP API το οποίο επιτρέπει τη διασύνδεση εξωτερικών συστημάτων, για αναζήτηση και παραγωγή απαντήσεων.

Τα endpoints για την αλληλεπίδραση με το chatbot είναι:

- [GET] /chat?q=<ερώτηση>
- [GET] /search?q=<ερώτηση>

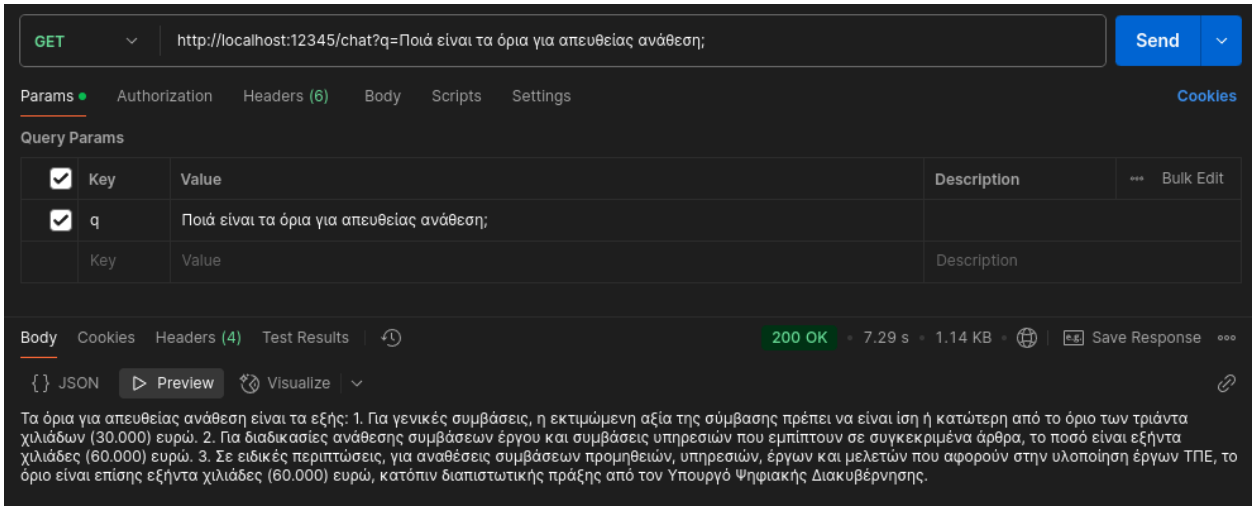
Το endpoint /chat παράγει μια απάντηση στο ερώτημα του χρήστη, ενώ το /search πραγματοποιεί αναζήτηση και επιστρέφει τα αποτελέσματα μαζί με τις πηγές τους.

6.7.1 Παραδείγματα και δοκιμές

Χρησιμοποιούμε την εφαρμογή Postman για να στείλουμε ερωτήματα στο HTTP API του chatbot και να παράγουμε τις απαιτούμενες απαντήσεις.

Ποιά είναι τα όρια για απευθείας ανάθεση;

Στην παρακάτω εικόνα χρησιμοποιούμε το endpoint /chat για να κάνουμε την ερώτηση “Ποιά είναι τα όρια για απευθείας ανάθεση;” στο chatbot:



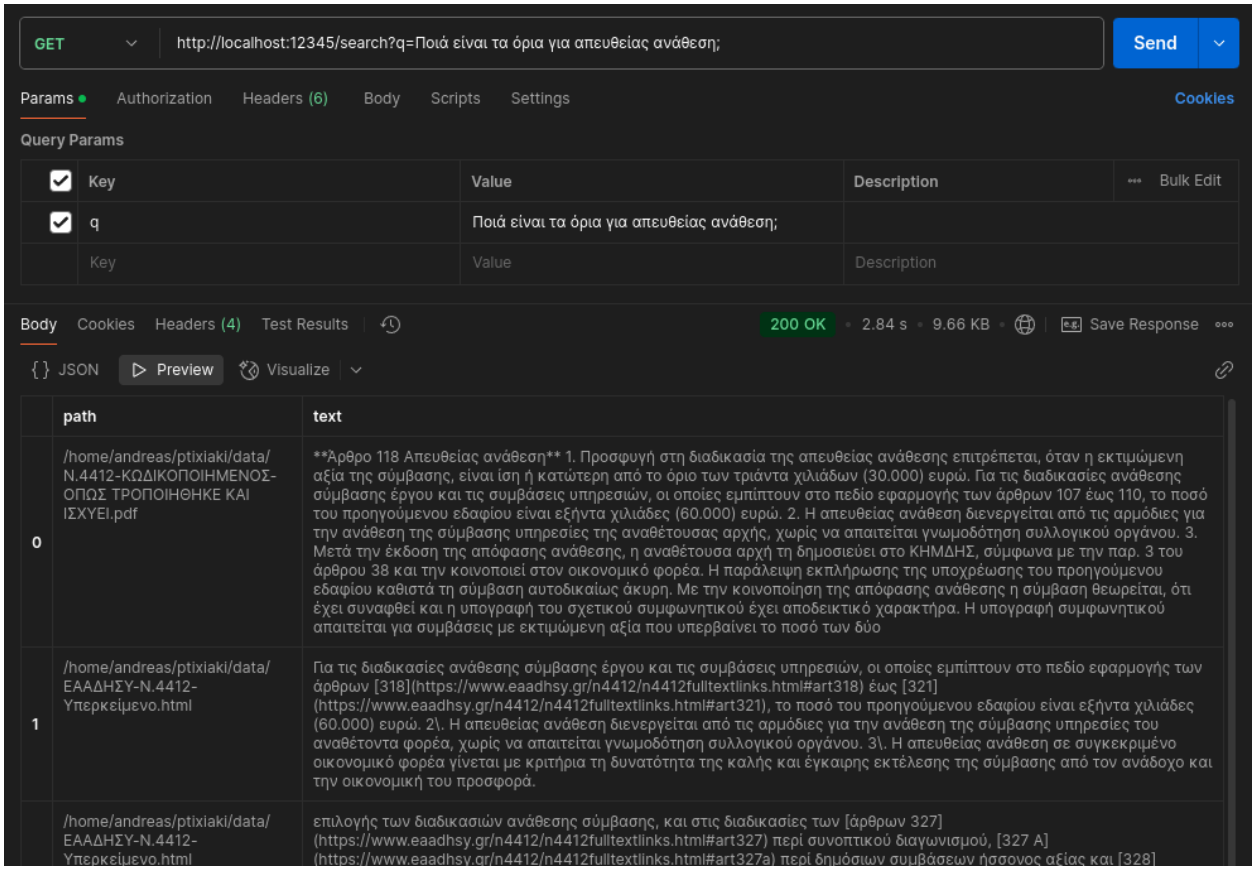
Απάντηση από το μοντέλο GPT 4o-mini

Μετά από ένα μικρό χρονικό διάστημα η απάντηση που παίρνουμε είναι:

Τα όρια για απευθείας ανάθεση είναι τα εξής:

1. *Για γενικές συμβάσεις, η εκτιμώμενη αξία της σύμβασης πρέπει να είναι ίση ή κατώτερη από το όριο των τριάντα χιλιάδων (30.000) ευρώ.*
2. *Για διαδικασίες ανάθεσης συμβάσεων έργου και συμβάσεις υπηρεσιών που εμπίπτουν σε συγκεκριμένα άρθρα, το ποσό είναι εξήντα χιλιάδες (60.000) ευρώ.*
3. *Σε ειδικές περιπτώσεις, για αναθέσεις συμβάσεων προμηθειών, υπηρεσιών, έργων και μελετών που αφορούν στην υλοποίηση έργων ΤΠΕ, το όριο είναι επίσης εξήντα χιλιάδες (60.000) ευρώ, κατόπιν διαπιστωτικής πράξης από τον Υπουργό Ψηφιακής Διακυβέρνησης.*

Κάνοντας την ίδια ερώτηση στο endpoint /search μπορούμε να δούμε τα αποτελέσματα της αναζήτησης και τα chunks που δόθηκαν στο LLM για την παραγωγή της απάντησης καθώς θα τα πηγαία αρχεία από τα οποία προέκυψαν αυτά τα chunks, έτσι ώστε να έχουμε την πηγή της απάντησης.

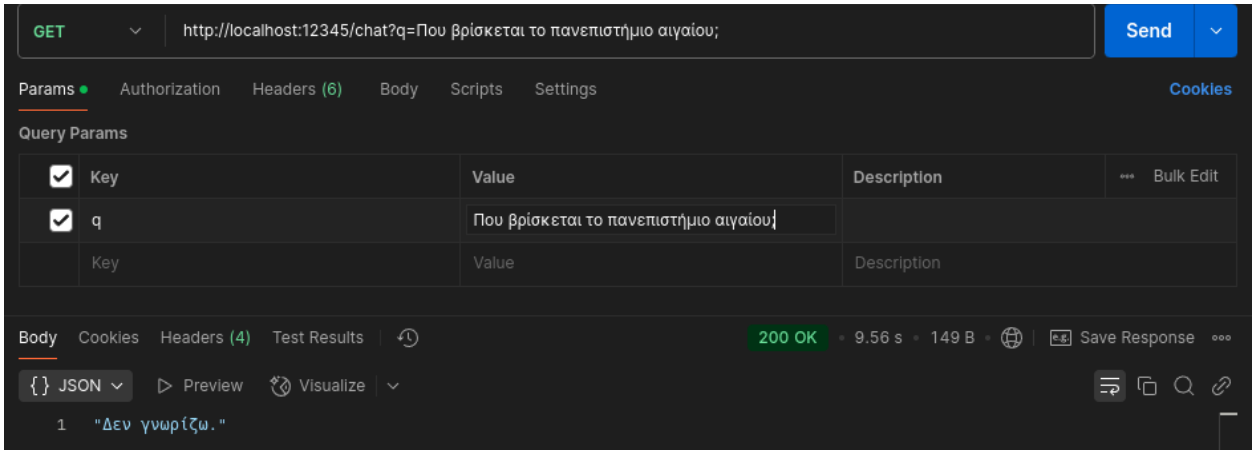


Αποτελέσματα αναζήτησης στα chunks

Όπως βλέπουμε στην εικόνα η απάντηση προέρχεται από το πρώτο αποτέλεσμα και βρίσκεται στο αρχείο ***Ν.4412-ΚΩΔΙΚΟΠΟΙΗΜΕΝΟΣ- ΟΠΩΣ ΤΡΟΠΟΙΗΘΗΚΕ ΚΑΙ ΙΣΧΥΕΙ.pdf***
Στο endpoint /chat το LLM παίρνει αυτά τα αποσπάσματα και παράγει την τελική απάντηση χρησιμοποιώντας μόνο τη σχετική, με την ερώτηση, πληροφορία.

Παράδειγμα με ερώτημα που δεν μπορεί να απαντηθεί από τα κείμενα

Για να επαληθεύσουμε ότι το chatbot δεν απαντάει σε ερωτήσεις που δεν υπάρχουν στα έγγραφα-πηγές που του δώσαμε, κάνουμε μια γενική ερώτηση που ξέρουμε ότι δεν αναγραφεται στις πηγές που δίνουμε στο LLM.



Απάντηση “Δεν γνωρίζω” από το LLM

Ρωτώντας “*Που βρίσκεται το πανεπιστήμιο αιγαίου;*” παίρνουμε την απάντηση “Δεν γνωρίζω.”. Προφανώς το LLM μπορεί να απαντήσει σε αυτή την ερώτηση, αλλά επειδή στο prompt μας του έχουμε δώσει την οδηγία να απαντάει μόνο με βάση τις πηγές που του περνάμε δεν φέρνει κάποια απάντηση. Με αυτόν τον τρόπο εξασφαλίζουμε ότι πάντοται απαντά με βάση τις πηγές και η πληροφορία που μας φέρνει είναι υπαρκτή σε αυτές.

6.8 Αξιολόγηση των αποτελεσμάτων

Μέσα από τη χρήση του chatbot, παρατηρούμε ότι οι απαντήσεις είναι ποιοτικά ανώτερες όταν οι ερωτήσεις μας είναι λεπτομερείς και στοχευμένες, επικεντρώνοντας σε συγκεκριμένα σημεία του κειμένου. Αντίθετα, οι γενικές ερωτήσεις συχνά οδηγούν στην απάντηση «Δεν γνωρίζω».

Αυτό πιθανόν οφείλεται στο μέγεθος των τμημάτων (chunks) που χρησιμοποιήσαμε, τα οποία έχουν μέγεθος έως 1024 tokens το καθένα. Μεγαλύτερα chunks ενδέχεται να καλύπτουν ευρύτερο πλαίσιο και να παρέχουν περισσότερες πληροφορίες. Τα μεγάλα γλωσσικά μοντέλα (LLMs), όπως το GPT-4o, μπορούν να διαχειριστούν context έως και 128.000 tokens, που σημαίνει ότι θα μπορούσαμε να χρησιμοποιήσουμε τμήματα πάνω από 10 φορές μεγαλύτερα.

6.8.1 Αξιολόγηση με F1@k, Precision@k και Recall@k

Εκτός από την εμπειρική αξιολόγηση πραγματοποιήθηκε και αξιολόγηση στο μηχανισμό ανάκτησης πληροφορίας (retrieval) χρησιμοποιώντας τις μεθόδους Precision@k, Recall@k και F1@k. Με αυτές τις μετρήσεις μπορούμε να εκτιμήσουμε την ποιότητα της αναζήτησης και την καταλληλότητα των ανακτημένων εγγράφων.

Ο μηχανισμός αναζήτησης επιστρέφει τα k πλησιέστερα αποτελέσματα στο ερώτημα που δίνεται. Για την αξιολόγηση έχουμε επιλέξει ως k τον αριθμό 5.

6.8.1.1 Ορισμοί των Μετρικών

Precision@k

Το Precision@k μετρά το ποσοστό των πραγματικά σχετικών τμημάτων (chunks) στα k κορυφαία αποτελέσματα που επέστρεψε το σύστημα:

$$\text{Precision@k} = \frac{\text{Αριθμός σχετικών chunks στα πρώτα } k}{k}$$

Το Precision@k αξιολογεί **πόσο ακριβείς** είναι οι κορυφαίες επιστροφές του συστήματος, αλλά δεν λαμβάνει υπόψη τα σχετικά chunks που ενδέχεται να μην έχουν ανακτηθεί.

Recall@k

Το Recall@k μετρά το ποσοστό των σχετικών chunks που έχουν ανακτηθεί μεταξύ όλων των πιθανών σχετικών chunks:

$$\text{Recall@k} = \frac{\text{Αριθμός σχετικών chunks στα πρώτα } k}{\text{Συνολικός αριθμός σχετικών chunks}}$$

Το Recall@k είναι σημαντικό για να διασφαλίσουμε ότι το σύστημα δεν παραλείπει κρίσιμες πληροφορίες.

F1@k Score

Ο F1@k Score είναι ο αρμονικός μέσος του Precision@k και Recall@k, παρέχοντας μια συνολική εικόνα της απόδοσης του retrieval μηχανισμού.

$$\text{F1@k} = 2 \times \frac{\text{Precision@k} \times \text{Recall@k}}{\text{Precision@k} + \text{Recall@k}}$$

Το F1@k είναι χρήσιμο όταν θέλουμε να ισορροπήσουμε την ακρίβεια και την πληρότητα του συστήματος.

6.8.1.2 Δημιουργία dataset αξιολόγησης

Η αξιολόγηση της ποιότητας των απαντήσεων του ψηφιακού βοηθού απαιτεί την ύπαρξη ενός αξιόπιστου συνόλου δεδομένων αναφοράς. Ωστόσο, λόγω της εξειδικευμένης φύσης των εγγράφων που χρησιμοποιούμε, δεν υπάρχει κάποιο έτοιμο dataset αξιολόγησης. Τα έγγραφα αυτά αποτελούνται κυρίως από νομοθετικά κείμενα, όπως νόμους που έχουν δημοσιευθεί στην Εφημερίδα της Κυβερνήσεως (ΦΕΚ), υπουργικές αποφάσεις και εγκυκλίους. Αυτά τα κείμενα παρουσιάζουν ιδιαίτερες προκλήσεις όσον αφορά τη δομή, τη γλώσσα και την πολυπλοκότητα

της πληροφορίας που περιέχουν, γεγονός που καθιστά δύσκολο τον έλεγχο της ακρίβειας και της συνάφειας των απαντήσεων του συστήματος.

Για την αντιμετώπιση αυτού του προβλήματος, δημιουργήθηκε ένα προσαρμοσμένο σύνολο δεδομένων αξιολόγησης. Η διαδικασία αυτή περιλάμβανε τη χρήση του μοντέλου GPT-4o-mini για τη δημιουργία ερωτήσεων με βάση τα επιμέρους τμήματα (chunks) των εγγράφων. Συγκεκριμένα, το μοντέλο ανέλαβε την ανάλυση των κειμένων και την αυτόματη παραγωγή ερωτήσεων που αντικατοπτρίζουν το περιεχόμενό τους.

Η αξιολόγηση του συστήματος πραγματοποιείται υποβάλλοντας αυτές τις ερωτήσεις στον ψηφιακό βοηθό και συγκρίνοντας τα αποτελέσματα της αναζήτησης με τα αναμενόμενα, δηλαδή με τα chunks από τα οποία δημιουργήθηκαν οι ερωτήσεις. Με αυτόν τον τρόπο, διαμορφώνουμε ένα αντικειμενικό και επαναλήψιμο πλαίσιο αξιολόγησης, που επιτρέπει τη συνεχή βελτίωση της απόδοσης του συστήματος και τη διασφάλιση της αξιοπιστίας του στις παρεχόμενες πληροφορίες.

6.8.1.3 Προβλήματα και παραδοχές

Η διαδικασία αξιολόγησης που περιγράφεται σε αυτή την ενότητα εστιάζει αποκλειστικά στην απόδοση του συστήματος ανάκτησης κειμενικών αποσπασμάτων (chunks) και όχι στην τελική απάντηση που παράγει το Γλωσσικό Μοντέλο (LLM). Συνεπώς, η μέτρηση της απόδοσης αφορά μόνο την ποιότητα της ανάκτησης των σχετικών πληροφοριών και όχι την ακρίβεια ή την πληρότητα της τελικής απόκρισης του chatbot.

Το dataset αξιολόγησης αποτελείται από ερωτήσεις όπου κάθε ερώτηση έχει αντιστοιχιστεί με ένα συγκεκριμένο chunk. Αυτό σημαίνει ότι για έναν προκαθορισμένο αριθμό k επιστρεφόμενων chunks (επιλέχθηκε k=5), το μέγιστο Precision@k που μπορεί να επιτευχθεί είναι 1/5 = 0.2. Το Recall@k, από την άλλη, λαμβάνει δυαδικές τιμές: είτε 0, αν το σχετικό chunk δεν περιλαμβάνεται στα αποτελέσματα, είτε 1, αν περιλαμβάνεται. Κατά συνέπεια, η μέγιστη δυνατή τιμή για τη μετρική F1-score σε αυτό το πλαίσιο είναι περίπου 0.33.

Στην πράξη, το Recall@k αποτελεί την πιο κρίσιμη μετρική, καθώς το LLM έχει τη δυνατότητα να συνθέσει μια ορθή απάντηση εφόσον του παρέχεται το κατάλληλο chunk στο context. Με άλλα λόγια, ακόμα και αν από τα k επιστρεφόμενα chunks μόνο ένα περιέχει τη σχετική πληροφορία, το LLM μπορεί να παράξει τη σωστή απάντηση βασιζόμενο αποκλειστικά σε αυτό. Επιπλέον, υπάρχουν περιπτώσεις όπου περισσότερα από ένα chunks περιέχουν σχετικές πληροφορίες. Σε αυτές τις περιπτώσεις, το LLM αξιοποιεί όλα τα διαθέσιμα σχετικά chunks για τη σύνθεση της τελικής απάντησης. Ωστόσο, το παρόν σύστημα αξιολόγησης δεν λαμβάνει υπόψη την επίδραση αυτής της πληροφορίας στη συνολική ποιότητα της απόκρισης, καθώς εστιάζει αποκλειστικά στην ακρίβεια της διαδικασίας ανάκτησης.

6.8.1.4 Διαδικασία αξιολόγησης

Αρχικά, για κάθε chunk παράγουμε αυτόματα ένα σύνολο περίπου 10 ερωτήσεων χρησιμοποιώντας το LLM (ο αριθμός των ερωτήσεων εξαρτάται από το LLM). Για λόγους

αποδοτικότητας, τόσο σε επίπεδο υπολογιστικών πόρων όσο και κόστους, επιλέγουμε μία από αυτές τις ερωτήσεις ως αντιπροσωπευτική για το συγκεκριμένο chunk. Στη συνέχεια, δημιουργούμε τα embeddings των επιλεγμένων ερωτήσεων, τα οποία χρησιμοποιούνται για την αναζήτηση στο σύστημα ανάκτησης του chatbot.

Για την αξιολόγηση της αποτελεσματικότητας του συστήματος, ελέγχουμε αν το αρχικό chunk, από το οποίο προήλθε η ερώτηση, περιλαμβάνεται στα αποτελέσματα της αναζήτησης μεταξύ των κορυφαίων k=5 επιστρεφόμενων chunks. Αυτή η διαδικασία επαναλαμβάνεται για όλες τις ερωτήσεις του dataset.

Τέλος, συγκεντρώνουμε όλα τα αποτελέσματα και υπολογίζουμε τις βασικές μετρικές αξιολόγησης: Recall@k, Precision@k και F1@k, οι οποίες αποτυπώνουν την ακρίβεια και την πληρότητα της διαδικασίας ανάκτησης πληροφοριών.

6.8.1.5 Αποτελέσματα

Παρακάτω παρουσιάζουμε τα αποτελέσματα της αξιολόγησης του συστήματος ανάκτησης πληροφοριών, όπως προέκυψαν από τη διαδικασία που περιγράφηκε προηγουμένως. Ο πίνακας που ακολουθεί περιλαμβάνει τις μετρικές Precision@k, Recall@k και F1@k, οι οποίες αποτυπώνουν την απόδοση του συστήματος ως προς την ικανότητά του να ανακτά τα σχετικά κειμενικά αποσπάσματα για κάθε ερώτηση.

Recall@k	0.6629
Precision@k	0.1325
F1 Score@k	0.2209

Τα αποτελέσματα της αξιολόγησης του συστήματος ανάκτησης πληροφοριών δείχνουν τα εξής:

- **Recall@k = 0.6629**: Το σύστημα καταφέρνει να εντοπίσει το σχετικό chunk σε ποσοστό περίπου **66.3%** των περιπτώσεων μεταξύ των κορυφαίων **k=5** αποτελεσμάτων. Αυτό υποδηλώνει ότι σε δύο στις τρεις ερωτήσεις το σύστημα επιστρέφει τουλάχιστον ένα σχετικό chunk, κάτι που είναι κρίσιμο για την παραγωγή των κατάλληλων απαντήσεων.
- **Precision@k = 0.1325**: Μόνο το **13.3%** των ανακτηθέντων chunks είναι σχετικά με την ερώτηση, από το σύνολο των chunks. Αυτό σημαίνει ότι η πλειονότητα των επιστρεφόμενων chunks δεν περιέχει την επιθυμητή πληροφορία, γεγονός που υποδηλώνει ότι υπάρχει περιθώριο βελτίωσης ως προς την ακρίβεια του συστήματος ανάκτησης. Παρόλα αυτά, το αποτέλεσμα αυτό είναι αναμενόμενο καθώς κάθε ερώτηση αντιστοιχεί σε ένα μόνο chunk. Επίσης, δεδομένου ότι το LLM μπορεί να εξάγει τη σωστή απάντηση από ένα και μόνο σχετικό chunk, η χαμηλή ακρίβεια δεν επηρεάζει απαραίτητα την τελική απόδοση του chatbot.
- **F1 Score@k = 0.2209**: Ο συνδυασμός Recall και Precision οδηγεί σε ένα F1-score της τάξης του **22.1%**, το οποίο είναι αναμενόμενο, δεδομένων των προηγούμενων τιμών.

Ενώ το Recall είναι αρκετά υψηλό, η χαμηλή ακρίβεια περιορίζει τη συνολική επίδοση της ανάκτησης.

6.8.1.6 Συμπέρασμα

Το σύστημα ανάκτησης έχει ικανοποιητική απόδοση όσον αφορά την ανάκληση (Recall), κάτι που είναι ιδιαίτερα σημαντικό για την ορθή λειτουργία του LLM. Ωστόσο, η χαμηλή ακρίβεια δείχνει ότι αρκετά από τα επιστρεφόμενα chunks είναι άσχετα, κάτι που οδηγεί σε σπατάλη του διαθέσιμου context του LLM.

Τέλος, για να έχουμε γνώση της πρακτικής απόδοσης του συστήματος θα πρέπει συλλέξουμε δεδομένα χρηστών, τα οποία θα κατηγοριοποιήσουμε χειροκίνητα και θα χρησιμοποιήσουμε για να πραγματοποιήσουμε την αξιολόγησή.

6.9 Ανάπτυξη Chunker/Document Processor για έγγραφα ΦΕΚ

Στο πλαίσιο της επεξεργασίας εγγράφων ΦΕΚ, αναπτύχθηκε και ένας εξειδικευμένος αλγόριθμος εξαγωγής κειμένου και τμηματοποίησης των εγγράφων.

6.9.1 Τεχνολογίες που χρησιμοποιήθηκαν:

- **pymupdf**: Για την ανάγνωση των αρχείων PDF και την εξαγωγή απλού κειμένου.
- **stanza**: Βιβλιοθήκη NLP από το Stanford. Χρησιμοποιήθηκε για τη διάσπαση του κειμένου σε προτάσεις. Το συγκεκριμένο εργαλείο επιλέχθηκε καθώς προσέφερε τα καλύτερα αποτελέσματα συγκριτικά με άλλες βιβλιοθήκες που δοκιμάστηκαν, όπως NLTK, SpaCy κ.α.
- **Embeddings μοντέλα**: Για τη σύγκριση των προτάσεων και τη δημιουργία chunks με τη semantic chunking τεχνική που περιγράφηκε στην προηγούμενη ενότητα.

6.9.2 Περιγραφή του αλγορίθμου

Ο αλγόριθμος είναι ένα υβρίδιο μεταξύ recursive chunker και semantic chunker και ακολουθεί τα εξής βήματα:

1. Εξαγωγή του κειμένου ανά σελίδα από το PDF.
2. Αφαίρεση κενών, περιττών χαρακτήρων και μη χρήσιμων συμβόλων.
3. Αφαίρεση κεφαλίδων και επαναλαμβανόμενων πληροφοριών (όπως τίτλοι, αριθμοί σελίδων).
4. Συνενοποίηση του κειμένου που διασπάται σε διαφορετικές σελίδες, ώστε να διατηρείται η συνοχή των πληροφοριών.
5. Τμηματοποίηση του κειμένου ανά πρόταση χρησιμοποιώντας το stanza.

6. Δημιουργία embeddings ανά πρόταση.
7. Συνένωση προτάσεων με βάση τη σημασιολογική εγγύτητα των embeddings, δημιουργώντας νοηματικά συνεκτικά chunks.

6.9.3 Πλεονεκτήματα του αλγορίθμου

- **Βελτιστοποιημένη τμηματοποίηση:** Δημιουργεί μεγάλα chunks, τα οποία καλύπτουν όλο το νοηματικά σχετικό περιεχόμενο του κειμένου.
- **Καλύτερη απόδοση σε chatbot:** Ένα chatbot που βασίζεται σε αυτά τα embeddings θα είχε πολύ ανώτερη ακρίβεια και συνοχή σε σχέση με άλλες τεχνικές chunking.
- **Ακριβής εντοπισμός πληροφοριών:** Κάθε chunk διατηρεί αναφορά στη θέση του στο αρχικό κείμενο, ακόμα και σε περιπτώσεις όπου το περιεχόμενό του εκτείνεται σε πολλαπλές σελίδες.

6.9.4 Μειονεκτήματα και περιορισμοί

- **Υψηλό κόστος σε επεξεργαστικούς πόρους:** Σε σύγκριση με απλούστερους αλγορίθμους, η παραγωγή των chunks απαιτεί μεγάλο υπολογιστικό χρόνο.
- **Αυξημένο κόστος σε tokens:** Για κάθε πρόταση απαιτείται διπλή επεξεργασία embeddings—μία για την πρόταση και μία για το τελικό chunk.
- **Περιορισμός σε αρχεία PDF:** Ο αλγόριθμος δεν έχει υλοποιηθεί για άλλους τύπους αρχείων (π.χ. Word, HTML).

6.9.5 Συμπέρασμα

Παρότι ο chunker παρήγαγε υψηλής ποιότητας τμηματοποίηση κειμένου, το υπολογιστικό κόστος και η αυξημένη κατανάλωση tokens οδήγησαν στην απόρριψή του για το τελικό chatbot. Αντί αυτού, προτιμήθηκαν απλούστερες τεχνικές chunking, οι οποίες απαιτούσαν λιγότερους πόρους.

Παρόλα αυτά κρίνοντας από την αξιολόγηση του τελικού chatbot, η χρήση αυτού του αλγορίθμου θα είχε ως αποτέλεσμα λίγοτερα τελικά chunks, τα οποία όμως θα κάλυπταν πληρέστερα ολόκληρη την πληροφορία του κειμένου. Αυτό θα οδηγούσε στην παραγωγή καλύτερων ερωτήσεων αξιολόγησης και επίσης θα αύξανε και την Precision@k μετρική, καταλήγοντας εν τέλη σε έναν καλύτερο ψηφιακό βοηθό.

6.10 Τεχνικές απαιτήσεις

6.10.1 Μοντέλα ML που χρησιμοποιήθηκαν

Τα μοντέλα μηχανικής μάθησης που χρησιμοποιήσαμε σε αυτή την εργασία, δηλαδή GPT 4o-mini για παραγωγή των απαντήσεων και text-embeddings-3-large για την δημιουργία των ενσωματώσεων κειμένου, χρειάζονται πολλούς υπολογιστικούς πόρους και η εκτέλεσή τους τοπικά σε έναν κοινό υπολογιστή είναι αδύνατη. Για αυτό το λόγο χρησιμοποιήθηκε το API της OpenAI μέσω του οποίου αποκτήθηκε πρόσβαση στα μοντέλα αυτά. Χρησιμοποιώντας το API η

παραγωγή των embeddings και των απαντήσεων γίνονται με μία απλή κλήση HTTP και συνεπώς η υπολογιστική δύναμη που απαιτείται είναι μηδαμινή. Το API αυτό είναι επί πληρωμή και πρέπει να διατεθούν τουλάχιστον 5 αμερικανικά δολάρια για να επιτραπεί η χρήση του. Για την εργασία μας ξοδέψαμε συνολικά 13 δολάρια, 5 δολάρια για την αρχική παραγωγή του ευρετηρίου χρησιμοποιώντας το μοντέλο embeddings και τα υπόλοιπα χρησιμοποιήθηκαν για την διαδικασία αξιολόγησης.

Υπάρχει και η δυνατότητα χρήσης πιο μικρών μοντέλων, όμως θα πρέπει να γίνει αξιολόγηση της καταλληλότητας του μοντέλου για την εκάστοτε διεργασία που θέλουμε να πραγματοποιήσουμε. Για παράδειγμα ένα GenAI μοντέλο που θα μπορούσε να χρησιμοποιηθεί τοπικά για την υλοποίηση του ψηφιακού βοηθού της εργασίας είναι το Llama-Krikri-8B-Base (https://huggingface.co/ilsp/Llama-Krikri-8B-Base) το οποίο έχει εκπαιδευτεί σε ελληνικά κείμενα συμπεριλαμβανομένης και της ελληνικής νομοθεσίας. Σε κάθε περίπτωση η εκτέλεση αυτών των μοντέλων τοπικά περιορίζεται από την διαθέσιμη μνήμη και την ύπαρξη ισχυρού επεξεργαστή ή κάρτας γραφικών.

Ακόμα, μπορούν να χρησιμοποιηθούν και υπηρεσίες όπως το Google Collab ή το Kaggle, τα οποία παρέχουν δωρεάν πρόσβαση σε υπολογιστικούς πόρους και επιτρέπουν την εκτέλεση κώδικα Python μέσα από Jupyter Notebooks. Οι υπηρεσίες αυτές προσφέρουν και τη δυνατότητα επέκτασης των διαθέσιμων πόρων με σχετικά μικρό κόστος.

6.10.2 Αποθήκευση των embeddings και μηχανισμός αναζήτησης

Το συνολικό μέγεθος του index των embeddings είναι 740 MB. Συνολικά είχαμε 26170 chunks, τα οποία μετατράπηκαν σε embeddings με το μοντέλο text-embedding-3-large. Χρησιμοποιώντας το μοντέλο text-embedding-3-small, το κόστος θα ήταν πάνω 10 φορές μικρότερο και ο απαιτούμενος αποθηκευτικός χώρος θα μειωνόταν στο μισό.

Όσον αφορά τη διαδικασία αναζήτησης χρησιμοποιήθηκε k-nn brute force αλγόριθμός, ο οποίος ελέγχει και ταξινομεί ολόκληρη τη λίστα με τα embeddings για κάθε ερώτημα. Το θετικό αυτού του αλγορίθμου είναι ότι έχει μεγάλη ανάκληση (recall = 1) κατα την αναζήτηση των chunks, όμως έχει μεγάλο υπολογιστικό κόστος τόσο σε χρόνο εκτέλεσης όσο και σε κατανάλωση μνήμης. Χρησιμοποιώντας ένα από τα συστήματα και τις βιβλιοθήκες που αναφέραμε στην ενότητα 4.3.2 Vector Databases θα κάναμε πολύ πιο αποδοτική την αναζήτηση των embeddings με μία πολύ μικρή απώλεια στην ανάκληση των αποτελεσμάτων.

6.10.3 Διαδικασία Αξιολόγησης

Η διαδικασία της αξιολόγησης ήταν το πιο απαιτητικό κομμάτι της εργασίας καθώς απαιτούσε την παραγωγή ερωτήσεων μέσω LLM για κάθε ένα chunk και στην συνέχεια την μετατροπή των ερωτήσεων σε embeddings και την εκτέλεση της τελικής αναζήτησης.

Για κάθε chunk το LLM παράγει κατά μέσο όρο 10-15 ερωτήσεις. Στην τελική αξιολόγηση χρησιμοποιήθηκε μόνο μία ερώτηση ανά chunk. Παρόλα αυτά δημιουργήθηκαν embeddings για όλες τις ερωτήσεις, ώστε να μπορεί να γίνει extra πειραματισμός. Στο τέλος της διαδικασίας τα

embeddings χρησιμοποίησαν περίπου 7 GB αποθηκευτικού χώρου. Κατά τη διάρκεια της παραγωγής τους το πρόγραμμα χρησιμοποιούσε περίπου 10 GB μνήμης για την επεξεργασία των ερωτήσεων για 5000 chunks. Καθώς δεν ήταν εφικτή η επεξεργασία ολόκληρου του συνόλου αξιολόγησης, των 26170 chunks, απευθείας, λόγω περιορισμού στη μνήμη του υπολογιστή που χρησιμοποιήθηκε, η επεξεργασία έγινε σε τμήματα των 5000 chunks.

7. Επίλογος

Η αξιοποίηση της OpenAI και της Generative AI τεχνολογίας, όπως το ChatGPT, έχει αναδείξει νέες δυνατότητες στην αλληλεπίδραση ανθρώπου-μηχανής. Η ενσωμάτωση προηγμένων γλωσσικών μοντέλων (LLMs) σε chatbots επιτρέπει τη βελτίωση της επικοινωνίας, την αυτοματοποίηση διαδικασιών και την παροχή εξατομικευμένων απαντήσεων με υψηλή ταχύτητα και ακρίβεια. Η χρήση εργαλείων όπως το Retrieval-Augmented Generation (RAG) επιτρέπει την παροχή επικαιροποιημένων και τεκμηριωμένων απαντήσεων, μειώνοντας τα σφάλματα που μπορεί να προκύψουν από αποκλειστικά προεκπαιδευμένα μοντέλα.

7.1 Μειονεκτήματα και κίνδυνοι

Ωστόσο, η υλοποίηση τέτοιων συστημάτων απαιτεί προσεκτικό σχεδιασμό, καθώς και συνεχή αξιολόγηση της απόδοσής τους.

Ένα από τα βασικά προβλήματα των GPT μοντέλων είναι η αδυναμία τους να προσφέρουν σε πραγματικό χρόνο ενημερωμένες πληροφορίες. Δεδομένου ότι εκπαιδεύονται σε στατικά σύνολα δεδομένων, δεν γνωρίζουν πρόσφατα γεγονότα ή αλλαγές στη νομοθεσία. Αυτό καθιστά τα RAG συστήματα απαραίτητα, ώστε τα μοντέλα να αντλούν πληροφορίες από αξιόπιστες πηγές. Παράλληλα, η ανάγκη περιορισμού του διαθέσιμου context window απαιτεί αποτελεσματικούς μηχανισμούς ανάκτησης των σχετικών αποσπασμάτων.

Επιπλέον, τα μοντέλα GPT μπορεί να παράγουν λανθασμένες πληροφορίες (hallucinations), ειδικά όταν δεν έχουν πρόσβαση σε σχετικό περιεχόμενο. Αυτή η αδυναμία καθιστά κρίσιμη την ανάπτυξη μεθόδων για την επαλήθευση της πληροφορίας που παράγεται από τα chatbots, προκειμένου να αυξηθεί η αξιοπιστία τους. Παράλληλα, τα μοντέλα παρουσιάζουν ευαισθησία στη διατύπωση των ερωτήσεων (prompt sensitivity), κάτι που σημαίνει ότι διαφορετικές διατυπώσεις μπορεί να οδηγήσουν σε διαφορετικές απαντήσεις.

Η χρήση των chatbots, ιδιαίτερα σε περιβάλλοντα που αφορούν τον δημόσιο τομέα, απαιτεί αυστηρές προδιαγραφές ασφάλειας και προστασίας προσωπικών δεδομένων. Καθώς αυτά τα συστήματα αλληλεπιδρούν με πολίτες και διαχειρίζονται ευαίσθητες πληροφορίες, η διασφάλιση της ακεραιότητας, της εμπιστευτικότητας και της διαθεσιμότητας των δεδομένων είναι ζωτικής σημασίας.

7.2 Πλεονεκτήματα και χρησιμότητα

Η εξέλιξη της τεχνητής νοημοσύνης και ειδικότερα των Generative AI τεχνολογιών, όπως το ChatGPT της OpenAI, αναδιαμορφώνει τον τρόπο με τον οποίο αλληλεπιδρούμε με τα συστήματα πληροφορικής. Η ικανότητα αυτών των μοντέλων να κατανοούν και να παράγουν φυσική γλώσσα δημιουργεί νέες δυνατότητες για αυτοματοποίηση, βελτίωση της επικοινωνίας και παροχή πληροφοριών σε πραγματικό χρόνο.

Στον δημόσιο τομέα, η ενσωμάτωση προηγμένων chatbot μπορεί να προσφέρει μεγαλύτερη διαφάνεια, αποτελεσματικότητα και προσβασιμότητα στις υπηρεσίες που παρέχονται στους πολίτες. Η χρήση αυτών των τεχνολογιών σε φορείς όπως δήμοι, υπουργεία, νομοθετικές υπηρεσίες και οργανισμούς δημόσιας διοίκησης μπορεί να οδηγήσει σε αυτοματοποιημένη και άμεση ενημέρωση, μειώνοντας τον διοικητικό φόρτο και εξοικονομώντας πόρους.

Οφέλη από την υιοθέτηση Generative AI στα δημόσια chatbots

- **Αναβάθμιση της εξυπηρέτησης πολιτών**: Τα chatbots μπορούν να προσφέρουν προσωποποιημένες απαντήσεις, μειώνοντας τον χρόνο αναμονής και βελτιώνοντας την εμπειρία των χρηστών.
- **Αποτελεσματικότητα & εξοικονόμηση πόρων**: Η διαχείριση πολλαπλών αιτημάτων ταυτόχρονα μειώνει την ανάγκη για ανθρώπινη παρέμβαση, απελευθερώνοντας πόρους για πιο σύνθετα καθήκοντα.
- **Διαρκής διαθεσιμότητα**: Σε αντίθεση με τους ανθρώπινους υπαλλήλους, τα chatbots λειτουργούν 24/7, εξασφαλίζοντας άμεση πρόσβαση σε πληροφορίες και υπηρεσίες.
- **Τεχνολογική ανάπτυξη και καινοτομία**: Η υιοθέτηση AI-driven συστημάτων προωθεί την ψηφιακή πρόοδο και ενθαρρύνει τη χρήση καινοτόμων λύσεων σε όλο το εύρος της δημόσιας διοίκησης.

7.3 Ανάγκη για περαιτέρω ανάλυση

Σε μελλοντική εργασία θα μπορούσε να αναλυθεί η χρήση Agents και Agentic AI στα chatbots, άλλες τεχνικές αξιολόγησης όπως το RAGAS, αλλά και διαφορετικού είδους LLM.

7.3.1 Ανάπτυξη και εξερεύνηση Agents

Οι agents, ως αυτοματοποιημένες μονάδες που αναλαμβάνουν να ολοκληρώσουν σύνθετα καθήκοντα, μπορούν να ενσωματωθούν με τα υπάρχοντα chatbots για να παρέχουν υπηρεσίες που ξεπερνούν τα όρια της απλής απάντησης σε ερωτήματα. Η περαιτέρω ανάλυση της δυναμικής τους και η αξιολόγηση της αποτελεσματικότητάς τους θα είναι σημαντική για τη βελτίωση των λειτουργιών των chatbots στον δημόσιο τομέα.

7.3.2 Εναλλακτικές Τεχνικές Αξιολόγησης

Η εξέταση άλλων μετρικών και μεθόδων αξιολόγησης, οι οποίες λαμβάνουν υπόψη τη συνολική απόδοση του chatbot και όχι μόνο την ικανότητά του να ανακτά το σωστό κείμενο, μπορεί να προσφέρει μια πληρέστερη εικόνα για την αποτελεσματικότητα των συστημάτων που βασίζονται σε LLMs και RAG.
Η ανάπτυξη νέων μεθόδων αξιολόγησης που να εξετάζουν την ποιότητα του διαλόγου, τη σημασιολογική ακεραιότητα των απαντήσεων και τη συνεκτικότητα των συνομιλιών μπορεί να βελτιώσει σημαντικά τις επιδόσεις των συστημάτων.

7.3.3 Εξέταση και άλλων μοντέλων

Εκτός από τα μοντέλα της OpenAI, υπάρχουν επίσης μοντέλα από άλλες εταιρείες όπως το Gemini της Google, το Llama της Meta, το Claude της Anthropic, κ.α. Επιπλέον, υπάρχουν και πολλά μοντέλα ανοικτού κώδικα, όπως το Deepseek ή το Llama-Krikri, το οποίο έχει εκπαιδευτεί σε ελληνικό περιεχόμενο. Η συγκριτική αξιολόγηση αυτών των μοντέλων και η εξέταση της καταλληλότητάς τους για χρήση σε ψηφιακούς βοηθούς στον δημόσιο και ιδιωτικό τομέα θα μπορούσε να παρέχει πολύτιμα ευρήματα για την περαιτέρω βελτίωση των συστημάτων εξυπηρέτησης, δίνοντας προτεραιότητα σε μοντέλα που ανταποκρίνονται καλύτερα στις ανάγκες των χρηστών σε διάφορους τομείς.

Βιβλιογραφία

"Πελάτες & Συνεργάτες." n.d. Botakis. https://ai.botakis.com/partners.

"ΣΧΕΤΙΚΑ ΜΕ ΤΟ ΕΡΓΟ." n.d. The Pythia Project. https://thepythiaproject.com/schetika-me-to-ergo-about-the-project/.

Αλεξόπουλος, Χάρης. n.d. "Η τεχνολογία chatbot στην Ελλάδα," Εκδήλωση παρουσίασης του έργου ΠΥΘΙΑ, thepithyaproject.com με θέμα "Η τεχνολογία chatbot στην Ελλάδα". https://docs.google.com/presentation/d/1o4PtX9QctpDRc0kATZkRIsaP6Yf6sXsL/edit#slide=id.p1.

Androutsopoulou, Aggeliki, Nikos Karacapilidis, Euripidis Loukis, and Yannis Charalabidis. 2019. "Transforming the communication between citizens and government through AI-guided chatbots." *Government Information Quarterly* 36, no. 2 (April): 358-367. https://doi.org/10.1016/j.giq.2018.10.001.

"API platform." n.d. OpenAI. Accessed October 6, 2024. https://openai.com/api/.

Bouras, Christos, Damianos Diasakos, Chrysostomos Katsigiannis, Vasileios Kokkinos, Apostolos Gkamas, Nikos Karacapilidis, Yannis Charalabidis, et al. n.d. "On the Development of a Novel Chatbot Generator Architecture: Design and Assessment Issues." https://telematics.upatras.gr/wp-content/uploads/2024/06/MCSI2023_Bouras-2.pdf.

"CHATBOT | English meaning - Cambridge Dictionary." n.d. Cambridge Dictionary. Accessed October 5, 2024. https://dictionary.cambridge.org/dictionary/english/chatbot.

Floridi, Luciano, and Massimo Chiriatti. 2020. "GPT-3: Its Nature, Scope, Limits, and Consequences." *Minds and Machines* 30:681–694. https://link.springer.com/article/10.1007/s11023-020-09548-1.

"GPT-4o mini: advancing cost-efficient intelligence." 2024. OpenAI. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

Griffith, Erin. 2024. “OpenAI Is Growing Fast and Burning Through Piles of Money.” *The New York Times*, September 27, 2024. https://www.nytimes.com/2024/09/27/technology/openai-chatgpt-investors-funding.html.

“Introducing ChatGPT.” 2022. OpenAI. https://openai.com/index/chatgpt/.

Izacard, Gautier, and Edouard Grave. 2020. “Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering.” https://arxiv.org/abs/2007.01282.

Kamradt, Greg. 2024. “5 Levels Of Text Splitting.” 5 Levels Of Text Splitting. https://github.com/FullStackRetrieval-com/RetrievalTutorials/blob/main/tutorials/LevelsOfTextSplitting/5_Levels_Of_Text_Splitting.ipynb.

Karpukhin, Vladimir, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. “Dense Passage Retrieval for Open-Domain Question Answering.” arXiv. https://arxiv.org/abs/2004.04906.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. “Deep learning.” *Nature* 521:436–444. https://www.nature.com/articles/nature14539.

Lewis, Patrick, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, et al. 2020. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.” https://arxiv.org/abs/2005.11401.

“Πολιτική χρήσης mAIgov.” n.d. Gov.gr. Accessed March 9, 2025. https://www.gov.gr/info/politiki-xrisis-maigov.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “[1301.3781] Efficient Estimation of Word Representations in Vector Space.” arXiv. https://arxiv.org/abs/1301.3781.

Mitchell, Tom M. 1997. *Machine Learning*. N.p.: McGraw-Hill Education.

“New embedding models and API updates.” 2024. OpenAI. https://openai.com/index/new-embedding-models-and-api-updates/.

Radford, Alec, and Karthik Narasimhan. 2018. “Improving Language Understanding by Generative Pre-Training.”

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. n.d. “Attention Is All You Need.” https://arxiv.org/abs/1706.03762.

Weizenbaum, Joseph. 01 January 1966. “ELIZA—a computer program for the study of natural language communication between man and machine.” *Communications of the ACM* Volume 9, (Issue 1): 36 - 45. https://dl.acm.org/doi/10.1145/365153.365168.

“What is a large language model (LLM)?” n.d. Cloudflare. Accessed October 6, 2024. https://www.cloudflare.com/learning/ai/what-is-large-language-model/.