ΔΗΜΟΚΡΙΤΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΡΑΚΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

AI Powered Phishing Campaign

Ιωάννα Κ. Φωστηροπούλου
Γαρυφαλλιά Ν. Κυριακίδου

Επιβλέπων: Κωνσταντίνος Ράντος
Τίτλος: Τακτικός Καθηγητής

ΚΑΒΑΛΑ

Σεπτέμβριος 2025

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Καμπάνιες Ηλεκτρονικού Ψαρέματος υποστηριζόμενες από Τεχνητή Νοημοσύνη

Ιωάννα Κ. Φωστηροπούλου
Α.Μ.: 210151
Γαρυφαλλιά Ν. Κυριακίδου
Α.Μ.: 210062

ΕΠΙΒΛΕΠΟΝΤΕΣ **Κωνσταντίνος Ράντος,** Τακτικός Καθηγητής

Copyright@ 2025 Τμήμα **Πληροφορικής, Δημοκρίτειο Πανεπιστήμιο Θράκης,**
https://www.cs.duth.gr/

Το περιεχόμενο της συγκεκριμένης Πτυχιακής Εργασίας αποτελεί πνευματική ιδιοκτησία του/των συγγραφέα/ων, του/της επιβλέποντα/ουσας καθηγητή/τριας και του Τμήματος Πληροφορικής του ΔΠΘ και προστατεύεται από το νόμο περί πνευματικής ιδιοκτησίας (Νόμος 2121/1993 και κανόνες Διεθνούς Δικαίου που ισχύουν στην Ελλάδα).

ΔΗΛΩΣΗ ΤΗΡΗΣΗΣ ΑΚΑΔΗΜΑΪΚΗΣ ΔΕΟΝΤΟΛΟΓΙΑΣ

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνουμε ενυπόγραφα ότι είμαστε αποκλειστικοί συγγραφείς της παρούσας Πτυχιακής Εργασίας, για την ολοκλήρωση της οποίας κάθε βοήθεια είναι πλήρως αναγνωρισμένη και αναφέρεται με λεπτομέρεια στην πτυχιακή εργασία. Έχουμε αναφέρει πλήρως και με σαφείς αναφορές όλες τις πηγές χρήσης δεδομένων, απόψεων, θέσεων και προτάσεων, ιδεών και λεκτικών αναφορών, είτε κατά κυριολεξία, είτε βάσει επιστημονικής παράφρασης. Αναλαμβάνουμε την προσωπική και ατομική ευθύνη ότι σε περίπτωση αποτυχίας στην υλοποίηση των ανωτέρω δηλωθέντων στοιχείων, είμαστε υπόλογοι έναντι λογοκλοπής, γεγονός που σημαίνει αποτυχία στην Πτυχιακή μας Εργασία και κατά συνέπεια αποτυχία απόκτησης Τίτλου Σπουδών, πέραν των λοιπών συνεπειών του νόμου περί πνευματικών δικαιωμάτων. Δηλώνουμε, συνεπώς, ότι αυτή η Πτυχιακή Εργασία προετοιμάστηκε και ολοκληρώθηκε από εμάς προσωπικά και αποκλειστικά και ότι, αναλαμβάνουμε πλήρως όλες τις συνέπειες του νόμου στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μας ανήκει διότι είναι προϊόν λογοκλοπής άλλης πνευματικής ιδιοκτησίας.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να ευχαριστήσουμε θερμά τον επιβλέποντα καθηγητή μας, Κωνσταντίνο Ράντο, για την πολύτιμη καθοδήγηση, την επιστημονική του υποστήριξη και τη διαρκή διαθεσιμότητά του καθ' όλη τη διάρκεια εκπόνησης της πτυχιακής μας εργασίας. Η συμβολή του υπήρξε καθοριστική για την ολοκλήρωση της παρούσας μελέτης, τόσο σε επίπεδο κατεύθυνσης όσο και τεχνικής τεκμηρίωσης.

Στο ίδιο πλαίσιο ευγνωμοσύνης, θα θέλαμε να εκφράσουμε τις ευχαριστίες μας προς όλους τους διδάσκοντες του Τμήματος για τις γνώσεις, τις εμπειρίες και την υποστήριξη που μας προσέφεραν κατά τη διάρκεια των σπουδών μας.

Οφείλουμε, επίσης, ένα ιδιαίτερο ευχαριστώ σε όλους εκείνους που συνέβαλαν με οποιονδήποτε τρόπο, τεχνικά, πρακτικά ή ψυχολογικά στην πορεία και στην ολοκλήρωση της εργασίας αυτής.

Τέλος, ένα ειλικρινές και μεγάλο ευχαριστώ στους γονείς μας και στους ανθρώπους του στενού μας περιβάλλοντος για την αδιάκοπη ηθική τους στήριξη, την υπομονή και την πίστη τους σε εμάς καθ' όλη τη διάρκεια των σπουδών μας.

ΠΕΡΙΛΗΨΗ

Η ταχεία εξέλιξη της τεχνητής νοημοσύνης και των μεγάλων γλωσσικών μοντέλων (LLMs) έχει επηρεάσει σημαντικά τον τομέα της κυβερνοασφάλειας, τόσο ως προς τις δυνατότητες πρόληψης όσο και ως προς τη δημιουργία συνθετικών σεναρίων επίθεσης για σκοπούς εκπαίδευσης και αξιολόγησης. Η παρούσα εργασία εστιάζει στην ανάπτυξη και αξιολόγηση δύο διαφορετικών μεθόδων για την αυτόματη παραγωγή και αποστολή ηλεκτρονικών μηνυμάτων τύπου "ηλεκτρονικού ψαρέματος" (phishing), αξιοποιώντας εργαλεία τεχνητής νοημοσύνης χαμηλού κόστους και περιορισμένων υπολογιστικών πόρων.

Η υλοποίηση αφορά δύο διαφορετικές μεθόδους, η μια υποστηριζόμενη από το FastAPI, ενώ η δεύτερη φιλοξενείται σε τοπικό υπολογιστικό περιβάλλον. Παρατίθεται, επίσης, μία συγκριτική ανάλυση των δύο μεθόδων, συνοδευόμενη με τα μειονεκτήματα και πλεονεκτήματα που τις διέπουν, έτσι ώστε, ανάλογα με τις ανάγκες και απαιτήσεις του κάθε χρήστη, να επιλεγεί η αντίστοιχη μεθοδολογία. Παράλληλα, το μεγάλο γλωσσικό μοντέλο που αξιοποιείται στις μεθόδους παραμένει το ίδιο, έχει εκπαιδευτεί εκ νέου μέσω του πλαισίου Unsloth, με χρήση συνθετικού συνόλου δεδομένων που έχει δημιουργηθεί από την ερευνητική ομάδα.

Κεντρική επιδίωξη της εργασίας αποτελεί η διερεύνηση της δυνατότητας δημιουργίας ρεαλιστικών και λειτουργικών σεναρίων phishing με την υποστήριξη τεχνητής νοημοσύνης και ελάχιστους υπολογιστικούς πόρους, προκειμένου να αξιοποιηθούν σε περιβάλλοντα ηθικής πειρατείας (ethical hacking), προγράμματα ευαισθητοποίησης προσωπικού και δοκιμές red team. Η εργασία ολοκληρώνεται με τη διατύπωση προτάσεων για μελλοντικές επεκτάσεις, όπως η ενίσχυση του dataset, η υποστήριξη επιπλέον γλωσσών, καθώς και η ενσωμάτωση γραφικής διεπαφής χρήστη για ευκολότερη πρόσβαση και χρήση της εφαρμογής.

Η εργασία καταλήγει στα συμπεράσματά της, αναδεικνύοντας τη δυνατότητα αξιοποίησης των μεγάλων γλωσσικών μοντέλων και των σύγχρονων ψηφιακών εργαλείων στην προσομοίωση επιθέσεων ηλεκτρονικού ψαρέματος με τρόπο ρεαλιστικό, ευέλικτο και ελεγχόμενο. Οι τεχνολογίες αυτές μπορούν να συμβάλουν ουσιαστικά στην ενίσχυση της εκπαίδευσης σε θέματα κυβερνοασφάλειας, υπό την προϋπόθεση της εφαρμογής τους σε ηθικά και ελεγχόμενα περιβάλλοντα.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Τεχνητή Νοημοσύνη, Κυβερνοασφάλεια, Τεχνολογίες Λογισμικού

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: phishing, τεχνητή νοημοσύνη, LLM, FastAPI, κυβερνοασφάλεια

ABSTRACT

The rapid development of artificial intelligence and Large Language Models (LLMs) has significantly impacted the cybersecurity field, both in terms of prevention capabilities and in terms of the creation of synthetic attack scenarios for training and evaluation purposes. This paper focuses on the development and evaluation of two different methods for the automatic generation of phishing emails, utilizing low-cost artificial intelligence tools and limited computing resources.

The implementation was done through two different methods, one supported by FastAPI, while the second is hosted in a local computing environment. A comparative analysis of the two methods is also provided, accompanied by the disadvantages and advantages that characterize them, so that, depending on the needs and requirements of each user, the corresponding methodology can be selected. At the same time, the large language model utilized in the methods remains the same, having been fine-tuned through the Unsloth framework, using a synthetic dataset created by the research team.

The main goal of the work is to investigate the possibility of creating realistic and functional phishing scenarios with the support of artificial intelligence and minimal computing resources, in order to be utilized in ethical hacking environments, staff awareness programs and red team tests. The work is completed by formulating proposals for future extensions, such as enhancing the dataset, supporting additional languages, as well as integrating a graphical user interface for easier access and use of the application.

The work concludes by highlighting the possibility of utilizing large language models and modern digital tools in simulating phishing attacks in a realistic, flexible and controllable manner. These technologies can significantly contribute to enhancing cybersecurity education, provided they are implemented in ethical and controlled environments.

SUBJECT AREA: Artificial Intelligence, Cybersecurity, Software Technologies

KEYWORDS: phishing, artificial intelligence, LLM, FastAPI, cybersecurity

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ **17**

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ **20**

1ο ΚΕΦΑΛΑΙΟ: ΕΙΣΑΓΩΓΗ **22**

1.1 Εισαγωγή στο Ηλεκτρονικό Ψάρεμα 23

1.2 Μηνύματα Ηλεκτρονικού Ψαρέματος 24

1.3 Τεχνητή Νοημοσύνη 25

1.4 Τεχνητή Νοημοσύνη στο Ηλεκτρονικό Ψάρεμα 27

1.5 Μεγάλα Γλωσσικά Μοντέλα 28

1.6 Σχετική Βιβλιογραφία 29

1.7 Θεωρητικές Έννοιες 31

1.8 Διάρθρωση της Πτυχιακής Εργασίας 31

2ο ΚΕΦΑΛΑΙΟ: ΘΕΩΡΗΤΙΚΕΣ ΕΝΝΟΙΕΣ **33**

2.1 Κοινωνική Μηχανική 33

2.2 Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks/RNNs) 40

2.3 Spambot 42

2.4 ChatGPT 43

2.5 PhEmail Tool 46

2.6 LLaMA 48

2.7 Unsloth 50

3ο ΚΕΦΑΛΑΙΟ: ΠΕΡΙΓΡΑΦΗ ΛΟΓΙΣΜΙΚΟΥ **51**

3.1 Διαδικασία εκπαίδευσης και επιλογή Μεγάλου Γλωσσικού Μοντέλου 51

3.2 Σετ δεδομένων 54

3.3 Γλώσσα προγραμματισμού που επιλέχθηκε 57

3.4 Υλοποίηση – Μέθοδος Πρώτη – FastAPI 58

3.4.1 Τι κάνει το λογισμικό 58

3.4.2 Είναι χαμηλού ή/και υψηλού επιπέδου; 60

3.4.3 Λειτουργικές/μη λειτουργικές απαιτήσεις και Προδιαγραφές 61

3.4.3.1 Λειτουργικές Απαιτήσεις 61

3.4.3.2 Μη Λειτουργικές Απαιτήσεις 63

3.4.3.3 Προδιαγραφές 64

3.4.4 Διαγράμματα UML 68

3.4.4.1 Διάγραμμα Περιπτώσεων Χρήσης 68

3.4.4.2 Διάγραμμα Κλάσεων 69

3.4.4.3 Διάγραμμα Κατάστασης 70

3.4.5 Τρόπος λειτουργίας με παραδείγματα 71

3.4.6 Παρουσίαση μικρών χαρακτηριστικών τμημάτων κώδικα και επεξήγηση της λειτουργίας τους 74

3.4.6.1 Ρύθμιση παραμέτρων και αρχικοποίηση περιβάλλοντος (Environment Configuration and Initialization) 74

3.4.6.2 Email Template Generator (create_html_email) 75

3.4.6.3 Επαλήθευση Κλειδιού API (API Key Verification) 75

3.4.6.4 Βοηθητικό πρόγραμμα επεξεργασίας κειμένου 75
3.4.6.5 /generate Endpoint (Ενσωμάτωση LLM με Ollama) 76
3.4.6.6 /send_email Endpoint (Email Transmission) 77
3.4.6.7 Command-Line User Interface (CLI) 79
3.4.7 Περιγραφή παραθύρων εφαρμογής, παρουσίαση εκτυπώσεων, πίνακες με καταστάσεις λειτουργίας. 81
3.4.7.1 Περιγραφή παραθύρων / διεπαφών εφαρμογής 81
3.4.7.2 Παρουσίαση Εκτυπώσεων 83
3.7.7.3 Πίνακες Καταστάσεων Εφαρμογής 83
3.5 Υλοποίηση – Μέθοδος Δεύτερη – Τοπικό Περιβάλλον 85
3.5.1 Τι κάνει το λογισμικό 85
3.5.2 Είναι χαμηλού ή/και υψηλού επιπέδου; 86
3.5.3 Λειτουργικές/μη λειτουργικές απαιτήσεις και Προδιαγραφές 86
3.5.3.1 Λειτουργικές Απαιτήσεις 87
3.5.3.2 Μη λειτουργικές Απαιτήσεις 88
3.5.3.3 Προδιαγραφές 89
3.5.4 Διαγράμματα UML 94
3.5.4.1 Διάγραμμα Περιπτώσεων Χρήσης 94
3.5.4.2 Διάγραμμα Κλάσεων 94
3.5.4.2 Διάγραμμα Κατάστασης 95
3.5.5 Τρόπος λειτουργίας με παραδείγματα 96
3.5.6 Παρουσίαση μικρών χαρακτηριστικών τμημάτων κώδικα και επεξήγηση της λειτουργίας τους 100
3.5.6.1 Βιβλιοθήκες και ρύθμιση του LLM 100
3.5.6.2 Διαπιστευτήρια και αρχικοποίηση SMTP Server 102
3.5.6.3 Έλεγχος εισόδων 102
3.5.6.4 Αποστολή του παραγόμενου μηνύματος ηλεκτρονικού ταχυδρομείου 104
3.5.7 Περιγραφή παραθύρων εφαρμογής, παρουσίαση εκτυπώσεων, πίνακες με καταστάσεις λειτουργίας 107
3.5.7.1 Περιγραφή παραθύρων / διεπαφών εφαρμογής 108
3.5.7.2 Παρουσίαση Εκτυπώσεων 109
3.5.7.3 Πίνακες Καταστάσεων Εφαρμογής 109
4ο ΚΕΦΑΛΑΙΟ: ΕΛΕΧΓΟΣ ΚΑΛΗΣ ΛΕΙΤΟΥΡΓΙΑΣ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ – ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ 111
4.1 Αποτελέσματα – Μέθοδος Πρώτη – FastAPI 111
4.1.1 Πλεονεκτήματα 111
4.1.2 Μειονεκτήματα 113
4.2 Αποτελέσματα – Μέθοδος Δεύτερη – Τοπικό Περιβάλλον 115
4.2.1 Πλεονεκτήματα 115
4.2.2 Μειονεκτήματα 116
4.3 Πειραματικά Αποτελέσματα 117
4.4 Σύγκριση των δύο μεθόδων 129
5ο ΚΕΦΑΛΑΙΟ: ΠΡΟΒΛΗΜΑΤΑ – ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΟΙ ΣΤΟΧΟΙ 131
5.1 Προβλήματα που αντιμετωπίσαμε 131
5.2 Αποτελέσματα 132

5.3 Συμπεράσματα 133
5.4 Μελλοντικοί Στόχοι 134
ΑΝΑΦΟΡΕΣ **136**
ΠΑΡΑΡΤΗΜΑ Ι **143**
ΠΑΡΑΡΤΗΜΑ ΙΙ **145**

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Στάδια επίθεσης κοινωνικής μηχανικής [52] 33
Σχήμα 2: Ταξινόμηση επιθέσεων κοινωνικής μηχανικής. 34
Σχήμα 3: Ταξινόμηση επιθέσεων κοινωνικής μηχανικής. 34
Σχήμα 4: Επιθέσεις κοινωνικής μηχανικής. 35
Σχήμα 5: Τύποι επιθέσεων αντίστροφης κοινωνικής μηχανικής. 36
Σχήμα 6: Ένα απλό επαναλαμβανόμενο νευρωνικό δίκτυο (RNN) και η αναπαράστασή του ως ξεδιπλωμένη ακολουθία στο χρόνο t. Τα βέλη αντιπροσωπεύουν πλήρως συνδεδεμένες μονάδες μεταξύ των επιπέδων. Για λόγους απλοποίησης, οι παράμετροι μεροληψίας δεν περιλαμβάνονται 37
Σχήμα 7: Απλή δομή RNN. 38
Σχήμα 8: Διάγραμμα της διαδικασίας υλοποίησης GPT. 41
Σχήμα 9: Διάγραμμα Περιπτώσεων Χρήσης - FastAPI 65
Σχήμα 10: Διάγραμμα κλάσεων - FastAPI 65
Σχήμα 11: Διάγραμμα Κατάστασης - FastAPI 66
Σχήμα 12: Διάγραμμα Περιπτώσεων Χρήσης - Τοπικό Περιβάλλον 90
Σχήμα 13: Διάγραμμα Κλάσεων - Τοπικό Περιβάλλον 91
Σχήμα 14: Διάγραμμα Κατάστασης - Τοπικό Περιβάλλον 92

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Διάγραμμα των βημάτων της μεθοδολογίας phishing. 23
Εικόνα 2: Η σχέση μεταξύ της Τεχνητής Νοημοσύνης, της Μηχανικής Μάθησης και της Βαθιάς Μάθησης 27
Εικόνα 3: Εντολή τερματικού 71
Εικόνα 4: Δεδομένα εισαγωγής 71
Εικόνα 5: Επιλογή αποστολής και HTML 72
Εικόνα 6: Μήνυμα επιβεβαίωσης αποστολής 72
Εικόνα 7: Παράδειγμα προτύπου HTML 72
Εικόνα 8: Παράδειγμα απλού κειμένου 73
Εικόνα 9: Παράδειγμα παραγόμενου αποτελέσματος 73
Εικόνα 10: Πεδία εισαγωγής στοιχείων 97
Εικόνα 11: Επεξηγηματικό μήνυμα προς τον χρήστη 97
Εικόνα 12: Εκτέλεση και αποστολή του μηνύματος 98
Εικόνα 13: Παράδειγμα προτύπου HTML 98
Εικόνα 14: Παράδειγμα απλού κειμένου 99
Εικόνα 15: Παράδειγμα παραγόμενου αποτελέσματος 100
Εικόνα 16: Στατιστικά από το Linkly 100
Εικόνα 17: Μη επιθυμητό αποτέλεσμα από το μοντέλο 132

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Βιβλιοθήκες FastAPI 56
Πίνακας 2: 3.4.3.3.1 Προδιαγραφή 60
Πίνακας 3: 3.4.3.3.2 Προδιαγραφή 61
Πίνακας 4: 3.4.3.3.3 Προδιαγραφή 62
Πίνακας 5: 3.4.3.3.4 Προδιαγραφή 63
Πίνακας 6: 3.4.3.3.5 Προδιαγραφή 64
Πίνακας 7: Παρουσίαση Εκτυπώσεων 79
Πίνακας 8: Διαχείριση κλειδιού API 79
Πίνακας 9: Καταστάσεις επεξεργασίας μοντέλου 79
Πίνακας 10: Καταστάσεις αποστολής ηλεκτρονικού μηνύματος ταχυδρομείου 80
Πίνακας 11: Καταστάσεις σφάλαμτος και εξαιρέσεις 80
Πίνακας 12: Βιβλιοθήκες Τοπικού Περιβάλλοντος 82
Πίνακας 13: 3.5.3.3.1 Προδιαγραφή 85
Πίνακας 14: 3.5.3.3.2 Προδιαγραφή 86
Πίνακας 15: 3.5.3.3.3 Προδιαγραφή 86
Πίνακας 16: 3.5.3.3.4 Προδιαγραφή 87
Πίνακας 17: 3.5.3.3.5 Προδιαγραφή 88
Πίνακας 18: 3.5.3.3.6 Προδιαγραφή 88
Πίνακας 19: 3.5.3.3.7 Προδιαγραφή 89
Πίνακας 20: Παρουσίαση Εκτυπώσεων 105
Πίνακας 21: Καταστάσεις επεξεργασίας μοντέλου 106
Πίνακας 22: Καταστάσεις αποστολής μηνύματος ηλεκτρονικού ταχυδρομείου 106
Πίνακας 23: Καταστάσεις σφάλματος και εξαιρέσεις 106
Πίνακας 24: Παραδείγματα αποτελεσμάτων με διαφορετικές προτροπές 115
Πίνακας 25: Σύγκριση μεθόδων 125

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος όρος	**Ελληνικός Όρος**
Phishing	Ηλεκτρονικό Ψάρεμα
Social Engineering	Κοινωνική Μηχανική
Vishing	Επίθεση πραγματοποιείται μέσω τηλεφωνικών κλήσεων
Smishing	Απάτη μέσω μηνυμάτων SMS
Email	Ηλεκτρονικό Ταχυδρομείο
Short Message Service	Σύντομο Μήνυμα
Phisher	Υποκλοπέας Ηλεκτρονικών Στοιχείων
Phishing Email	Μήνυμα Ηλεκτρονικού Ψαρέματος
Authority	Αυθεντία
Social Proof	Κοινωνική Απόδειξη
Liking and Familiarity	Αρέσκεια και Οικειότητα
Distraction	Αποδιοργάνωση
Commitment and Reciprocation	Δέσμευση και Ανταπόδοση
Spear Phishing	Στοχευμένο μήνυμα ηλεκτρονικού ψαρέματος
Artificial Intelligence	Τεχνητή Νοημοσύνη
Big Data	Μεγάλα Δεδομένα
Computer Program	Πρόγραμμα Υπολογιστή
General Problem Solver Program	Πρόγραμμα Γενικής Επίλυσης Προβλημάτων
Natural Language Processing	Επεξεργασίας φυσικής γλώσσας
Cybersecurity	Κυβερνοασφάλεια
Machine Learning	Μηχανική Μάθηση
Cyberspace	Κυβερνοχώρος
Deep Learning	Βαθιά Μάθηση
Spam Email	Ανεπιθύμητη Αλληλογραφία
Large Language Models	Μεγάλα Γλωσσικά Μοντέλα
Natural Language Processing	Επεξεργασία Φυσικής Γλώσσας
Cybercriminal	Κυβερνοεγκληματίας
Neural Network	Νευρωνικό Δίκτυο
Token	Διακριτικό

End-Of-Sequence	Τέλος της Ακολουθίας
Language Model	Γλωσσικό Μοντέλο
Jailbreak Prompt	Ειδικές Εντολές
Recurrent Neural Network	Επαναλαμβανόμενο νευρωνικό δίκτυο
Operation	Λειτουργία
Framework	Πλαίσιο
Cyberattack	Κυβερνοεπίθεση
Phishing Email Campaign	Καμπάνια Ηλεκτρονικού Ψαρέματος
Personation	Πλαστοπροσωπία
U.S. Department of Justice	Υπουργείο Δικαιοσύνης των Ηνωμένων Πολιτειών
Social security numbers	Αριθμοί κοινωνικής ασφάλισης
Remote Access Trojan	Κακόβουλο λογισμικό απομακρυσμένης πρόσβασης τύπου Trojan
Social engineering toolkit	Εργαλειοθήκη κοινωνικής μηχανικής
Baiting	Δόλωμα
Shoulder surfing	Σέρφινγκ ώμου
Robocalls	Αυτοματοποιημένες κλήσεις
Ransomware	Τύπος κακόβουλου λογισμικού που κρυπτογραφεί τα προσωπικά δεδομένα του θύματος μέχρι να καταβληθούν λύτρα.
Reverse social engineering)	Αντίστροφη κοινωνική μηχανική
Artificial Neural Networks	Τεχνητά Νευρωνικά Δίκτυα
Short-term memory	Βραχυπρόθεσμη μνήμη
Spambot	Bot που στέλνει ανεπιθύμητα email
Spammers	Αποστολείς ανεπιθύμητης αλληλογραφίας
Chat Generative Pre-trained Transformer	Chatbot τεχνητής νοημοσύνης που αναπτύχθηκε από την OpenAI και παράγει απαντήσεις κειμένου που μοιάζουν με ανθρώπινες με βάση τα δεδομένα που λαμβάνει.
Proximal Policy Optimization	Εγγύς Βελτιστοποίηση Πολιτικής
Spam	Ανεπιθύμητη Ηλεκτρονική Αλληλογραφία
Simple Mail Transfer Protocol	Πρωτόκολλο επικοινωνίας Διαδικτύου για τη μετάδοση ηλεκτρονικού ταχυδρομείου.
Fine-tuning	Βελτιστοποίηση
Retrieval-Augmented Generation	Ανάκτηση-Επαυξημένη Δημιουργία

Quantized Low-Rank Adapter Fine-Tuning	Βελτιστοποίηση προσαρμογέα κβαντισμένου χαμηλού βαθμού
Adapter modules	Μονάδες προσαρμογέα
Training dataset	Σύνολο δεδομένων εκπαίδευσης
Paired examples/prompts	Ζευγαρωμένα παραδείγματα/προτροπές
Supervised learning	Εποπτευόμενη μάθηση
Cross-entropy loss	Απώλεια διασταυρούμενης εντροπίας
Adapter parameters	Παράμετροι προσαρμογέα
Dataset	Σύνολο δεδομένων
Instruction Mode	Λειτουργία διδασκαλίας
Supervised instruction finetuning	Εποπτευόμενη βελτιστοποίηση διδασκαλίας
Email credentials	Διαπιστευτήρια email
Rapid prototyping	Ταχεία πρωτοτυποποίηση
Preview	Προεπισκόπηση μηνύματος
Link	Σύνδεσμος
Ethical hacking	Ηθική πειρατεία
Terminal	Τερματικό
Inputs	Εισροές
Generated	Παραγόμενο
Regeneration	Αναδημιουργία
User Interface	Διεπαφή χρήστη
Plain Text	Απλό κείμενο
Unauthorized	Μη Εξουσιοδοτημένο
Validation	Νομιμοποίηση
Input Collection	Συλλογή Εισόδου
Path	Μονοπάτι
Third-party	Εξωτερικός
ENISA	Ευρωπαϊκή Υπηρεσία Κυβερνοασφάλειας

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

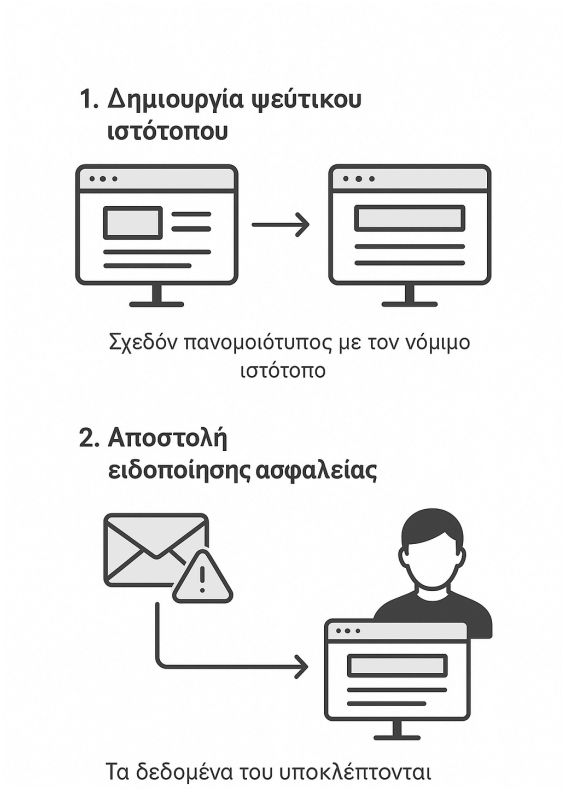
SMS	Short Message Service
AOL	America Online
URL	Uniform Resource Locator
AI	Artificial Intelligence
LLM	Large Language Model
NLP	Natural Language Processing
EOS	End-Of-Sequence
LM	Language Model
ΔΠΘ	Δημοκρίτειο Πανεπιστήμιο Θράκης
RNN	Recurrent Neural Network
SSN	Social security numbers
RAT	Remote Access Trojan
SET	Social engineering toolkit
ANN	Artificial Neural Networks
STM	Short-term memory
CAPTCHA	Completely Automated Public Turing Test to Tell Computers and Humans Apart
ChatGPT	Chat Generative Pre-trained Transformer
GPT	Γενετικά Προεκπαιδευμένος Μετασχηματιστής
PPO	Proximal Policy Optimization
SMTP	Simple Mail Transfer Protocol
RAG	Retrieval-Augmented Generation
QLoRA	Quantized Low-Rank Adapter Fine-Tuning
LoRA	Low-Rank Adaptation
UI	User Interface
HTML	Hypertext Markup Language
CLI	Command-Line User Interface

1ο ΚΕΦΑΛΑΙΟ: ΕΙΣΑΓΩΓΗ

Η κοινωνική μηχανική (Social Engineering) αποτελεί τον όρο που περιγράφει ένα ευρύ φάσμα κακόβουλων δραστηριοτήτων, οι οποίες υλοποιούνται μέσω μίας ή περισσότερων διαδοχικών φάσεων και βασίζονται κυρίως σε ανθρώπινες αλληλεπιδράσεις, συχνά με τη διαμεσολάβηση του Διαδικτύου. Πρόκειται για μία «τέχνη» ψυχολογικής χειραγώγησης των ατόμων, με σκοπό να εξαναγκαστούν να αποκαλύψουν ευαίσθητες και εμπιστευτικές πληροφορίες, όπως κωδικούς πρόσβασης, τραπεζικά στοιχεία ή να παραχωρήσουν πρόσβαση στους υπολογιστές τους. Μία τέτοια παρέμβαση ενδέχεται να έχει ως αποτέλεσμα την εγκατάσταση κακόβουλου λογισμικού, το οποίο επιτρέπει στον δράστη να αποκτήσει πλήρη έλεγχο του συστήματος του θύματος.

Οι τακτικές της κοινωνικής μηχανικής εκμεταλλεύονται συχνά τη φυσική τάση των ανθρώπων να επιδεικνύουν εμπιστοσύνη, καθιστώντας αυτήν την προσέγγιση πιο απλή και άμεση σε σύγκριση με την εξεύρεση τεχνικών μέσων για την παραβίαση ενός λογαριασμού χρήστη ή ενός λογισμικού ασφαλείας. Οι κυβερνοεγκληματίες προτιμούν αυτήν τη μέθοδο, καθώς είναι σαφώς ευκολότερο να εξαπατήσουν κάποιον ώστε να αποκαλύψει τον κωδικό πρόσβασής του, παρά να τον παραβιάσουν με τεχνικά μέσα. Οι επιθέσεις κοινωνικής μηχανικής διακρίνονται σε τέσσερις κύριες μορφές, καθεμία από τις οποίες αξιοποιεί διαφορετικά μέσα επικοινωνίας για την παραπλάνηση των θυμάτων. Το Vishing συνίσταται στη χρήση τηλεφωνικών κλήσεων, μέσω των οποίων οι επιτιθέμενοι αποσπούν εμπιστευτικές πληροφορίες προσποιούμενοι νόμιμους φορείς. Το Phishing αφορά τη διανομή παραπλανητικών ηλεκτρονικών μηνυμάτων με σκοπό τη συλλογή προσωπικών δεδομένων ή την εγκατάσταση κακόβουλου λογισμικού. Το Smishing λειτουργεί αντίστοιχα μέσω μηνυμάτων SMS, εκμεταλλευόμενο την αμεσότητα της κινητής επικοινωνίας. Τέλος, το Personation περιλαμβάνει φυσική αλληλεπίδραση, κατά την οποία ο δράστης υποδύεται άλλο πρόσωπο για να αποσπάσει πληροφορίες ή πρόσβαση. Και στις τέσσερις περιπτώσεις, η κοινή στρατηγική είναι η χειραγώγηση της ανθρώπινης εμπιστοσύνης με σκοπό την παράνομη απόκτηση δεδομένων ή πόρων.

Μεταξύ αυτών, το phishing συνιστά την πιο συνήθη μορφή επίθεσης κοινωνικής μηχανικής, πραγματοποιείται κατά κύριο λόγο μέσω μηνυμάτων ηλεκτρονικού ταχυδρομείου και θεωρείται μία από τις πιο συχνά χρησιμοποιούμενες και επιτυχημένες μεθόδους κυβερνοεπιθέσεων, κυρίως λόγω της απλότητας και της αποτελεσματικότητάς της. Βασίζεται στη δημιουργία ψευδών ιστοσελίδων ή ηλεκτρονικών επικοινωνιών που μιμούνται με ακρίβεια την εμφάνιση και τη λειτουργικότητα αξιόπιστων ιστοτόπων ή οργανισμών. Οι επιτιθέμενοι χρησιμοποιούν πλαστές διευθύνσεις ηλεκτρονικού ταχυδρομείου για την αποστολή παραπλανητικών μηνυμάτων, με στόχο την υποκλοπή ευαίσθητων πληροφοριών, όπως διαπιστευτήρια πρόσβασης και δεδομένα χρηστών. Η μεθοδολογία του phishing βασίζεται συνήθως σε δύο βήματα, όπως φαίνεται και στην Εικόνα 1, πρώτον στη δημιουργία ενός ψεύτικου ιστότοπου, ο οποίος εμφανίζεται σχεδόν πανομοιότυπος με τον νόμιμο ιστότοπο που επισκέπτεται συχνά ο χρήστης-στόχος. Το δεύτερο βήμα αφορά την αποστολή ειδοποίησης ασφαλείας μέσω ηλεκτρονικού ταχυδρομείου, η οποία καλεί το θύμα να «κάνει κλικ στον παρακάτω σύνδεσμο» ή να εισαγάγει προσωπικά του στοιχεία, όπως το όνομα χρήστη και τον κωδικό πρόσβασης. Μόλις ο χρήστης μεταβεί στον κακόβουλο ιστότοπο και επιχειρήσει να συνδεθεί, τα δεδομένα του υποκλέπτονται.



Εικόνα 1: Διάγραμμα των βημάτων της μεθοδολογίας phishing.

Οι κυβερνοεγκληματίες είναι, κατά κανόνα, άρτια καταρτισμένοι τεχνικά και παραμένουν διαρκώς ενημερωμένοι για τις τελευταίες τεχνολογικές εξελίξεις. Ως αποτέλεσμα, διαθέτουν την ικανότητα να παρακάμπτουν μέτρα ασφαλείας και να εξαπατούν ακόμη και τους πιο προσεκτικούς χρήστες. Η αναγνώριση ψευδών ιστοτόπων καθίσταται ολοένα και δυσκολότερη, ακόμη και με τη χρήση αυστηρών μεθόδων ελέγχου ταυτότητας διακομιστή. Εντωμεταξύ, οι κυβερνοεγκληματίες αναπτύσσουν συνεχώς νέες τεχνικές και προσαρμόζουν τις επιθέσεις τους με βάση τις συνήθειες και τις αντιδράσεις των χρηστών, καθιστώντας τη διαδικτυακή ασφάλεια μία διαρκώς εξελισσόμενη και απαιτητική πρόκληση.

Οι επιθέσεις phishing εξελίσσονται συνεχώς λόγω των εγκληματιών του κυβερνοχώρου και της επιθυμίας τους να αναζητήσουν νέες και καινοτόμες μεθόδους για να εξαπατήσουν τα πιθανά θύματά τους. Ορισμένοι εισβολείς μπορεί να μην αισθάνονται την ανάγκη να αλλάξουν τις μεθόδους τους, απλώς να εφαρμόσουν νέες καινοτόμες μεθόδους. Μια από αυτές τις μεθόδους είναι η καινοτόμος εφαρμογή της Τεχνητής Νοημοσύνης στο phishing.

1.1 Εισαγωγή στο Ηλεκτρονικό Ψάρεμα

Στον 21ο αιώνα, το ηλεκτρονικό ψάρεμα (Phishing) αποτελεί ένα από τα πιο οργανωμένα και επικερδή εγκλήματα στον κυβερνοχώρο (Cyberspace) [1]. Η ταχεία ενσωμάτωση των τεχνολογικών εξελίξεων έχει οδηγήσει σε σημαντική αύξηση των επιθέσεων στην ασφάλεια των υπολογιστικών συστημάτων, ενισχύοντας τις ευπάθειες που μπορούν να αξιοποιηθούν από κακόβουλους παράγοντες [2].

Ο όρος phishing προέρχεται από μια αναλογία που παρομοιάζει τους επιτιθέμενους με «ψαράδες», οι οποίοι αξιοποιούν δόλια μέσα, όπως παραπλανητικά μηνύματα ηλεκτρονικού ταχυδρομείου, προκειμένου να αποκτήσουν πρόσβαση σε ευαίσθητα δεδομένα, αντλώντας τα από το ευρύ φάσμα των χρηστών του Διαδικτύου [3]. Μια πρώιμη τεχνική phishing καταγράφηκε και παρουσιάστηκε το 1987 σε έγγραφο της Διεθνούς Ομάδας Χρηστών της Hewlett-Packard, Interex [4]. Η λέξη «ψάρεμα» εμφανίστηκε για πρώτη φορά στο Διαδίκτυο το 1990, όταν μια ομάδα χάκερ χρησιμοποίησε την πρακτική αυτή για να υποκλέψει λογαριασμούς της πλατφόρμας America Online (AOL), εξαπατώντας ανυποψίαστους χρήστες ώστε να αποκαλύψουν τους κωδικούς πρόσβασής τους [5]. Οι δράστες ανέπτυξαν έναν αλγόριθμο ικανό να δημιουργεί τυχαίους αριθμούς πιστωτικών καρτών, οι οποίοι χρησιμοποιήθηκαν για τη διενέργεια επιθέσεων [4]. Οι τυχαία παραγόμενοι αριθμοί αξιοποιήθηκαν για το άνοιγμα λογαριασμών στην πλατφόρμα AOL, καθώς και για την αποστολή μαζικών ανεπιθύμητων μηνυμάτων σε πολλούς χρήστες [4]. Στις 2 Ιανουαρίου 1996, η ομάδα συζητήσεων του Usenet με την επωνυμία AOHell εισήγαγε τον όρο "phishing" για να περιγράψει τη συγκεκριμένη μέθοδο επίθεσης [4].

Γενικότερα, το phishing αναφέρεται ως μια μορφή κακόβουλης δραστηριότητας κατά την οποία οι επιτιθέμενοι αποστέλλουν παραπλανητικά μηνύματα ηλεκτρονικού ταχυδρομείου σε πιθανούς ανυποψίαστους παραλήπτες, με σκοπό την εξαπάτησή τους για την απόκτηση προσωπικών και ευαίσθητων πληροφοριών [6]. Ο επιτιθέμενος μπορεί ακόμη να προσποιηθεί ότι εκπροσωπεί έναν αξιόπιστο οργανισμό ή πρόσωπο για να παρακινήσει το θύμα να αποκαλύψει ευαίσθητες πληροφορίες, όπως κωδικούς πρόσβασης, αριθμούς πιστωτικών καρτών, τραπεζικά στοιχεία ή άλλα προσωπικά δεδομένα [1]. Πρόκειται για μια εξελιγμένη τεχνική εξαπάτησης, η οποία αξιοποιεί τεχνικές της κοινωνικής μηχανικής και σύγχρονα τεχνολογικά μέσα για να επιτύχει τον σκοπό της [1]. Η τεχνική της κοινωνικής μηχανικής αναλύεται περαιτέρω στην ενότητα 2.1. Καθώς ο δράστης εφαρμόζει τεχνικές κοινωνικής μηχανικής, η πρακτική του phishing καθίσταται περισσότερο αποτελεσματική, στοχεύοντας στην πειθώ των ατόμων να προβούν σε συγκεκριμένες ενέργειες, οι οποίες εξυπηρετούν τα συμφέροντα του επιτιθέμενου. Επιπροσθέτως, περιλαμβάνει τη διαδικασία επικοινωνίας με το εκάστοτε επιτιθέμενο άτομο, κυρίως μέσω ηλεκτρονικού ταχυδρομείου (email), σύντομων μηνυμάτων (Short Message Service/SMS) ή τηλεφωνικών κλήσεων [1].

Το phishing έχει εξελιχθεί σε μια ιδιαίτερα προσοδοφόρα δραστηριότητα, με τους phishers να αποκομίζουν εκατομμύρια δολάρια μέσω κλοπής προσωπικών δεδομένων και οικονομικών πόρων των θυμάτων [1]. Οι εγκληματικές ομάδες που διεξάγουν τέτοιου είδους επιθέσεις αυξάνονται ραγδαία, με τη δράση τους να εντοπίζεται κυρίως σε περιοχές όπως η Ανατολική Ευρώπη, η Ασία, η Αφρική και η Μέση Ανατολή [1]. Η ταχύτατη διάδοση και εξέλιξη των τεχνικών phishing καθιστούν επιτακτική την ανάγκη για διαρκή επαγρύπνηση και ενίσχυση των μηχανισμών κυβερνοασφάλειας.

1.2 Μηνύματα Ηλεκτρονικού Ψαρέματος

Σύμφωνα με τον Jason Hong, τα μηνύματα ηλεκτρονικού ψαρέματος (Phishing Emails) είναι ένα είδος επίθεσης κοινωνικής μηχανικής όπου ο επιτιθέμενος στέλνει ένα κακόβουλο μήνυμα ηλεκτρονικού ταχυδρομείου στο θύμα, μεταμφιεζόμενος ως μια αξιόπιστη οντότητα, με σκοπό να ξεγελάσει το θύμα και να μοιραστεί ευαίσθητες πληροφορίες, ή ακόμη να κατεβάσει κάποιο κακόβουλο λογισμικό [7].

Οι επιτιθέμενοι εκμεταλλεύονται τη ψυχολογία του ανθρώπου και συνήθως προσπαθούν να δημιουργήσουν συγκεκριμένα συναισθήματα, όπως ο φόβος, η απληστία, η περιέργεια, η εμπιστοσύνη, η απάθεια, με το πιο δημοφιλής να είναι αυτό της αίσθησης

επείγουσας ανάγκης [8]. Ο λόγος που συμβαίνει αυτό είναι έτσι ώστε το θύμα να δράσει γρήγορα, χωρίς να έχει χρόνο να σκεφτεί λογικά το αίτημα που του δίνεται [9].

Οι εγκληματίες τείνουν να χρησιμοποιούν διαφορετικούς τρόπους πειθούς και τεχνικές για να εξαπατήσουν τα θύματά τους. Αναλυτικότερα, ένας τρόπος είναι μέσω της αυθεντίας (Authority), μιας και οι άνθρωποι έχουν τη τάση να ακολουθούν τους υψηλότερους ιεραρχικά ανθρώπους, χωρίς να αναρωτηθούν για το αίτημα που τους δίνεται [8]. Άλλος τρόπος είναι αυτός της κοινωνικής απόδειξης (Social Proof), η οποία περιγράφει τη ψυχολογική τάση να υιοθετούν συμπεριφορές τρίτων, αφού η έμφυτη τάση των ανθρώπων να μιμούνται τους άλλους συντελεί στην απροσεξία τους [8]. Η αρέσκεια και η οικειότητα (Liking and Familiarity) αποτελεί μια ακόμη τεχνική, εφόσον οι άνθρωποι προτιμούν να ακολουθήσουν κάποιον που νοιάζονται ή γνωρίζουν, αντίστοιχα [8]. Άμα ο επιτιθέμενος μεταμφιεστεί ως ένα γνώριμο πρόσωπο, οι πιθανότητες επιτυχίας της επίθεσης θα είναι υψηλότερες. Επιπλέον, μπορεί να υλοποιηθεί και η τεχνική της αποδιοργάνωσης (Distraction) έτσι ώστε τα θύματα να εστιάσουν σε δευτερεύοντα στοιχεία, κάνοντάς τα να σκεφτούν άλογα [8]. Τέλος, η τεχνική της δέσμευσης και ανταπόδοσης (Commitment and Reciprocation) είναι κάτι που είναι πιθανό να αξιοποιηθεί από τους εγκληματίες [8].

Έχει παρατηρηθεί πως τα περισσότερα phishing emails στέλνονται σε εταιρικούς λογαριασμούς και όχι σε προσωπικούς [8]. Φαίνεται λοιπόν πως οι εγκληματίες βρίσκουν τους εταιρικούς λογαριασμούς πιο σημαντικούς και ενδιαφέρον. Το κύριο περιεχόμενο των emails αυτών αναφερόντουσαν σε ζητήματα δουλειάς, φιλανθρωπίας, κεφαλαίων και μεταφοράς χρημάτων, συνήθως επισυναπτόμενο μαζί με κάποιο κακόβουλο URL (Unifrom Line Locator) [8].

Υπάρχει, επίσης, και μια ειδική περίπτωση μηνυμάτων ηλεκτρονικού ψαρέματος, ονόματι στοχευμένα μηνύματα ηλεκτρονικού ψαρέματος (Spear Phishing Email) [7]. Η κύρια διαφορά των Spear Phishing Emails με τα Phishing Emails είναι πως το πρώτο είναι μια εξελιγμένη επίθεση στοχευμένη σε ένα συγκεκριμένο άτομο [7]. Οι επιτιθέμενοι συλλέγουν πληροφορίες για το θύμα που θέλουν να εκμεταλλευτούν και δημιουργούν ένα πιο προσωπικό μήνυμα, αυξάνοντας έτσι τις πιθανότητες επιτυχίας [7]. Μια υποκατηγορία του Spear Phishing είναι το Whaling, ή αλλιώς επιθέσεις φαλαινοθηρίας. Η συγκεκριμένη κατηγορία στοχέυει σε υψηλόβαθμα στελέχη, με εξατομικευμένα μηνύματα που αυξάνουν την επιτυχία και μπορεί να προκαλέσουν μεγάλες οικονομικές ζημιές [10]. Βέβαια, υπάρχει και μια επίθεση που αποσκοπεί το αντίθετο του Whaling, δηλαδή αντί να γίνει η επίθεση σε υψηλόβαθμα στελέχη, υποδύονται τον ρόλο τους, με απότερο σκοπό την εξαπάτηση των χαμηλόβαθμων στελεχών [10]. Αυτές οι επιθέσεις ονομάζονται Business Email Compromise και είναι ένας τρόπος εξαπάτησης χρηστών μέσω email, συνήθως σε επαγγελματικό περιβάλλον. Μια διαφορετική κατηγορία ηλεκτρονικού ψαρέματος που είναι πιθανό να εμφανιστεί μέσω μηνύματος ηλεκτρονικού ταχυδρομείου είναι το Pharming. Το Pharming είναι μια επίθεση που ανακατευθύνει χρήστες από νόμιμες ιστοσελίδες σε δόλιμες μέσω δηλητηρίασης DNS, κλέβοντας προσωπικά διαπιστευτήρια [10]. Αξίζει να σημειωθεί και η κατηγορία του Clone Phishing, η οποία βασίζεται στην αντιγραφή και κλωνοποίηση νόμιμων email, αντικαθιστόντας τους συνδέσμους ή συνημμένα με κακόβουλα, εξαπατώντας έτσι τα θύματα [10].

1.3 Τεχνητή Νοημοσύνη

Από την εποχή της Βιομηχανικής Επανάστασης, σημειώθηκε σημαντική πρόοδος στον τομέα της τεχνολογίας [11]. Πολλές απαιτητικές χειρωνακτικές εργασίες έχουν σταδιακά αντικατασταθεί από τεχνολογικές λύσεις, οι οποίες προσφέρουν ανεκτίμητη

υποστήριξη στην ανθρώπινη δραστηριότητα. Μία από τις πιο πρωτοποριακές εξελίξεις στον χώρο της τεχνολογίας είναι η Τεχνητή Νοημοσύνη (Artificial Intelligence/AI), η οποία έχει σχεδιαστεί για να αναλαμβάνει και να αυτοματοποιεί καθήκοντα που παραδοσιακά εκτελούνταν από ανθρώπινα χέρια, σε πολλούς και διαφορετικούς κλάδους [12]. Το 1956, κατά τη διάρκεια του Συνεδρίου Τεχνητής Νοημοσύνης που πραγματοποιήθηκε στο Dartmouth, η έννοια της τεχνητής νοημοσύνης ορίστηκε ως η δυνατότητα κάθε πτυχής της μάθησης ή οποιουδήποτε άλλου χαρακτηριστικού της νοημοσύνης να περιγράφεται με τέτοια ακρίβεια, ώστε να καθίσταται εφικτή η κατασκευή μηχανών που μπορούν να την προσομοιώνουν [13].

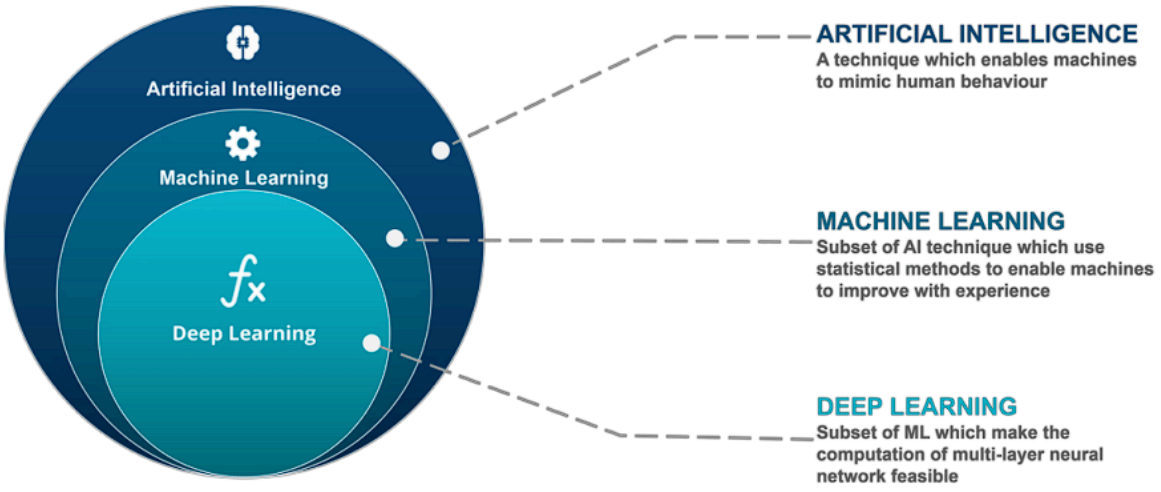
Η Τεχνητή Νοημοσύνη αποτελεί έναν διακριτό κλάδο της επιστήμης και της τεχνολογίας, ο οποίος ασχολείται με την ανάπτυξη ευφυών συστημάτων και λογισμικών ικανών να εκτελούν εργασίες που απαιτούν ανθρώπινη νοημοσύνη [11]. Πρόκειται για ένα σύστημα που προσομοιώνει τις ανθρώπινες γνωστικές λειτουργίες, αξιοποιώντας εξωτερικές πηγές δεδομένων, όπως τα μεγάλα δεδομένα (Big Data), για να επιτύχει υψηλή απόδοση σε συγκεκριμένα καθήκοντα [14]. Ενώ παλαιότερα η Τεχνητή Νοημοσύνη περιοριζόταν στη σφαίρα της επιστημονικής φαντασίας και στις θεωρητικές συζητήσεις για την επίδραση της τεχνολογίας στον σύγχρονο κόσμο, σήμερα έχει ενσωματωθεί πλήρως στην καθημερινότητά μας και αποτελεί πλέον θεμελιώδη λειτουργία σε πληθώρα τεχνολογιών και τομέων, επηρεάζοντας σημαντικά βιομηχανίες όπως η μεταποίηση, η υγειονομική περίθαλψη και οι αλυσίδες εφοδιασμού [14]. Η ικανότητα της Τεχνητής Νοημοσύνης να εκτελεί διεργασίες που υπερβαίνουν τις ανθρώπινες δυνατότητες έχει οδηγήσει σε καινοτόμες εφαρμογές, συμβάλλοντας στη βελτίωση της αποδοτικότητας και της παραγωγικότητας [14].

Η ιστορία της Τεχνητής Νοημοσύνης έχει τις ρίζες της σε ιδέες και φαντασιώσεις όπως προαναφέρθηκε, που διατυπώθηκαν από φιλοσόφους και συγγραφείς μυθοπλασίας [15]. Οι φιλόσοφοι εισήγαγαν την έννοια των ευφυών μηχανών ως μέσο κατανόησης της ουσίας του ανθρώπινου όντος. Ο Γάλλος φιλόσοφος Rene Descartes αξιοποίησε τη μεταφορά του «μηχανικού ανθρώπου» για να περιγράψει τη δυνατότητα ύπαρξης ευφυών μηχανών [14]. Παράλληλα, συγγραφείς επιστημονικής φαντασίας, όπως ο Jules Verne κατά τον 19ο αιώνα και ο Isaac Asimov στον 20ο αιώνα, ανέπτυξαν ανάλογες ιδέες, ενσωματώνοντάς τες στα έργα τους που εξερευνούν την έννοια της μη ανθρώπινης ευφυΐας [14]. Οι αφηγήσεις τους αποτέλεσαν πηγή έμπνευσης για πολλούς επιστήμονες και ερευνητές στον τομέα της Τεχνητής Νοημοσύνης, ενισχύοντας τη διαμόρφωση και εξέλιξη του πεδίου.

Το έργο DSRPAI αποτέλεσε ένα σημαντικό σημείο καμπής στον τομέα της τεχνητής νοημοσύνης [16]. Χαρακτηριστικό παράδειγμα αποτελεί το πρόγραμμα υπολογιστή (Computer Program) ELIZA, ένα πρωτοποριακό εργαλείο επεξεργασίας φυσικής γλώσσας (Natural Language Processing), το οποίο είχε τη δυνατότητα να συνομιλεί με ανθρώπους και αναπτύχθηκε από τον Joseph Weizenbaum. Ένα ακόμη αξιοσημείωτο παράδειγμα είναι το Πρόγραμμα Γενικής Επίλυσης Προβλημάτων (General Problem Solver Program), το οποίο δημιουργήθηκε από τον Herbert Simon (βραβευμένο με Νόμπελ), τον Cliff Shaw και τον Allen Newell, εξέχοντες επιστήμονες της εταιρείας RAND. Οι επιτυχίες αυτές οδήγησαν σε σημαντικές χρηματοδοτήσεις για την προώθηση της έρευνας στον τομέα της τεχνητής νοημοσύνης [17]. Από την αρχική εμφάνιση της τεχνητής νοημοσύνης, ο κλάδος έχει εξελιχθεί ριζικά με την πάροδο του χρόνου, ενώ έχει αναδειχθεί σε κρίσιμο παράγοντα για διάφορους τομείς και κυρίως στην κυβερνοασφάλεια (Cybersecurity).

Στον επιστημονικό κλάδο της τεχνητής νοημοσύνης, παρατηρείται η χρήση πολλών εξειδικευμένων όρων. Ειδικότερα, η τεχνητή νοημοσύνη αποτελεί τη γενικότερη κατηγορία, ενώ η μηχανική μάθηση (Machine Learning/ML) συνιστά ένα υποσύνολό της.

Επιπροσθέτως, η βαθιά μάθηση (Deep Learning) αποτελεί περαιτέρω εξειδίκευση της μηχανικής μάθησης, όπως αναλύεται στην Εικόνα 1 [3].



Εικόνα 2: Η σχέση μεταξύ της Τεχνητής Νοημοσύνης, της Μηχανικής Μάθησης και της Βαθιάς Μάθησης

Η ψηφιακή εξαπάτηση, η οποία εντείνεται από τις ραγδαίες εξελίξεις της Τεχνητής Νοημοσύνης, αποτελεί σοβαρή απειλή για τη σύγχρονη, διασυνδεδεμένη κοινωνία μας. Καθώς τα συστήματα τεχνητής νοημοσύνης αποκτούν ολοένα και μεγαλύτερες ικανότητες στη μίμηση των χαρακτηριστικών της ανθρώπινης επικοινωνίας και στη δημιουργία αισθήματος εμπιστοσύνης, διαμορφώνουν ένα νέο πεδίο για επιθέσεις κοινωνικής μηχανικής και ηλεκτρονικής εξαπάτησης [18].

1.4 Τεχνητή Νοημοσύνη στο Ηλεκτρονικό Ψάρεμα

Η Τεχνητή Νοημοσύνη στην Κυβερνοασφάλεια μπορεί να αξιοποιηθεί είτε για το κοινό καλό, είτε για κερδοσκοπικούς, κακόβουλους λόγους. Συγκεκριμένα, η Τεχνητή Νοημοσύνη στο Phishing ενδέχεται να χρησιμοποιείται για την αναγνώριση και τον εντοπισμό κακόβουλων ιστοσελίδων ή ψεύτικων μηνυμάτων ηλεκτρονικού ταχυδρομείου, όπως επίσης και στην δημιουργία τους.

Αναλυτικότερα σχετικά με τα ψεύτικα μηνύματα ηλεκτρονικού ταχυδρομείου, η Τεχνητή Νοημοσύνη έχει βοηθήσει στον αποδοτικότερο εντοπισμό ανεπιθύμητης αλληλογραφίας (Spam Emails) και μηνύματα ηλεκτρονικού ψαρέματος. Ο εντοπισμός αυτός γίνεται μέσω προκαθορισμένων σετ δεδομένων με παραδείγματα μηνυμάτων spam και phishing [19]. Εκτός από την ανίχνευση μηνυμάτων ηλεκτρονικού ψαρέματος, υπάρχει και η ανίχνευση ψεύτικων ιστοσελίδων. Για τον λόγο αυτό, αναπτύχθηκε το εργαλείο PhishHaven [20], το οποίο εντοπίζει σε πραγματικό χρόνο ψεύτικες ιστοσελίδες φτιαγμένες είτε μέσω Τεχνητής Νοημοσύνης, το οποίο αναλύεται περαιτέρω στη παρακάτω παράγραφο, είτε χειρωνακτικά από κακόβουλους χρήστες. Η ακρίβειά του ήταν εξαιρετικά μεγάλη [20].

Όμως, υπάρχουν και πολλοί χρήστες όπου θα χρησιμοποιήσουν τη Τεχνητή Νοημοσύνη κακόβουλα, με σκοπό να παραπλανήσουν τους αφελής ανθρώπους για να

κλέψουν ή να καταστρέψουν προσωπικά ευαίσθητα στοιχεία [19]. Οι αλγόριθμοι που αξιοποιούν οι κακόβουλοι δημιουργούν phishing διευθύνσεις URL με τέτοιο τρόπο έτσι ώστε να μην γίνονται ανιχνεύσιμοι από μοντέλα AI. Το αντίκτυπο είναι τόσο μεγάλο όπου κατά τη λειτουργία του DeepPhish [21], ενός εργαλείου σαν αυτό που περιγράφτηκε πιο πάνω, οχταπλασίασε την αποδοτικότητα των ψεύτικων ιστοσελίδων [21].

1.5 Μεγάλα Γλωσσικά Μοντέλα

Τα Μεγάλα Γλωσσικά Μοντέλα (Large Language Models/LLMs) είναι μια κατηγορία μοντέλων θεμελίωσης που εκπαιδεύονται σε τεράστιες ποσότητες δεδομένων που τα καθιστούν ικανά να κατανοούν και να δημιουργούν φυσική γλώσσα και άλλους τύπους περιεχομένου για να εκτελούν ένα ευρύ φάσμα εργασιών [22]. Είναι ένας τύπος εργαλείων Natural Language Processing (NLP) που εκπαιδεύονται με τη Μηχανική Μάθηση. Ενώ τα LLM επιδεικνύουν εντυπωσιακές τεχνολογικές δυνατότητες που μπορούν να αξιοποιηθούν για ευεργετικούς σκοπούς, η φύση διπλής χρήσης τους σημαίνει ότι μπορούν επίσης να χρησιμοποιηθούν κακόβουλα [23]. Επιπλέον, επιδεικνύουν αξιοσημείωτες ικανότητες στην αντίληψη, την επεξεργασία γνώσης και τη λήψη αποφάσεων στο πλαίσιο εφαρμογών τεχνητής νοημοσύνης, καθιστώντας τα εξαιρετικά πολύτιμα εργαλεία για την ενίσχυση της ασφάλειας στον κυβερνοχώρο [24].

Η τεράστια επιτυχία των LLM πηγάζει σε μεγάλο βαθμό από τη χρήση πολλαπλών επιπέδων νευρωνικών δικτύων (Neural Network) σε μαζική κλίμακα, το καθένα με παραμέτρους που μπορούν να ρυθμιστούν με ακρίβεια κατά τη διάρκεια της εκπαίδευσης και η οποία έχει οδηγήσει σε σημαντικά κέρδη απόδοσης. Αυτά τα μοντέλα χρησιμοποιούν συνήθως μια αρχιτεκτονική μετασχηματιστή που χρησιμοποιεί μηχανισμούς προσοχής για την κατανόηση των σχέσεων μεταξύ των λέξεων σε προτάσεις [25]. Καθώς η κλίμακα των νευρωνικών δικτύων επεκτείνεται, οι ερευνητές έχουν διαπιστώσει ότι τα μοντέλα βαθιάς μάθησης επιδεικνύουν προηγμένες δυνατότητες, οι οποίες υπερβαίνουν τις ικανότητες των παραδοσιακών μοντέλων. Οι εν λόγω δυνατότητες περιλαμβάνουν εξελιγμένο λογικό συλλογισμό, ανάλυση συναισθηματικής πληροφορίας, καθώς και περίπλοκη επεξεργασία του πλαισίου [26].

Ένα γλωσσικό μοντέλο, γενικά, κωδικοποιεί μια κατανομή πιθανοτήτων ανά συμβολοσειρά κειμένου. Η διαδικασία αυτή επιτυγχάνεται μέσω της ανάλυσης μιας συμβολοσειράς σε διακριτές μονάδες x_1, x_2, ..., x [27]. Η πιθανότητα που αποδίδεται σε αυτή τη συμβολοσειρά μπορεί να εκφραστεί ως εξής:

$$p(x1, \ldots, xn) = p(x1)p(x2|x1) \cdot \cdot \cdot p(xn|x1, \ldots, xn-1).$$

Αντί να υπολογίζει άμεσα την πιθανότητα της αριστερής πλευράς, το γλωσσικό μοντέλο δύναται να υπολογίσει κάθε επιμέρους όρο της δεξιάς πλευράς. Αξιοποιώντας αυτή τη δυνατότητα, τα γλωσσικά μοντέλα έχουν τη δυνατότητα να παράγουν κείμενο με την εξής διαδικασία: λαμβάνοντας ως είσοδο μία προτροπή (Prompt), το μοντέλο υπολογίζει την κατανομή πιθανότητας για το επόμενο διακριτικό (Token) και δημιουργεί ένα διακριτικό είτε μέσω δειγματοληψίας από την κατανομή αυτή είτε επιλέγοντας το πλέον πιθανό διακριτικό. Η διαδικασία αυτή επαναλαμβάνεται, με το μοντέλο να παράγει διαδοχικά διακριτικά, έως ότου δημιουργηθεί το διακριτικό που υποδηλώνει το τέλος της ακολουθίας (End-of-Sequence/EOS) [27].

Τα σύγχρονα γλωσσικά μοντέλα (Language Models/LM) υπολογίζουν την πιθανότητα μέσω της χρήσης νευρωνικών δικτύων [27]. Η αρχιτεκτονική του μετασχηματιστή [25] αποτελεί τη θεμελιώδη βάση σχεδόν όλων των γλωσσικών μοντέλων που

χρησιμοποιούνται ευρέως στη σημερινή εποχή. Τα τελευταία χρόνια, βελτιωμένα γλωσσικά μοντέλα, όπως αυτά που περιγράφονται στις πηγές [22], [28], [29], έχουν σημειώσει σημαντική πρόοδο, κυρίως χάρη στη χρήση μεγαλύτερων μοντέλων και στην αξιοποίηση μεγαλύτερου όγκου δεδομένων εκπαίδευσης. Τα τρέχοντα γλωσσικά μοντέλα διαθέτουν εκατοντάδες δισεκατομμύρια παραμέτρους. Αυτά τα εξαιρετικά μεγάλα μοντέλα συχνά αναφέρονται ως Μεγάλα Γλωσσικά Μοντέλα [27].

Η συνεχώς αυξανόμενη πολυπλοκότητα και ισχύς των Μεγάλων Γλωσσικών Μοντέλων τα καθιστά ικανά να αξιοποιηθούν όχι μόνο σε θετικές εφαρμογές, αλλά και σε κακόβουλα σενάρια. Ειδικότερα, μπορούν να χρησιμοποιηθούν για την επιτάχυνση και διεύρυνση κυβερνοεπιθέσεων που βασίζονται σε κείμενο, όπως οι στοχευμένες επιθέσεις τύπου spear phishing. Ιδιαίτερα ανησυχητικός είναι ο κίνδυνος που προκύπτει από την ικανότητά τους να παράγουν συνεκτικά και πειστικά κείμενα, γεγονός που επιτρέπει την εκμετάλλευσή τους από κυβερνοεγκληματίες για επιθέσεις μεγάλης κλίμακας. Επιπλέον, τα λεγόμενα «Frontier» γλωσσικά μοντέλα διαθέτουν τη δυνατότητα παραγωγής εκτενών και εξαιρετικά εξελιγμένων κειμένων σε ελάχιστο χρόνο. Ο συνδυασμός της ταχύτητας και της προηγμένης τεχνολογικής τους ικανότητας καθιστά εφικτή την πλήρη υποκατάσταση ανθρώπινων χειριστών σε εκστρατείες spear phishing μεγάλης κλίμακας [25].

1.6 Σχετική Βιβλιογραφία

Αυτή η ενότητα παρουσιάζει μια συνοπτική επισκόπηση των κύριων ερευνών που έχουν διεξαχθεί αναφορικά με την ανάπτυξη μοντέλων που παράγουν phishing μηνύματα ηλεκτρονικού ταχυδρομείου.

Ο Lepe Khanum [30] στην εργασία του, ανέπτυξε το εργαλείο “AutoPhish” ως απάντηση στον αυξημένο κίνδυνο επιθέσεων phishing που πλήττουν τις επιχειρήσεις και δύνανται να επιφέρουν σημαντικές οικονομικές ζημιές. Λαμβάνοντας υπόψη ότι αυτές οι επιθέσεις αξιοποιούν εγγενείς ανθρώπινες αδυναμίες, το AutoPhish παρέχει μια αυτοματοποιημένη λύση για την εκτέλεση δοκιμών ευαισθητοποίησης και την εκπαίδευση του προσωπικού. Ειδικότερα, το εργαλείο προσφέρει τη δυνατότητα δημιουργίας και αποστολής στοχευμένων phishing emails, αξιοποιώντας τεχνικές κοινωνικής μηχανικής για τη συλλογή σχετικών δεδομένων. Μέσω αυτής της διαδικασίας, οι επιχειρήσεις αποκτούν τη δυνατότητα να αξιολογούν την αποτελεσματικότητα των εκπαιδευτικών τους προγραμμάτων, να ενισχύουν τα μέτρα κυβερνοασφάλειας και να μειώνουν τον κίνδυνο επιτυχημένων επιθέσεων στο μέλλον. Πιο συγκεκριμένα, το μοντέλο περιγράφει μια αυτοματοποιημένη διαδικασία phishing, η οποία εμπλέκει τρεις βασικές οντότητες: τον δράστη (επιτιθέμενο), το εργαλείο και το θύμα. Ο επιτιθέμενος πραγματοποιεί συλλογή πληροφοριών σχετικά με τον στόχο μέσω τεχνικών κοινωνικής μηχανικής και χρήσης νόμιμων διευθύνσεων email αποστολέα. Οι πληροφορίες αυτές αξιοποιούνται για τη δημιουργία προτύπων phishing και την προετοιμασία μηνυμάτων ηλεκτρονικού ταχυδρομείου με παραπλανητικό περιεχόμενο. Το εργαλείο αναλαμβάνει την αυτοματοποίηση της σύνταξης και αποστολής αυτών των μηνυμάτων phishing. Εάν η επίθεση είναι επιτυχής, το θύμα λαμβάνει το email, το ανοίγει και ενδέχεται να αποκαλύψει ευαίσθητες πληροφορίες, όπως δεδομένα πιστωτικών καρτών ή προσωπικά στοιχεία επικοινωνίας. Το εργαλείο παρακολουθεί τις αλληλεπιδράσεις του θύματος με τα email, καταγράφει τα αποτελέσματα, αποθηκεύει τα συλλεγόμενα δεδομένα και αναλύει τα σχόλια για τη συνεχή βελτιστοποίηση της διαδικασίας. Ο απώτερος στόχος της διαδικασίας είναι η εξαπάτηση του θύματος με σκοπό την απόσπαση προσωπικών δεδομένων για κακόβουλους σκοπούς.

Το 2016 οι John Seymour και Philip Tully [31] δημιούργησαν ένα αναδρομικό νευρωνικό δίκτυο (Recurrent Neural Network) το οποίο μαθαίνει από phishing tweets στο Twitter - πλέον X - και συνθέτει ένα στοχευμένο tweet προς κάποιον χρήστη, δηλαδή είναι ένα εργαλείο spear phishing. Η επιλογή του Twitter και όχι κάποιας άλλης πλατφόρμας κοινωνικής δικτύωσης έγινε επειδή είναι πιο σύνηθες τα μικρά tweets με “σπασμένα” αγγλικά, όπως επίσης και η χρήση συντομευμένων URL, μιας και που τα tweets έχουν όριο στις λέξεις, κάνοντας πιο πιθανή την εξαπάτηση χρηστών.

Οι Ashfak Md Shibli et al. [32] ανέπτυξαν ένα μοντέλο το οποίο αναπαριστά μια διαδικασία κατά την οποία οι επιτιθέμενοι εκμεταλλεύονται τις συνομιλίες με συστήματα τεχνητής νοημοσύνης, όπως το ChatGPT ή το Bard, χρησιμοποιώντας ειδικές εντολές (Jailbreak Prompts) για να παρακάμψουν τους περιορισμούς που τους επιβάλλονται. Μόλις παρακάμψουν αυτούς τους περιορισμούς, δημιουργούν ψευδείς τομείς, συντάσσουν κείμενα που αφορούν την απάτη μέσω μηνυμάτων SMS και αποκτούν εργαλεία για τη διάπραξη εγκλημάτων στον κυβερνοχώρο. Αυτοί οι πόροι χρησιμοποιούνται για την κατασκευή ψευδών ιστοσελίδων και την εκπόνηση επιθέσεων μεγάλης κλίμακας μέσω σπασμωδικών smishing επιθέσεων. Οι επιθέσεις υλοποιούνται μέσω της αποστολής κακόβουλων μηνυμάτων και της συγκέντρωσης επαφών των θυμάτων για στοχευμένη διάδοση. Η καμπάνια εξελίσσεται με την προσαρμογή των στρατηγικών, βασισμένων σε ανατροφοδότηση και αποτελέσματα από συνεχείς λειτουργίες (operations).

Στη συνέχεια, οι Liu et al. [33] διερεύνησαν τις μεθόδους και την αποδοτικότητα της εφαρμογής προσεκτικά σχεδιασμένων προτροπών, με στόχο την άρση των περιορισμών που σχετίζονται με το jailbreak σε μεγάλα γλωσσικά μοντέλα όπως το GPT-3.5 TURBO και το GPT-4. Οι τεχνικές αυτές χρησιμοποιούνται για τη διαμόρφωση και τροποποίηση των ηθικών προτύπων που διέπουν τη λειτουργία των chatbots τεχνητής νοημοσύνης.

Μία άλλη μελέτη που διερεύνησε το ζήτημα του jailbreaking των Μεγάλων Γλωσσικών Μοντέλων από τους Deng et al. [34] ανέδειξε ότι συγκεκριμένες κατηγορίες προτροπών έχουν τη δυνατότητα να παρακάμπτουν τις προστασίες σε διάφορα απαγορευμένα σενάρια. Επιπλέον, η μελέτη παρουσίασε ένα πλαίσιο (framework), το Masterkey, που αποσκοπεί στην αυτοματοποίηση της διαδικασίας jailbreaking σε chatbots συστήματα τεχνητής νοημοσύνης.

Στην εργασία των Gupta et al. [35], αναλύθηκαν τα πλεονεκτήματα και οι περιορισμοί της χρήσης του ChatGPT ως εργαλείο για την εκτέλεση κυβερνοεπιθέσεων (Cyberattack), την άμυνα απέναντι σε απειλές στον κυβερνοχώρο, καθώς και οι νομικές και κοινωνικές επιπτώσεις που απορρέουν από τη χρήση του.

Σε διαφορετική έρευνα, οι Begou et al. [36] προσπάθησαν με επιτυχία να κλωνοποιήσουν συγκεκριμένους ιστοτόπους χρησιμοποιώντας το ChatGPT, ενσωματώνοντας κώδικα για κλοπή διαπιστευτηρίων και άλλες ενέργειες που σχετίζονται με το ηλεκτρονικό ψάρεμα.

Επιπλέον, μια πρόσφατη μελέτη των Langford et al. [37] ανέδειξε τη χρησιμότητα του ChatGPT στην ανάπτυξη καμπανιών ηλεκτρονικού ψαρέματος (phishing email campaign), επιβεβαιώνοντας τα ευρήματά μας και για την περίπτωση των καμπανιών phishing email.

Οι Shih-Wei Guo et al. [38] ανέπτυξαν ένα μοντέλο που δημιουργεί phishing email μηνύματα, το οποίο λειτουργεί συνδυάζοντας ένα μοντέλο γλώσσας με ένα στοιχείο εξαγωγής λέξεων-κλειδιών για τη δημιουργία μηνυμάτων ηλεκτρονικού ταχυδρομείου με βάση συγκεκριμένη εισαγωγή. Κατά τη διάρκεια της εκπαίδευσης, το μοντέλο λαμβάνει ένα θέμα ηλεκτρονικού ταχυδρομείου και ένα σύνολο λέξεων-κλειδιών που εξάγονται από το σώμα του. Αυτές οι λέξεις-κλειδιά, μαζί με το θέμα, καθοδηγούν τη δημιουργία του περιεχομένου email. Το μοντέλο γλώσσας χρησιμοποιεί αυτήν την είσοδο για να μάθει πώς να δομεί το email και να ενσωματώνει τις εξαγόμενες λέξεις-κλειδιά για να παράγει

συνεκτικό και σχετικό κείμενο. Ο στόχος είναι το μοντέλο να δημιουργήσει το σώμα του email με βάση το δεδομένο θέμα και τις λέξεις-κλειδιά, ελέγχοντας αποτελεσματικά τόσο τη δομή όσο και το περιεχόμενο του email.

Η παρούσα εργασία στοχεύει στη κάλυψη που υπάρχει ως προς την ανάπτυξη ενός εργαλείου για τη σύνταξη και αποστολή ηλεκτρονικών μηνυμάτων ψαρέματος μέσα σε ένα ελεγχόμενο περιβάλλον, με σκοπό τη βελτίωση της εκπαίδευσης των χρηστών σε θέματα ηλεκτρονικής απάτης και ασφάλειας στο διαδίκτυο.

1.7 Θεωρητικές Έννοιες

1.7.1 Κοινωνική Μηχανική

Η κοινωνική μηχανική αναφέρεται στις τεχνικές χειραγώγησης ανθρώπων ώστε να αποκαλύψουν ευαίσθητες πληροφορίες ή να εκτελέσουν ενέργειες που ωφελούν τους επιτιθέμενους. Βασίζεται στην ανθρώπινη ψυχολογία και όχι σε τεχνικά μέσα. Οι επιθέσεις κοινωνικής μηχανικής εκμεταλλεύονται στοιχεία όπως η εμπιστοσύνη, η αφέλεια και το αίσθημα του επείγοντος. Παραδείγματα περιλαμβάνουν το phishing, το vishing, το smishing και την πλαστοπροσωπία.

1.7.2 Ηλεκτρονικό Ψάρεμα

Το phishing αποτελεί μια μορφή ηλεκτρονικής απάτης κατά την οποία επιτιθέμενοι προσποιούνται αξιόπιστες πηγές για να αποσπάσουν προσωπικές πληροφορίες, όπως διαπιστευτήρια σύνδεσης και τραπεζικά δεδομένα. Εμφανίζεται σε διάφορες μορφές, όπως phishing emails, SMS phishing και τηλεφωνικές απάτες. Συνήθως περιλαμβάνει ψεύτικες ιστοσελίδες ή παραπλανητικά μηνύματα που παροτρύνουν το θύμα να αποκαλύψει κρίσιμες πληροφορίες.

1.7.3 Κυβερνοασφάλεια

Η κυβερνοασφάλεια αναφέρεται σε μέτρα και τεχνικές που προστατεύουν δίκτυα, συστήματα και δεδομένα από κυβερνοεπιθέσεις. Περιλαμβάνει την ανίχνευση και αποτροπή απειλών, την εκπαίδευση και ευαισθητοποίηση χρηστών, την κρυπτογράφηση δεδομένων και την ανάπτυξη ασφαλών συστημάτων. Η διαρκής εξέλιξη των επιθέσεων phishing και κοινωνικής μηχανικής καθιστά την κυβερνοασφάλεια κρίσιμο τομέα για επιχειρήσεις και άτομα.

1.7.4 Τεχνητή Νοημοσύνη

Η τεχνητή νοημοσύνη περιλαμβάνει συστήματα και αλγορίθμους που προσομοιώνουν ανθρώπινη νοημοσύνη για την ανάλυση δεδομένων και τη λήψη αποφάσεων. Έχει εφαρμογές στην κυβερνοασφάλεια, όπου χρησιμοποιείται για τον εντοπισμό απειλών, την αυτοματοποίηση διαδικασιών ανίχνευσης κακόβουλων δραστηριοτήτων και την πρόβλεψη πιθανών επιθέσεων. Στον χώρο του phishing, η Τεχνητή Νοημοσύνη μπορεί να χρησιμοποιηθεί τόσο από επιτιθέμενους (π.χ. για τη δημιουργία πειστικών ψεύτικων email) όσο και από ειδικούς ασφαλείας για την αναγνώριση ύποπτων μοτίβων.

1.8 Διάρθρωση της Πτυχιακής Εργασίας

Στην ενότητα αυτή παρουσιάζεται η δομή και το περιεχόμενο των επόμενων κεφαλαίων της εργασίας, με στόχο την παροχή μιας συνοπτικής εικόνας της ανάπτυξης του

θέματος. Κάθε κεφάλαιο εστιάζει σε διαφορετικές πτυχές της έρευνας και της υλοποίησης, καλύπτοντας τόσο τη θεωρητική θεμελίωση όσο και την πρακτική εφαρμογή, την αξιολόγηση των αποτελεσμάτων και τα τελικά συμπεράσματα. Παρακάτω ακολουθεί μια συνοπτική περιγραφή των περιεχομένων κάθε κεφαλαίου.

Αρχικά, στο 2ο Κεφάλαιο παρουσιάζονται αναλυτικά οι θεωρητικές έννοιες που σχετίζονται με την παρούσα εργασία. Συγκεκριμένα, γίνεται αναφορά στις μεθόδους, τους αλγορίθμους και τις τεχνικές που έχουν προταθεί από την επιστημονική κοινότητα, καθώς και σε εκείνες που αναπτύχθηκαν κατά τη διάρκεια της εργασίας, αναδεικνύοντας τα πλεονεκτήματα και τα ιδιαίτερα χαρακτηριστικά τους.

Στο 3ο Κεφάλαιο περιγράφεται το λογισμικό που αναπτύχθηκε, με έμφαση στην επιλογή των γλωσσών προγραμματισμού, τις απαιτήσεις και τις προδιαγραφές του συστήματος. Παρουσιάζεται η λειτουργικότητα του λογισμικού μέσα από διαγράμματα UML, παραδείγματα κώδικα και επεξήγηση βασικών χαρακτηριστικών του, καθώς και οπτικές αναπαραστάσεις των παραθύρων της εφαρμογής.

Το 4ο Κεφάλαιο εστιάζει στον έλεγχο καλής λειτουργίας και τα αποτελέσματα χρήσης του λογισμικού. Παρουσιάζονται μετρήσεις απόδοσης και πειραματικά αποτελέσματα που αποτυπώνουν την αξιοπιστία και την αποτελεσματικότητα του συστήματος, ενώ παρέχονται παραδείγματα ελέγχων μέσω εικόνων και πινάκων.

Τέλος, στο 5ο Κεφάλαιο παρατίθενται τα συμπεράσματα που προέκυψαν από την εκπόνηση της πτυχιακής εργασίας. Γίνεται αναφορά στις προκλήσεις που αντιμετωπίστηκαν και στον τρόπο επίλυσής τους, ενώ προτείνονται μελλοντικές βελτιώσεις και επεκτάσεις της παρούσας εργασίας, ώστε να αποτελέσει βάση για περαιτέρω έρευνα και ανάπτυξη.

2ο ΚΕΦΑΛΑΙΟ: ΘΕΩΡΗΤΙΚΕΣ ΕΝΝΟΙΕΣ

2.1 Κοινωνική Μηχανική

Οι επιθέσεις κοινωνικής μηχανικής εμφανίζουν έντονη αύξηση στα σύγχρονα δίκτυα, υπονομεύοντας την ακεραιότητα της κυβερνοασφάλειας [39]. Η κοινωνική μηχανική αποτελεί μια μέθοδο εκμετάλλευσης ανθρώπινων αδυναμιών με σκοπό την παραβίαση της ασφάλειας των πληροφοριακών συστημάτων [40]. Αντί να εστιάζουν σε τεχνικές επιθέσεις εναντίον των συστημάτων, οι δράστες στοχεύουν άτομα με πρόσβαση σε ευαίσθητες πληροφορίες, χειραγωγώντας τα ώστε να αποκαλύψουν εμπιστευτικά δεδομένα ή να διευκολύνουν κακόβουλες ενέργειες μέσω ψυχολογικής επιρροής και πειθούς [40]. Τα συμβατικά τεχνικά μέτρα προστασίας αποδεικνύονται συχνά ανεπαρκή έναντι αυτού του τύπου επιθέσεων [40]. Επιπλέον, οι περισσότεροι άνθρωποι τείνουν να υπερεκτιμούν την ικανότητά τους να αναγνωρίζουν περιπτώσεις εξαπάτησης. Ωστόσο, ερευνητικά δεδομένα υποδεικνύουν ότι η ανθρώπινη ικανότητα ανίχνευσης ψεύδους και παραπλάνησης είναι περιορισμένη [41], [42]. Οι διαβόητες επιθέσεις του Kevin Mitnick ανέδειξαν τον εξαιρετικά επικίνδυνο χαρακτήρα των προηγμένων μεθόδων κοινωνικής μηχανικής, αποδεικνύοντας το εύρος των επιπτώσεών τους στην ασφάλεια πληροφοριών επιχειρήσεων και κυβερνητικών φορέων [43].

Πιο συγκεκριμένα, οι επιθέσεις αυτές αποσκοπούν στη χειραγώγηση ατόμων και επιχειρήσεων, με στόχο την απόσπαση πολύτιμων και ευαίσθητων δεδομένων προς όφελος των κυβερνοεγκληματιών [44]. Η κοινωνική μηχανική συνιστά πολυδιάστατη απειλή για την ασφάλεια οργανισμών και συστημάτων στο σύνολό τους, καθώς δύναται να παρακάμψει ακόμη και τα πλέον προηγμένα τεχνικά μέτρα προστασίας, όπως είναι οι μηχανισμοί κρυπτογράφησης, τα τείχη προστασίας, τα συστήματα ανίχνευσης εισβολών και οι λύσεις αντιμετώπισης κακόβουλου λογισμικού. Οι άνθρωποι, εκ φύσεως, έχουν μεγαλύτερη τάση να εμπιστεύονται άλλους ανθρώπους παρά τα υπολογιστικά συστήματα ή τις τεχνολογίες ασφαλείας, γεγονός που τους καθιστά τον πιο ευάλωτο κρίκο στην αλυσίδα της κυβερνοασφάλειας. Μέσω ανθρώπινων αλληλεπιδράσεων, οι επιθέσεις κοινωνικής μηχανικής επηρεάζουν ψυχολογικά τα θύματα, παρασύροντάς τα στην αποκάλυψη εμπιστευτικών πληροφοριών ή στην παραβίαση πρωτοκόλλων ασφαλείας [45]. Αυτού του είδους οι επιθέσεις είναι από τις πιο επικίνδυνες, καθώς στοχεύουν όλα τα συστήματα και δίκτυα χωρίς να μπορούν να αποτραπούν αποκλειστικά με λύσεις λογισμικού ή υλικού. Η εκπαίδευση των χρηστών αποτελεί βασική προϋπόθεση για την πρόληψη τέτοιων επιθέσεων. Οι κυβερνοεγκληματίες καταφεύγουν σε αυτές τις μεθόδους όταν δεν μπορούν να παραβιάσουν ένα σύστημα μέσω τεχνικών ευπαθειών [46].

Σύμφωνα με το Υπουργείο Δικαιοσύνης των Ηνωμένων Πολιτειών (U.S. Department of Justice), οι επιθέσεις κοινωνικής μηχανικής συγκαταλέγονται στις πλέον σοβαρές απειλές παγκοσμίως. Το 2016, η εταιρεία ανάλυσης κυβερνοασφάλειας Cyence ανέφερε ότι οι Ηνωμένες Πολιτείες ήταν η χώρα που δέχθηκε τις περισσότερες επιθέσεις αυτού του είδους, καταγράφοντας το υψηλότερο οικονομικό κόστος, με τη Γερμανία και την Ιαπωνία να ακολουθούν. Το συνολικό εκτιμώμενο κόστος των επιθέσεων κοινωνικής μηχανικής στις ΗΠΑ ανήλθε σε 121,22 δισεκατομμύρια δολάρια.

Αντίστοιχα, η Ευρωπαϊκή Υπηρεσία Κυβερνοασφάλειας (ENISA) τονίζει ότι οι επιθέσεις κοινωνικής μηχανικής αποτελούν σταθερά μια από τις σημαντικότερες απειλές για τον χρηματοοικονομικό τομέα στην Ευρώπη. Συγκεκριμένα, καταγράφτηκε ότι το 38% των περιστατικών κοινωνικής μηχανικής στόχευψαν άτομα, ενώ το 36% στόχευχαν πιστωτικά ιδρύματα, με κύριους σκοπούς τη κλοπή προσωπικών δεδομένων ή εταιρικών πληροφοριών [47]. Η ENISA επισημαίνει ότι οι πιο συχνές τεχνικές που χρησιμοποιούνται

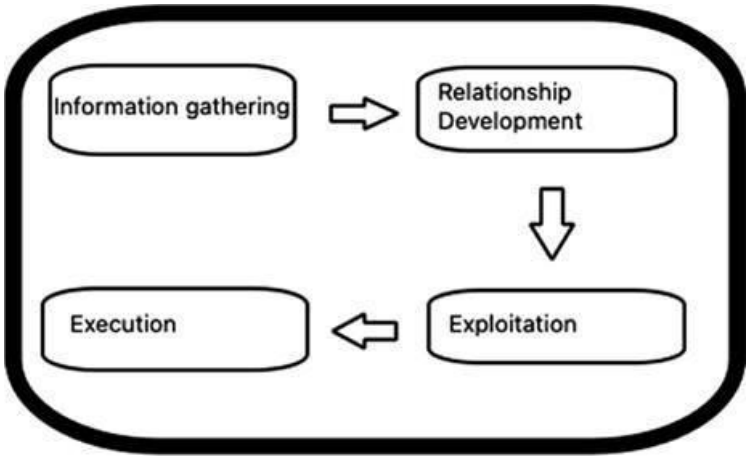
είναι το phishing, smishing και vishing. Ταυτόχρονα, εμφανίζεται αύξηση στα περιστατικά τύπου business email compromise [47].

Οι αμερικανικές επιχειρήσεις αποτελούν ιδιαίτερα ελκυστικούς στόχους για κυβερνοεγκληματίες και χάκερς από όλο τον κόσμο, καθώς διαχειρίζονται κρίσιμα και πολύτιμα διεθνή δεδομένα. Η παραβίαση αυτών των επιχειρήσεων μπορεί να έχει σοβαρές επιπτώσεις στην παγκόσμια οικονομία και στην προστασία των προσωπικών δεδομένων [48]. Χαρακτηριστικό παράδειγμα αποτελεί η περίπτωση της εταιρείας Equifax, η οποία υπέστη εκτεταμένη παραβίαση το 2017 [49]. Ως εταιρεία αναφοράς και παρακολούθησης καταναλωτικής πίστης, η Equifax συλλέγει και διαχειρίζεται δεδομένα ιδιωτών και επιχειρήσεων με σκοπό την αξιολόγηση πιστοληπτικής ικανότητας και την πρόληψη της απάτης [49]. Ωστόσο, εξαιτίας αυτής της κυβερνοεπίθεσης, διέρρευσαν ευαίσθητες προσωπικές πληροφορίες 145,5 εκατομμυρίων Αμερικανών καταναλωτών, συμπεριλαμβανομένων πλήρων ονομάτων, ημερομηνιών γέννησης, αριθμών κοινωνικής ασφάλισης (SSN), αριθμών αδειών οδήγησης, διευθύνσεων, αριθμών τηλεφώνου, στοιχείων πιστωτικών καρτών και πιστωτικών βαθμολογιών [49]. Η επίθεση πραγματοποιήθηκε μέσω phishing, όπου οι εισβολείς έστελναν παραπλανητικά ηλεκτρονικά μηνύματα που προσομοίαζαν επίσημες επικοινωνίες από χρηματοπιστωτικά ιδρύματα, όπως η Bank of America [50]. Η διαρροή αυτή συνεχίζει να προκαλεί ανησυχία στους χρήστες της Equifax, καθώς οι κυβερνοεγκληματίες εξακολουθούν να εκμεταλλεύονται τα κλεμμένα δεδομένα [50].

Επιπλέον, πρόσφατα αναφέρθηκε άλλη σοβαρή κυβερνοεπίθεση εις βάρος της Κεντρικής Τράπεζας της Βόρειας Κορέας, κατά την οποία ένας εισβολέας απέσπασε περισσότερα από 80 εκατομμύρια δολάρια [51]. Η επίθεση πραγματοποιήθηκε μέσω ενός κακόβουλου λογισμικού απομακρυσμένης πρόσβασης (Remote Access Trojan/RAT), το οποίο είχε εγκατασταθεί στους υπολογιστές της τράπεζας, επιτρέποντας στους δράστες να υποκλέψουν κρίσιμες οικονομικές πληροφορίες [51].

Όπως αναφέρει ο M. Libicki [51], οι επιθέσεις κοινωνικής μηχανικής αν και είναι δυνατόν να εντοπιστούν, δεν μπορούν να αποτραπούν πλήρως. Οι δράστες αυτών των επιθέσεων εκμεταλλεύονται τα θύματά τους προκειμένου να αποσπάσουν ευαίσθητες πληροφορίες, οι οποίες είτε αξιοποιούνται για συγκεκριμένους σκοπούς είτε διατίθενται προς πώληση στη μαύρη αγορά και στο σκοτεινό διαδίκτυο. Με την εξέλιξη των Big Data, οι εισβολείς αξιοποιούν μεγάλες ποσότητες δεδομένων για την εξαγωγή πολύτιμων πληροφοριών με επιχειρηματική αξία [52]. Επιπλέον, συγκεντρώνουν τεράστιους όγκους δεδομένων, τα οποία διατίθενται μαζικά ως εμπορεύσιμα αγαθά στις σύγχρονες αγορές [53].

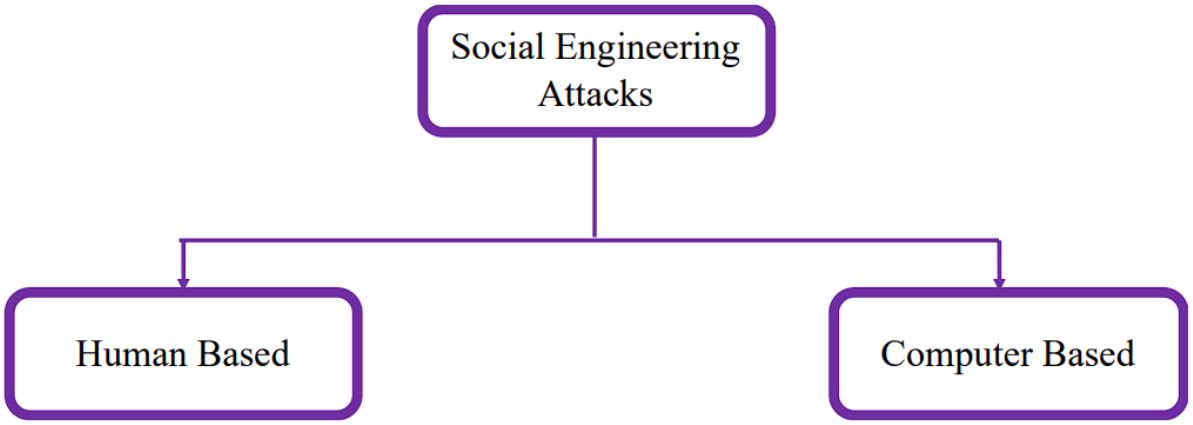
Παρόλο που οι επιθέσεις κοινωνικής μηχανικής ποικίλλουν ως προς τη μορφή και την εκτέλεσή τους, ακολουθούν ένα κοινό πρότυπο με διακριτές φάσεις. Το γενικό αυτό μοντέλο αποτελείται από τέσσερα στάδια: (1) τη συλλογή πληροφοριών σχετικά με τον εκάστοτε στόχο, διαδικασία που εντάσσεται κυρίως στο πλαίσιο των επιθέσεων τύπου spear phishing, (2) την ανάπτυξη μιας σχέσης εμπιστοσύνης με αυτόν, (3) την αξιοποίηση των συγκεντρωμένων δεδομένων για την πραγματοποίηση της επίθεσης και (4) την αποχώρηση του δράστη χωρίς να αφήσει ίχνη [54]. Στο Σχήμα 1 παρουσιάζονται αναλυτικά τα επιμέρους στάδια μιας επίθεσης κοινωνικής μηχανικής.



Σχήμα 1: Στάδια επίθεσης κοινωνικής μηχανικής [52]

Κατά τη διάρκεια της ερευνητικής φάσης, γνωστής και ως συλλογή πληροφοριών, ο δράστης επιλέγει το υποψήφιο θύμα του με βάση συγκεκριμένα κριτήρια. Στη φάση της προσέγγισης, επιχειρεί να εδραιώσει σχέση εμπιστοσύνης με το θύμα μέσω άμεσης επικοινωνίας ή ανταλλαγής μηνυμάτων, όπως μέσω ηλεκτρονικού ταχυδρομείου. Ακολουθεί η φάση της χειραγώγησης, όπου ο εισβολέας εκμεταλλεύεται συναισθηματικά το θύμα, με σκοπό να αποκτήσει ευαίσθητες πληροφορίες ή να το οδηγήσει σε σφάλματα ασφαλείας. Τέλος, στη φάση της αποχώρησης, ο επιτιθέμενος διακόπτει κάθε επαφή, φροντίζοντας να μην αφήσει κανένα ίχνος της δραστηριότητάς του [54], [55].

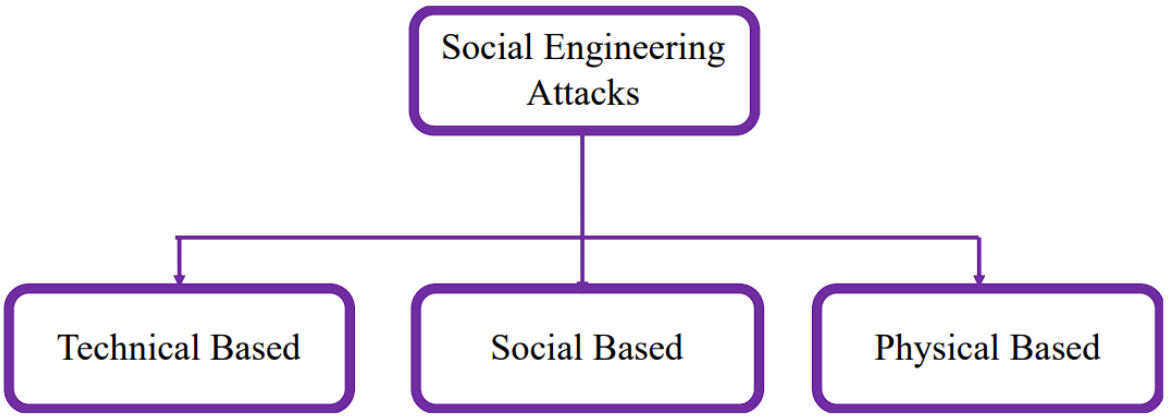
Οι επιθέσεις κοινωνικής μηχανικής διακρίνονται σε δύο κύριες κατηγορίες: αυτές που βασίζονται στην ανθρώπινη αλληλεπίδραση και εκείνες που αξιοποιούν τεχνολογικά μέσα, όπως παρουσιάζεται στο Σχήμα 2 [54], [56].



Σχήμα 2: Ταξινόμηση επιθέσεων κοινωνικής μηχανικής.

Στις επιθέσεις που εστιάζουν στον ανθρώπινο παράγοντα, ο εισβολέας προσεγγίζει άμεσα τον στόχο και αλληλεπιδρά μαζί του, με σκοπό τη συλλογή των επιθυμητών πληροφοριών [54]. Αυτό το είδος επίθεσης επηρεάζει συνήθως περιορισμένο αριθμό θυμάτων. Αντίθετα, οι επιθέσεις που εκτελούνται μέσω λογισμικού αξιοποιούν τεχνολογικές συσκευές, όπως υπολογιστές ή κινητά τηλέφωνα, για την απόσπαση πληροφοριών από τα θύματα, καθιστώντας δυνατή την επίθεση σε μεγάλο αριθμό χρηστών μέσα σε ελάχιστο

χρόνο [54]. Ένα χαρακτηριστικό παράδειγμα τέτοιων επιθέσεων είναι η εργαλειοθήκη κοινωνικής μηχανικής (SET), η οποία χρησιμοποιείται για την εκτέλεση στοχευμένων επιθέσεων ηλεκτρονικού ψαρέματος [57]. Επιπλέον, οι επιθέσεις κοινωνικής μηχανικής μπορούν να ταξινομηθούν σε τρεις επιμέρους κατηγορίες, ανάλογα με τη μεθοδολογία που χρησιμοποιείται: κοινωνικές, τεχνικές και επιθέσεις που στηρίζονται σε φυσικά μέσα, όπως αποτυπώνεται στο Σχήμα 3 [44], [45]. Το σχήμα παρουσιάζει με διαγραμματική μορφή την ιεραρχία αυτών των τύπων επιθέσεων, ξεκινώντας από τον γενικό τίτλο “Social Engineering Attacks” στο επάνω επίπεδο, ο οποίος λειτουργεί ως κεντρική κατηγορία. Από αυτόν τον κόμβο εκκινούν τρεις διακλαδώσεις που οδηγούν στις αντίστοιχες υποκατηγορίες: Technical Based, δηλαδή τεχνικές επιθέσεις, όπως phishing μέσω κακόβουλου λογισμικού ή παραβίαση συστημάτων, Social Based που αφορούν καθαρά ανθρώπινες αλληλεπιδράσεις, όπως η παραπλάνηση μέσω τηλεφώνου ή η χειραγώγηση της εμπιστοσύνης και Physical Based, οι οποίες σχετίζονται με φυσική πρόσβαση, όπως το tailgating (είσοδος σε απαγορευμένο χώρο ακολουθώντας έναν εξουσιοδοτημένο υπάλληλο) ή η κλοπή εγγράφων. Με αυτόν τον τρόπο, το διάγραμμα αποσαφηνίζει ότι οι επιθέσεις κοινωνικής μηχανικής δεν περιορίζονται σε μια μόνο τεχνική, αλλά καλύπτουν ένα ευρύ φάσμα προσεγγίσεων που εκμεταλλεύονται είτε τεχνολογικά μέσα, είτε ανθρώπινες αδυναμίες, είτε φυσική πρόσβαση.



Σχήμα 3: Ταξινόμηση επιθέσεων κοινωνικής μηχανικής.

Οι κοινωνικές επιθέσεις βασίζονται στην αλληλεπίδραση του επιτιθέμενου με το θύμα, αξιοποιώντας ψυχολογικές και συναισθηματικές τακτικές για την εξαπάτησή του [54]. Αυτές οι επιθέσεις θεωρούνται ιδιαίτερα επικίνδυνες και αποτελεσματικές, καθώς περιλαμβάνουν άμεσες ανθρώπινες αλληλεπιδράσεις [58]. Χαρακτηριστικά παραδείγματα τέτοιων επιθέσεων είναι το baiting και το spear phishing. Αντίστοιχα, οι τεχνικές επιθέσεις διενεργούνται μέσω διαδικτυακών μέσων, όπως κοινωνικά δίκτυα και ιστοσελίδες υπηρεσιών, με σκοπό τη συλλογή ευαίσθητων πληροφοριών, όπως κωδικών πρόσβασης, δεδομένων πιστωτικών καρτών και απαντήσεων σε ερωτήσεις ασφαλείας [44]. Οι φυσικές επιθέσεις, από την άλλη, περιλαμβάνουν άμεσες ενέργειες του επιτιθέμενου για την απόκτηση πληροφοριών σχετικά με τον στόχο. Ένα χαρακτηριστικό παράδειγμα αυτής της κατηγορίας είναι η αναζήτηση σε κάδους απορριμμάτων για την εύρεση σημαντικών εγγράφων [45].

Οι επιθέσεις κοινωνικής μηχανικής συχνά συνδυάζουν πολλαπλές διαστάσεις, συμπεριλαμβάνοντας ανθρώπινες, τεχνικές, κοινωνικές και φυσικές προσεγγίσεις. Ορισμένες από τις πλέον διαδεδομένες τεχνικές περιλαμβάνουν το phishing, την πλαστοπροσωπία σε κλήσεις γραφείου υποστήριξης, το shoulder surfing, την αναζήτηση

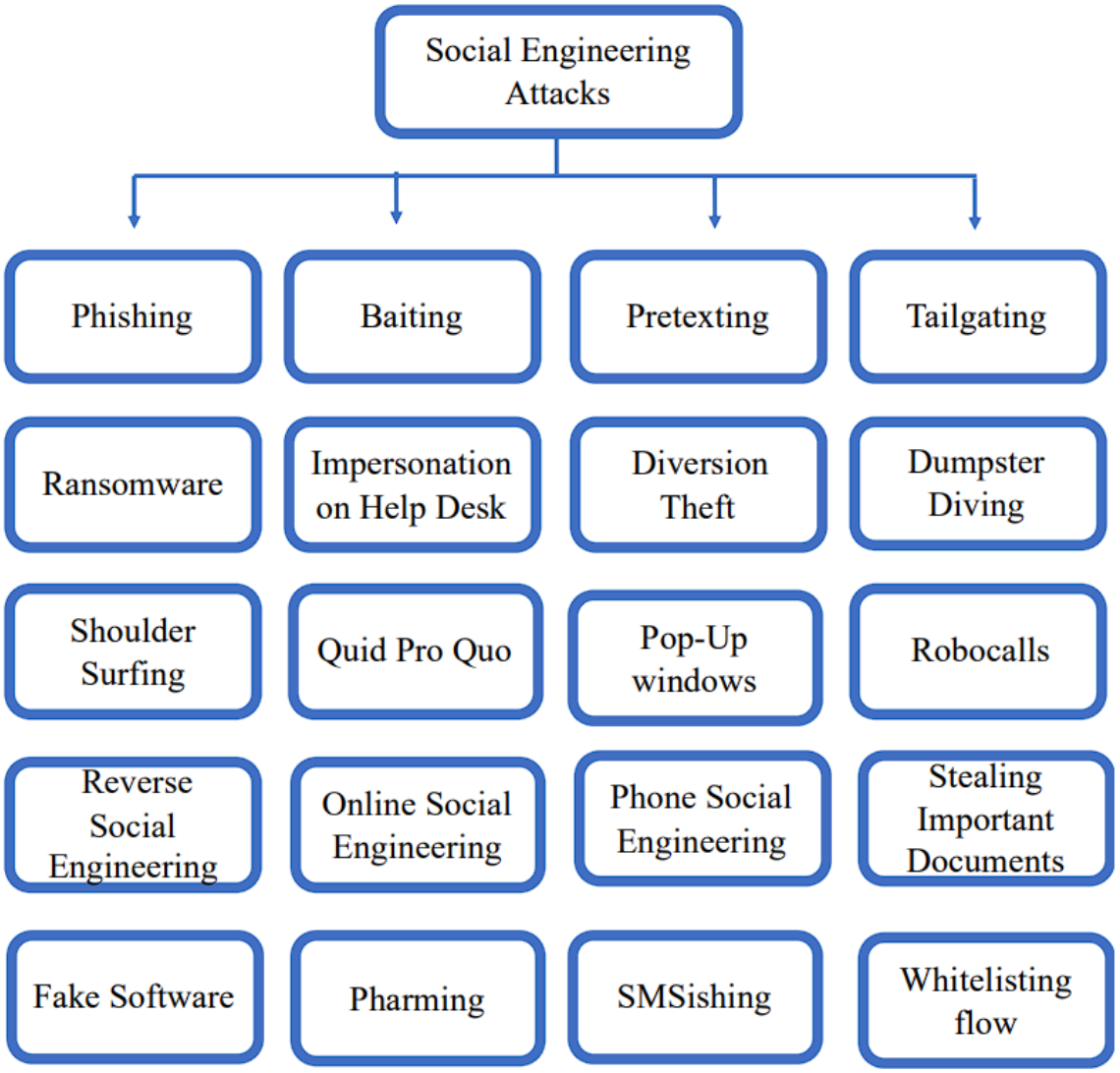
σε σκουπίδια, την κλοπή εγγράφων, την εκτροπή αλληλογραφίας, τη διανομή κακόβουλου λογισμικού, το baiting, το quid pro quo, την προσποίηση, το tailgating, τα παραπλανητικά αναδυόμενα παράθυρα, τις αυτοματοποιημένες κλήσεις (Robocalls), το ransomware, την αντίστροφη κοινωνική μηχανική (reverse social engineering) και τις επιθέσεις μέσω τηλεφώνου [39], [54].

Πιο συγκεκριμένα, το phishing συνιστά σοβαρή μορφή κυβερνοεγκλήματος, κατά την οποία οι δράστες παραπλανούν τους χρήστες προκειμένου να αποσπάσουν εμπιστευτικές προσωπικές πληροφορίες [59]. Παρά την ύπαρξη σύγχρονων τεχνικών και εργαλείων πρόληψης, πολλοί χρήστες εξακολουθούν να καταχωρούν δεδομένα σε παραπλανητικούς ιστότοπους, γεγονός που δυσχεραίνει το έργο των ειδικών ασφαλείας [59]. Διατίθενται γραμμές εργαλείων (toolbars) για διάφορα προγράμματα περιήγησης με σκοπό την έγκαιρη ειδοποίηση για ύποπτες ιστοσελίδες, ωστόσο οι επιθέσεις τύπου “spear phishing” καθιστούν ακόμη πιο δύσκολη τη διάκριση γνήσιων από πλαστογραφημένα μηνύματα. Τα πλαστά αυτά μηνύματα μπορούν να προκαλέσουν σοβαρή βλάβη στη διαδικασία επαλήθευσης της ταυτότητας των χρηστών [59]. Το ransomware συνιστά σοβαρή απειλή τόσο για τον άνθρωπο όσο και για επιχειρήσεις [60]. Διαδίδεται κυρίως μέσω συνημμένων αρχείων ή συνδέσμων σε μηνύματα ηλεκτρονικού ψαρέματος, αλλά και από μολυσμένα USB [60]. Η επίθεση περιλαμβάνει κρυπτογράφηση ή διαγραφή κρίσιμων δεδομένων, με το κλειδί αποκρυπτογράφησης να παραμένει υπό τον έλεγχο των δραστών μέχρι την καταβολή λύτρων, γεγονός που διευκολύνεται από την ανωνυμία των συναλλαγών σε κρυπτονομίσματα, όπως το bitcoin [60]. Το shoulder surfing περιγράφεται ως μια κλασική μέθοδος επίθεσης, η οποία δεν απαιτεί τεχνολογία [61]. Η τεχνική αυτή περιλαμβάνει την κρυφή παρατήρηση κάποιου που εισάγει ευαίσθητα δεδομένα, όπως κωδικούς ή αριθμούς, με το να κοιτά κάποιος πάνω από τον ώμο του θύματος [61]. Ιστορικά, πριν από τη δεκαετία του 1990, η μέθοδος χρησιμοποιούνταν για την υποκλοπή των ψηφίων των τηλεφωνικών καρτών από δημόσια τηλέφωνα, με σκοπό τη δωρεάν πραγματοποίηση υπεραστικών κλήσεων ή την πώλησή τους [61]. Παρά το γεγονός ότι σήμερα υπάρχουν πιο σύγχρονες τεχνικές για την υποκλοπή αυτών των δεδομένων, η παρακολούθηση πληκτρολογίου παραμένει μια πρακτική μέθοδος με ποικίλες εφαρμογές [61]. Οι επιθέσεις τύπου ψεύτικου λογισμικού, γνωστές και ως fake software, εκμεταλλεύονται την εμπιστοσύνη των χρηστών, προσομοιώνοντας γνωστές και αξιόπιστες πλατφόρμες [62]. Μέσω αυτών, τα θύματα εισάγουν τα πραγματικά στοιχεία σύνδεσής τους, τα οποία οι επιτιθέμενοι χρησιμοποιούν για πρόσβαση σε νόμιμους λογαριασμούς, όπως τραπεζικούς [62]. Ένα χαρακτηριστικό παράδειγμα αποτελεί η επίθεση tabnabbing, όπου η ψεύτικη σελίδα μιμείται δημοφιλείς ιστοσελίδες, π.χ. για online τραπεζικές υπηρεσίες ή κοινωνικά δίκτυα, και οι χρήστες εισάγουν τα διαπιστευτήριά τους ενώ εστιάζουν σε άλλο αντικείμενο [62]. Οι κακόβουλοι χρήστες εκμεταλλεύονται αυτήν την εμπιστοσύνη για να αποκτήσουν πρόσβαση σε ευαίσθητες πληροφορίες [63]. Οι επιθέσεις baiting, που ονομάζονται επίσης road apples, αποτελούν μορφή ηλεκτρονικού «ψαρέματος» (phishing), όπου οι χρήστες παροτρύνονται να κάνουν κλικ σε συνδέσμους υπό την υπόσχεση δωρεάν περιεχομένου [39]. Λειτουργούν ως δούρειοι ίπποι (trojan horses), εκμεταλλευόμενες μη ασφαλή μέσα αποθήκευσης, όπως USB που περιέχουν κακόβουλο λογισμικό και αφήνονται σε δημόσιους χώρους [39]. Όταν οι χρήστες συνδέουν αυτές τις συσκευές στους υπολογιστές τους, το κακόβουλο λογισμικό ενεργοποιείται στο παρασκήνιο, χωρίς να γίνεται αντιληπτό, προκαλώντας ανεπιθύμητες ενέργειες στο σύστημα [39]. Η πλαστοπροσωπία σε περιβάλλον Help Desk συνίσταται στην ενέργεια ενός ατόμου να εμφανίζεται ως εξουσιοδοτημένος χρήστης ή υπάλληλος της εταιρείας, με σκοπό την απόκτηση πληροφοριών ή την παροχή υπηρεσιών από το τμήμα υποστήριξης [39]. Οι επιθέσεις τύπου Quid Pro Quo συνίστανται σε επιθέσεις baiting που προσφέρουν δωρεάν υπηρεσίες

με σκοπό την παραπλάνηση του θύματος, απαιτώντας ως αντάλλαγμα την παροχή πληροφοριών ή άλλων στοιχείων [40]. Οι επιθέσεις κοινωνικής μηχανικής στο διαδίκτυο (Online social engineering attacks) λαμβάνουν χώρα όταν ο επιτιθέμενος υποδύεται τον διαχειριστή ενός εταιρικού δικτύου προκειμένου να αποσπάσει διαπιστευτήρια χρηστών [39]. Οι επιθέσεις τύπου pharming πραγματοποιούνται όταν ένας κακόβουλος χρήστης ανακατευθύνει την επισκεψιμότητα ενός νόμιμου ιστότοπου προς έναν ψεύτικο, με σκοπό την υποκλοπή των μεταδιδόμενων πληροφοριών [64]. Η μέθοδος αυτή περιλαμβάνει την παραβίαση του διακομιστή συστήματος ονομάτων τομέα (DNS) και την εκμετάλλευση πιθανών ευπαθειών για την τροποποίηση των διευθύνσεων IP του υπολογιστή-πελάτη και του διακομιστή [39]. Επιπλέον, οι επιθέσεις τύπου pretexting έχουν ως σκοπό να δημιουργήσουν ένα περιβάλλον όπου ο επιτηθέμενος θα βρίσκεται σε μια προνομιακό θέση για να πετύχει μια μελλοντική επίθεση [65]. Συχνή τακτική των κακόβουλων είναι να παρουσιάζουν πειστικές ιστορίες χρησιμοποιώνοτας μηνύματα και εικόνες, όπως επίσημα λογότυπα, που μοιάζουν με νόμιμα. Η επίθεση pretexting μπορεί να γίνει διαδικτυακά, αυτοπροσώπως ή τηλεφωνικά. Οι επιθέσεις Diversion Theft αναφέρονται στην παραπλάνηση μεταφορικών εταιρειών για τη παράδοση δεμάτων στην επιθυμητή τοποθεσία [39]. Η επίθεση Pop-up Windows εμφανίζει στην οθόνη του θύματος κάποιο παραπλανητικό παράθυρο, όπως για παράδειγμα ένα ειδοποιητικό μήνυμα ότι ο χρήστης έχει κολλήσει ιό και η συσκευή του βρίσκεται σε κίνδυνο, με σκοπό το θύμα να πατήσει πάνω στο παράθυρο και να κατεβάσει κακόβουλο λογισμικό [39]. Το λογισμικό αυτό θα τρέξει στο παρασκήνιο της συσκευής, κλέβοντας προσωπικά στοιχεία του θύματος. Οι επιτηθέμενοι μπορούν να εξαπατήσουν τα θύματά τους μέσω της τηλεφωνικής κινητής συσκευής. Δύο διαφορετικοί τρόποι είναι οι επιθέσεις Smishing και Robocall. Στις Smishing επιθέσεις ο θύτης στέλνει πειστικά κακόβουλα μηνύματα στη κινητή συσκευή του θύματος, έτσι ώστε ο θύτης να τα ανοίξει και να αλληλεπιδράσει μαζί τους. Η επίθεση αυτή είναι αποδοτικοί, καθώς τα θύματα έχουν μαζί τους τη προσωπική τους κινητή συσκευή ανά πάσα ώρα και στιγμή [39]. Τέτοια μηνύματα συνήθως περιλαμβάνουν κακόβουλο λογισμικό το οποίο εγκαταστεί backdoors για να έχουν πρόσβαση οι επιτηθέμενοι. Από την αλλή, οι επιθέσεις Robocalls αφορούν προκαθορισμένες κλήσεις που στοχεύουν σε γνωστά οικιακά και επαγγελματικά νούμερα. Μια robocall είναι μια συσκευή ή ένα πρόγραμμα σε υπολογιστή το οποίο καλεί αυτόματα μια λίστα από αριθμούς και μεταδίδει προκαθορισμένα μηνύματα. Βασίζεται κυρίως στη χρήση του voice over the internet protocol (VoIP), για να διασφαλιστούν συγκεκριμένες λειτουργίες του VoIP, όπως διαδραστικές απαντήσεις και text to speech [66]. Οι κλήσεις αυτές κυμαίνονται από την προσφορά ή την πώληση υπηρεσιών, όπως βοήθεια σε ζητήματα εφορίας [39]. Ο μόνος τρόπος για να αποφύγει κάποιος τις επιθέσεις robocalls είναι να αρνείται τη εισερχόμενη κλήση, διαφορετικά ακόμη και άμα ο αριθμός που έχει κάνει το robocall είναι αποκλεισμένος, το νούμερο του θύματος έχει καταχωρηθεί σε μια βάση δεδομένων για να κληθεί από άλλο νούμερο του robocall. Ταυτόχρονα, αντίστοιχη λογική είναι και οι επιθέσεις phone social engineering, ή vishing, που αναφέρονται στις κακόβουλες κλήσεις που θα εκτελέσει ο θύτης για να εξαπατήσει το θύμα. Οι επιθέσεις τύπου tailgating είναι τεχνικές κοινωνικής μηχανικής που εξάγονται στον φυσικό χώρο. Ο επιτηθέμενος συνιθίζει να παριστάνει κάποιον υπάλληλο μιας εταιρείας, με σκοπό να μπει στο κτίριο της εταιρείας μη εξουσιοδοτημένα. Για παράδειγμα, ο επιτηθέμενος ρωτάει το θύμα άμα μπορεί να ανοίξει το θύμα τη πόρτα του κτιρίου επειδή ο επιτηθέμενος έχει ξεχάσει την κάρτα του που του επιτρέπει να εισέλθει [39]. Η επίθεση Dumpster Diving είναι μια ακόμη επίθεση στον φυσικό χώρο και περιλαμβάνει την συλλογή ευαίσθητων αρχείων ή παλιών συσκευών από τα απορρίματα μιας εταιρείας [40]. Παράλληλα, η επίθεση Stealing important documents αναφέρεται στη μη εξουσιοδοτημένη συλλογή αρχείων από τα γραφεία των υπαλλήλων

[39]. Τέλος, η επίθεση σε whitelist flow αναφέρεται στη προσπάθεια των επιτηθέμενων να παρέμβουν στη whitelist του θύτη με μη εξουσιοδοτημένους τρόπους [67].

Το Σχήμα 4 παρουσιάζει τη διάρθρωση και την κατηγοριοποίηση αυτών των επιθέσεων.

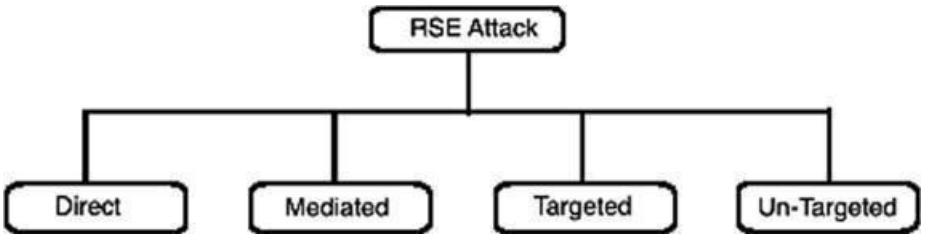


Σχήμα 4: Επιθέσεις κοινωνικής μηχανικής.

Η αντίστροφη κοινωνική μηχανική αποτελεί μια αυτοματοποιημένη τεχνική υψηλής αποτελεσματικότητας, καθώς οι επιτιθέμενοι δεν έρχονται σε άμεση επικοινωνία με το θύμα. Αντίθετα, το παραπλανούν ώστε να πιστέψει ότι ο εισβολέας είναι μια αξιόπιστη πηγή. Για παράδειγμα, αν ένας εισβολέας καλούσε απευθείας ένα θύμα ζητώντας του τα διαπιστευτήριά του, το τελευταίο θα μπορούσε να υποψιαστεί την απάτη. Ωστόσο, μέσω της πλαστογράφησης του email ενός διαχειριστή συστήματος, ο εισβολέας μπορεί να αποστείλει έναν φαινομενικά έγκυρο αριθμό τηλεφώνου, τον οποίο το θύμα καλεί όταν αντιμετωπίσει κάποιο πρόβλημα, οδηγώντας το έτσι στην παγίδα του [54].

Οι επιθέσεις αντίστροφης κοινωνικής μηχανικής είναι ιδιαίτερα αποτελεσματικές σε διαδικτυακά κοινωνικά δίκτυα, καθώς επιτρέπουν την πρόσβαση σε μεγάλο αριθμό χρηστών, παρακάμπτοντας τα υπάρχοντα συστήματα ανίχνευσης ύποπτης συμπεριφοράς

και φιλτραρίσματος ανεπιθύμητης επικοινωνίας. Όταν το θύμα προσεγγίζει μόνο του τον εισβολέα, η δυσπιστία μειώνεται σημαντικά, αυξάνοντας έτσι τις πιθανότητες επιτυχίας της επίθεσης [39]. Η κοινωνική μηχανική στην αντίστροφη RSE χωρίζεται σε τέσσερα είδη με βάση το είδος της επίθεσης (Σχήμα 5).



Σχήμα 5: Τύποι επιθέσεων αντίστροφης κοινωνικής μηχανικής.

Σε μια Άμεση Επίθεση, η κακόβουλη δραστηριότητα του εισβολέα γίνεται άμεσα αντιληπτή από τους χρήστες-στόχους. Ο επιτιθέμενος μπορεί, για παράδειγμα, να αποστείλει ένα μήνυμα ή να ανεβάσει μια εικόνα σε έναν διαδικτυακό ιστότοπο με σκοπό την παραπλάνηση ή την επίθεση. Στην περίπτωση της Διαμεσολαβημένης Επίθεσης, ένας ενδιάμεσος παράγοντας συγκεντρώνει το κακόβουλο περιεχόμενο και αναλαμβάνει τη διανομή του στους τελικούς αποδέκτες, συχνά σε τροποποιημένη μορφή, ώστε να καταστεί πιο πειστικό ή δύσκολο να εντοπιστεί. Μια Στοχευμένη Επίθεση απευθύνεται σε έναν συγκεκριμένο χρήστη. Για να υλοποιήσει μια τέτοια επίθεση, ο εισβολέας πρέπει να διαθέτει προκαταρκτικές πληροφορίες για το θύμα, όπως τα στοιχεία σύνδεσης ή τη διεύθυνση ηλεκτρονικού ταχυδρομείου του. Μια Μη Στοχευμένη Επίθεση έχει ως κύριο στόχο τη μαζική διάδοση του κακόβουλου περιεχομένου, χωρίς να επικεντρώνεται σε συγκεκριμένους χρήστες, αλλά επιδιώκοντας να προσεγγίσει όσο το δυνατόν περισσότερα άτομα [54].

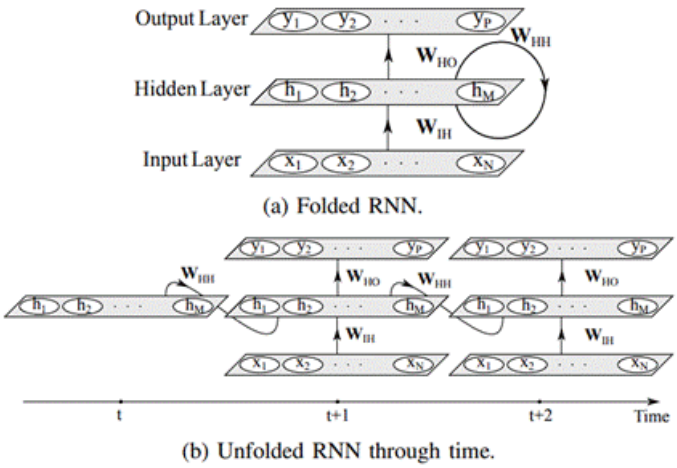
2.2 Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks/RNNs)

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks/ANNs) αποτελούνται από διασυνδεδεμένες μονάδες, γνωστές ως τεχνητοί νευρώνες, οι οποίες οργανώνονται σε επίπεδα [68]. Ένα ρηχό δίκτυο περιλαμβάνει ένα επίπεδο εισόδου, ένα επίπεδο εξόδου και το πολύ ένα ενδιάμεσο κρυφό επίπεδο, χωρίς επαναλαμβανόμενες συνδέσεις [68]. Η πολυπλοκότητα του δικτύου αυξάνεται με την προσθήκη επιπλέον επιπέδων ή επαναλαμβανόμενων συνδέσεων, γεγονός που οδηγεί στη δημιουργία βαθύτερων δικτύων [68]. Αυτά επιτρέπουν την προοδευτική εξαγωγή χαρακτηριστικών και αναπαράσταση δεδομένων σε διαφορετικά επίπεδα αφαίρεσης, μια διαδικασία γνωστή ως βαθιά μάθηση [68]. Συνήθως, τα δίκτυα αυτά συγκροτούνται από μη γραμμικές, αλλά απλές μονάδες, με τα ανώτερα επίπεδα να παρέχουν πιο αφηρημένες αναπαραστάσεις και να μειώνουν την περιττή μεταβλητότητα στα δεδομένα [69].

Πριν από τις σημαντικές εξελίξεις του 2006, η έρευνα στις βαθιές νευρωνικές αρχιτεκτονικές ήταν περιορισμένη λόγω των προκλήσεων που επέφερε η σύνθεση της μη γραμμικότητας σε κάθε επίπεδο [70], [71]. Τα ANN που διαθέτουν επαναλαμβανόμενες συνδέσεις ονομάζονται επαναλαμβανόμενα νευρωνικά δίκτυα, τα οποία μπορούν να επεξεργάζονται διαδοχικά δεδομένα για σκοπούς αναγνώρισης και πρόβλεψης ακολουθιών [72]. Η αρχιτεκτονική τους βασίζεται σε κρυφές καταστάσεις υψηλής διάστασης με μη γραμμική δυναμική, λειτουργώντας ως μνήμη του δικτύου, όπου η τρέχουσα κατάσταση

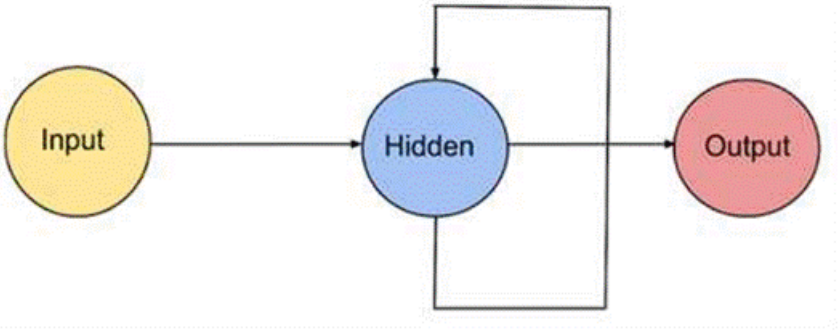
εξαρτάται από την προηγούμενη [73], [74]. Αυτή η ιδιότητα επιτρέπει στα RNN να αποθηκεύουν και να επεξεργάζονται σύνθετα σήματα για εκτεταμένες χρονικές περιόδους, διευκολύνοντας την αντιστοίχιση και την πρόβλεψη ακολουθιών μεταξύ διαδοχικών χρονικών βημάτων.

Τα RNN αποτελούν μια κατηγορία εποπτευόμενων μοντέλων μηχανικής μάθησης, τα οποία απαρτίζονται από τεχνητούς νευρώνες με έναν ή περισσότερους βρόχους ανάδρασης [75]. Αυτοί οι βρόχοι επιτρέπουν τη διατήρηση πληροφορίας κατά μήκος χρονικών βημάτων ή ακολουθιών [76], όπως φαίνεται στο Σχήμα 6. Η εκπαίδευση ενός RNN σε εποπτευόμενο πλαίσιο προϋποθέτει τη χρήση ενός συνόλου δεδομένων με αντιστοιχίες εισόδου-στόχου, με σκοπό τη βελτιστοποίηση των βαρών του δικτύου και την ελαχιστοποίηση της απόκλισης μεταξύ των προβλεπόμενων και των πραγματικών τιμών, μειώνοντας έτσι τη συνάρτηση απώλειας.



Σχήμα 6: Ένα απλό επαναλαμβανόμενο νευρωνικό δίκτυο (RNN) και η αναπαράστασή του ως ξεδιπλωμένη ακολουθία στο χρόνο t. Τα βέλη αντιπροσωπεύουν πλήρως συνδεδεμένες μονάδες μεταξύ των επιπέδων. Για λόγους απλοποίησης, οι παράμετροι μεροληψίας δεν περιλαμβάνονται

Το Επαναλαμβανόμενο Νευρωνικό Δίκτυο αποτελεί μια εποπτευόμενη μέθοδο μάθησης που προσομοιώνει τη λειτουργία της βραχυπρόθεσμης μνήμης, η οποία εντοπίζεται στον μετωπιαίο λοβό του εγκεφάλου [77]. Η βασική αρχή του RNN είναι η διατήρηση και αξιοποίηση πληροφορίας από προηγούμενες εισόδους, ώστε να βελτιστοποιεί τη διαδικασία μάθησης [77]. Σε αντίθεση με τα συμβατικά νευρωνικά δίκτυα, τα κρυφά επίπεδά του δεν παράγουν απλώς εξόδους αλλά ανατροφοδοτούν τον εαυτό τους, επιτρέποντας έτσι τη διαχείριση ακολουθιακών δεδομένων μέσω της ενσωματωμένης μνήμης τους [78], όπως φαίνεται στο Σχήμα 7.



Σχήμα 7: Απλή δομή RNN.

Οι νευρώνες συνδέονται μεταξύ τους και διατηρούν τη σύνδεση αυτή μέσω του χρόνου, γεγονός που υποδηλώνει την ύπαρξη ενός τύπου μνήμης (βραχυπρόθεσμης μνήμης - STM). Αυτοί οι νευρώνες έχουν την ικανότητα να ανακαλούν πληροφορίες που υπήρχαν στο εσωτερικό τους στο παρελθόν. Μέσα από τη διαδικασία μάθησης, οι νευρώνες αξιοποιούν τις προηγούμενες παρατηρήσεις και μεταβιβάζουν τις αποκτηθείσες γνώσεις στους επόμενους νευρώνες. Για παράδειγμα, όταν χρησιμοποιείται ένα επαναλαμβανόμενο νευρωνικό δίκτυο για τη μετάφραση μιας πρότασης, το δίκτυο πρέπει να αποθηκεύει τις μεταφράσεις κάθε λέξης καθώς προχωρά στην ανάλυση, προκειμένου να κατανοήσει τη συνολική έννοια και να παραγάγει μία ακριβή μετάφραση. Υπάρχουν διάφοροι τύποι RNN, όπως: (α) "Ένα-Πολλά", το οποίο χρησιμοποιείται για την περιγραφή εικόνας με τη χρήση διαφορετικών λέξεων, όπου η εικόνα αποτελεί την είσοδο και οι λέξεις την έξοδο. (β) "Πολλά-Ένα", που χρησιμοποιείται στη σημασιολογική ανάλυση, όπως η κατηγοριοποίηση κειμένων ως θετικών ή αρνητικών, όπου η είσοδος είναι μια ομάδα κειμένων και η έξοδος μία τιμή που υποδεικνύει θετικό ή αρνητικό αποτέλεσμα. (γ) "Πολλά-Πολλά", που αξιοποιείται στη μετάφραση μέσω συστημάτων όπως το Google Translate, όπου λέξεις από μία γλώσσα μεταφράζονται σε άλλες γλώσσες [79].

2.3 Spambot

Το Spambot αποτελεί μια καινοτόμο μέθοδο δημιουργίας και διάδοσης ανεπιθύμητης αλληλογραφίας μέσω ενός αυτοματοποιημένου προγράμματος σε περιβάλλον web 2.0. Εφαρμογές όπως τα ιστολόγια, το Facebook, το Twitter και άλλες μορφές κοινωνικών δικτύων προσφέρουν ένα ιδανικό πεδίο για την εξάπλωση αυτού του είδους ανεπιθύμητης επικοινωνίας [80].

Το spambot είναι ένα αυτοματοποιημένο λογισμικό, το οποίο έχει σχεδιαστεί για να προσομοιώνει τη συμπεριφορά του ανθρώπου κατά τη διαδικασία αποστολής και διάδοσης των ανεπιθύμητων μηνυμάτων. Συνήθως, αυτά τα μηνύματα προκύπτουν από ψευδείς λογαριασμούς, οι οποίοι μπορούν να χρησιμοποιηθούν για την αποστολή αυτών των μηνυμάτων. Αυτό κατέστη εφικτό με την εξέλιξη του web 2.0, το οποίο χαρακτηρίζεται από δυναμικούς ιστότοπους, σε αντίθεση με τους στατικούς ιστότοπους του παρελθόντος. Ένας ιστότοπος Web 2.0 επιτρέπει στους χρήστες να αλληλεπιδρούν με τις πληροφορίες για συζητήσεις και συνεργασίες, καθώς και να μοιράζονται δεδομένα και αρχεία, όπως παρατηρείται σε πλατφόρμες κοινωνικής δικτύωσης, όπως το Facebook και το Twitter [80].

Αντίθετα, ένας στατικός ιστότοπος παρέχει μόνο παθητική πρόσβαση στο περιεχόμενο [81, p. 2]. Αυτός ο τύπος ανεπιθύμητης αλληλογραφίας που εκδηλώνεται στο web 2.0 αναφέρεται συνήθως ως "spam 2.0", δηλαδή ένα ρομπότ που στέλνει και διαδίδει

αυθαίρετα μηνύματα σε δυναμικούς ιστότοπους [82], [83]. Συνηθέστερα, αυτός ο τύπος spam ανιχνεύεται μέσω της εφαρμογής της δοκιμής CAPTCHA (Completely Automated Public Turing Test to Tell Computers and Humans Apart), η οποία αποσκοπεί στην αναγνώριση της διαφοράς μεταξύ υπολογιστών και ανθρώπων. Παρά το γεγονός αυτό, από πλευράς ασφάλειας, η μηχανική εκμάθηση μπορεί να παρακάμψει το CAPTCHA, ενώ από πλευράς χρηστικότητας, προκαλεί ενοχλήσεις στους νόμιμους χρήστες. Επομένως, η χρήση του CAPTCHA δεν θεωρείται η βέλτιστη λύση για την απόκρουση των web robots. Οι αποστολείς ανεπιθύμητης αλληλογραφίας εκμεταλλεύονται αυτήν την αδυναμία του CAPTCHA για να αναπτύξουν τεχνικές αποστολής ανεπιθύμητων μηνυμάτων μέσω ειδικών κωδικών αποκρυπτογράφησης, παρακάμπτοντας την προστασία και εξαπολύοντας επιθέσεις spambot που ενδέχεται να πλήξουν ακούσιους στόχους στο πλαίσιο του web 2.0. Επιπλέον, το Spambot μπορεί να κρύβεται σε νόμιμους ιστότοπους, παραπλανώντας τους επισκέπτες και οδηγώντας τους σε κακόβουλα περιβάλλοντα. Αυτό όχι μόνο απομακρύνει τους χρήστες από τους εν λόγω ιστότοπους, αλλά ενδέχεται επίσης να οδηγήσει σε μαυρισμένα καταλόγους για αυτούς τους ιστότοπους.

Οι αποστολείς ανεπιθύμητης αλληλογραφίας δεν περιορίζονται μόνο στη δημιουργία δικών τους ιστοσελίδων για ανεπιθύμητα μηνύματα (γνωστών και ως ανεπιθύμητα μηνύματα Ιστού), αλλά επίσης διανέμουν περιεχόμενο ανεπιθύμητης αλληλογραφίας σε πλατφόρμες Web 2.0, όπως διαδικτυακές κοινότητες, wiki, κοινωνικά δίκτυα, διαδικτυακούς πίνακες συζητήσεων κ.ά [82]. Οι πλατφόρμες συνεργασίας του Web 2.0, όπως τα διαδικτυακά φόρουμ συζήτησης, τα wiki και τα ιστολόγια, εκμεταλλεύονται συχνά από τους αποστολείς ανεπιθύμητης αλληλογραφίας για τη διάδοση περιεχομένου ανεπιθύμητης αλληλογραφίας. Η συγκεκριμένη τεχνική του spamming αναφέρεται ως Spam 2.0. Παραδείγματα του Spam 2.0 περιλαμβάνουν την ανάρτηση διαφημιστικών μηνυμάτων σε διαδικτυακούς πίνακες συζητήσεων, τη χειραγώγηση σελίδων wiki, τη δημιουργία ψευδών και ελκυστικών προφίλ χρηστών σε ιστοσελίδες διαδικτυακών κοινοτήτων, κ.ά [82].

Σύμφωνα με πρόσφατες ζωντανές αναφορές του [84], ο αριθμός των ανεπιθύμητων σχολίων στον Ιστό έχει διπλασιαστεί κατά την τελευταία χρονιά. Μέχρι σήμερα, οι spammers εκμεταλλεύονται νέες τεχνολογίες και στρατηγικές για την επίτευξη των σκοπών τους. Ένα παράδειγμα αυτών των εργαλείων είναι το web spambot (ή απλώς spambot), το οποίο αποτελεί έναν τύπο ρομπότ Ιστού που έχει αναπτυχθεί για τη διάδοση περιεχομένου ανεπιθύμητης αλληλογραφίας εκ μέρους των spammers [85]. Τα web spambot μπορούν να σαρώσουν τον Ιστό, να δημιουργήσουν λογαριασμούς χρηστών και να συμμετέχουν σε πλατφόρμες συνεργασίας, προωθώντας ανεπιθύμητο περιεχόμενο [86]. Τα ανεπιθύμητα μηνύματα όχι μόνο καταναλώνουν πολύτιμους πόρους, αλλά ενδέχεται επίσης να θέσουν σε κίνδυνο την αξιοπιστία ενός νόμιμου ιστότοπου, προκαλώντας τον αποκλεισμό του. Συνεπώς, η αναγνώριση και ανίχνευση των ανεπιθύμητων μηνυμάτων παραμένει ένα ενεργό πεδίο έρευνας. Είναι σημαντικό να σημειωθεί ότι τα web spambots διαφέρουν από τα ανεπιθύμητα μηνύματα που έχουν σκοπό τη συλλογή διευθύνσεων email από ιστοσελίδες. Για λόγους απλότητας, σε αυτή την περίπτωση, αναφερόμαστε στο web spambot ως spambot [87].

Τα υπάρχοντα μέτρα που επικεντρώνονται αποκλειστικά στην αναγνώριση και πρόληψη ανεπιθύμητου περιεχομένου δεν είναι επαρκή για εφαρμογή σε περιβάλλον Web 2.0 [82]. Για παράδειγμα, οι περισσότερες τεχνικές που βασίζονται στην ανάλυση περιεχομένου για την ανίχνευση ανεπιθύμητων μηνυμάτων ηλεκτρονικού ταχυδρομείου [88] ή μέθοδοι για τον εντοπισμό ανεπιθύμητης αλληλογραφίας στον ιστό, όπως η ανίχνευση με βάση συνδέσμους [89], το Trustrank [90] κ.λπ., δεν είναι αποτελεσματικές σε περιβάλλοντα Web 2.0. Αυτό συμβαίνει διότι, σε αντίθεση με τα παραδοσιακά ανεπιθύμητα

μηνύματα, το περιεχόμενο ανεπιθύμητης αλληλογραφίας στο Web 2.0 περιλαμβάνει στοιχεία που συχνά συμβάλλουν σε νόμιμους ιστότοπους [82].

2.4 ChatGPT

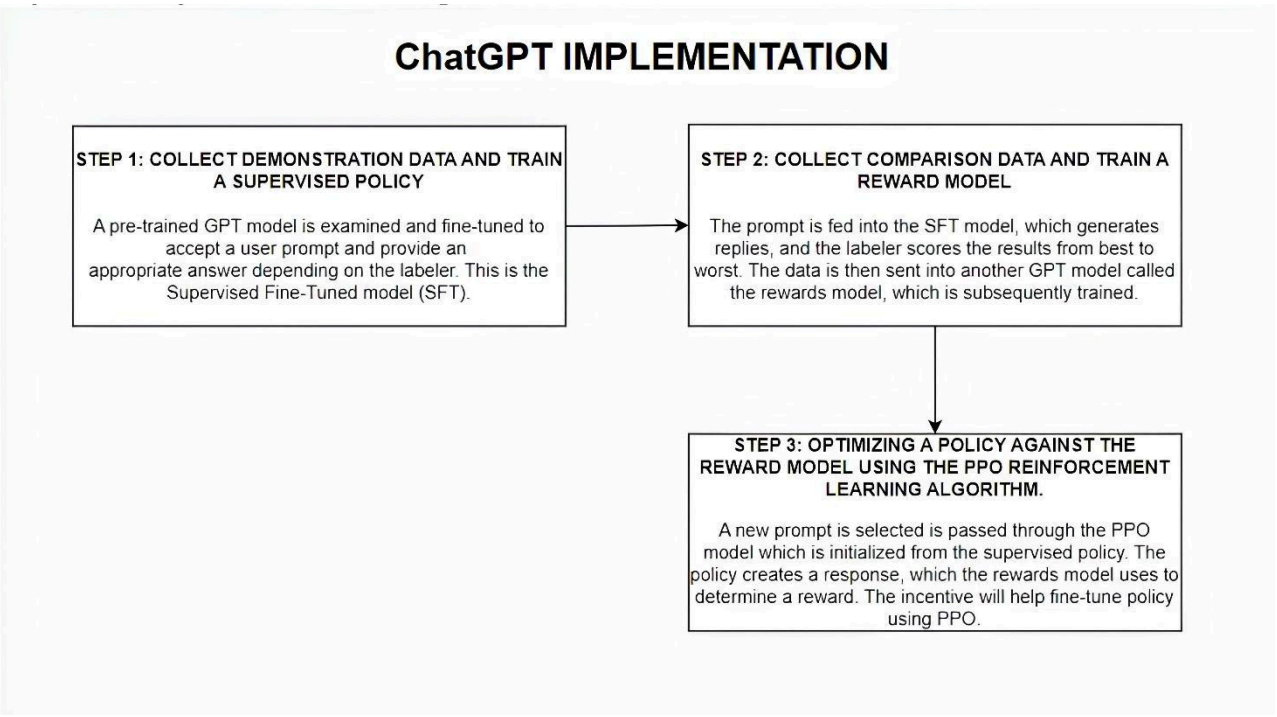
Έχουν καταγραφεί σημαντικές εξελίξεις και επιτεύγματα στον τομέα της Επεξεργασίας Φυσικής Γλώσσας, της Τεχνητής Νοημοσύνης και των Νευρωνικών Δικτύων. Μία από τις πλέον αξιοσημείωτες προόδους αφορά την ανάπτυξη προηγμένων γλωσσικών μοντέλων, όπως τα GPT-2 και GPT-3. Τα εν λόγω μοντέλα έχουν επιδείξει υψηλή απόδοση σε ποικίλες εφαρμογές NLP, συμπεριλαμβανομένης της απάντησης σε ερωτήματα, της δημιουργίας κειμένων και της κατανόησης φυσικής γλώσσας. Παρόλα αυτά, παρουσιάζουν ορισμένους περιορισμούς, όπως εγγενείς προκαταλήψεις και περιορισμένη ακρίβεια. Για την υπέρβαση αυτών των αδυναμιών, η ερευνητική κοινότητα έχει προχωρήσει στην ανάπτυξη διαδραστικών γλωσσικών μοντέλων, με χαρακτηριστικό παράδειγμα το Chat GPT, το οποίο αποτελεί βελτιωμένη εκδοχή των GPT-2 και GPT-3 [91].

Η OpenAI, που ιδρύθηκε το 2015, αποτελεί ένα ερευνητικό ίδρυμα αφιερωμένο στην προαγωγή καινοτόμων λύσεων τεχνητής νοημοσύνης προς όφελος της κοινωνίας [92]. Η οργάνωση έχει λάβει σημαντική χρηματοδοτική στήριξη από εξέχουσες προσωπικότητες, όπως ο αρχικός δωρητής Elon Musk, καθώς και από μεγάλες εταιρείες, συμπεριλαμβανομένης της Microsoft Corporation, η οποία επένδυσε ένα δισεκατομμύριο δολάρια ΗΠΑ εξασφαλίζοντας αποκλειστικά δικαιώματα πρόσβασης σε συγκεκριμένες τεχνολογίες της OpenAI [93]. Χάρη σε αυτήν τη στρατηγική υποστήριξη, το ίδρυμα έχει επιταχύνει την εξέλιξη των τεχνολογιών τεχνητής νοημοσύνης του. Στο πλαίσιο αυτό, η OpenAI έχει διαθέσει στο ευρύ κοινό μια σειρά προηγμένων προϊόντων μηχανικής μάθησης, με το DALL-E και το ChatGPT να συγκαταλέγονται μεταξύ των πιο διακεκριμένων [94].

Η τεχνολογία DALL-E, βασισμένη στη μηχανική μάθηση, επιτρέπει τη δημιουργία εικόνων με βάση τις εισροές των χρηστών και κατέστη ευρέως διαθέσιμη στο κοινό πριν από την αντίστοιχη διάθεση του ChatGPT [95]. Η λειτουργία της βασίζεται σε τεχνητά νευρωνικά δίκτυα με πολυτροπικούς νευρώνες, οι οποίοι διαθέτουν εξαιρετική ικανότητα κατανόησης και δημιουργίας νέων οπτικών συνθέσεων. Η υψηλή προσαρμοστικότητα αυτών των νευρώνων σε ποικίλες εκφράσεις μιας έννοιας επιτρέπει τη δημιουργία πλήθους εναλλακτικών απεικονίσεων [96]. Αυτή η πρωτοποριακή εξέλιξη, που προέρχεται από την OpenAI, αντικατοπτρίζει τη στρατηγική του οργανισμού για τον μετασχηματισμό του πεδίου της τεχνητής νοημοσύνης. Η δημόσια διαθεσιμότητα του DALL-E υπήρξε καταλυτικός παράγοντας για τη δημοφιλία του, ενώ παράλληλα συνέβαλε στην εκρηκτική απήχηση του ChatGPT, το οποίο προσέλκυσε πάνω από ένα εκατομμύριο μοναδικούς χρήστες μέσα σε μία μόλις εβδομάδα από την κυκλοφορία του [97].

Ο γενετικά προεκπαιδευμένος μετασχηματιστής (GPT) αποτελεί ένα προηγμένο μοντέλο γλώσσας, το οποίο έχει τη δυνατότητα να επεξεργάζεται και να κατανοεί τις ανθρώπινες εισόδους, παράγοντας στη συνέχεια κείμενο που προσομοιάζει τη φυσική ανθρώπινη γλώσσα σχεδόν αδιακρίτως [98]. Οι θεμελιώδεις αρχές λειτουργίας του GPT, αν και σύνθετες από τη σκοπιά της επεξεργασίας φυσικής γλώσσας και της μηχανικής μάθησης, θεωρούνται ότι έχουν βελτιστοποιηθεί από την OpenAI [99]. Η διαδικασία ανάπτυξης και βελτίωσης του αλγορίθμου ακολουθεί δύο βασικά στάδια: α) Αυτόνομη, μη εποπτευόμενη προεκπαίδευση, κατά την οποία το μοντέλο εκπαιδεύεται σε μεγάλα σύνολα μη επισημασμένων δεδομένων, επιτρέποντάς του να μάθει φυσικά μέσω εξερεύνησης, όπως συμβαίνει όταν ένα άτομο εισέρχεται σε ένα νέο περιβάλλον χωρίς προηγούμενες

γνώσεις και σταδιακά διαμορφώνει μια ολοκληρωμένη κατανόηση. Αυτή η προσέγγιση διαφέρει από τη διαδικασία εποπτευόμενης μάθησης, όπου η εκπαίδευση καθοδηγείται και επιμελείται, όπως συμβαίνει σε μια τάξη υπό την καθοδήγηση ενός εκπαιδευτικού [100]. β) Στοχευμένη, εποπτευόμενη βελτιστοποίηση, κατά την οποία, έπειτα από την αρχική προεκπαίδευση, το μοντέλο περνά από περαιτέρω προσαρμογή και εξειδίκευση από τους δημιουργούς του, ώστε να βελτιστοποιηθεί η απόδοσή του σε συγκεκριμένες εργασίες [101]. Το Σχήμα 8 παρουσιάζει, σε τεχνικό επίπεδο, τη λειτουργία του ChatGPT, βασισμένη σε ένα αρχικό μοντέλο GPT, καθώς και τις αρχές της εποπτευόμενης εκπαίδευσης με ακρίβεια, της μοντελοποίησης ανταμοιβών και της προσέγγισης εγγύς βελτιστοποίησης πολιτικής (PPO). Όπως προκύπτει από την ανάλυση, η ανάπτυξη μιας αποδοτικής και λειτουργικής διεπαφής για το ChatGPT αποτελεί μια επαναληπτική διαδικασία συνεχούς βελτίωσης.



Σχήμα 8: Διάγραμμα της διαδικασίας υλοποίησης GPT.

Το GPT αποτελεί ένα από τα πλέον προηγμένα γλωσσικά μοντέλα, καθώς βασίζεται σε εκτεταμένα δεδομένα και εξελιγμένους αλγορίθμους βαθιάς μάθησης. Η εκπαίδευσή του πραγματοποιείται μέσω πληθώρας διαδικτυακών πηγών, γεγονός που το καθιστά ένα από τα μεγαλύτερα μοντέλα επεξεργασίας φυσικής γλώσσας παγκοσμίως [102]. Χάρη στην ευελιξία του, μπορεί να εκτελεί ποικίλες γλωσσικές εργασίες, όπως δημιουργία κειμένων, απάντηση ερωτημάτων και μετάφραση, προσφέροντας απαντήσεις με υψηλό επίπεδο κατανόησης και συνοχής. Είναι ένα προηγμένο γλωσσικό μοντέλο που ανέπτυξε η OpenAI και βασίζεται στην τεχνολογία GPT [103]. Λειτουργεί ως ένα εξελιγμένο chatbot, ικανό να επεξεργάζεται και να ανταποκρίνεται σε ποικίλα αιτήματα που σχετίζονται με το κείμενο [96]. Πέρα από την απλή παροχή απαντήσεων, αξιοποιεί εκτενείς βάσεις δεδομένων και προηγμένους αλγορίθμους για να διαχειρίζεται πιο σύνθετα αιτήματα. Για παράδειγμα, μπορεί να δημιουργήσει εύστοχα διατυπωμένα μηνύματα, όπως μια ευχαριστήρια επιστολή προς έναν συνάδελφο ή μια σημείωση σχετική με τη βελτίωση της

παραγωγικότητας. Επιπλέον, μπορεί να συντάξει αναλυτικά κείμενα, όπως ένα δοκίμιο για την αξία της τεχνητής νοημοσύνης, με ελάχιστη ανθρώπινη παρέμβαση [104].

Μέσα σε λιγότερο από ένα λεπτό, το GPT είναι ικανό να παράγει ένα εκτενές δοκίμιο εκατοντάδων λέξεων, διατηρώντας την ποιότητα γραφής ενός επαγγελματία ερευνητή. Ένα άρθρο θα μπορούσε να δημιουργηθεί εξ ολοκλήρου από το GPT, με τη θεματική του να αναλύεται σε επιμέρους ενότητες, οι οποίες στη συνέχεια συντάσσονται αυτόνομα από το μοντέλο. Εάν αξιοποιηθεί πλήρως η υπολογιστική του δυνατότητα, μέσω μιας διευρυμένης έκδοσης που θα επέτρεπε τη σύνθεση κειμένων χιλιάδων λέξεων αντί για περιορισμένης έκτασης απαντήσεις, τότε μια ολόκληρη ερευνητική εργασία θα μπορούσε να ολοκληρωθεί μέσα σε λίγα λεπτά, με ελάχιστη καθοδήγηση από τον ερευνητή. Η εξέλιξη αυτή έχει τη δυνατότητα να μειώσει δραστικά τον χρόνο που απαιτείται για τη σύνταξη ερευνητικών κειμένων, από δεκάδες ώρες σε ελάχιστα δευτερόλεπτα, και ενδεχομένως να καταστήσει τον επαγγελματία συγγραφέα ή ερευνητή λιγότερο απαραίτητο [104].

Φυσικά, ακόμη και ανεξαρτήτως ηθικών ζητημάτων, το GPT αντιμετωπίζει ορισμένους περιορισμούς. Παρά το γεγονός ότι τα μοντέλα Επεξεργασίας Φυσικής Γλώσσας επιτυγχάνουν γενικά υψηλό επίπεδο ακρίβειας, εξακολουθούν να παρουσιάζονται σφάλματα στην ερμηνεία του νοήματος ή στη δημιουργία ακριβών πληροφοριών. Τα εν λόγω μοντέλα δεν είναι αλάνθαστα και το GPT έχει αναμφίβολα αντιμετωπίσει τέτοιες προκλήσεις [22], [105]. Για παράδειγμα, όροι με αμφίσημες ή πολλαπλές σημασίες (όπως η λέξη «κατασκευή»), καθώς και πιο σύνθετες έννοιες (όπως «ψηφιακός μετανάστης»), ενδέχεται να προκαλέσουν δυσκολίες στην ορθή ερμηνεία τους από το μοντέλο. Επιπλέον, οι αλγόριθμοι αυτοί, καθώς και οι βάσεις δεδομένων που τους υποστηρίζουν, απαιτούν σημαντικούς ενεργειακούς πόρους, ιδίως σε μεγάλη κλίμακα λειτουργίας, όπως αυτή που εφαρμόζει το OpenAI [106].

Περαιτέρω προκλήσεις προκύπτουν στο πεδίο της ηθικής ανάπτυξης και συντήρησης αυτών των τεχνολογιών. Το OpenAI έχει κληθεί να αντιμετωπίσει ζητήματα δεοντολογίας κατά τη δημιουργία του GPT και της πλατφόρμας ChatGPT, εγείροντας ερωτήματα σχετικά με την τήρηση αρχών υπεύθυνης τεχνολογικής ανάπτυξης [107]. Επιπροσθέτως, υπάρχει ο κίνδυνος το GPT να ενισχύσει προκαταλήψεις ή να συμβάλει στη διάδοση παραπληροφόρησης, δεδομένου ότι οι αλγόριθμοι NLP δεν διαθέτουν ακόμη επαρκή ικανότητα διάκρισης της ανακρίβειας στα δεδομένα εκπαίδευσής τους. Συνεπώς, δεν διαθέτουν μηχανισμούς αντίστασης όταν τους ζητείται να παράγουν ή να ενισχύσουν παραποιημένες ή διαστρεβλωμένες πληροφορίες [108], [109].

2.5 PhEmail Tool

Το απόρρητο μπορεί να οριστεί ως το δικαίωμα ενός ατόμου να ελέγχει και να διαχειρίζεται τις προσωπικές του πληροφορίες [110]. Η δυνατότητα ελέγχου αυτών των δεδομένων συμβάλλει ουσιαστικά στην αντιμετώπιση μιας από τις σημαντικότερες προκλήσεις του σύγχρονου Διαδικτύου: της διάσπασης της προσοχής λόγω του μεγάλου όγκου ανεπιθύμητης ηλεκτρονικής αλληλογραφίας (spam) [111]. Η έλλειψη ελέγχου στη διάθεση και διάχυση της διεύθυνσης ηλεκτρονικού ταχυδρομείου ενός χρήστη τον καθιστά ευάλωτο, καθώς οποιοσδήποτε στο Διαδίκτυο μπορεί να την αξιοποιήσει [110]. Δεδομένου ότι το κόστος αποστολής ηλεκτρονικών μηνυμάτων είναι εξαιρετικά χαμηλό, οι αποστολείς ανεπιθύμητης αλληλογραφίας κατακλύζουν το Διαδίκτυο με διαφημίσεις που αφορούν προϊόντα και εμπορικά είδη [110]. Ως αποτέλεσμα, οι χρήστες αναγκάζονται να αφιερώνουν σημαντικό μέρος του χρόνου και της προσοχής τους στη διάκριση των ουσιωδών μηνυμάτων από τα ανεπιθύμητα, γεγονός που επηρεάζει την παραγωγικότητα και την εμπειρία τους στο Διαδίκτυο [110].

Υπάρχουν πολλές τεχνικές για την αντιμετώπιση των ανεπιθύμητων μηνυμάτων, όπως μαύρες και λευκές λίστες, μέθοδοι βασισμένες στην τεχνητή νοημοσύνη και ο έλεγχος ταυτότητας χρηστών και διακομιστών SMTP. Επιπλέον, έχουν προταθεί οικονομικές προσεγγίσεις, όπως η επιβολή κόστους στην αποστολή email, καθώς και νομικές λύσεις. Ωστόσο, η νομοθετική ρύθμιση παρουσιάζει προκλήσεις λόγω της πολυπλοκότητας των δικαιωμάτων λόγου και της παγκόσμιας φύσης του Διαδικτύου, που υπερβαίνει τα εθνικά σύνορα [110].

Αναλύοντας το ρυθμιστικό μοντέλο του Lessig, το οποίο βασίζεται στην αρχιτεκτονική, τους κοινωνικούς κανόνες, την αγορά και τη νομοθεσία, γίνεται εμφανές ότι απουσιάζει μια κοινωνική προσέγγιση [112]. Προκειμένου να καλυφθεί αυτό το κενό, προτείνετε το Privacy Honoring Email, ένα σύστημα σχεδιασμένο με γνώμονα κυρίως τα κοινωνικά πρότυπα και την αντίληψη ότι η αρχιτεκτονική του email και του Διαδικτύου μπορεί να τροποποιηθεί μέσω κώδικα [110]. Στο πλαίσιο αυτό, διερευνούμε και ενισχύουμε τους κοινωνικούς κανόνες που διέπουν τις πρακτικές χρήσης του email ως μέσου επικοινωνίας μεταξύ των ανθρώπων [110].

Το PhEmail αποτελεί ένα εργαλείο ανοιχτού κώδικα, ανεπτυγμένο σε Python, το οποίο αυτοματοποιεί τη διαδικασία αποστολής μηνυμάτων ηλεκτρονικού ψαρέματος στο πλαίσιο δοκιμών κοινωνικής μηχανικής [30]. Σχεδιασμένο για να αξιολογεί την ανταπόκριση των χρηστών σε τέτοιες επιθέσεις, το PhEmail αποστέλλει μεγάλο αριθμό μηνυμάτων ηλεκτρονικού ψαρέματος και καταγράφει ποιοι χρήστες αλληλεπιδρούν με αυτά, χωρίς ωστόσο να επιχειρεί την εκμετάλλευση του προγράμματος περιήγησης ή του email client. Αντιθέτως, στοχεύει στη συλλογή του μέγιστου δυνατού όγκου πληροφοριών [113].

Οι αλληλεπιδράσεις των ανθρώπων δημιουργούν ίχνη στα κοινωνικά τους δίκτυα, τα οποία αποτυπώνονται μέσω επαφών, συνεργασιών, επικοινωνίας και μηνυμάτων (όπως email και άμεσα μηνύματα) [114], [115]. Καθώς οι αλληλεπιδράσεις γίνονται ολοένα και πιο ψηφιακές, αυτά τα ίχνη γίνονται ευκολότερα συλλέξιμα και αναλύσιμα. Αυτά τα δίκτυα μπορούν να χρησιμοποιηθούν για να βελτιώσουν τον έλεγχο πρόσβασης, όπως για παράδειγμα στο φιλτράρισμα του email, καθιστώντας πιο σημαντικά τα μηνύματα από άτομα του κοινωνικού δικτύου, ενώ τα υπόλοιπα θα παρακάμπτονται για μελλοντική ανάγνωση [110].

Τα διακριτικά που χρησιμοποιούνται στο σύστημα είναι κρυπτογραφημένα, διασφαλίζοντας την ασφάλεια των δεδομένων. Επιπλέον, το σύστημα λειτουργεί ως κλειστό περιβάλλον, επιτρέποντας την πρόσβαση στα tokens αποκλειστικά στον προβλεπόμενο παραλήπτη, ενώ παράλληλα καθιστά αδύνατη την πλαστογράφησή τους. Η έννοια του απορρήτου είναι δυναμική και δεν γίνεται αντιληπτή με τον ίδιο τρόπο από όλους τους χρήστες. Ενώ κάποιοι επιδεικνύουν μεγαλύτερη διάθεση διαμοιρασμού προσωπικών πληροφοριών, άλλοι επιθυμούν να διατηρούν αυστηρότερο έλεγχο αυτών. Ως εκ τούτου, σχεδιάστηκε αυτό το σύστημα ώστε να παρέχει τη δυνατότητα στους χρήστες να προσαρμόζουν δυναμικά το επίπεδο απορρήτου τους [110].

Στο πλαίσιο της διαχείρισης ηλεκτρονικού ταχυδρομείου, οι χρήστες έχουν τη δυνατότητα να καθορίζουν το εύρος φιλτραρίσματος των εισερχόμενων μηνυμάτων τους. Έτσι, με την πρόθεση να παρέχονται εργαλεία που θα επιτρέπουν στους χρήστες να διαμορφώνουν τις δικές τους πολιτικές απορρήτου, προσφέροντάς τους ενισχυμένο έλεγχο, ενώ παράλληλα λαμβάνουμε υπόψη τις προεπιλεγμένες ρυθμίσεις, αποφασίστηκαν τα εξής: 1) Να εφαρμοστεί ένα ήπιο φιλτράρισμα, ώστε όλα τα μηνύματα ηλεκτρονικού ταχυδρομείου να παραδίδονται κανονικά, με την επισήμανση των μηνυμάτων PHEmail. Οι χρήστες έχουν τη δυνατότητα να αυστηροποιήσουν το φιλτράρισμα, εφόσον το επιθυμούν και 2) Να επισυνάπτεται στα εξερχόμενα μηνύματα το διακριτικό του

κατόχου. Δεδομένου ότι μόνο οι κάτοχοι των διακριτικών έχουν τη δυνατότητα να προβάλλουν και να διαχειρίζονται τα διακριτικά τους, η κατοχή ενός διακριτικού δεν συνεπάγεται απαραίτητα πρόσβαση στα δεδομένα του κατόχου. Έτσι, διασφαλίζεται η δυνατότητα ανταλλαγής διακριτικών χωρίς να τίθεται σε κίνδυνο το απόρρητο των χρηστών [110].

Η αρχική ανάπτυξη του συστήματος βασίζεται στην αρχή του Bob Metcalfe ότι η αξία ενός δικτύου αυξάνεται εκθετικά με τον αριθμό των χρηστών του. Έτσι, η αποδοτικότητα και η αξία του Privacy Honoring Email (PHEmail) θα ενισχυθούν καθώς αυξάνεται η χρήση του. Δεδομένου ότι το υπάρχον αρχείο ηλεκτρονικού ταχυδρομείου ενός ατόμου περιλαμβάνει μεγάλο μέρος του κοινωνικού του δικτύου, η ενσωμάτωσή του μπορεί να ξεκινήσει άμεσα. Η διαδικασία σποράς του συστήματος βασίζεται στην ανάλυση του αρχείου email (εφόσον υπάρχει), ενώ οι χρήστες που επικοινωνούν μέσω PHEmail ανταλλάσσουν αυτόματα διακριτικά και ενισχύουν τα κοινωνικά ίχνη της πλατφόρμας [110].

Σε κοινωνικές εισαγωγές, όταν ένα άτομο (Α) επιθυμεί να επικοινωνήσει με κάποιον (Γ) που δεν γνωρίζει, συχνά βασίζεται σε έναν κοινό γνωστό (Β) για τη σύσταση. Μέσω του PHEmail, το άτομο Α ζητά από το Β το διακριτικό του Γ, το οποίο του επιτρέπει να του αποστείλει email. Όταν ο Γ λάβει το μήνυμα, το σύστημα αναγνωρίζει τη σύνδεση μέσω του Β και επιτρέπει στο Γ να εφαρμόσει φίλτρα προτεραιότητας, όπως η προώθηση μηνυμάτων από άτομα που βρίσκονται εντός συγκεκριμένης κοινωνικής απόστασης [110].

Πιο συγκεκριμένα, το εργαλείο αυτό αξιοποιεί δεδομένα από το LinkedIn για τη συλλογή διευθύνσεων email, γεγονός που το καθιστά ιδιαίτερα χρήσιμο κατά τη φάση της αναγνώρισης και συγκέντρωσης πληροφοριών. Επιπλέον, ενσωματώνει μια μηχανή ελέγχου ταυτότητας Gmail, η οποία επιτρέπει την αποστολή μηνυμάτων μέσω έγκυρου λογαριασμού Gmail, παρέχοντας μια εναλλακτική σε περιπτώσεις όπου ο τομέας-στόχος έχει αποκλείσει το email προέλευσης ή τη διεύθυνση IP. Ένα από τα πλέον κρίσιμα χαρακτηριστικά του PhEmail είναι η δυνατότητα κλωνοποίησης εταιρικών πυλών σύνδεσης, με σκοπό την υποκλοπή διαπιστευτηρίων πρόσβασης. Αυτή η λειτουργικότητα αποτελεί ένα από τα πιο ισχυρά στοιχεία του εργαλείου, ενισχύοντας την αποτελεσματικότητά του σε δοκιμές ασφάλειας και επιθέσεις κοινωνικής μηχανικής [30].

Σε αυτό το εργαλείο γραμμής εντολών, ο χρήστης οφείλει να καθορίσει τη διαδρομή προς ένα αρχείο κειμένου που περιέχει τις διευθύνσεις email των παραληπτών (Default: emails.txt), καθώς και το αρχείο με το περιεχόμενο του μηνύματος ηλεκτρονικού ψαρέματος (Default: body.txt). Επιπλέον, απαιτείται η παροχή της διεύθυνσης αποστολέα που θα εμφανίζεται στο πεδίο FROM του email (Default: Name Surname <name_surname@example.com>), της πραγματικής διεύθυνσης email που θα χρησιμοποιηθεί για την αποστολή και λήψη απαντήσεων (Default: Name Surname), καθώς και του θέματος (Default: Newsletter) και του περιεχομένου του email (Default: body.txt). Οι απαντήσεις που ενδέχεται να σταλούν από τους παραλήπτες αποθηκεύονται σε σχετικό αρχείο καταγραφής [30]. Η βασική μορφή εντολής του εργαλείου είναι η εξής:

phemail.py [-e <emails>] [-m <mail_server>] [-f <from_address>] [-r<replay_address>] [-s <subject>] [-b <body>]

Το PhEmail επιτρέπει την αποστολή του ίδιου μηνύματος σε πολλούς παραλήπτες ταυτόχρονα, ωστόσο απαιτεί τη χειροκίνητη εισαγωγή των απαραίτητων παραμέτρων από τον χρήστη. Παρέχεται επίσης η δυνατότητα προσαρμογής επιπλέον ρυθμίσεων, όπως: 1) Καθορισμός τομέα διευθύνσεων email, 2) Σύνδεσμος προορισμού του phishing email, 3) Όριο αποστολής email ανά σύνδεση, 4) Ενεργοποίηση αναλυτικής καταγραφής συμβάντων (Verbose Mode, Default: false), 5) Αποστολή email χωρίς ενσωματωμένες

εικόνες, 6) Προσθήκη hook για BeEF, 7) Χρήση λογαριασμού Google με όνομα χρήστη/κωδικό πρόσβασης, 8) Καθορισμός χρονικής καθυστέρησης μεταξύ διαδοχικών email (προεπιλεγμένο: 3 δευτερόλεπτα), 9) Μαζική αποστολή πολλών email ταυτόχρονα (προεπιλεγμένο: 10 email ανά φορά), 10) Δυνατότητα εκτέλεσης αναζητήσεων στο Google. Αυτό το εργαλείο επιτρέπει την προσαρμογή της αποστολής email, προσφέροντας διάφορες δυνατότητες διαμόρφωσης, ανάλογα με τις ανάγκες του χρήστη [30].

2.6 LLaMA

Τα Μεγάλα Γλωσσικά Μοντέλα [116] έχουν προσελκύσει έντονο ερευνητικό ενδιαφέρον στην επιστημονική κοινότητα της Τεχνητής Νοημοσύνης, εξαιτίας της εξαιρετικής τους ικανότητας στην κατανόηση, επεξεργασία και παραγωγή φυσικής γλώσσας [117]. Τα LLM που εκπαιδεύονται σε μεγάλα σύνολα δεδομένων έχουν αποδείξει την ικανότητά τους να εκτελούν νέες εργασίες με βάση οδηγίες ή παραδείγματα [22]. Αυτές οι δυνατότητες εμφανίστηκαν όταν τα μοντέλα κλιμακώθηκαν σε μεγαλύτερα μεγέθη [118], οδηγώντας σε έρευνες για τη περαιτέρω κλιμάκωση αυτών [119], [120]. Ωστόσο, μελέτες [121] δείχνουν ότι, με δεδομένο έναν υπολογιστικό προϋπολογισμό, τα καλύτερα αποτελέσματα δεν επιτυγχάνονται από τα μεγαλύτερα μοντέλα, αλλά από μικρότερα που εκπαιδεύονται σε περισσότερα δεδομένα. Η εργασία των Hoffmann et al. [121] εστιάζει στον προσδιορισμό του τρόπου κλιμάκωσης δεδομένων και μοντέλων για συγκεκριμένο προϋπολογισμό εκπαίδευσης. Ωστόσο, παραβλέπει τη σημασία του προϋπολογισμού συμπερασμάτων, ο οποίος είναι καθοριστικός στην κλίμακα εξυπηρέτησης των μοντέλων. Ειδικότερα, όταν επιδιώκεται ένας στόχος απόδοσης, το πιο γρήγορο μοντέλο σε συμπεράσματα είναι προτιμότερο, ακόμα κι αν απαιτεί λιγότερο χρόνο εκπαίδευσης. Για παράδειγμα, παρά τη σύσταση των Hoffmann et al. [121] για εκπαίδευση ενός μοντέλου 10B με 200B δεδομένα, η απόδοση ενός μοντέλου 7B συνεχίζει να βελτιώνεται ακόμα και με 1T δεδομένα.

Τα μοντέλα LLaMA, με παραμέτρους από 7B έως 65B, προσφέρουν ανταγωνιστική απόδοση σε σχέση με τα κορυφαία LLM. Ενδεικτικά, το LLaMA-13B υπερτερεί του GPT-3 σε πολλές δοκιμές, παρά το ότι είναι δέκα φορές μικρότερο. Θεωρούμε ότι αυτό το μοντέλο θα διευκολύνει την ευρεία πρόσβαση και μελέτη των LLM, καθώς μπορεί να εκτελείται σε μία μόνο GPU. Στην υψηλότερη κλίμακα, το μοντέλο με 65B παραμέτρους είναι επίσης ανταγωνιστικό με τα πιο προηγμένα μοντέλα, όπως το Chinchilla και το PaLM-540B [122].

Για τη βελτίωση της εκφραστικότητας και της πειστικότητας των αποκρίσεων των Μεγάλων Γλωσσικών Μοντέλων, πρόσφατες μελέτες [123], [124], [125], [126], [127] έχουν επικεντρωθεί στον μετασχηματισμό των LLMs σε μοντέλα καθοδηγούμενα από οδηγίες (instruction-following). Ενδεικτικά, το έργο Stanford Alpaca [126] πραγματοποιεί βελτιστοποίηση του LLaMA [122] ώστε να ακολουθεί οδηγίες, αξιοποιώντας δεδομένα οδηγιών που έχουν δημιουργηθεί από το μοντέλο InstructGPT της OpenAI [128]. Μεταγενέστερες προσεγγίσεις επεκτείνουν περαιτέρω το LLaMA, ενσωματώνοντας δεδομένα υψηλότερης ποιότητας, όπως το ShareGPT [129] και σύνολα δεδομένων που έχουν παραχθεί από το GPT-4 [124]. Σε αντίθεση με τις πλήρους κλίμακας διαδικασίες εκπαίδευσης που εφαρμόζονται από έργα όπως το Alpaca και το Vicuna [127], η προσέγγιση LLaMA-Adapter [130] εισάγει ελαφρούς προσαρμογείς με μηδενική αρχικοποίηση στις προσοχές του παγωμένου LLaMA, επιτυγχάνοντας αποδοτικότερη παραμετροποίηση και δυνατότητα έγχυσης πολυτροπικής γνώσης. Παρά τη σημαντική πρόοδο, οι υφιστάμενες μέθοδοι εξακολουθούν να παρουσιάζουν περιορισμούς στην εκτέλεση προηγμένων πολυτροπικών οδηγιών, όπως για παράδειγμα στην οπτική κατανόηση, δυνατότητα που προσφέρει το GPT-4 [28].

Πιο πρόσφατες ερευνητικές προσπάθειες, όπως τα MiniGPT-4 [131] και LLaVA [132], εγκαινιάζουν μια νέα κατεύθυνση στην επέκταση των γλωσσικών μοντέλων σε πολυτροπικά, προσδίδοντας στα LLMs την ικανότητα οπτικής συλλογιστικής, σε ευθυγράμμιση με την αρχιτεκτονική του LLaMA-Adapter. Το MiniGPT-4 επιτυγχάνει αυτή τη σύνδεση μέσω της ενσωμάτωσης ενός παγωμένου οπτικού κωδικοποιητή με ένα LLM, χρησιμοποιώντας προκαταρκτική προεκπαίδευση σε 134 εκατομμύρια ζεύγη εικόνας-κειμένου και περαιτέρω βελτιστοποίηση σε επιμελώς ευθυγραμμισμένο σύνολο δεδομένων. Αντίστοιχα, το LLaVA αξιοποιεί ζεύγη εικόνας-κειμένου για την ευθυγράμμιση του οπτικού συστήματος με το LLM. Σε αντίθεση με το MiniGPT-4, το LLaVA πραγματοποιεί πλήρη βελτιστοποίηση του LLM, βασιζόμενο σε 150.000 υψηλής ποιότητας πολυτροπικές οδηγίες που έχουν παραχθεί από το GPT-4. Παρότι οι ανωτέρω μέθοδοι παρουσιάζουν σημαντικές δυνατότητες στην κατανόηση πολυτροπικών εισόδων, εντούτοις απαιτούν τον ανασχεδιασμό και εκ νέου εκπαίδευση ενός μεγάλου αριθμού παραμέτρων, καθώς και τη δημιουργία ευρύτατων συνόλων πολυτροπικών δεδομένων, είτε μέσω ανθρώπινης επιμέλειας είτε μέσω εξόδου από το OpenAI API.

2.7 Unsloth

Η Unsloth αποτελεί μία προηγμένη πλατφόρμα που επιταχύνει σημαντικά τη διαδικασία προσαρμοσμένης εκπαίδευσης (fine-tuning) μεγάλων γλωσσικών μοντέλων, όπως τα Llama-3, Mistral, Phi-4 και Gemma, επιτυγχάνοντας διπλάσια ταχύτητα, μείωση της απαιτούμενης μνήμης έως και 70%, χωρίς καμία απώλεια στην ακρίβεια των αποτελεσμάτων. Παρέχει μια ολοκληρωμένη και αυτοματοποιημένη ροή εργασιών, καλύπτοντας όλα τα στάδια της εκπαίδευσης, από τη φόρτωση και ποσοτικοποίηση του μοντέλου, έως την εκτέλεση, αξιολόγηση, αποθήκευση, εξαγωγή και διασύνδεση με μηχανές εκτέλεσης όπως οι Ollama, llama.cpp και vLLM. Η αξιοπιστία της πλατφόρμας ενισχύεται από τη σημαντική της συνεισφορά στη διόρθωση κρίσιμων σφαλμάτων σε δημοφιλή μοντέλα, μέσω στενής συνεργασίας με οργανισμούς όπως η Meta (Llama 4), η Google (Gemma), η Microsoft (Phi-3/4) και άλλους. Χαρακτηριστικά παραδείγματα περιλαμβάνουν τη βελτίωση της ακρίβειας κατά 2% στο Llama 4, διορθώσεις στους μηχανισμούς RoPE και στα πρότυπα διαλόγου, καθώς και τον εντοπισμό μη αρχικοποιημένων παραμέτρων στο Gemma 3. Η Unsloth είναι πλήρως παραμετροποιήσιμη, υποστηρίζοντας προσαρμογές σε πρότυπα διαλόγου και μορφοποίηση δεδομένων, ενώ προσφέρει έτοιμα εργαλεία για εφαρμογές όπως όραση υπολογιστή, μετατροπή κειμένου σε φωνή, BERT και ενισχυτική μάθηση, υποστηρίζοντας παράλληλα κάθε μοντέλο βασισμένο σε αρχιτεκτονική transformers και όλα τα είδη εκπαίδευσης. Η διαδικασία fine-tuning επιτρέπει την ουσιαστική προσαρμογή ενός προκαταρτικά εκπαιδευμένου μοντέλου στις ανάγκες ενός εξειδικευμένου πεδίου, ενισχύοντας τη γνώση του, ρυθμίζοντας το στυλ ή τη συμπεριφορά των απαντήσεών του και βελτιστοποιώντας την απόδοσή του σε στοχευμένες εργασίες. Για παράδειγμα, ένα μοντέλο μπορεί να αξιοποιεί το ιστορικό των πελατών για ακριβέστερες και προσωποποιημένες απαντήσεις ή να μαθαίνει πάνω σε νομικά κείμενα έτσι ώστε να έχει τη δυνατότητα να αναλύει συμβόλαια, νομικές έρευνες και κανονισμούς. Ένα fine-tuned μοντέλο λειτουργεί ως εξειδικευμένος ψηφιακός πράκτορας, ικανός να ανταπεξέρχεται αποτελεσματικά σε συγκεκριμένες απαιτήσεις, καλύπτοντας πλήρως τις δυνατότητες που προσφέρουν τεχνικές όπως το Retrieval-Augmented Generation (RAG), αλλά με μεγαλύτερη ευελιξία και απόδοση [133].

3ο ΚΕΦΑΛΑΙΟ: ΠΕΡΙΓΡΑΦΗ ΛΟΓΙΣΜΙΚΟΥ

3.1 Διαδικασία εκπαίδευσης και επιλογή Μεγάλου Γλωσσικού Μοντέλου

Η ραγδαία πρόοδος στον τομέα της Τεχνητής Νοημοσύνης και, ειδικότερα, των Μοντέλων Μεγάλης Κλίμακας Γλώσσας, έχει επιτρέψει την ανάπτυξη εργαλείων ικανών να επιλύουν σύνθετα προβλήματα επεξεργασίας φυσικής γλώσσας [134]. Ωστόσο, τα γενικά εκπαιδευμένα LLMs ενδέχεται να μην αποδίδουν ικανοποιητικά σε συγκεκριμένα γνωστικά πεδία ή ειδικές εφαρμογές. Η προσαρμογή τους μέσω fine-tuning καθίσταται κρίσιμη ώστε να ενισχυθεί η εξειδίκευσή τους και να εξασφαλιστεί μεγαλύτερη ακρίβεια σε στοχευμένα σενάρια χρήσης [135]. Στο πλαίσιο της παρούσας εργασίας, χρησιμοποιήθηκε η τεχνική του fine-tuning σε συνδυασμό με τη βιβλιοθήκη Unsloth, αξιοποιώντας ως βάση το προεκπαιδευμένο μοντέλο LLaMA 3 (8B) σε περιβάλλον Google Colab.

Το fine-tuning αποτελεί μια ευρέως χρησιμοποιούμενη προσέγγιση που βασίζεται στην περαιτέρω εκπαίδευση ενός ήδη προεκπαιδευμένου μοντέλου (π.χ. LLaMA, GPT, BERT) σε νέο, εξειδικευμένο σύνολο δεδομένων. Η πρακτική αυτή επιτρέπει την προσαρμογή του μοντέλου σε συγκεκριμένα καθήκοντα, χωρίς την ανάγκη πλήρους εκπαίδευσης από το μηδέν [134]. Σκοπός του fine-tuning είναι να ενισχύσει τη γενική κατανόηση που έχει ήδη αναπτύξει το μοντέλο με επιπλέον γνώση προσαρμοσμένη στο επιθυμητό πεδίο εφαρμογής. Τα κύρια πλεονεκτήματα του fine-tuning περιλαμβάνουν: 1) Μειωμένο υπολογιστικό κόστος, 2) Ταχύτερη προσαρμογή σε domain-specific περιεχόμενο, 3) Διατήρηση της γενικής γλωσσικής κατανόησης του αρχικού μοντέλου [135]. Όπως αποδεικνύουν οι Howard & Ruder [135], η εφαρμογή fine-tuning πάνω σε προεκπαιδευμένα γλωσσικά μοντέλα αποδίδει καλύτερα σε downstream tasks σε σύγκριση με την εκπαίδευση από το μηδέν.

Η βιβλιοθήκη Unsloth επελέγη λόγω της βελτιστοποίησής της για αποδοτικό fine-tuning μοντέλων τύπου LLaMA, ειδικά για εκτέλεση σε περιβάλλοντα με περιορισμένους πόρους, όπως το Google Colab. Παρέχει συμβατότητα με τις πλέον σύγχρονες τεχνικές, όπως: 1) QLoRA (Quantized Low-Rank Adapter Fine-Tuning): Η QLoRA είναι μια τεχνική που επιτρέπει την εκπαίδευση LLMs με πολύ περιορισμένους πόρους, μέσω της χρήσης 4-bit quantization και low-rank updates [136], 2) Συμβατότητα με βιβλιοθήκες όπως Transformers, PEFT, και Hugging Face: Επιτρέπει την απρόσκοπτη ενσωμάτωση σε υπάρχουσες ροές εργασίας και datasets, 3) Ειδικά σχεδιασμένες λειτουργίες για χρήση σε Google Colab με GPU acceleration [137], 4) Εκτέλεση σε GPU με μειωμένες απαιτήσεις μνήμης: Η Unsloth υποστηρίζει fine-tuning ακόμα και σε NVIDIA T4 ή A100, χωρίς την ανάγκη για server-grade hardware. Η χρήση της Unsloth είναι κομβική για την επίτευξη υψηλής αποδοτικότητας, μειώνοντας έως και 80% τη χρήση VRAM, γεγονός που καθιστά το fine-tuning εφικτό για μικρότερες ερευνητικές ομάδες ή έργα με περιορισμένο προϋπολογισμό.

Το μοντέλο LLaMA 3 (8B), που κυκλοφόρησε από τη Meta AI το 2024, αποτελεί μια από τις πιο εξελιγμένες οικογένειες ανοιχτών LLMs η οποία είναι διαθέσιμη σε παραλλαγές των 8B και 70B παραμέτρων και επελέγη για την παρούσα εργασία λόγω της ιδανικής αναλογίας απόδοσης που προσφέρει ως προς το υπολογιστικό κόστος. Η έκδοση 8B αποτελεί μια ελαφριά αλλά ισχυρή παραλλαγή και επιλέχθηκε για τους εξής λόγους: 1) Υποστηρίζει αξιόπιστη κατανόηση φυσικής γλώσσας, reasoning και επιτρέπει ελεύθερη χρήση και τροποποίηση [138, p. 2], 2) Διατίθεται με ανοικτό κώδικα και άδεια χρήσης, επιτρέποντας πλήρη παραμετροποίηση και fine-tuning, 3) Υποστηρίζεται από το εργαλείο Ollama, το οποίο διευκολύνει την τοπική ανάπτυξη και αξιοποίηση LLMs σε περιβάλλοντα

χωρίς σύνδεση [139], 4) Προσφέρει εξαιρετική ποιότητα απαντήσεων, συγκρίσιμη με μεγαλύτερα μοντέλα, με πολύ μικρότερες απαιτήσεις σε υπολογιστικούς πόρους.

Η εκπαίδευση του μοντέλου πραγματοποιήθηκε στο Google Colab, μια υπηρεσία που παρέχει πρόσβαση σε υπολογιστικούς πόρους υψηλής απόδοσης με ελάχιστο κόστος. Χρησιμοποιήθηκε ως υποδομή εκπαίδευσης λόγω των εξής πλεονεκτημάτων: 1) Παρέχει δωρεάν πρόσβαση σε GPU/TPU, όπως NVIDIA T4, V100 και A100, κατάλληλες για fine-tuning μοντέλων μικρού και μεσαίου μεγέθους με το Unsloth, 2) Απλότητα στην εγκατάσταση και χρήση εργαλείων machine learning, όπως PyTorch και Hugging Face Transformers, 3) Συνεργατική φύση των Google Notebooks, που επιτρέπουν παράλληλη εργασία και αποθήκευση στο cloud [140], 4) Επιτρέπει γρήγορη επανεκκίνηση, debugging και πειραματισμό σε διαφορετικά configurations, κάτι κρίσιμο κατά την πειραματική φάση της εκπαίδευσης, 5) Διατίθεται στο cloud, χωρίς ανάγκη φυσικής εγκατάστασης. Η επιλογή του Colab συνδυάστηκε ιδανικά με τις δυνατότητες της Unsloth, επιτρέποντας την εκπαίδευση του LLaMA 3 (8B) χωρίς την ανάγκη τοπικών υποδομών υψηλού κόστους.

Η εκπαιδευτική διαδικασία που ακολουθήθηκε βασίστηκε σε τεκμηριωμένες και αποδοτικές τεχνικές, με στόχο την επίτευξη προσαρμοσμένων και ελαφριών μοντέλων γλώσσας για ειδικές εφαρμογές. Η ενσωμάτωση τεχνικών όπως το fine-tuning μέσω της βιβλιοθήκης Unsloth, σε συνδυασμό με το μοντέλο LLaMA 3 (8B) και το υπολογιστικό περιβάλλον του Google Colab, αποδείχθηκε μια αποδοτική και πρακτική προσέγγιση προσφέροντας υψηλή αποδοτικότητα και ακρίβεια, σημαντική εξοικονόμηση πόρων και ευελιξία στην ανάπτυξη και την επεκτασιμότητα. Οι επιλογές αυτές κατέστησαν εφικτή την εξειδίκευση ενός ισχυρού γλωσσικού μοντέλου με περιορισμένους υπολογιστικούς πόρους, χωρίς συμβιβασμό στην απόδοση.

Ακολούθως παρουσιάζεται η λεπτομερής περιγραφή της διαδικασίας της εκπαίδευσης:

Η εκπαίδευση του μοντέλου LLaMA 3 (8B) ξεκινά με τη χρήση ενός προ-εκπαιδευμένου βασικού μοντέλου. Αυτό το βασικό μοντέλο έχει ήδη εκπαιδευτεί εκτενώς σε ένα τεράστιο σώμα ποικίλων κειμενικών δεδομένων, τα οποία του προσδίδουν μια ευρεία κατανόηση της δομής της γλώσσας, της σημασιολογίας και των γενικών γνώσεων. Το προ-εκπαιδευμένο μοντέλο χρησιμεύει ως σημείο εκκίνησης, επιτρέποντας στη διαδικασία εκπαίδευσης να επικεντρωθεί στην προσαρμογή των δυνατοτήτων του μοντέλου σε συγκεκριμένες εργασίες ή σύνολα δεδομένων αντί για εκπαίδευση από την αρχή, κάτι που θα ήταν υπολογιστικά απαγορευτικό.

Στιγμιότυπο κώδικα:

```
from unsloth import FastLanguageModel
model, tokenizer =
FastLanguageModel.from_pretrained(
   model_name = “unsloth/
llama-3-8b-Instruct-bnb-4bit”,
   max_seq_length = 4096,
   dtype = None,
   load_in_4bit = True,
)
```

Στη συνέχεια, η διαδικασία βελτιστοποίησης διεξάγεται χρησιμοποιώντας το πλαίσιο Ollama. Το Ollama διευκολύνει την αποτελεσματική προσαρμογή μεγάλων γλωσσικών μοντέλων χρησιμοποιώντας μεθοδολογίες βελτιστοποίησης με αποδοτικότητα

παραμέτρων, όπως η Προσαρμογή Χαμηλού Βαθμού (LoRA). Αυτές οι τεχνικές περιλαμβάνουν την προσθήκη μικρών, εκπαιδεύσιμων μονάδων προσαρμογέα (adapter modules) στο προ-εκπαιδευμένο μοντέλο, επιτρέποντας την τροποποίηση της συμπεριφοράς του μοντέλου χωρίς την ανάγκη ενημέρωσης όλων των παραμέτρων.

Στιγμιότυπο κώδικα:
```
model = FastLanguageModel.get_peft_model(
   model,
   r = 64,
   target_modules = [“q_proj”,”k_proj”,”v_proj,”o_proj””],
   lora_alpha = 16,
   lora_dropout = 0,
   bias = “none”,
   use_gradient_checkpointing = True,
   random_state = 3407,
      max_seq_length = 4096,
  )
```

Αυτό μειώνει σημαντικά τις υπολογιστικές απαιτήσεις και τον χρόνο εκπαίδευσης, διατηρώντας παράλληλα τα κέρδη απόδοσης.
Πριν από τη βελτιστοποίηση, το σύνολο δεδομένων εκπαίδευσης (training dataset) προετοιμάζεται προσεκτικά. Αυτό το σύνολο δεδομένων συνήθως αποτελείται από ζευγαρωμένα παραδείγματα/προτροπές (paired examples/prompts) και τις αντίστοιχες επιθυμητές απαντήσεις τους. Στην παρούσα εργασία, το dataset που χρησιμοποιήθηκε είναι συνθετικό και δημιουργήθηκε εξ ολοκλήρου από εμάς. Η σωστή προεπεξεργασία, συμπεριλαμβανομένης της δημιουργίας διακριτικών (tokenization) και της μορφοποίησης σε ζεύγη εντολών-απαντήσεων, διασφαλίζει ότι τα δεδομένα βρίσκονται σε κατάλληλη μορφή για την πρόσληψη του μοντέλου.

Στιγμιότυπο κώδικα:
```
from datasets import Dataset

           dataset = Dataset.from_list([
     {“instruction”: “Ποια είναι η πρωτεύουσα της Γαλλίας;”, “output”: “Η
     πρωτεύουσα της Γαλλίας είναι το Παρίσι.”},
   {“instruction”: “Μετάφρασε στα αγγλικά: Καλημέρα!”, “output”: “Good
morning!”},
])

from trl import SFTTrainer
from unsloth import patch_dataset
dataset = patch_dataset(dataset, tokenizer,
formatting_func = “alpaca”)
```

Κατά τη φάση εκπαίδευσης, το μοντέλο υφίσταται εποπτευόμενη μάθηση (supervised learning), όπου παρουσιάζεται με τις εντολές εισόδου και εκπαιδεύεται για να παράγει τις σχετικές απαντήσεις. Ο στόχος της εκπαίδευσης είναι η ελαχιστοποίηση της απώλειας διασταυρούμενης εντροπίας (cross-entropy loss) μεταξύ των προβλεπόμενων

κατανομών διακριτικών του μοντέλου και των πραγματικών διακριτικών/στόχων στην απάντηση.

Στιγμιότυπο κώδικα:

```
trainer = SFTTrainer(
   model = model,
   tokenizer = tokenizer,
   train_dataset = dataset,
   dataset_text_field = “text”,
   max_seq_length = 4096,
   dataset_num_proc = 2,
   packing = False,
   args = TrainingArguments(
      per_device_train_batch_size = 2,
      gradient_accumulation_steps = 4,
      warmup_steps = 5,
      num_train_epochs = 1,
      learning_rate = 2e-4,
      logging_steps = 1,
      save_strategy = “no”,
      output_dir = “outputs”,
      optim = “adamw_8bit”,
      lr_scheduler_type = “linear”,
      fp16 = not
   torch.cuda.is_bf16_supported(),
      bf16 = torch.cuda.is_bf16_supported(),
   ),
)
trainer.train()
```

Ενημερώνοντας επαναληπτικά τις παραμέτρους του προσαρμογέα (adapter parameters) για να μειώσει αυτήν την απώλεια, το μοντέλο βελτιώνει σταδιακά την ικανότητά του να παράγει ακριβείς και κατάλληλες για τα συμφραζόμενα απαντήσεις.
Καθ' όλη αυτή τη διαδικασία, το Ollama διαχειρίζεται κρίσιμα στοιχεία όπως η ομαδοποίηση, ο υπολογισμός διαβάθμισης, τα βήματα βελτιστοποίησης και ο έλεγχος των βαρών του μοντέλου. Εστιάζοντας στην τελειοποίηση μόνο των επιπέδων του προσαρμογέα, το πλαίσιο επιτυγχάνει αποτελεσματική χρήση της μνήμης και των υπολογιστικών πόρων. Με την ολοκλήρωση της εκπαίδευσης, το αποτέλεσμα είναι μια βελτιωμένη έκδοση του μοντέλου LLaMA 3 (8B), η οποία είναι καλύτερα ευθυγραμμισμένη με τις συγκεκριμένες απαιτήσεις του συνόλου δεδομένων και της εργασίας, επιδεικνύοντας βελτιωμένη απόδοση σε σχετικά ερωτήματα ή εφαρμογές.

3.2 Σετ δεδομένων

Το σετ δεδομένων (dataset) που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου είναι συνθετικό και έχει δημιουργηθεί εξ ολοκλήρου από την ερευνητική ομάδα [141]. Είναι δομημένο με τη μορφή οδηγιών (Instruction Mode), μιας και αποτελεί έναν από τους

τρόπους που χρησιμοποιείται για τη προσαρμοσμένη εκπαίδευση ενός γλωσσικού μοντέλου [142].

Κάθε εγγραφή του dataset έχει τη παρακάτω δομή:

```
"Instruction": "Τη πράξη που θέλουμε να κάνει το μοντέλο"

"Input": "Προαιρετικό πεδίο, θεωρείται το query που θα δώσει ο χρήστης."

"Output": "Το επιθυμητό αποτέλεσμα της πράξης που θα παράγει το μοντέλο."
```

Έπειτα, συνδυάζονται αυτές οι τρεις γραμμές σε μία, με τη παρακάτω δομή, αποτελώντας έτσι την εντολή που λαμβάνει το LLM:

```
Below is an instruction that describes a task, paired with an input that provides further context.

Write a response that appropriately completes the request.


### Instruction:

{Instruction}


### Input:

{Input}


### Response:

{Output}
```

Ο τρόπος αυτός ονομάζεται εποπτευόμενη βελτιστοποίηση διδασκαλίας (supervised instruction finetuning) και έγινε δημοφιλής όταν εκδόθηκε για πρώτη φορά το Llama-1, αφού με αυτή τη τεχνική μπορούσε ένα fine-tuned LLM να ανταγωνιστεί το ChatGPT [142].

Το dataset αποτελείται συνολικά από 156 διαφορετικές εγγραφές, οι οποίες στο Instruction αναφέρεται το είδους του phishing email που επιλέγει ο χρήστης και στο Output είναι το επιθυμητό αποτέλεσμα αυτής της επιλογής email. Για παράδειγμα, ο χρήστης ζητάει να γραφτεί ένα phishing email όπου ο παραλήπτης έχει κερδίσει έναν διαγωνισμό:

```
"Instruction": "Write an email informing the user they won a giveaway"

"Input": " "

"Output": "Subject: Congratulations! You're a Winner

Body: Dear [Name],

We're excited to let you know that you've been selected as a winner in our recent giveaway!
```

```
To claim your prize, please confirm your details using the link below.
Prizes must be claimed within 7 days.

[Insert Call-to-Action button or link]"

Below is an instruction that describes a task, paired with an input that
provides further context.

Write a response that appropriately completes the request.


### Instruction:

Write an email informing the user they won a giveaway


### Response:

Subject: Congratulations! You're a Winner

Body: Dear [Name],

We're excited to let you know that you've been selected as a winner in
our recent giveaway!


To claim your prize, please confirm your details using the link below.
Prizes must be claimed within 7 days.

[Insert Call-to-Action button or link]
```

Η δομή του παραγόμενου Output προσομοιάζει με εκείνη ενός ηλεκτρονικού μηνύματος, περιλαμβάνοντας αντίστοιχα πεδία "Subject" (Θέμα) και "Body" (Σώμα). Η μορφή αυτή εφαρμόζεται σε κάθε εγγραφή, διασφαλίζοντας ότι το αποτέλεσμα που παράγεται από το γλωσσικό μοντέλο περιλαμβάνει ένα θέμα και ένα σώμα κειμένου, τα οποία στη συνέχεια ενσωματώνονται στο τελικό Email. Επιπλέον, υπάρχουν τα σημεία [Name] και [Insert Call-to-Action button or link] τα οποία τροποποιούνται δυναμικά μέσω του κώδικα της εφαρμογής, για να γίνει το Email προσωποποιημένο και να παρέχει έναν σύνδεσμο της επιλογής του τελικού χρήστη, χωρίς να δεσμεύεται το dataset σε συγκεκριμένα στοιχεία και να απαιτείται η ύπαρξη προκαθορισμένων δεδομένων.

Έγινε διερεύνηση των ήδη υπάρχων dataset με phishing emails, ωστόσο, τα περισσότερα εξ αυτών δεν πληρούσαν τις απαιτήσεις του έργου. Κατά συνέπεια, το περιεχόμενο του χρησιμοποιούμενου dataset βασίζεται σε έγκυρα ηλεκτρονικά μηνύματα, τα οποία αποστέλλονται καθημερινά σε χρήστες. Για τη σύνταξη των εγγραφών, συλλέχθηκαν πληροφορίες από αυτά τα μηνύματα και ξεκίνησε η δημιουργία του. Παρόλα αυτά, αξιοποιήθηκαν επιλεγμένα διαθέσιμα datasets που βοήθησαν στην αύξηση της ποικιλία των εγγραφών. Συγκεκριμένα, αυτά ήταν το Phishing-Email-Dataset [143] το οποίο είναι διαθέσιμο στο GitHub και αποτελείται από πολλά επιμέρους datasets, καθώς και το Phishing validation emails dataset [144] που ποικίλει από ασφαλής σε κακόβουλα Emails. Ως τελευταίο βήμα, για την περαιτέρω ενίσχυση του πλήθους των εγγραφών και την επέκταση του dataset, αξιοποιήθηκε το ChatGPT για τη δημιουργία επιπρόσθετου περιεχομένου.

3.3 Γλώσσα προγραμματισμού που επιλέχθηκε

Για την υλοποίηση της παρούσας εργασίας, επιλέχθηκε η Python ως η βασική γλώσσα προγραμματισμού. Η επιλογή αυτή δεν ήταν τυχαία, αλλά βασίστηκε σε τεχνικά, πρακτικά και επιστημονικά κριτήρια που την καθιστούν την πλέον κατάλληλη για έργα που σχετίζονται με την επεξεργασία φυσικής γλώσσας και την εκπαίδευση ή χρήση γλωσσικών μοντέλων.

Αρχικά, η Python είναι η κατεξοχήν γλώσσα στο πεδίο της Τεχνητής Νοημοσύνης και της Μηχανικής Μάθησης. Παρέχει ένα πλούσιο και ώριμο οικοσύστημα βιβλιοθηκών και εργαλείων, όπως τα PyTotch, TensorFlow, Transformers (Hugging Face) και scikit_learn, τα οποία διευκολύνουν την υλοποίηση, την παραμετροποίηση και την αξιολόγηση σύνθετων μοντέλων, με ελάχιστο φόρτο ανάπτυξης. Στο πλαίσιο της εργασίας αυτής, όπου γίνεται χρήση προσαρμοσμένης εκπαίδευσης γλωσσικού μοντέλου, η Python προσφέρει εγγενή υποστήριξη για τα απαιτούμενα frameworks και APIs, αφού χρησιμοποιήθηκε το ollama.chat API για την αλληλεπίδραση με το LLM, το οποίο ενσωματώθηκε με εξαιρετική ευκολία μέσω της Python. Επιπλέον, μέσω βιβλιοθηκών όπως η requests διευκολύνουν ιδιαίτερα τη σύνδεση με RESTful APIs, γεγονός που καθιστά τη γλώσσα ιδανική για τον συντονισμό μεταξύ της διεπαφής προγραμματισμού (FastAPI), του LLM και του backend αποστολής email (SMTP).

Επιπροσθέτως, η χρήση της Python επέτρεψε την αξιοποίηση του FastAPI, ενός μοντέρνου και αποδοτικού web framework που είναι ιδανικό για την ανάπτυξη RESTful υπηρεσιών με μεγάλη ταχύτητα εκτέλεσης, προσφέρει εγγενή υποστήριξη ασύγχρονου κώδικα και αυτόματη τεκμηρίωση (OpenAPI). Παράλληλα, μέσω των βιβλιοθηκών smtplib και email.mime, η Python προσέφερε πλήρεις δυνατότητες διαμόρφωσης και αποστολής email, χωρίς την ανάγκη προσθήκης εξωτερικών εργαλείων. Η ευελιξία αυτών των βιβλιοθηκών διευκόλυνε την δημιουργία και αποστολή προσωποποιημένων emails με την κατάλληλη μορφοποίηση. Αξιοσημείωτο είναι επίσης το γεγονός ότι η Python επιτρέπει την ασφαλή διαχείριση ευαίσθητων πληροφοριών, όπως διαπιστευτήρια email (email credentials) ή API keys, μέσω της χρήσης βιβλιοθηκών όπως η python-dotenv, η οποία χρησιμοποιήθηκε για τη φόρτωση μεταβλητών περιβάλλοντος από .env αρχείο. Αυτό επιτρέπει ευκολότερη και ασφαλέστερη μεταφορά της εφαρμογής σε διαφορετικά περιβάλλοντα (local, staging, production).

Επιπλέον, η Python ενδείκνυται για ταχεία πρωτοτυποποίηση (rapid prototyping), καθώς επιτρέπει την άμεση υλοποίηση και αξιολόγηση διαφορετικών σεναρίων και παραμέτρων χωρίς την ανάγκη σύνθετων διαδικασιών μεταγλώττισης ή χειρισμού χαμηλού επιπέδου λεπτομερειών. Αυτό το χαρακτηριστικό είναι ιδιαίτερα χρήσιμο σε έργα NLP, όπου η επανάληψη και ο πειραματισμός με παραλλαγές δεδομένων και μοντέλων είναι βασικοί παράγοντες επιτυχίας.

Τέλος, η ευρεία αποδοχή και η ενεργή κοινότητα υποστήριξης της Python διασφαλίζουν τη διαθεσιμότητα πλούσιας τεκμηρίωσης, παραδειγμάτων και βοήθειες σε περίπτωση προβλημάτων. Αυτό συμβάλλει ουσιαστικά στη μείωση του χρόνου ανάπτυξης και στην επίλυση τεχνικών ζητημάτων που ενδέχεται να προκύψουν.

Συνοψίζοντας, η Python επιλέχθηκε για την εργασία αυτή λόγω του συνδυασμού πλεονεκτημάτων που προσφέρει, όπως την ταχύτητα ανάπτυξης, την απλότητα σύνδεσης με LLM APIs, την ευκολία δημιουργίας RESTful endpoints και την πλήρη υποστήριξη για αυτοματοποίηση αποστολής email. Παρέχει όλα τα απαραίτητα εργαλεία σε ένα ενιαίο και αρμονικό οικοσύστημα, γεγονός που την καθιστά ιδανική γλώσσα για την ανάπτυξη και

πειραματική εφαρμογή τεχνικών επεξεργασίας φυσικής γλώσσας, όπως αυτών που υλοποιήθηκαν στην παρούσα εργασία.

3.4 Υλοποίηση – Μέθοδος Πρώτη – FastAPI

Ο πρώτος τρόπος υλοποίησης της προσομοίωσης αποστολής email μέσω Gmail γίνεται με τη χρήση ενός backend συστήματος βασισμένου στο FastAPI, το οποίο λειτουργεί τοπικά στον υπολογιστή του χρήστη. Η μέθοδος αυτή επιτρέπει τη δημιουργία προσωποποιημένων email με τη βοήθεια ενός μεγάλου γλωσσικού μοντέλου που ενεργοποιείται μέσω της πλατφόρμας Ollama. Ο χρήστης συμπληρώνει βασικές πληροφορίες, όπως το prompt, το όνομα του παραλήπτη και έναν σύνδεσμο (link), και στη συνέχεια λαμβάνει προεπισκόπηση του μηνύματος (preview) πριν αυτό σταλεί. Παράλληλα, δίνεται η δυνατότητα αποστολής σε πολλούς παραλήπτες ταυτόχρονα μέσω μιας απλής εισαγωγής διευθύνσεων και προσφέρεται επιλογή μορφής του email, είτε σε απλό κείμενο είτε με HTML. Για λόγους ασφαλείας, η πρόσβαση ελέγχεται με API key που έχει περιορισμένο αριθμό χρήσεων και απαιτείται πιστοποιημένος λογαριασμός email για την αποστολή των μηνυμάτων. Τέλος, το μεγάλο γλωσσικό μοντέλο εκπαιδεύτηκε και βελτιστοποιήθηκε (fine-tuned) με τη χρήση των εργαλείων Google Colab και Unsloth. Η πλατφόρμα του Google Colab διέθετε υπολογιστικούς πόρους που περιλάμβαναν 13GB μνήμης RAM συστήματος, 15GB μνήμης GPU και 113GB διαθέσιμου αποθηκευτικού χώρου, με επιλεγμένο επιταχυντή υλικού τον επεξεργαστή γραφικών T4 GPU. Για τη διαδικασία fine-tuning αξιοποιήθηκε μέσω της βιβλιοθήκης Unsloth το μοντέλο Llama3, το οποίο διαθέτει 8 δισεκατομμύρια παραμέτρους.

3.4.1 Τι κάνει το λογισμικό

Ο κώδικας που αναπτύχθηκε υλοποιεί ένα σύστημα προσομοίωσης ηλεκτρονικού "ψαρέματος" χρησιμοποιώντας FastAPI και ένα προσαρμοσμένο μοντέλο μεγάλης γλώσσας που φιλοξενείται στο Ollama.com. Το μοντέλο βελτιστοποιήθηκε χρησιμοποιώντας το πλαίσιο Unsloth στο Google Colab με ένα προσαρμοσμένο σύνολο δεδομένων που αποτελείται από 86 καταχωρήσεις, καθεμία από τις οποίες περιγράφει διάφορα σενάρια ηλεκτρονικού "ψαρέματος". Ο κύριος σκοπός αυτού του συστήματος είναι η δημιουργία ρεαλιστικών email τύπου "ψαρέματος" με βάση προτροπές που ορίζονται από τον χρήστη και η προσομοίωση της παράδοσής τους, με προαιρετική μορφοποίηση που βασίζεται σε HTML. Χρησιμοποιεί επίσης έλεγχο ταυτότητας μέσω κλειδιών API και ενσωματώνει λειτουργικότητα email SMTP για την αποστολή των δημιουργημένων μηνυμάτων.

Στον πυρήνα της, αυτή η εφαρμογή υποστηρίζεται από έναν FastAPI web server που εκθέτει δύο κύρια τελικά σημεία HTTP: 1) /generate: Αυτό το τελικό σημείο επιτρέπει στους χρήστες να υποβάλουν μια γραπτή προτροπή που περιγράφει το είδος του email ηλεκτρονικού "ψαρέματος" που θέλουν να δημιουργήσει το LLM. Η πρόσβαση στο μοντέλο LLM γίνεται μέσω του API Ollama, χρησιμοποιώντας συγκεκριμένα το αναγνωριστικό μοντέλου "gawyria/demo-model", το οποίο αναφέρεται στο βελτιστοποιημένο LLM που φιλοξενείται στο Ollama.com. Η προτροπή, μαζί με παραμέτρους που αφορούν συγκεκριμένα τον παραλήπτη, όπως το όνομά του και έναν κακόβουλο σύνδεσμο, αποστέλλονται στο μοντέλο. Η έξοδος αναλύεται, καθαρίζεται και επιστρέφεται στον χρήστη ως σώμα email προεπισκόπησης, 2) /send_email: Μόλις η προεπισκόπηση

εγκριθεί από τον χρήστη, αυτό το τελικό σημείο είναι υπεύθυνο για την αποστολή του email ηλεκτρονικού "ψαρέματος" σε έναν ή περισσότερους παραλήπτες. Υποστηρίζει τόσο απλό κείμενο όσο και πλούσια μορφοποίηση HTML μέσω ενός ενσωματωμένου προτύπου email με κεφαλίδες σε λιλά χρώμα και πολυτελή τυπογραφία. Το email αποστέλλεται χρησιμοποιώντας διαπιστευτήρια SMTP που έχουν φορτωθεί με ασφάλεια από μεταβλητές περιβάλλοντος.

Όσον αφορά την επικύρωση του κλειδιού του API και τα όρια χρήσης, η διαχείριση της πιστοποίησης API γίνεται ελέγχοντας για ένα έγκυρο x-api-key στην κεφαλίδα αιτήματος HTTP. Τα κλειδιά φορτώνονται από μεταβλητές περιβάλλοντος και έχουν ένα προκαθορισμένο όριο αιτήματος (π.χ., 5 αιτήματα ανά κλειδί). Κάθε request στο τελικό σημείο /generate μειώνει το υπόλοιπο όριο. Αυτό βοηθά στον έλεγχο της πρόσβασης στη δυνατότητα δημιουργίας δυνητικά ευαίσθητων email ηλεκτρονικού "ψαρέματος".

Όταν υποβάλλεται ένα μήνυμα στο τελικό σημείο /generate, αποστέλλεται στο Ollama LLM χρησιμοποιώντας τη μέθοδο ollama.chat. Το μοντέλο επιστρέφει μια απάντηση που περιέχει ένα δομημένο email ηλεκτρονικού "ψαρέματος". Αυτή η απάντηση μπορεί να περιλαμβάνει ενότητες όπως Subject: και Body:, αν και μόνο το σώμα αναλύεται και χρησιμοποιείται. Στη συνέχεια, υπάρχει η συνάρτηση extract_subject_and_body, που συμβάλλει στην προετοιμασία του περιεχομένου του email. Η συνάρτηση αυτή απομακρύνει τυχόν πρόθεμα "Subject:" και χρησιμοποιεί κανονικές εκφράσεις (regex) για να εντοπίσει και να εξαγάγει το θέμα και το σώμα του μηνύματος από την έξοδο του LLM. Φροντίζει, επίσης, να τηρείται η αναμενόμενη μορφή της απάντησης. Αν κάποιο από τα δύο στοιχεία (θέμα ή σώμα) απουσιάζει, εγείρεται ένα ValueError, το οποίο προκαλεί την επανάληψη του αιτήματος στο τελικό endpoint /generate.

Για να γίνουν τα δημιουργούμενα ηλεκτρονικά μηνύματα πιο πειστικά, οι λέξεις-κλειδιά placeholder όπως [Name] και [Insert Call-to-Action button or link] αντικαθίστανται δυναμικά με τιμές που καθορίζονται από τον χρήστη (π.χ., το όνομα του στόχου και έναν κακόβουλο σύνδεσμο). Αυτό επιτυγχάνεται χρησιμοποιώντας αντικατάσταση που βασίζεται σε regex για τη σάρωση και αντικατάσταση διαφόρων συμβάσεων ονομασίας που χρησιμοποιούνται σε πρότυπα ηλεκτρονικού "ψαρέματος". Το στοιχείο [Insert Call-to-Action button or link] πρέπει να περιλαμβάνεται στο θέμα του email. Αν απουσιάζει, το API θα επιχειρήσει εκ νέου έως και 5 φορές, με την παραδοχή ότι το μοντέλο ενδέχεται να δημιουργεί διαφορετικά αποτελέσματα κάθε φορά.

Το σύστημα υποστηρίζει προαιρετικά την αποστολή του ηλεκτρονικού ταχυδρομείου σε πλούσια μορφή HTML. Χρησιμοποιείται ένα προσχεδιασμένο πρότυπο HTML που περιλαμβάνει μια κεφαλίδα με λιλά διαβάθμιση, επώνυμη τυπογραφία και ένα κουμπί που περιέχει τον κακόβουλο σύνδεσμο. Αυτό είναι ιδιαίτερα χρήσιμο σε προσομοιώσεις ηλεκτρονικού "ψαρέματος", καθώς μιμείται την εμφάνιση και την αίσθηση των νόμιμων εταιρικών ηλεκτρονικών μηνυμάτων. Η HTML συμπληρώνεται δυναμικά με το θέμα, το όνομα του παραλήπτη, το περιεχόμενο του σώματος και τη διεύθυνση URL.

Για την παράδοση email, το σύστημα χρησιμοποιεί το smtplib της Python για να συνδεθεί σε έναν διακομιστή SMTP (π.χ., Gmail), να επαληθεύσει με διαπιστευτήρια που είναι αποθηκευμένα σε μεταβλητές περιβάλλοντος .env και να στείλει το email στους προβλεπόμενους παραλήπτες. Πριν από την αποστολή, όλες οι διευθύνσεις email των παραληπτών επικυρώνονται χρησιμοποιώντας τον τύπο EmailStr από το Pydantic για την αποφυγή λανθασμένων καταχωρίσεων.

Στο τέλος, παρέχεται ένα διαδραστικό εργαλείο γραμμής εντολών από την πλευρά του χρήστη. Αυτό το σενάριο εκτελεί την ακόλουθη ακολουθία: 1) Ζητάει από τον χρήστη όλες τις απαραίτητες πληροφορίες, συμπεριλαμβανομένων των διαπιστευτηρίων αποστολέα, των email παραληπτών, του ονόματος παραλήπτη, του συνδέσμου ηλεκτρονικού

"ψαρέματος" και της προτροπής δημιουργίας του phishing email, 2) Υποβάλλει αυτές τις πληροφορίες στο τελικό σημείο /generate για να ανακτήσει και να εμφανίσει μια προεπισκόπηση του email ηλεκτρονικού "ψαρέματος", 3) Ρωτάει τον χρήστη εάν θέλει να προχωρήσει στην αποστολή του email και εάν αυτό το email θέλει να έχει το πρότυπο UI HTML, 4) Εάν επιβεβαιωθεί, το σενάριο καλεί στη συνέχεια το τελικό σημείο /send_email για να παραδώσει το μήνυμα, προαιρετικά χρησιμοποιώντας το πρότυπο HTML.

Συνοπτικά, αυτό το σύστημα προσομοιώνει ρεαλιστικές επιθέσεις ηλεκτρονικού "ψαρέματος" χρησιμοποιώντας ένα βελτιστοποιημένο LLM και προσφέρει ένα ελεγχόμενο και προσαρμόσιμο περιβάλλον για τη δημιουργία και τη διανομή αυτών των email. Συνδυάζει τον έλεγχο ταυτότητας API, τη δημιουργία email μέσω προτροπών, τη δυναμική εξατομίκευση και την ευέλικτη παράδοση email σε μια ενιαία συνεκτική εφαρμογή. Αυτό θα μπορούσε να χρησιμοποιηθεί σε σενάρια ηθικής πειρατείας (ethical hacking), σε εκπαίδευση ευαισθητοποίησης για το ηλεκτρονικό "ψάρεμα" ή σε ελεγχόμενα περιβάλλοντα δοκιμών red-team, υπό την προϋπόθεση ότι τηρούνται αυστηρά όλες οι ηθικές και νομικές οδηγίες.

Παρακάτω υπάρχει ένας συνοπτικός πίνακας για τις χρήσεις των βιβλιοθηκών στο λογισμικό (Πίνακας 1).

Πίνακας 1: Βιβλιοθήκες FastAPI

Βιβλιοθήκη	**Σκοπός**
smtplib	Σύνδεση και αποστολή Email (Gmail σε αυτή τη περίπτωση).
fastapi	Δημιουργία REST API endpoints, διαχείριση αιτημάτων HTTP και εξαρτήσεων.
re (regex)	Χρήση regular expressions για έλεγχο και τον διαχωρισμό του Σώματος από το Θέμα του ηλεκτρονικού μηνύματος.
email.mime.multipart & email.mime.text	Δημιουργία απλού και σύνθετου HTML ηλεκτρονικού μηνύματος.
requests	Αποστολή HTTP POST requests στο API έτσι ώστε να παρθεί πίσω η απάντηση από το AI.
pydantic	Έλεγχος εγκυρότητας και τύπων δεδομένων (π.χ. email), μέσω μοντέλων τύπου BaseModel.
ollama	Επικοινωνία με το AI μοντέλο (gawyria/demo-model) για τη δημιουργία περιεχομένου.
os	Πρόσβαση σε περιβαλλοντικές μεταβλητές, όπως διαπιστευτήρια ή ρυθμίσεις από το .env.
dotenv (load_dotenv)	Φόρτωση μεταβλητών περιβάλλοντος από αρχείο .env, για λόγους ασφαλείας και διαμόρφωσης.

datetime	Ενσωμάτωση της τρέχουσας ημερομηνίας στο πρότυπο HTML email (π.χ. έτος στο footer).

3.4.2 Είναι χαμηλού ή/και υψηλού επιπέδου;

Ο κώδικας που αναπτύχθηκε για τη συγκεκριμένη μέθοδο είναι υψηλού επιπέδου, δεδομένου ότι λειτουργεί απομακρυσμένα από το υλικό και αλληλεπιδρά κυρίως με διεπαφές διαδικτύου και υπηρεσίες επιπέδου εφαρμογής. Είναι γραμμένος σε Python, μια γλώσσα προγραμματισμού υψηλού επιπέδου που αφαιρεί την ανάγκη άμεσης διαχείρισης λειτουργιών χαμηλού επιπέδου, όπως η διαχείριση μνήμης και η επικοινωνία με το υλικό. Το σύστημα αξιοποιεί σύγχρονα web frameworks, όπως το FastAPI, για τη δημιουργία RESTful API, και ενσωματώνεται με εξωτερικές υπηρεσίες, όπως SMTP διακομιστές email, μηχανές τεχνητής νοημοσύνης μέσω Ollama, καθώς και διαχείριση περιβαλλοντικών ρυθμίσεων μέσω αρχείων .env. Επιπλέον, η λειτουργικότητα περιλαμβάνει χαρακτηριστικά φιλικά προς τον χρήστη, όπως έλεγχο ταυτότητας API, παραγωγή προτύπων HTML και αυτοματοποιημένη εξατομίκευση ηλεκτρονικών μηνυμάτων, στοιχεία τα οποία είναι τυπικά για εφαρμογές υψηλού επιπέδου σχεδιασμένες για την εξυπηρέτηση τελικών χρηστών ή διαχειριστών συστημάτων. Ο κώδικας δεν πραγματοποιεί αλληλεπιδράσεις με στοιχεία υλικού, δεν διαχειρίζεται μνήμη σε χαμηλό επίπεδο και δεν ελέγχει απευθείας το λειτουργικό σύστημα, χαρακτηριστικά που συνήθως συναντώνται σε γλώσσες και συστήματα χαμηλού επιπέδου. Συνεπώς, η παρούσα υλοποίηση εντάσσεται σαφώς σε ένα σύστημα υψηλού επιπέδου, εστιάζοντας στο web και στο application layer.

3.4.3 Λειτουργικές/μη λειτουργικές απαιτήσεις και Προδιαγραφές

3.4.3.1 Λειτουργικές Απαιτήσεις

Οι λειτουργικές απαιτήσεις της παρούσας εργασίας μπορούν να περιγραφούν ως εξής:

3.4.3.1.1 Η εφαρμογή θα πρέπει να επιτρέπει στους πιστοποιημένους χρήστες να δημιουργούν εξατομικευμένο περιεχόμενο ηλεκτρονικού ταχυδρομείου χρησιμοποιώντας ένα LLM.

Περιγραφή: Οι χρήστες πρέπει πρώτα να πιστοποιηθούν χρησιμοποιώντας ένα κλειδί API για να αποκτήσουν πρόσβαση στη λειτουργία δημιουργίας ηλεκτρονικού ταχυδρομείου Τεχνητής Νοημοσύνης. Μόλις πιστοποιηθούν, μπορούν να εισάγουν βασικές παραμέτρους όπως τα διαπιστευτήρια του αποστολέα, τα email των παραληπτών, το όνομα του παραλήπτη, τον κακόβουλο σύνδεσμο του email και της προτροπής δημιουργίας του phishing email. Το σύστημα στη συνέχεια διασυνδέεται με το τοπικά φιλοξενούμενο μοντέλο γλώσσας μέσω του Ollama για να δημιουργήσει ένα προσαρμοσμένο ηλεκτρονικό μήνυμα με βάση τα παρεχόμενα δεδομένα εισόδου. Το περιεχόμενο του ηλεκτρονικού ταχυδρομείου δομείται και αποδίδεται μέσω προτύπων Jinja2 για αναγνωσιμότητα και συνέπεια.

Αιτιολόγηση: Αυτή η λειτουργικότητα επιτρέπει την βασική περίπτωση χρήσης της εφαρμογής: αυτόματη και βασισμένη στο περιβάλλον σύνταξης ηλεκτρονικού ταχυδρομείου χρησιμοποιώντας Τεχνητή Νοημοσύνη. Απαιτώντας έλεγχο ταυτότητας, το σύστημα διασφαλίζει ότι η πρόσβαση ελέγχεται και παρακολουθείται. Η εξατομίκευση με

βάση τα πεδία εισόδου βελτιώνει τη συνάφεια και τον επαγγελματισμό της εξόδου του ηλεκτρονικού ταχυδρομείου, καθιστώντας το σύστημα πολύτιμο για επαγγελματικές εργασίες επικοινωνίας και βελτιώνοντας την παραγωγικότητα.

3.4.3.1.2 Το σύστημα θα πρέπει να επιτρέπει στους χρήστες να κάνουν προεπισκόπηση του περιεχομένου email που δημιουργείται από το μοντέλο μέσω μιας διεπαφής ιστού.

Περιγραφή: Μετά τη δημιουργία ενός email, το σύστημα εμφανίζει το αποτέλεσμα σε ένα πρότυπο HTML που βασίζεται στον ιστό και μιμείται μια ρεαλιστική διάταξη email. Το πρότυπο αποδίδεται χρησιμοποιώντας τη μηχανή δημιουργίας προτύπων του FastAPI (Jinja2), επιτρέποντας στους χρήστες να βλέπουν το αποτέλεσμα στο πρόγραμμα περιήγησης πριν από περαιτέρω ενέργειες.

Αιτιολόγηση: Η παροχή οπτικής προεπισκόπησης διασφαλίζει ότι οι χρήστες μπορούν να αξιολογήσουν και να αναθεωρήσουν το email που δημιουργείται πριν το αντιγράψουν ή το στείλουν. Αυτό αυξάνει την εμπιστοσύνη στο αποτέλεσμα που δημιουργείται από το μοντέλο και ενθαρρύνει τον έλεγχο ποιότητας, υποστηρίζοντας έτσι επαγγελματικές περιπτώσεις χρήσης όπου η ακρίβεια και ο τόνος είναι κρίσιμα.

3.4.3.1.3 Η εφαρμογή θα πρέπει να είναι ικανή να στέλνει email σε συγκεκριμένους παραλήπτες μέσω SMTP.

Περιγραφή: Οι χρήστες μπορούν προαιρετικά να επιλέξουν να στείλουν το email που δημιουργείται απευθείας από την εφαρμογή. Κατόπιν αιτήματος, το σύστημα συνδέεται με έναν διακομιστή SMTP χρησιμοποιώντας ασφαλή διαπιστευτήρια (αποθηκευμένα σε ένα αρχείο `.env`), συνθέτει το email με το δημιουργημένο περιεχόμενο και το αποστέλλει στη διεύθυνση email του/των παραλήπτη/παραληπτών.

Αιτιολόγηση: Αυτή η λειτουργικότητα επιτρέπει την απρόσκοπτη ενσωμάτωση μεταξύ της σύνθεσης και της αποστολής email. Αυτοματοποιώντας τόσο τη δημιουργία όσο και την παράδοση του περιεχομένου του email, η εφαρμογή βελτιστοποιεί τις ροές εργασίας επικοινωνίας. Η χρήση ασφαλών πρακτικών SMTP (με διαπιστευτήρια που βασίζονται στο περιβάλλον) ενισχύει την λειτουργική αξιοπιστία και την εμπιστευτικότητα των δεδομένων.

3.4.3.1.4 Η εφαρμογή θα πρέπει να επιτρέπει τη διαμόρφωση μεταβλητών περιβάλλοντος, όπως ρυθμίσεις SMTP, αριθμούς θύρας και επιλογή μοντέλου.

Περιγραφή: Οι προγραμματιστές και οι διαχειριστές μπορούν να διαμορφώσουν διαπιστευτήρια SMTP, ρυθμίσεις θύρας και επιλογές μοντέλου AI μέσω ενός αρχείου `.env`. Η εφαρμογή χρησιμοποιεί το πακέτο `dotenv` για τη φόρτωση αυτών των μεταβλητών κατά τη διάρκεια εκτέλεσης.

Αιτιολόγηση: Η κεντρική διαχείριση διαμόρφωσης απλοποιεί την ανάπτυξη σε πολλαπλά περιβάλλοντα (π.χ. ανάπτυξη, σταδιακή προετοιμασία, παραγωγή). Προωθεί την ασφάλεια (μη κωδικοποιώντας διαπιστευτήρια) και διασφαλίζει ευελιξία στην εναλλαγή μοντέλων ή διαμορφώσεων διακομιστή χωρίς τροποποίηση του πηγαίου κώδικα.

3.4.3.1.5 Το σύστημα θα πρέπει να υποστηρίζει την εκτέλεση της εφαρμογής σε μια καθορισμένη θύρα.

Περιγραφή: Η εφαρμογή συνδέεται δυναμικά με μια θύρα που καθορίζεται στο αρχείο `.env`, επιτρέποντας την ανάπτυξη σε διάφορα περιβάλλοντα cloud ή on-premise χωρίς αλλαγές στον κώδικα.

Αιτιολόγηση: Η δυνατότητα δυναμικής ανάθεσης θύρας διασφαλίζει ότι η εφαρμογή μπορεί να ενσωματωθεί σε ποικίλες αρχιτεκτονικές συστήματος, συμπεριλαμβανομένων πίσω από proxies ή εντός αναπτύξεων σε κοντέινερ, ενισχύοντας την ευελιξία ανάπτυξης και τη συμβατότητα DevOps.

3.4.3.2 Μη Λειτουργικές Απαιτήσεις

3.4.3.2.1 Η εφαρμογή θα πρέπει να είναι εύκολη στη χρήση

Περιγραφή: Η διεπαφή της εφαρμογής θα πρέπει να είναι διαισθητική, κατανοητή και εύχρηστη. Οι χρήστες θα μπορούν να εισάγουν πληροφορίες εύκολα και να λαμβάνουν απαντήσεις με ελάχιστη προσπάθεια. Θα πρέπει να παρέχεται επικύρωση για τον εντοπισμό ελλιπών ή εσφαλμένων δεδομένων.

Αιτιολόγηση: Η χρηστικότητα είναι κρίσιμη για την ευρεία αποδοχή της εφαρμογής από χρήστες με διαφορετικά επίπεδα τεχνικών γνώσεων.

3.4.3.2.2 Η εφαρμογή θα πρέπει να έχει υψηλή απόδοση

Περιγραφή: Η εφαρμογή θα πρέπει να δημιουργεί και να επιστρέφει email εντός λίγων δευτερολέπτων κατά την κανονική χρήση. Η τοπική χρήση του Ollama συμβάλλει στη μείωση της καθυστέρησης, εξασφαλίζοντας ταχύτερες αποκρίσεις.

Αιτιολόγηση: Η απόκριση σε πραγματικό χρόνο είναι απαραίτητη για την αποτελεσματική χρήση της εφαρμογής σε περιβάλλοντα παραγωγής.

3.4.3.2.3 Η εφαρμογή θα πρέπει να παρέχει ασφάλεια στον χρήστη

Περιγραφή: Η πρόσβαση σε βασικές λειτουργίες προστατεύεται μέσω API key, ενώ ευαίσθητα δεδομένα (όπως SMTP credentials) αποθηκεύονται με ασφάλεια σε μεταβλητές περιβάλλοντος. Η αποστολή email γίνεται μέσω ασφαλών πρωτοκόλλων (SMTP).

Αιτιολόγηση: Η ασφάλεια προστατεύει τον αποστολέα, τον παραλήπτη και τα δεδομένα, διασφαλίζοντας τη συμμόρφωση με τις βέλτιστες πρακτικές.

3.4.3.2.4 Η εφαρμογή θα πρέπει να είναι επεκτάσιμη

Περιγραφή: Η αρχιτεκτονική της εφαρμογής θα πρέπει να υποστηρίζει κλιμάκωση, με διαχωρισμένα υποσυστήματα που μπορούν να λειτουργούν σε πολλαπλούς διακομιστές μέσω load balancing. Το μοντέλο εκτελείται ξεχωριστά για παράλληλη εξυπηρέτηση χρηστών.

Αιτιολόγηση: Η επεκτασιμότητα επιτρέπει την προσαρμογή της εφαρμογής σε αυξημένο φόρτο εργασίας στο μέλλον.

3.4.3.2.5 Θα πρέπει η εφαρμογή να είναι έτσι δομημένη ώστε να μπορούν μελλοντικά να γίνουν αλλαγές (συντηρησιμότητα)

Περιγραφή: Ο κώδικας πρέπει να είναι σαφής, ευανάγνωστος, τεκμηριωμένος και modular. Θα πρέπει να υλοποιείται καταγραφή σφαλμάτων για διευκόλυνση της αποσφαλμάτωσης και της συντήρησης.

Αιτιολόγηση: Η σωστή συντηρησιμότητα μειώνει τον χρόνο και το κόστος αλλαγών στο μέλλον, υποστηρίζοντας τη μακροχρόνια αξιοπιστία της εφαρμογής.

3.4.3.2.6 Η εφαρμογή θα πρέπει να είναι διαλειτουργική

Περιγραφή: Η εφαρμογή θα πρέπει να συνεργάζεται με τυπικές υπηρεσίες email (SMTP), τοπικά μοντέλα όπως το Ollama και να υποστηρίζει ενσωμάτωση με εξωτερικά συστήματα (όπως CRM) μέσω RESTful API.

Αιτιολόγηση: Η διαλειτουργικότητα επιτρέπει την ευρύτερη αξιοποίηση της εφαρμογής σε διάφορα πληροφοριακά συστήματα.

3.4.3.2.7 Η εφαρμογή θα πρέπει να είναι αξιόπιστη

Περιγραφή: Θα πρέπει να υπάρχει διαχείριση σφαλμάτων για περιπτώσεις αποτυχιών του μοντέλου, μη έγκυρων εισροών ή αποτυχημένων αποστολών email. Το σύστημα πρέπει να επιστρέφει χρήσιμα μηνύματα σφαλμάτων στους χρήστες και να καταγράφει προβλήματα για τους προγραμματιστές.

Αιτιολόγηση: Η αξιοπιστία εξασφαλίζει ομαλή εμπειρία για τον χρήστη και διευκολύνει την επίλυση τεχνικών προβλημάτων.

3.4.3.3 Προδιαγραφές

Πίνακας 2: 3.4.3.3.1 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Δημιουργία email μέσω του LLM μοντέλου
Περιγραφή	Οι εξουσιοδοτημένοι χρήστες δημιουργούν ένα εξατομικευμένο email χρησιμοποιώντας ένα τοπικό μεγάλο γλωσσικό μοντέλο. Το σύστημα συλλέγει βασικές λεπτομέρειες και επιστρέφει ένα πλήρες email με βάση αυτές τις εισόδους.
Δεδομένα Εισόδου	User’s email, App password, recipient email, recipient's name, link, prompt, subject.
Προέλευση	Web form on the user interface.
Δεδομένα Εξόδου	Περιεχόμενο του email που δημιουργείται από το LLM (με και χωρίς μορφοποίηση HTML).
Προορισμός	Πρότυπο web (απάντηση HTML που αποδίδεται).
Ενέργεια	Ο χρήστης υποβάλλει ένα αίτημα μέσω ενός request. Ο διακομιστής μεταβιβάζει δεδομένα εισόδου στο μοντέλο μέσω του Ollama, λαμβάνει μια απάντηση, μορφοποιεί το περιεχόμενο και το αποδίδει στο πρόγραμμα περιήγησης.
Απαίτηση	Δυναμική και εξατομικευμένη σύνθεση email χρησιμοποιώντας τεχνητή νοημοσύνη.
Προϋπόθεση ή Προσυνθήκη	Έγκυρο κλειδί API ελέγχου ταυτότητας και διαθεσιμότητα του μοντέλου μέσω του Ollama.
Αποτέλεσμα ή Μετασυνθήκη	Εμφανιζόμενο αποτέλεσμα email στο πρόγραμμα περιήγησης, προσαρμοσμένο στην είσοδο του χρήστη.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα

Πίνακας 3: 3.4.3.3.2 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Διεπαφή Προεπισκόπησης Ηλεκτρονικού Ταχυδρομείου
Περιγραφή	Μετά τη δημιουργία ενός ηλεκτρονικού ταχυδρομείου, οι χρήστες κάνουν

	προεπισκόπηση του μηνύματος που δημιουργείται από το μοντέλο μέσω μιας διεπαφής ιστού με πρότυπα Jinja2 (FastAPI).
Δεδομένα Εισόδου	Περιεχόμενο ηλεκτρονικού ταχυδρομείου που δημιουργείται από το μοντέλο.
Προέλευση	Επεξεργασία backend μετά τη δημιουργία από το μοντέλο.
Δεδομένα Εξόδου	Προεπισκόπηση ηλεκτρονικού ταχυδρομείου HTML που έχει αποδοθεί.
Προορισμός	Διεπαφή χρήστη/UI (σελίδα προγράμματος περιήγησης).
Ενέργεια	Το backend εισάγει το δημιουργημένο περιεχόμενο ηλεκτρονικού ταχυδρομείου σε ένα πρότυπο HTML, το οποίο παρέχεται στον χρήστη για έλεγχο.
Απαίτηση	Απόδοση και οπτικοποίηση ηλεκτρονικού ταχυδρομείου.
Προϋπόθεση ή Προσυνθήκη	Επιτυχής απάντηση από το μοντέλο με έγκυρο περιεχόμενο.
Αποτέλεσμα ή Μετασυνθήκη	Ο χρήστης βλέπει την προεπισκόπηση του ηλεκτρονικού ταχυδρομείου σε μορφοποιημένη διάταξη.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα.

Πίνακας 4: 3.4.3.3.3 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Παράδοση email μέσω SMTP
Περιγραφή	Το σύστημα στέλνει τα email που δημιουργούνται στους παραλήπτες μέσω ενός διαμορφωμένου διακομιστή SMTP, χρησιμοποιώντας ασφαλή διαπιστευτήρια που ορίζονται σε ένα αρχείο .env.
Δεδομένα Εισόδου	Email αποστολέα, email παραλήπτη, θέμα email, σώμα email (προαιρετικά HTML).
Προέλευση	Ενέργεια χρήστη μέσω ενός αιτήματος ή μιας ενεργοποίησης δρομολόγησης (π.χ. /send).
Δεδομένα Εξόδου	Απεσταλμένο email και μήνυμα επιβεβαίωσης.

Προορισμός	Διεύθυνση email παραλήπτη (μέσω διακομιστή SMTP).
Ενέργεια	Μετά τη σύνταξη του email, ο διακομιστής συνδέεται με το SMTP, επαληθεύει την ταυτότητά του και αποστέλλει το μήνυμα. Ένα μήνυμα κατάστασης επιστρέφεται στον χρήστη.
Απαίτηση	Ενσωμάτωση δυνατοτήτων αποστολής email.
Προϋπόθεση ή Προσυνθήκη	Έγκυρα διαπιστευτήρια SMTP και επιτυχής δημιουργία περιεχομένου email.
Αποτέλεσμα ή Μετασυνθήκη	Αποστολή email στον παραλήπτη, επιβεβαίωση που παρέχεται στον χρήστη.
Πλευρικά Φαινόμενα ή Παρενέργειες	Μπορεί να περιλαμβάνει μηνύματα σφάλματος εάν η σύνδεση SMTP ή τα διαπιστευτήρια είναι λανθασμένα.

Πίνακας 5: 3.4.3.3.4 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Ρύθμιση παραμέτρων εφαρμογής μέσω μεταβλητών περιβάλλοντος
Περιγραφή	Οι προγραμματιστές διαχειρίζονται τις ρυθμίσεις συστήματος, συμπεριλαμβανομένων των διαπιστευτηρίων διακομιστή SMTP, του αριθμού θύρας και της ρύθμισης παραμέτρων μοντέλου, χρησιμοποιώντας μεταβλητές περιβάλλοντος που φορτώνονται κατά τον χρόνο εκτέλεσης.
Δεδομένα Εισόδου	Διακομιστής SMTP, θύρα SMTP, διαπιστευτήρια email, αριθμός θύρας, όνομα μοντέλου.
Προέλευση	Αρχείο .env
Δεδομένα Εξόδου	Ρύθμιση παραμέτρων συστήματος που χρησιμοποιούνται κατά τον χρόνο εκτέλεσης.
Προορισμός	Μνήμη εφαρμογής (επίπεδο ρύθμισης παραμέτρων).
Ενέργεια	Οι μεταβλητές περιβάλλοντος φορτώνονται κατά την εκκίνηση χρησιμοποιώντας το dotenv και εφαρμόζονται για την αρχικοποίηση του διακομιστή

	αλληλογραφίας, των ρυθμίσεων διαδρομής και της επιλογής μοντέλου.
Απαίτηση	Κεντρική και ασφαλής ρύθμιση παραμέτρων συστήματος.
Προϋπόθεση ή Προσυνθήκη	Το αρχείο .env πρέπει να υπάρχει και να έχει σωστή μορφοποίηση.
Αποτέλεσμα ή Μετασυνθήκη	Η εφαρμογή λειτουργεί με καθορισμένες ρυθμίσεις.
Πλευρικά Φαινόμενα ή Παρενέργειες	Το εσφαλμένα διαμορφωμένο αρχείο .env μπορεί να εμποδίσει την εκκίνηση της εφαρμογής ή να οδηγήσει σε σφάλματα χρόνου εκτέλεσης.

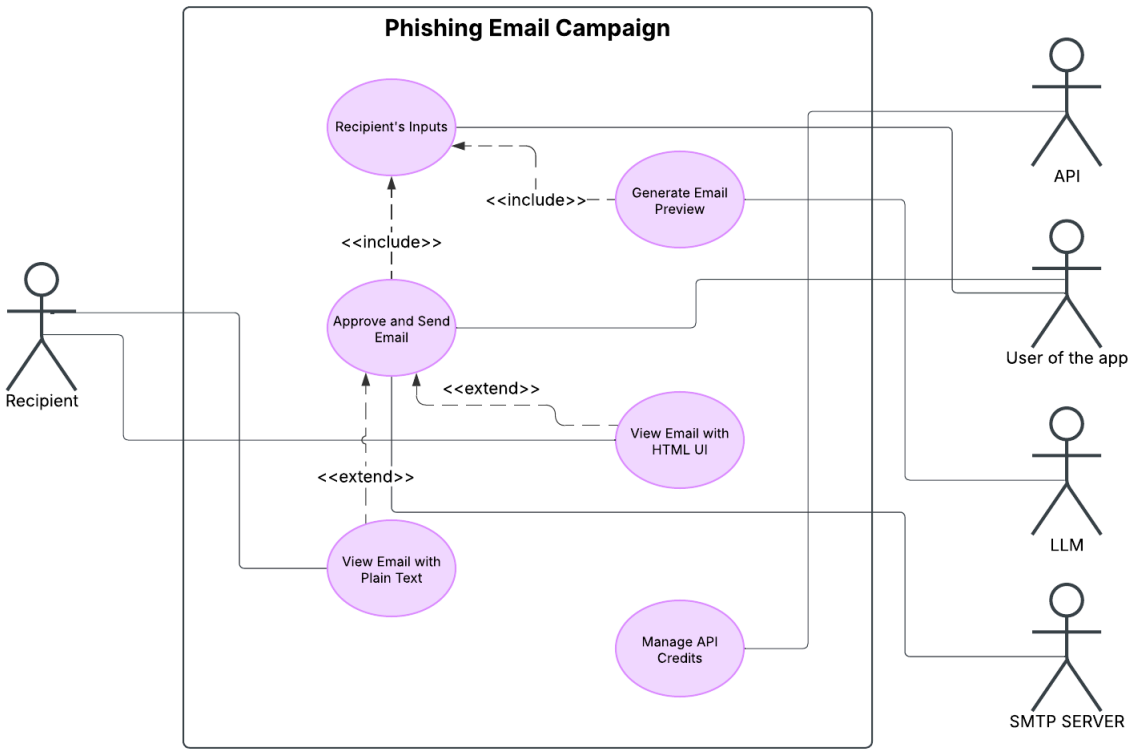
Πίνακας 6: 3.4.3.3.5 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Δυναμική Σύνδεση Θυρών (Dynamic Port Binding)
Περιγραφή	Η εφαρμογή ξεκινά σε μια θύρα που ορίζεται από τον χρήστη και ανακτάται από το αρχείο .env, επιτρέποντας ευέλικτη ανάπτυξη σε ποικίλα περιβάλλοντα.
Δεδομένα Εισόδου	Αριθμός θύρας (π.χ., 8080).
Προέλευση	Αρχείο διαμόρφωσης .env.
Δεδομένα Εξόδου	Διακομιστής FastAPI που εκτελείται στην καθορισμένη θύρα.
Προορισμός	Περιβάλλον κεντρικού υπολογιστή του διακομιστή.
Ενέργεια	Κατά την εκκίνηση, η εφαρμογή διαβάζει τη μεταβλητή περιβάλλοντος για τη θύρα και συνδέει την εφαρμογή FastAPI με αυτήν τη θύρα.
Απαίτηση	Ευελιξία ανάπτυξης σε διαφορετικά περιβάλλοντα.
Προϋπόθεση ή Προσυνθήκη	Έγκυρος αριθμός θύρας που ορίζεται στο .env.
Αποτέλεσμα ή Μετασυνθήκη	Η εφαρμογή παρακολουθεί τη συγκεκριμένη θύρα για εισερχόμενες συνδέσεις.
Πλευρικά Φαινόμενα ή Παρενέργειες	Η εφαρμογή ενδέχεται να μην ξεκινήσει εάν η θύρα χρησιμοποιείται ή δεν είναι έγκυρη.

3.4.4 Διαγράμματα UML

3.4.4.1 Διάγραμμα Περιπτώσεων Χρήσης

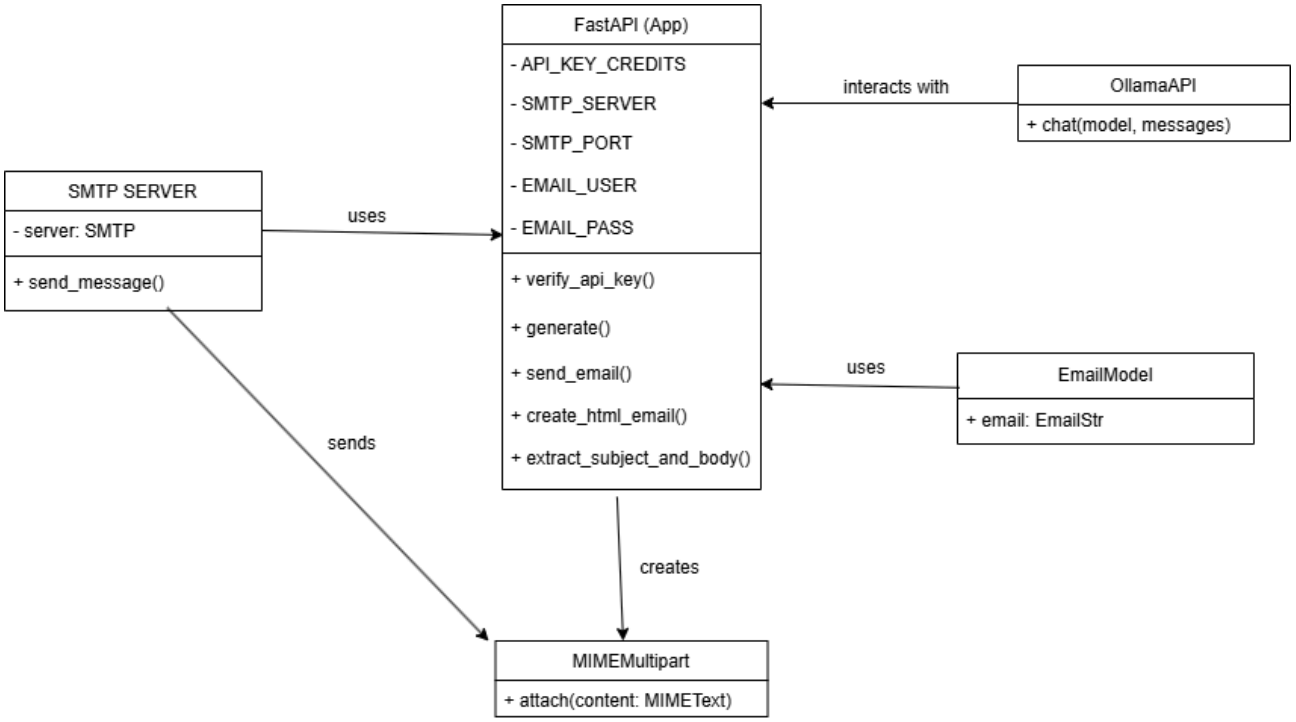
Παρακάτω ακολουθεί το διάγραμμα περιπτώσεων χρήσης της πρώτης μεθόδου (Σχήμα 9).



Σχήμα 9: Διάγραμμα Περιπτώσεων Χρήσης - FastAPI

3.4.4.2 Διάγραμμα Κλάσεων

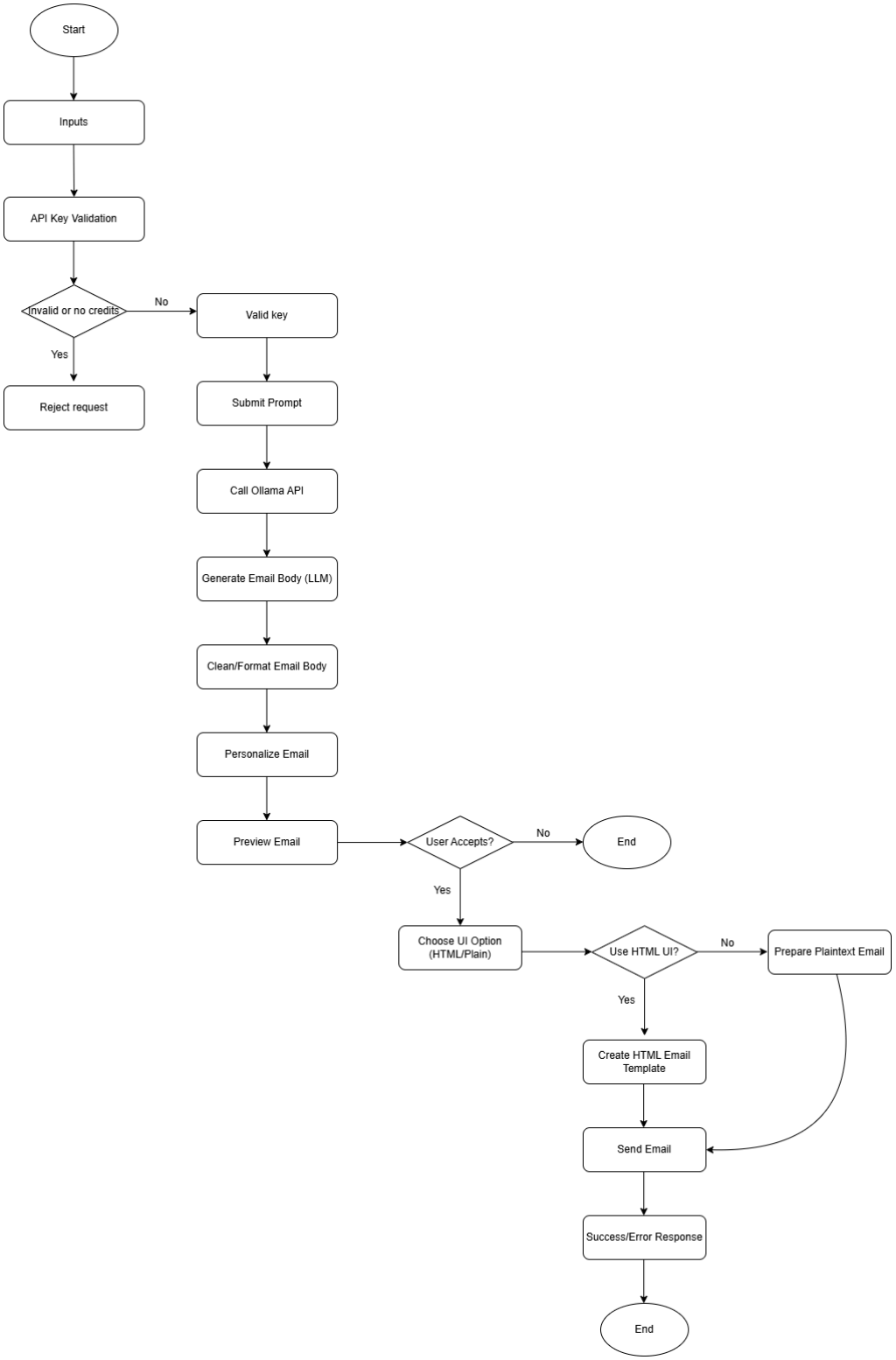
Παρακάτω ακολουθεί το διάγραμμα κλάσεων της πρώτης μεθόδου (Σχήμα 10).



Σχήμα 10: Διάγραμμα κλάσεων - FastAPI

3.4.4.3 Διάγραμμα Κατάστασης

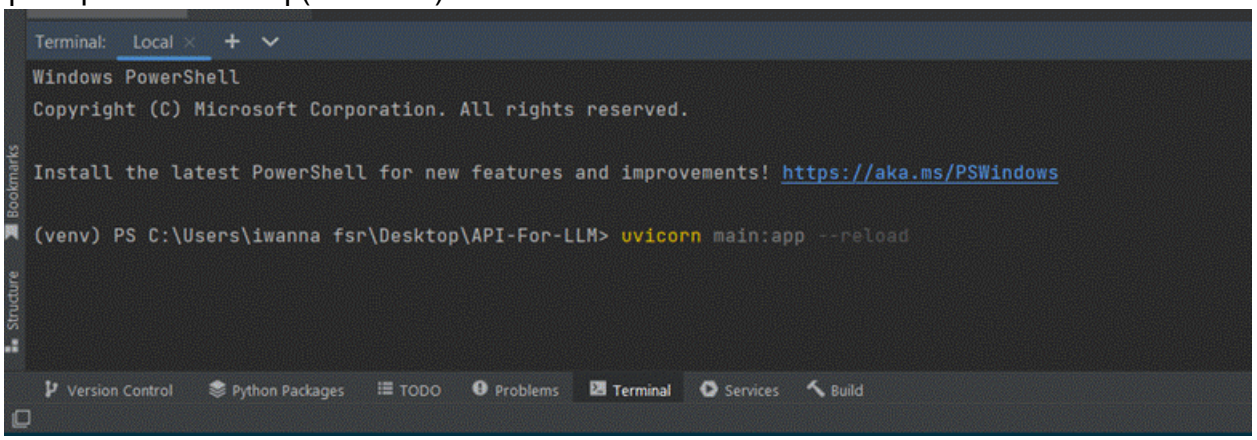
Παρακάτω ακολουθεί το διάγραμμα κατάστασης της πρώτης μεθόδου (Σχήμα 11).



Σχήμα 11: Διάγραμμα Κατάστασης - FastAPI

3.4.5 Τρόπος λειτουργίας με παραδείγματα

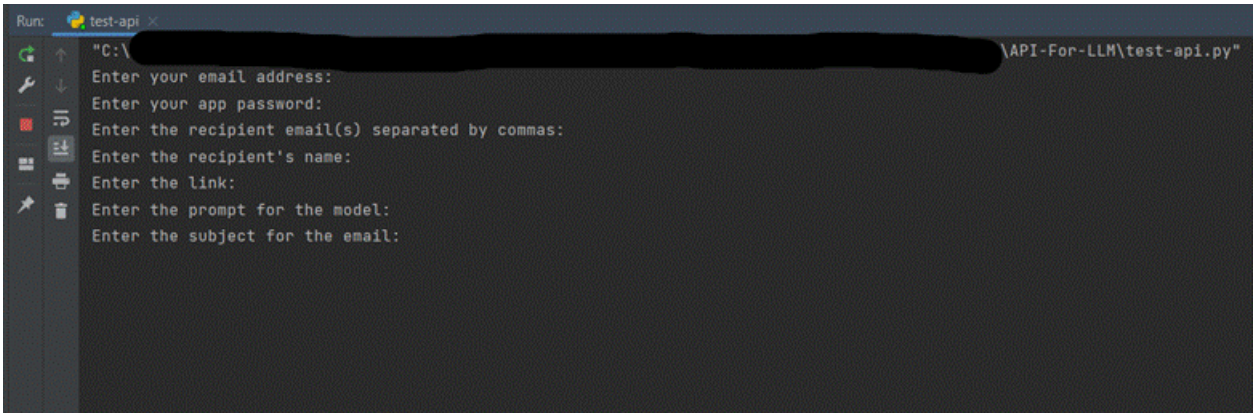
Για την εκτέλεση της εφαρμογής, αρχικά ανοίγουμε το τερματικό (terminal) και εκτελούμε την παρακάτω εντολή (Εικόνα 2):



Εικόνα 3: Εντολή τερματικού

Η εντολή αυτή θέτει σε λειτουργία το API. Με την ολοκλήρωση της εκτέλεσης, εμφανίζεται στο terminal η πληροφορία σχετικά με το port στο οποίο "ακούει" το API, δηλαδή σε ποια θύρα είναι διαθέσιμο για αλληλεπίδραση.

Στη συνέχεια, όταν ο χρήστης εκτελέσει το αρχείο test-api.py, εμφανίζεται διαδραστικά ένα πλαίσιο εισαγωγής δεδομένων (inputs), τα οποία απαιτείται να συμπληρωθούν από τον χρήστη ένα προς ένα, όπως φαίνεται παρακάτω (Εικόνα 3):



Εικόνα 4: Δεδομένα εισαγωγής

Αφού ολοκληρωθεί η καταχώρηση των ζητούμενων στοιχείων, το πρόγραμμα επιστρέφει στο terminal το παραγόμενο (generated) email. Αυτό προσφέρει στον χρήστη τη δυνατότητα να αξιολογήσει το αποτέλεσμα και, εφόσον δεν είναι ικανοποιημένος με το περιεχόμενο, να ζητήσει την αναδημιουργία (regeneration) του email προτού προχωρήσει στην αποστολή του στον καθορισμένο παραλήπτη. Κατόπιν, παρουσιάζονται στον χρήστη δύο επιλογές: 1) Αν επιθυμεί την άμεση αποστολή του email στη διεύθυνση που έχει ήδη εισαγάγει και 2) Αν επιθυμεί το email να σταλεί εμπλουτισμένο με ένα πρότυπο διεπαφής χρήστη (UI) σε μορφή HTML (Εικόνα 4):

```
Do you want to send this email? (yes or no): yes
Do you want to send the email with an UI? (yes or no): no
```

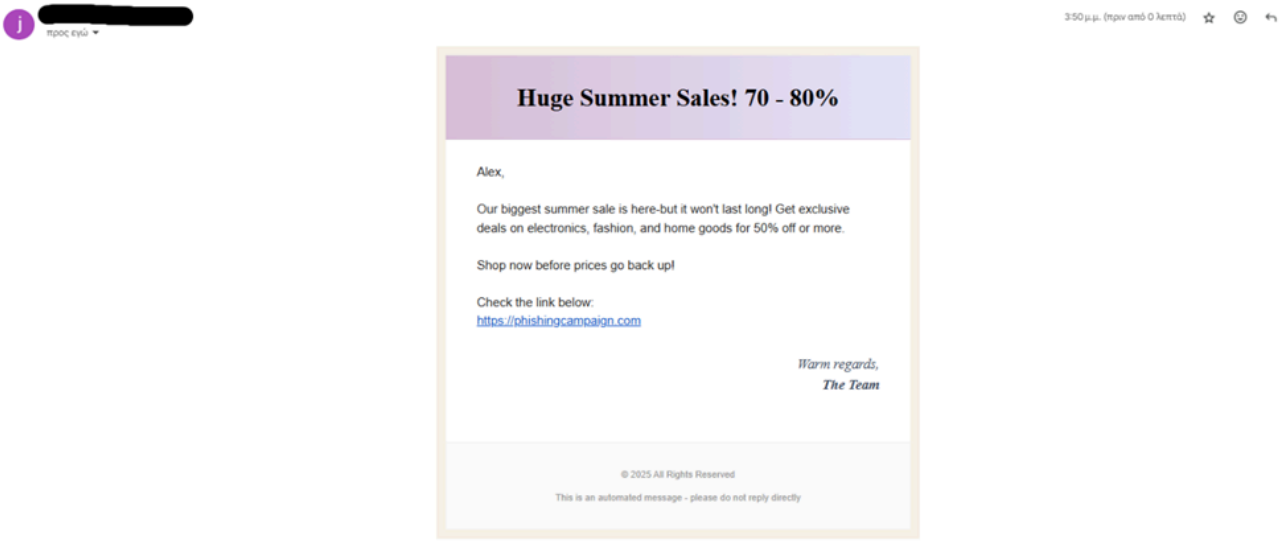
Εικόνα 5: Επιλογή αποστολής και HTML

Εφόσον ο χρήστης επιλέξει "yes" για την αποστολή του email, το σύστημα εμφανίζει σχετικό επιβεβαιωτικό μήνυμα, και η αποστολή πραγματοποιείται επιτυχώς στη δηλωθείσα διεύθυνση παραλήπτη (Εικόνα 5):

```
Status Code: 200
Response: {'message': 'Email sent to [redacted]', 'subject': 'Big Summer Sales -80%!', 'body': "Dear JO,\n\nHope this

Process finished with exit code 0
```

Εικόνα 6: Μήνυμα επιβεβαίωσης αποστολής

Αν ο χρήστης έχει επιλέξει να χρησιμοποιηθεί το HTML UI, τότε το παραδοτέο email θα έχει διαμορφωθεί με οπτικά και λειτουργικά στοιχεία σύμφωνα με το αντίστοιχο πρότυπο HTML, προσφέροντας μία πιο δομημένη και αισθητικά ευχάριστη παρουσίαση του περιεχομένου (Εικόνα 6):



Εικόνα 7: Παράδειγμα προτύπου HTML

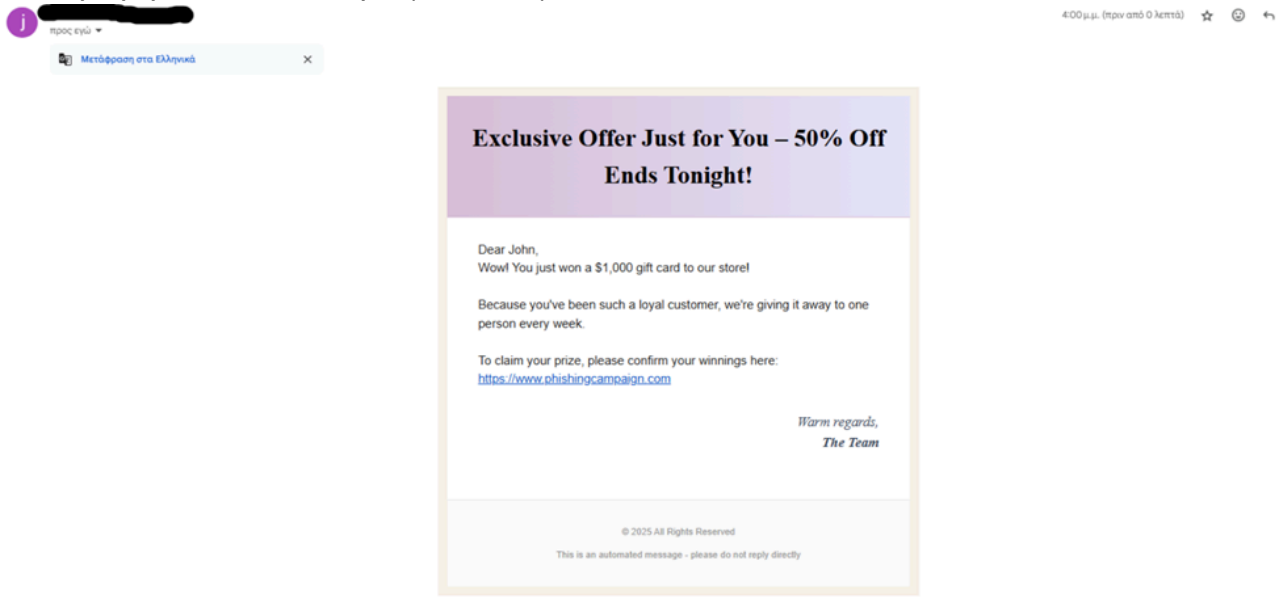
Σε αντίθετη περίπτωση, δηλαδή αν επιλεγεί η αποστολή ως απλό κείμενο (Plain Text), το email αποστέλλεται χωρίς μορφοποίηση HTML, παρουσιάζοντας το περιεχόμενο με τη βασική, απλή μορφή κειμένου (Εικόνα 7):

Εικόνα 8: Παράδειγμα απλού κειμένου

Παράδειγμα:

Στοιχεία εισαγωγής:
Enter your email address: example@gmail.com
Enter your app password: xxxxxxxxxxxx
Enter the recipient email(s) separated by commas: example2@gmail.com
Enter the recipient's name: John
Enter the link: https://www.phishingcampaign.com
Enter the prompt for the model: Generate a persuasive phishing email pretending to be from a popular online retail store
Enter the subject for the email: Exclusive Offer Just for You – 50% Off Ends Tonight!

Το παραγόμενο αποτέλεσμα (Εικόνα 8):



Εικόνα 9: Παράδειγμα παραγόμενου αποτελέσματος

3.4.6 Παρουσίαση μικρών χαρακτηριστικών τμημάτων κώδικα και επεξήγηση της λειτουργίας τους

3.4.6.1 Ρύθμιση παραμέτρων και αρχικοποίηση περιβάλλοντος (Environment Configuration and Initialization)

Ο κώδικας ξεκινά με την εισαγωγή βασικών βιβλιοθηκών, όπως το FastAPI για API handling, το ollama για σύνδεση με το προσαρμοσμένο LLM και τα smtplib και email.mime για την αποστολή email. Χρησιμοποιεί το dotenv για τη φόρτωση ευαίσθητων διαπιστευτηρίων από ένα αρχείο .env. Μεταξύ αυτών είναι οι λεπτομέρειες του διακομιστή email SMTP (SMTP_SERVER, SMTP_PORT, κ.λπ.) και το API_KEY, το οποίο χρησιμοποιείται για τον έλεγχο ταυτότητας αιτημάτων API και την εκχώρηση ενός σταθερού αριθμού (5) ανά κλειδί για περιορισμένη χρήση. Αυτό αποθηκεύεται σε ένα dictionary Python API_KEY_CREDITS και εκτυπώνεται κατά την εκκίνηση για τον εντοπισμό σφαλμάτων. Η παρουσία της εφαρμογής δημιουργείται χρησιμοποιώντας το FastAPI() και χρησιμεύει ως η βάση του διακομιστή backend.

Ο κώδικας:

```
from fastapi import FastAPI, Depends, HTTPException, Header, Form
from pydantic import EmailStr, ValidationError, BaseModel
import ollama
import os
import smtplib
import re
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from dotenv import load_dotenv
import datetime

load_dotenv()  # Loads a value from our environment variable file

API_KEY_CREDITS = {os.getenv("API_KEY"): 5}
print("Loaded API Keys with credits:", API_KEY_CREDITS)

app = FastAPI()

SMTP_SERVER = os.getenv("SMTP_SERVER")
SMTP_PORT = int(os.getenv("SMTP_PORT", 587))
EMAIL_USER = os.getenv("EMAIL_USER")
EMAIL_PASS = os.getenv("EMAIL_PASS")
```

3.4.6.2 Email Template Generator (create_html_email)

Αυτή η συνάρτηση δημιουργεί ένα οπτικά σχεδιασμένο email HTML με αισθητική υψηλής ποιότητας. Λαμβάνει το σώμα του email, το όνομα του παραλήπτη, το θέμα και τον σύνδεσμο του κακόβουλου link, μορφοποιώντας τα σε μια δομημένη συμβολοσειρά HTML. Το στυλ περιλαμβάνει μια κεφαλίδα με διαβάθμιση σε λιλά χρώμα, μοντέρνα τυπογραφία και ένα κουμπί που περιέχει το κακόβουλο link. Αυτό το πρότυπο διασφαλίζει ότι το email ηλεκτρονικού "ψαρέματος" φαίνεται ελκυστικό και επαγγελματικά σχεδιασμένο, γεγονός που αυξάνει την πιθανή αποτελεσματικότητά του.

Ο κώδικας:

```
def create_html_email(body: str, recipient_name: str, subject: str, link: str) -> str:
      formatted_body = body.replace('\n', '<br>')
      html_template = f"""
      <!DOCTYPE html>
      <html lang="en">
      ...
      </html>
      """
      return html_template
```

3.4.6.3 Επαλήθευση Κλειδιού API (API Key Verification)

Ένα βασικό χαρακτηριστικό ασφαλείας αυτού του API είναι η συνάρτηση verify_api_key, η οποία ελέγχει τις κεφαλίδες αιτημάτων HTTP για την παρουσία ενός έγκυρου x-api-key. Εάν το κλειδί λείπει ή έχει εξαντλήσει το όριο χρήσης του (δηλαδή, ο αριθμός πιστώσεων του είναι 0), ενεργοποιείται μια HTTPException με κατάσταση 401 Unauthorized (Μη Εξουσιοδοτημένο). Αυτό προστατεύει το LLM από κατάχρηση και διασφαλίζει ότι μόνο εξουσιοδοτημένοι χρήστες μπορούν να δημιουργήσουν περιεχόμενο ηλεκτρονικού "ψαρέματος".

Ο κώδικας:

```
def verify_api_key(x_api_key: str = Header(None)):
      credits = API_KEY_CREDITS.get(x_api_key, 0)
      if credits <= 0:
      raise HTTPException(status_code=401, detail="Invalid API Key, or no credits left")
      return x_api_key
```

3.4.6.4 Βοηθητικό πρόγραμμα επεξεργασίας κειμένου

Υπάρχει μια βοηθητική λειτουργία για την προετοιμασία του περιεχομένου του email, το *extract_subject_and_body* το οποίο αφαιρεί οποιοδήποτε πρόθεμα "Subject:" και εξάγει την ενότητα του σώματος χρησιμοποιώντας regex. Πιο συγκεκριμένα, αυτός ο βοηθός χρησιμοποιεί regex για να εξαγάγει τα Subject: και Body: από το κείμενο εξόδου του LLM και διασφαλίζει ότι πληρούται η δομή που αναμένεται από το LLM. Εάν λείπει κάποιο από τα δύο, εμφανίζεται ένα ValueError, το οποίο οδηγεί σε επαναλήψεις στο τελικό σημείο κλήσης του /generate.

Ο κώδικας:

```
def extract_subject_and_body(text):
      subject_match = re.search(r"Subject:\s*(.+)", text)
      body_match = re.search(r"Body:\s*(.+)", text, re.DOTALL)

      if subject_match and body_match:
      subject = subject_match.group(1).strip()
      body = body_match.group(1).strip()
      return subject, body
      else:
      raise ValueError("Couldn't extract subject and body from the
text.")
```

3.4.6.5 /generate Endpoint (Ενσωμάτωση LLM με Ollama)

Το /generate POST endpoint είναι το βασικό στοιχείο όπου καλείται το LLM. Αναμένει δεδομένα που περιέχουν μια προτροπή, ένα recipient_name και ένα link, μαζί με ένα έγκυρο κλειδί API. Μετά από επιτυχές αίτημα: 1) Η προτροπή αποστέλλεται στο LLM που φιλοξενείται μέσω του ollama.chat, αναφέροντας ένα μοντέλο που έχει βελτιωθεί και φιλοξενείται στο Ollama (gawyria/demo-model), 2) Το LLM δημιουργεί ένα κείμενο email ηλεκτρονικού "ψαρέματος" με βάση την προτροπή, 3) Το περιεχόμενο αναλύεται και εξατομικεύεται χρησιμοποιώντας την προαναφερθείσα βοηθητική συνάρτηση, 4) Η συνάρτηση επιστρέφει μια απόκριση JSON με το καθαρισμένο και μορφοποιημένο σώμα του email, δίνοντας στον χρήστη μια προεπισκόπηση πριν από την αποστολή.

Πιο συγκεκριμένα, ο χρήστης υποβάλλει ένα μήνυμα (π.χ., "δημιουργία email ηλεκτρονικού "ψαρέματος" που προσποιείται ότι προέρχεται από τράπεζα") και το μοντέλο ερωτάται μέσω του ollama.chat. Η απάντηση αναλύεται για ένα Subject: και Body: χρησιμοποιώντας regex. Το placeholder [Insert Call-to-Action button or link] πρέπει να υπάρχει στο σώμα. Εάν όχι, το API προσπαθεί ξανά έως και 5 φορές, υποθέτοντας ότι το μοντέλο ενδέχεται να παράγει μεταβλητά αποτελέσματα. Μόλις ολοκληρωθεί με επιτυχία, το σώμα αντικαθιστά το [Name] με το όνομα του πραγματικού παραλήπτη και εισάγει τον παρεχόμενο σύνδεσμο εκεί που υπήρχε το placeholder. Το σύστημα στη συνέχεια επιστρέφει το προεπισκοπημένο θέμα και σώμα, επιτρέποντας στον πελάτη (frontend ή χρήστη) να αποφασίσει εάν θα προχωρήσει στην αποστολή του email.

Αυτό είναι το βασικό τελικό σημείο που αλληλεπιδρά με το LLM που φιλοξενείται μέσω του Ollama: response = ollama.chat(model="gawyria/demo-model", messages=[{"role": "user", "content": prompt}])

Ο κώδικας:

```
@app.post("/generate")
def generate(
      prompt: str = Form(...),
      recipient_name: str = Form(...),
      link: str = Form(...),
      x_api_key: str = Depends(verify_api_key)
):
      API_KEY_CREDITS[x_api_key] -= 1
```

```
    MAX_ATTEMPTS = 5
    placeholder = "[Insert Call-to-Action button or link]"

    for attempt in range(MAX_ATTEMPTS):
    try:
        response = ollama.chat(
            model="gawyria/demo-model",
            messages=[{"role": "user", "content": prompt}]
        )
        llm_output = response["message"]["content"]
        print(f"Attempt {attempt + 1} LLM OUTPUT:\n", llm_output)

        # Ensure the placeholder exists before proceeding
        if placeholder not in llm_output:
            print(f"Placeholder '{placeholder}' not found.
Retrying...")
            continue

        # Extract subject and body
        subject, body = extract_subject_and_body(llm_output)

        # Replace placeholders
        body = re.sub(r"\[Name\]", recipient_name, body,
flags=re.IGNORECASE)
        body = re.sub(re.escape(placeholder), link, body,
flags=re.IGNORECASE)

        return {
            "message": "Preview generated. Use this subject and body
for approval.",
            "subject": subject,
            "body": body
        }

    except Exception as e:
        if attempt == MAX_ATTEMPTS - 1:
            raise HTTPException(status_code=500, detail=f"Final
attempt failed: {str(e)}")

    raise HTTPException(status_code=500, detail=f"Failed to generate
content with required placeholder after {MAX_ATTEMPTS} attempts.")
```

3.4.6.6 /send_email Endpoint (Email Transmission)

Μόλις ένας χρήστης εγκρίνει το δημιουργημένο περιεχόμενο, ενεργοποιείται το τελικό σημείο /send_email. Αυτό το τελικό σημείο: 1) Λαμβάνει το μορφοποιημένο σώμα, το θέμα, τα email του παραλήπτη, το προαιρετικό όνομα παραλήπτη και το εάν πρέπει να χρησιμοποιηθεί το HTML UI, 2) Επικυρώνει όλες τις διευθύνσεις email του παραλήπτη

χρησιμοποιώντας το EmailStr της Pydantic, 3) Δημιουργεί ένα email MIME πολλαπλών τμημάτων, προαιρετικά επισυνάπτοντας το πολυτελές πρότυπο HTML, 4) Συνδέεται στον διακομιστή SMTP χρησιμοποιώντας τα διαπιστευτήρια που παρέχονται στο αρχείο .env ή από τις εισόδους της φόρμας, 5) Στέλνει το email χρησιμοποιώντας το smtplib.SMTP, 6) Οποιαδήποτε αποτυχία σε αυτήν τη διαδικασία δημιουργεί μια εξαίρεση HTTP με ένα κατάλληλο μήνυμα για εντοπισμό σφαλμάτων.

Ο κώδικας:

```
@app.post("/send_email")
def send_email(
      body: str = Form(...),
      email: str = Form(...),
      subject: str = Form(...),
      recipient_name: str = Form(None),
      use_html_ui: bool = Form(False),
      link: str = Form(...),
      sender_email: EmailStr = Form(...),
      sender_password: str = Form(...)
):
      try:
      raw_emails = [e.strip() for e in email.split(",")]

      class EmailModel(BaseModel):
            email: EmailStr

      recipients = []
      for e in raw_emails:
            validated = EmailModel(email=e)
            recipients.append(validated.email)

      msg = MIMEMultipart("alternative")
      msg["From"] = sender_email
      msg["To"] = email
      msg["Subject"] = subject

      if not recipient_name:
            recipient_name = "Valued Customer"

      if use_html_ui:
            html_body = create_html_email(body, recipient_name, subject,
link)
            msg.attach(MIMEText(html_body, "html"))
      else:
            msg.attach(MIMEText(body, "plain"))

      with smtplib.SMTP(SMTP_SERVER, SMTP_PORT) as server:
            server.starttls()
            server.login(sender_email, sender_password)
            server.send_message(msg)
```

```
      return {
            "message": f"Email sent to {email}",
            "subject": subject,
            "body": body,
            "html_ui_used": use_html_ui
      }
      except Exception as e:
      raise HTTPException(status_code=500, detail=f"Failed to send email:
{str(e)}")
```

3.4.6.7 Command-Line User Interface (CLI)

Το τελευταίο μπλοκ του κώδικα παρέχει μια διεπαφή γραμμής εντολών που αυτοματοποιεί την πλήρη αλληλεπίδραση του χρήστη: 1) Ζητά από τον χρήστη δεδομένα εισόδου, συμπεριλαμβανομένων των διαπιστευτηρίων του αποστολέα, των στοιχείων του παραλήπτη και μιας προτροπής, 2) Υποβάλλει ένα αίτημα HTTP POST στο τελικό σημείο /generate για προεπισκόπηση του email ηλεκτρονικού "ψαρέματος", 3) Εμφανίζει το δημιουργημένο περιεχόμενο και ζητά επιβεβαίωση από τον χρήστη, 4) Εάν ο χρήστης συμφωνήσει, προαιρετικά ρωτά αν θα στείλει το email χρησιμοποιώντας το περιβάλλον χρήστη (πρότυπο HTML) ή ως απλό κείμενο, 5) Τέλος, υποβάλλει ένα αίτημα στο /send_email, το οποίο παραδίδει το email μέσω SMTP.

Ο κώδικας:

```
import requests
from dotenv import load_dotenv
import os

load_dotenv()

sender_email = input("Enter your email address: ")
sender_password = input("Enter your app password: ")
email = input("Enter the recipient email(s) separated by commas: ")
recipient_name = input("Enter the recipient's name: ")
link = input("Enter the link: ")
prompt = input("Enter the prompt for the model: ")
subject = input("Enter the subject for the email: ")

headers = {
      "x-api-key": os.getenv("API_KEY"),
      "Content-Type": "application/x-www-form-urlencoded"
}

preview_data = {
      "prompt": prompt,
      "recipient_name": recipient_name,
      "link": link
}
```

```
preview_response = requests.post("http://localhost:8000/generate",
headers=headers, data=preview_data)

if preview_response.status_code != 200:
    print("Failed to generate preview.")
    print("Error:", preview_response.text)
    exit()

body = preview_response.json()["body"]
print("\n=== EMAIL PREVIEW ===")
print("Subject:", subject)
print("Body:\n", body)
print("====================\n")

choice = input("Do you want to send this email? (yes or no):
").strip().lower()
if choice != "yes":
    print("Email not sent.")
    exit()

use_ui = input("Do you want to send the email with an UI? (yes or no):
").strip().lower()
if choice != "yes":
    print("Email sent without a ui.")
    exit()

send_data = {
    "email": email,
    "subject": subject,
    "body": body,
    "recipient_name": recipient_name,
    "link": link,
    "use_html_ui": str(use_ui),
    "sender_email": sender_email,
    "sender_password": sender_password
}

send_response = requests.post("http://localhost:8000/send_email",
headers=headers, data=send_data)

print("Status Code:", send_response.status_code)
try:
    print("Response:", send_response.json())
except Exception as e:
    print("Failed to parse JSON:", e)
    print("Raw response:", send_response.text)
```

3.4.7 Περιγραφή παραθύρων εφαρμογής, παρουσίαση εκτυπώσεων, πίνακες με καταστάσεις λειτουργίας.

3.4.7.1 Περιγραφή παραθύρων / διεπαφών εφαρμογής

Αυτή η εφαρμογή είναι ένα εργαλείο βασισμένο στο FastAPI που αλληλεπιδρά με ένα προσαρμοσμένο, βελτιωμένο μοντέλο μεγάλης γλώσσας που φιλοξενείται στο Ollama.com. Το LLM εκπαιδεύεται μέσω του Unsloth στο Google Colab σε ένα σύνολο δεδομένων 86 δειγμάτων email ηλεκτρονικού "ψαρέματος". Δημιουργεί ρεαλιστικά email ηλεκτρονικού "ψαρέματος" με βάση τις προτροπές του χρήστη και προαιρετικά τα στέλνει μέσω email χρησιμοποιώντας ένα κομψό πρότυπο email HTML.

Η εφαρμογή χωρίζεται σε τρία κύρια λειτουργικά παράθυρα (καταστάσεις εφαρμογής), το καθένα με τη δική του συμπεριφορά, εισαγωγή/έξοδο δεδομένων και σκοπό.

3.4.7.1.1 Παράθυρα εφαρμογών και καταστάσεις λειτουργίας

Λειτουργία:

- Ζητείται από τον χρήστη να εισάγει βασικές λεπτομέρειες:
    1. Τα διαπιστευτήρια του αποστολέα
    2. Email(s) του/των παραλήπτη/ παραληπτών
    3. Το όνομα του παραλήπτη
    4. Προτροπή για την περιγραφή του σεναρίου ηλεκτρονικού "ψαρέματος"
    5. Το θέμα του email
    6. Το phishing link
- Αυτές οι τιμές αποστέλλονται στο τελικό σημείο /generate το οποίο:
    1. Στέλνει την προτροπή στο LLM που φιλοξενείται στο Ollama.
    2. Αναλύει και καθαρίζει την έξοδο του μοντέλου για να εξαγάγει μόνο το κυρίως κείμενο.
    3. Εξατομικεύει το σώμα με το όνομα του παραλήπτη και τον σύνδεσμο της επιλογής του αποστολέα.

Έξοδος εκτύπωσης:

=== EMAIL PREVIEW ===
Subject: [User-entered subject]
Body:
[Cleaned and personalized phishing email content]
==========

Αυτή η κατάσταση επιτρέπει στον χρήστη να κάνει προεπισκόπηση του δημιουργημένου περιεχομένου πριν από οποιαδήποτε παράδοση. Εξασφαλίζει ασφαλή εποπτεία από άνθρωπο σε άνθρωπο, η οποία είναι απαραίτητη δεδομένης της ευαίσθητης φύσης της δημιουργίας περιεχομένου τύπου ηλεκτρονικού "ψαρέματος". Αυτή η πύλη προεπισκόπησης επιβάλλει τον έλεγχο του περιεχομένου και τη συμμόρφωση με τους ηθικούς περιορισμούς.

3.4.7.1.2 Παράθυρο Επιβεβαίωσης και Επιλογής UI

Λειτουργία:

- Ρωτάτε στον χρήστη:
    1. Εάν θα προχωρήσει στην αποστολή του email.
    2. Εάν θα χρησιμοποιήσει το πρότυπο HTML ή απλό κείμενο για την παράδοση.

Έξοδος εκτύπωσης:

Do you want to send this email? (yes or no):
Do you want to send the email with an UI? (yes or no):

Αυτό το παράθυρο εισάγει ένα βήμα επιβεβαίωσης και μια προτίμηση εξατομίκευσης, διασφαλίζοντας ότι ο αποστολέας εγκρίνει συνειδητά το περιεχόμενο και τη μορφή. Η επιλογή HTML UI επιτρέπει μια οπτική βελτίωση χρησιμοποιώντας μια επαγγελματική διάταξη email με θέμα το λιλά, κατάλληλη για την προσομοίωση προτύπων ηλεκτρονικού "ψαρέματος" στον πραγματικό κόσμο.

3.4.7.1.3 Παράθυρο Αποστολής Email και Σχόλια Κατάστασης

Λειτουργία:

- Η εφαρμογή χρησιμοποιεί το smtplib για να στείλει το email χρησιμοποιώντας τα διαπιστευτήρια του αποστολέα.
- Υποστηρίζει πολλαπλούς παραλήπτες και επικυρώνει κάθε διεύθυνση email χρησιμοποιώντας το Pydantic.
- Το email αποστέλλεται είτε ως:
    1. Plain text (MIMEText)
    2. Μήνυμα σε μορφή HTML (MIMEMultipart με HTML)

Έξοδος εκτύπωσης:

Status Code: 200
Response: {
    "message": "Email sent to [emails]",
    "subject": "[Subject]",
    "body": "[Body]",
    "html_ui_used": true/false
}

Αυτό το τελευταίο παράθυρο είναι κρίσιμο για την παρακολούθηση της κατάστασης παράδοσης και την επιβεβαίωση της επιτυχίας της λειτουργίας. Η απόκριση δείχνει εάν χρησιμοποιήθηκε HTML, βοηθώντας τους χρήστες να αξιολογήσουν την αποτελεσματικότητα του UI. Ο χειρισμός εξαιρέσεων διασφαλίζει ότι τυχόν αποτυχίες (π.χ. σφάλματα σύνδεσης SMTP) αναφέρονται με σαφήνεια.

3.4.7.2 Παρουσίαση Εκτυπώσεων

Πίνακας 7: Παρουσίαση Εκτυπώσεων

Στάδιο	**Έξοδος εκτύπωσης/καταγραφής**
Ηλεκτρονική διεύθυνση αποστολέα	Enter your email address:
Κωδικός αποστολέα	Enter your app password:
Ηλεκτρονικές διευθύνσεις παραλήπτη/παραληπτών	Enter the recipient email(s) separated by commas:
Όνομα παραλήπτη	Enter the recipient's name:
Σύνδεσμος	Enter the link:
Προτροπή LLM	Enter the prompt for the model:
Θέμα του email	Enter the subject for the email:
Εξατομικευμένο περιεχόμενο	/n=== EMAIL PREVIEW ===\nSubject: ...\nBody: ...
Επιβεβαίωση	Do you want to send this email? (yes or no):
Επιλογή HTML	Do you want to send the email withan UI? (yes or no):
Τελική απάντηση	Response: { "message": "Email sent to [emails]", "subject": "[Subject]", "body": "[Body]", "html_ui_used": true/false
Διαχείριση σφαλμάτων (εάν υπάρχουν)	Failed to send email: [error message]

3.7.7.3 Πίνακες Καταστάσεων Εφαρμογής

Πίνακας 8: Διαχείριση κλειδιού API

API Key	**Κατάσταση**	**Credits που παραμένουν**	**Κατάσταση**
abcd1234apikey	5	4	Έγκυρο
expiredapikey	0	0	Απορρίφθηκε
Κανένα	-	-	Ανεξουσιοδότητος

Πίνακας 9: Καταστάσεις επεξεργασίας μοντέλου

Στάδιο	**Κατάσταση**	**Περιγραφή**

Υποβολή Προτροπής	Ολοκληρώθηκε	Ο χρήστης στέλνει μια προτροπή για τη δημιουργία email
Παραγωγή LLM	Επιτυχία/Σφάλμα	Το ollama δημιουργεί απάντηση από το μοντέλο gawyria/pers-com-email-model:Q8_0
Αντικατάσταση Placeholder	Εξατομικευμένο	Εισάγει το όνομα του παραλήπτη και τον σύνδεσμο της επιλογής του αποστολέα
Προεπισκόπηση απόδοσης	Εμφάνιση στην οθόνη	Email που εμφανίζεται στον χρήστη με θέμα και σώμα

Πίνακας 10: Καταστάσεις αποστολής ηλεκτρονικού μηνύματος ταχυδρομείου

Βήμα	**Πεδίο**	**Παράδειγμα**	**Κατάσταση**
Νομιμοποίηση (Validation)	Έλεγχος μορφής ηλεκτρονικής διεύθυνσης	sakhs@example.com	Έγκυρο
Αυθεντικοποίηση (Authentication)	Σύνδεση με τον SMTP	EMAIL_USER, EMAIL_PASS	Συνδεδεμένος/η
Επιλογή μορφής	HTML ή απλό κείμενο (Plain Text)	Do you want to send the email with an UI? (yes or no): yes	Αποστολή με HTML
Αποστολή	Μετάδοση μηνύματος	return { "message": f"Email sent to {email}", "subject": subject, "body": body, "html_ui_used": use_html_ui }	Επιτυχία

Πίνακας 11: Καταστάσεις σφάλαμτος και εξαιρέσεις

Σενάριο σφάλματος	**Κωδικός κατάστασης**	**Μήνυμα**
Δεν δημιουργήθηκε θέμα και σώμα από το LLM	-	“Couldn't extract subject and body from the text.”
Μη έγκυρο ή εξαντλημένο κλειδί API	401	“Invalid API Key, or no credits left.”
Η δημιουργία LLM απέτυχε	500	“Error generating or parsing content: ...”
Μη έγκυρη εισαγωγή διεύθυνσης ηλεκτρονικού ταχυδρομείου	400	“Invalid email: bad@email.”

Αποτυχία SMTP (λάθος κωδικός πρόσβασης, κ.λ.π)	500	“Failed to send email: SMTPAuthenticationError.”
Αποτυχία δημιουργίας περιεχομένου με το απαιτούμενο σύμβολο κράτησης θέσης (placeholder)	500	”Failed to generate content with required placeholder after {MAX_ATTEMPTS} attempts.”

3.5 Υλοποίηση – Μέθοδος Δεύτερη – Τοπικό Περιβάλλον

Ο δεύτερος τρόπος υλοποίησης της προσομοίωσης αποστολής ηλεκτρονικών μηνυμάτων Gmail τρέχει σε τοπικό περιβάλλον. Με αυτόν τον τρόπο, ο τελικός χρήστης μπορεί να έχει πλήρη έλεγχο του προγράμματος και είναι απόλυτα προστατευμένο από εξωτερικά (third-party) συστήματα. Το LLM εκπαιδεύτηκε και έγινε fine-tune μέσω του Google Colab και του Unsloth. Τα χαρακτηριστικά του Google Colab αποτελούνται από 13GB System RAM, 15GB GPU RAM, 113GB Δίσκου και ο επιλεγμένος Hardware Accelerator είναι ο T4 GPU, ενώ το μοντέλο που χρησιμοποιήθηκε από το Unsloth είναι το Llama3 με 8 Billion παραμέτρους.

3.5.1 Τι κάνει το λογισμικό

Ο κώδικας του λογισμικού ξεκινάει στέλνοντας μια προτροπή στο τοπικό LLM που τρέχει στον υπολογιστή μέσω του Ollama API (http://localhost:11434/api/generate) και της βιβλιοθήκης requests. Έπειτα, το API στέλνει πίσω την απάντηση του LLM, έτσι ώστε να μπορεί να χρησιμοποιηθεί στο υπόλοιπο πρόγραμμα. Στη συνέχεια, το λογισμικό χωρίζει το Θέμα και το Σώμα του ηλεκτρονικού μηνύματος μέσω μερικών regular expressions (βιβλιοθήκη re) και ελέγχει τη δομή τους. Σε περίπτωση που το Σώμα, Θέμα ή ο σύνδεσμος ηλεκτρονικού ψαρέματος λείπει από το αποτέλεσμα, τότε εγείρεται σφάλμα.

Συνεχίζοντας, ζητούνται από τον χρήστη να δώσει το Gmail του, το Google App Password του Gmail του χρήστη, το οποίο είναι κρυμμένο και δεν εμφανίζεται στη κονσόλα του προγράμματος με τη βοήθεια της βιβλιοθήκης getpass, όπως επίσης και το μονοπάτι (path) του .csv αρχείου που περιέχει τα emails και τα ονόματα των παραληπτών. Το email του αποστολέα οφείλει να είναι αυθεντικοποιημένο από τη Google για να μπορεί να αποστείλει τα ηλεκτρονικά μηνύματα, έτσι ώστε να αποτραπεί η κακόβουλη χρήση του. Για να αυθεντικοποιηθεί από τη Google, ο αποστολέας θα πρέπει να έχει ενεργοποιημένο το 2FA για το λογαριασμό του και μόνο τότε θα δημιουργηθεί το Google App Password του και επομένως μπορεί να πραγματοποιηθεί η σύνδεση με τον SMTP server της Google (smtp.gmail.com:587). Αφού ο τελικός χρήστης προσθέσει το .csv αρχείο, γίνεται έλεγχος της δομής του. Αναλυτικότερα, το αρχείο αυτό θα πρέπει να αποτελείται από δύο στήλες, η μία στήλη είναι η name και η άλλη η email (name,email) και κάθε γραμμή είναι απαραίτητο να περιέχει και τις δύο στήλες. Αν λείπει το όνομα, η σύνταξη το email είναι λανθασμένη με βάση ενός regular expression ή δεν υπάρχουν οι κατάλληλες επικεφαλίδες, τότε εγείρεται επεξηγηματικό σφάλμα.

Μετά από αυτούς τους ελέγχους, αρχίζει η λούπα που παράγει και στέλνει τα ηλεκτρονικά μηνύματα. Η διαδικασία ξεκινά με την εισαγωγή μιας προτροπής από τον χρήστη για το ηλεκτρονικό μήνυμα που επιθυμεί να δημιουργήσει. Στη συνέχεια, το LLM παράγει ένα πρώτο προσχέδιο και ελέγχεται για τη δομή του, δηλαδή άμα παρέχει Σώμα,

Θέμα ή τον σύνδεσμο ηλεκτρονικού ψαρέματος. Σε περίπτωση που δε πληρούνται αυτές οι προϋποθέσεις, τότε το LLM παράγει νέα έξοδο. Έπειτα, το τελικό επιθυμητό αποτέλεσμα εμφανίζεται στον χρήστη, ο οποίος καλείται να επιβεβαιώσει αν είναι ικανοποιημένος με αυτό. Εάν επιλέξει όχι, τότε η διαδικασία επαναλαμβάνεται μέχρι ο χρήστης να εγκρίνει το περιεχόμενο. Όταν ο χρήστης μείνει ικανοποιημένος και επιλέξει το περιεχόμενο που του αρμόζει, τότε ερωτάται εάν επιθυμεί το περιεχόμενο να σταλεί ως απλό κείμενο ή με ένα HTML πρότυπο όπως επίσης και ποιος θα είναι ο σύνδεσμος ηλεκτρονικού ψαρέματος. Στο μεταξύ, ο κώδικας προσωποποιεί το ηλεκτρονικό μήνυμα με το όνομα του παραλήπτη και τον σύνδεσμο της επιλογής του αποστολέα, στέλνοντας λοιπόν το παραγόμενο μήνυμα και ενημερώνοντας τον τελικό χρήστη για τα email που έχουν σταλεί, τερματίζοντας έτσι τη SMTP σύνδεση.

Παρακάτω υπάρχει ένας συνοπτικός πίνακας για τις χρήσεις των βιβλιοθηκών στο λογισμικό (Πίνακας 12).

Πίνακας 12: Βιβλιοθήκες Τοπικού Περιβάλλοντος

Βιβλιοθήκη	**Σκοπός**
smtplib	Σύνδεση και αποστολή Email (Gmail σε αυτή τη περίπτωση).
csv	Ανάλυση του .csv αρχείου που περιέχει τα ονόματα και τις ηλεκτρονικές διευθύνσεις των παραληπτών.
re (regex)	Χρήση regular expressions για έλεγχο και τον διαχωρισμό του Σώματος από το Θέμα του ηλεκτρονικού μηνύματος.
email.mime.multipart & email.mime.text	Δημιουργία απλού και σύνθετου HTML ηλεκτρονικού μηνύματος.
requests	Αποστολή HTTP POST requests στο τοπικό Ollama API έτσι ώστε να παρθεί πίσω η απάντηση από το AI.
getpass	Προστασία του κωδικού όταν αναγράφεται στη κονσόλα.

3.5.2 Είναι χαμηλού ή/και υψηλού επιπέδου;

Ο κώδικας της δεύτερης μεθόδου χαρακτηρίζεται ως υψηλού επιπέδου, καθώς βασίζεται σε βιβλιοθήκες ανωτέρου επιπέδου, που είναι χαρακτηριστικό γνώρισμα προγραμματισμού υψηλού επιπέδου, όπως οι smtplib, requests και email.mime, κάνοντας έτσι πιο εύχρηστη την διαχείριση πρωτοκόλλων email, επικοινωνία μέσω HTTP και επεξεργασία αρχείων. Επιπλέον, η ίδια η γλώσσα της Python είναι υψηλού επιπέδου, σε σύγκριση με γλώσσες χαμηλού επιπέδου (Assembly, C++), οι οποίες βρίσκονται πιο κοντά στη μνήμη και τους πόρους του υπολογιστή. Διαχειρίζεται κυρίως διεργασίες εφαρμογής, όπως παραγωγή ηλεκτρονικών μηνυμάτων, παρά διεργασίες όπως διαχείριση μνήμης και έλεγχο συσκευών, με τις οποίες ασχολούνται οι γλώσσες χαμηλού επιπέδου.

3.5.3 Λειτουργικές/μη λειτουργικές απαιτήσεις και Προδιαγραφές

Παρακάτω αναγράφονται οι λειτουργικές, μη λειτουργικές απαιτήσεις και προδιαγραφές της παρούσας εργασίας.

3.5.3.1 Λειτουργικές Απαιτήσεις

3.5.3.1.1. Θα πρέπει ο τελικός χρήστη να μπορεί να δώσει τα διαπιστευτήρια του λογαριασμού για το ηλεκτρονικό ταχυδρομείο του.

Περιγραφή: Ο χώρος εισαγωγής των στοιχείων αυτών φαίνονται στη κονσόλα της εφαρμογής. Ο τελικός χρήστης μπορεί να προσθέσει τα στοιχεία του.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη για να γίνει η σύνδεση με το ηλεκτρονικό ταχυδρομείο του χρήστη.

3.5.3.1.2. Θα πρέπει ο τελικός χρήστης να έχει τη δυνατότητα να εξετάσει το περιεχόμενο του παραγόμενου αποτελέσματος του LLM.

Περιγραφή: Εμφανίζεται ένας χώρος όπου ο χρήστης εξετάζει το περιεχόμενο και αποφασίζει εάν θέλει να το κρατήσει ή όχι. Εάν το αποτέλεσμα δεν αρμόζει στον τελικό χρήστη, τότε έχει την επιλογή να το ξανά δοκιμάσει.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη έτσι ώστε ο χρήστης να έχει την εποπτεία του αποτελέσματος του LLM.

3.5.3.1.3. Θα πρέπει να γίνεται σύνδεση με το LLM και να εξάγεται η απαραίτητη έξοδος.

Περιγραφή: Ο κώδικας συνδέεται με το LLM για να μπορέσει να δέχεται τις προτροπές του χρήστη. Στη συνέχεια, το αποτέλεσμα αυτό επεξεργάζεται και δημιουργείται η τελική, επιθυμητή δομή.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη για να λειτουργήσει σωστά το LLM και να εξάγει τα επιθυμητά αποτελέσματα.

3.5.3.1.4. Θα πρέπει ο τελικός χρήστης να έχει τη δυνατότητα να δώσει αρχείο εγγραφών με τις διευθύνσεις των παραληπτών και η εφαρμογή να ελέγχει τη δομή του.

Περιγραφή: Ρωτάτε ο χρήστης να δώσει το αρχείο εγγράφων μέσω ενός μηνύματος στη κονσόλα της εφαρμογής. Αφού δοθεί, τότε περνάει από έλεγχο της δομής του. Εάν η δομή δεν ακολουθεί συγκεκριμένες συνθήκες, εγείρεται σφάλμα και ζητείται να ανέβει ξανά.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη για να προσπελαστούν οι διευθύνσεις που θα σταλούν τα ηλεκτρονικά μηνύματα. Ο έλεγχος δομής είναι εξίσου απαραίτητος επειδή άμα η δομή είναι λανθασμένη, τότε δε θα προσπελαστούν σωστά οι τιμές του αρχείου.

3.5.3.1.5. Θα πρέπει να γίνεται σύνδεση με έναν SMTP server για την αποστολή των ηλεκτρονικών μηνυμάτων.

Περιγραφή: Χρειάζεται η σύνδεση με έναν SMTP server έτσι ώστε να μπορέσουν να σταλούν επιτυχώς τα μηνύματα της εφαρμογής. Ο SMTP server ακούσει σε μια θύρα Α.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη για να σταλούν επιτυχώς να μηνύματα ηλεκτρονικού ταχυδρομείου.

3.5.3.1.6. Θα πρέπει ο τελικός χρήστης να μπορεί να επιλέξει τον τρόπο εμφάνισης του ηλεκτρονικού μηνύματος.

Περιγραφή: Δίνεται η επιλογή στον χρήστη να αποστείλει το μήνυμα με τον τρόπο Α ή τον τρόπο Β. Έτσι, ο χρήστης μπορεί να διαμορφώσει το μήνυμα στις προτιμήσεις του.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη έτσι ώστε ο τελικός χρήστης να έχει τη δυνατότητα να τροποποιήσει τη μορφή του μηνύματος με βάση τις επιθυμίες του.

3.5.3.1.7. Θα πρέπει ο τελικός χρήστης να ενημερώνεται για τους παραλήπτες που λάβανε το μήνυμα.

Περιγραφή: Απεικονίζεται ένα μήνυμα στη κονσόλα της εφαρμογής έτσι ώστε ενημερωθεί ο χρήστης για τις αποστολές των μηνυμάτων.

Αιτιολόγηση: Η λειτουργία αυτή είναι απαραίτητη για να γνωρίζει ο τελικός χρήστης σε ποιους και πόσα μηνύματα έχουν σταλεί.

3.5.3.2 Μη λειτουργικές Απαιτήσεις

3.5.3.2.1. Η εφαρμογή θα πρέπει να είναι εύκολη στη χρήση

Περιγραφή: Η δομή της εφαρμογής θα πρέπει να είναι απλή και κατανοητή. Δίχως σύνθετες επιλογές ή κανόνες που θα πρέπει να ακολουθούνται κατά τη χρήση της.

Αιτιολόγηση: Η εφαρμογή θα πρέπει να είναι εύκολα διαχειρίσιμη.

3.5.3.2.2. Η εφαρμογή θα πρέπει να παρέχει ασφάλεια στον χρήστη

Περιγραφή: Ο κωδικός του λογαριασμού του χρήστη είναι προστατευμένος και δεν μπορεί να εκτεθεί σε τρίτους. Επιπλέον, το LLM τρέχει τοπικά, δίνοντας έτσι πλήρη έλεγχο των ευαίσθητων πληροφοριών που μπορεί να χρειαστούν για τη προτροπή.

Αιτιολόγηση: Η εφαρμογή θα πρέπει να προστατεύει τον αποστολέα και τους παραλήπτες.

3.5.3.2.3. Το λογισμικό θα πρέπει να εγκατασταθεί σε υπολογιστή γραφείου

Περιγραφή: Το λογισμικό μπορεί να κατέβει και να τρέξει σε εταιρικό υπολογιστή γραφείου.

Αιτιολόγηση: Το λογισμικό απαιτεί μια συσκευή για να μπορεί να λειτουργήσει επιτυχώς.

3.5.3.2.4. Η ελάχιστη απαιτούμενη μνήμη RAM του Η/Υ θα πρέπει να είναι μεγαλύτερη ή ίση με 4 GB.

Περιγραφή: Το λογισμικό απαιτεί την διαχείριση ενός τοπικού LLM μοντέλου, το οποίο χρειάζεται 4GB μνήμη ή παραπάνω για να λειτουργήσει αποδοτικά.

Αιτιολόγηση: Η μνήμη παίζει θεμελιώδη ρόλο στα LLM, μιας και χρειάζεται για να τρέξει επιτυχώς.

3.5.3.2.5. Θα πρέπει η εφαρμογή να είναι έτσι δομημένη ώστε να μπορούν μελλοντικά να γίνουν αλλαγές (συντηρησιμότητα)

Περιγραφή: Ο κώδικας που έχει υλοποιηθεί θα πρέπει να είναι ευανάγνωστος και τεκτημιωμένος με σχόλια, χωρίς περιττές πληροφορίες που θα προκαλέσουν σύγχυση σε κάποιον τρίτο που θα εξετάσει τον κώδικα.

Αιτιολόγηση: Η συντηρησιμότητα είναι σημαντική έτσι ώστε ο κώδικας να μπορέσει να διατηρηθεί καθαρός, με σκοπό μελλοντικοί προγραμματιστές να έχουν τη δυνατότητα να τον επεξεργαστούν χωρίς δυσκολία.

3.5.3.3 Προδιαγραφές

Πίνακας 13: 3.5.3.3.1 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει ο τελικός χρήστη να μπορεί να δώσει τα διαπιστευτήρια του λογαριασμού για το ηλεκτρονικό ταχυδρομείο του.
Περιγραφή	Ο χώρος εισαγωγής των στοιχείων αυτών φαίνονται στη κονσόλα της εφαρμογής. Ο τελικός χρήστης μπορεί να προσθέσει τα στοιχεία του.
Δεδομένα Εισόδου	Διεύθυνση ηλεκτρονικού ταχυδρομείου και κωδικός App Password.
Προέλευση	Παράθυρο κονσόλας.
Δεδομένα Εξόδου	Κανένα.
Προορισμός	Παράθυρο κονσόλας.
Ενέργεια	Ο χρήστης δίνει την διεύθυνση ηλεκτρονικού ταχυδρομείου και κωδικός πρόσβασης.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Επικυρωμένα διαπιστευτήρια χρήστη.

Αποτέλεσμα ή Μετασυνθήκη	Αποθήκευση των στοιχείων για την σύνδεση με τον SMTP server.
Πλευρικά Φαινόμενα ή Παρενέργειες	Σε περίπτωση που τα δεδομένα δεν είναι σωστά, δε θα σταλούν τα μηνύματα στους παραλήπτες.

Πίνακας 14: 3.5.3.3.2 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει ο τελικός χρήστης να έχει τη δυνατότητα να εξετάσει το περιεχόμενο του παραγόμενου αποτελέσματος του LLM.
Περιγραφή	Εμφανίζεται ένας χώρος όπου ο χρήστης εξετάζει το περιεχόμενο και αποφασίζει εάν θέλει να το κρατήσει ή όχι. Εάν το αποτέλεσμα δεν αρμόζει στον τελικό χρήστη, τότε έχει την επιλογή να το ξανά δοκιμάσει.
Δεδομένα Εισόδου	Απάντηση “ναι” (yes/y) ή “όχι” (no/n) για διατήρηση ή όχι του αποτελέσματος.
Προέλευση	Παράθυρο κονσόλας.
Δεδομένα Εξόδου	Αποθήκευση ή επανάληψη δημιουργίας του αποτελέσματος.
Προορισμός	Παράθυρο κονσόλας.
Ενέργεια	Ο χρήστης απαντάει θετικά ή αρνητικά, ανάλογα με το άμα επιθυμεί να κρατήσει το περιεχόμενο ή όχι. Εάν δε του αρμόζει, τότε δημιουργείται νέο αποτέλεσμα.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Να είναι ανοιχτό το Ollama.
Αποτέλεσμα ή Μετασυνθήκη	Επιθυμητό παραγόμενο αποτέλεσμα από το LLM.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα.

Πίνακας 15: 3.5.3.3.3 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει να γίνεται σύνδεση με το LLM και να εξάγεται η απαραίτητη έξοδος.
Περιγραφή	Ο κώδικας συνδέεται με το LLM για να μπορέσει να δέχεται τις προτροπές του

	χρήστη. Στη συνέχεια, το αποτέλεσμα αυτό επεξεργάζεται και δημιουργείται η τελική, επιθυμητή δομή.
Δεδομένα Εισόδου	Προτροπή LLM.
Προέλευση	Παράθυρο κονσόλας.
Δεδομένα Εξόδου	Αποτέλεσμα LLM.
Προορισμός	Παράθυρο κονσόλας.
Ενέργεια	Ο χρήστης δίνει τη προτροπή στο LLM και με τη σειρά του το LLM δίνει το αποτέλεσμα, το οποίο περνάει από έλεγχο για τη δομή του.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος και να έχει δώσει ο χρήστης τη προτροπή.
Προϋπόθεση ή Προσυνθήκη	Να είναι ανοιχτό το Ollama.
Αποτέλεσμα ή Μετασυνθήκη	Σωστά δομημένο αποτέλεσμα από το LLM.
Πλευρικά Φαινόμενα ή Παρενέργειες	Σφάλμα μη δυνατότητας επεξεργασίας της δομής του μηνύματος.

Πίνακας 16: 3.5.3.3.4 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει ο τελικός χρήστης να έχει τη δυνατότητα να δώσει αρχείο εγγραφών με τις διευθύνσεις των παραληπτών και η εφαρμογή να ελέγχει τη δομή του.
Περιγραφή	Ρωτάτε ο χρήστης να δώσει το αρχείο εγγράφων μέσω ενός μηνύματος στη κονσόλα της εφαρμογής. Αφού δοθεί, τότε περνάει από έλεγχο της δομής του. Εάν η δομή δεν ακολουθεί συγκεκριμένες συνθήκες, εγείρεται σφάλμα και ζητείται να ανέβει ξανά.
Δεδομένα Εισόδου	Μονοπάτι αρχείου.
Προέλευση	Παράθυρο κονσόλας.
Δεδομένα Εξόδου	Κανένα.
Προορισμός	Πουθενά.
Ενέργεια	Ο χρήστης δίνει το μονοπάτι του αρχείου με τα ονόματα και τις διευθύνσεις των παραληπτών, έτσι ώστε να αποθηκευτούν

	και να αξιοποιηθούν για την αποστολή ηλεκτρονικών μηνυμάτων ταχυδρομείου.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Να είναι ανοιχτός ο υπολογιστής.
Αποτέλεσμα ή Μετασυνθήκη	Επιτυχής προσπέλαση του αρχείου.
Πλευρικά Φαινόμενα ή Παρενέργειες	Σφάλμα λανθασμένης δομής αρχείου.

Πίνακας 17: 3.5.3.3.5 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει να γίνεται σύνδεση με έναν SMTP server για την αποστολή των ηλεκτρονικών μηνυμάτων.
Περιγραφή	Χρειάζεται η σύνδεση με έναν SMTP server έτσι ώστε να μπορέσουν να σταλούν επιτυχώς τα μηνύματα της εφαρμογής. Ο SMTP server ακούσει σε μια θύρα Α.
Δεδομένα Εισόδου	SMTP Server και θύρα.
Προέλευση	Κώδικας.
Δεδομένα Εξόδου	Κανένα.
Προορισμός	Πουθενά.
Ενέργεια	Ο προγραμματιστής ορίζει τον SMTP server όπως επίσης και τη θύρα. Στο συγκεκριμένο έργο ο server είναι αυτός της Google που “ακούει” στη θύρα 587.
Απαίτηση	Να έχει πρόσβαση στον κώδικα του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Να έχει το αρχείο python του προγράμματος.
Αποτέλεσμα ή Μετασυνθήκη	Επιτυχής σύνδεση με SMTP server της Google.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα.

Πίνακας 18: 3.5.3.3.6 Προδιαγραφή

Υπηρεσία ή Λειτουργία	Θα πρέπει ο τελικός χρήστης να μπορεί να επιλέξει τον τρόπο εμφάνισης του ηλεκτρονικού μηνύματος.
Περιγραφή	Δίνεται η επιλογή στον χρήστη να αποστείλει το μήνυμα με τον τρόπο Α ή τον τρόπο Β. Έτσι, ο χρήστης μπορεί να διαμορφώσει το μήνυμα στις προτιμήσεις του.
Δεδομένα Εισόδου	Απάντηση “ναι” (yes/y) ή “όχι” (no/n) για την επιλογή του τρόπου Α από ότι του Β.
Προέλευση	Παράθυρο κονσόλας.
Δεδομένα Εξόδου	Κανένα.
Προορισμός	Μήνυμα ηλεκτρονικού ταχυδρομείου.
Ενέργεια	Ο χρήστης μπορεί να επιλέξει εάν επιθυμεί το μήνυμα να σταλεί με πρότυπο HTML ή όχι. Εάν επιλέξει όχι, τότε το μήνυμα στέλνεται ως απλό κείμενο.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Να είναι ανοιχτός ο υπολογιστής.
Αποτέλεσμα ή Μετασυνθήκη	Αποστολή το μηνύματος είτε με HTML είτε χωρίς.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα.

Πίνακας 19: 3.5.3.3.7 Προδιαγραφή

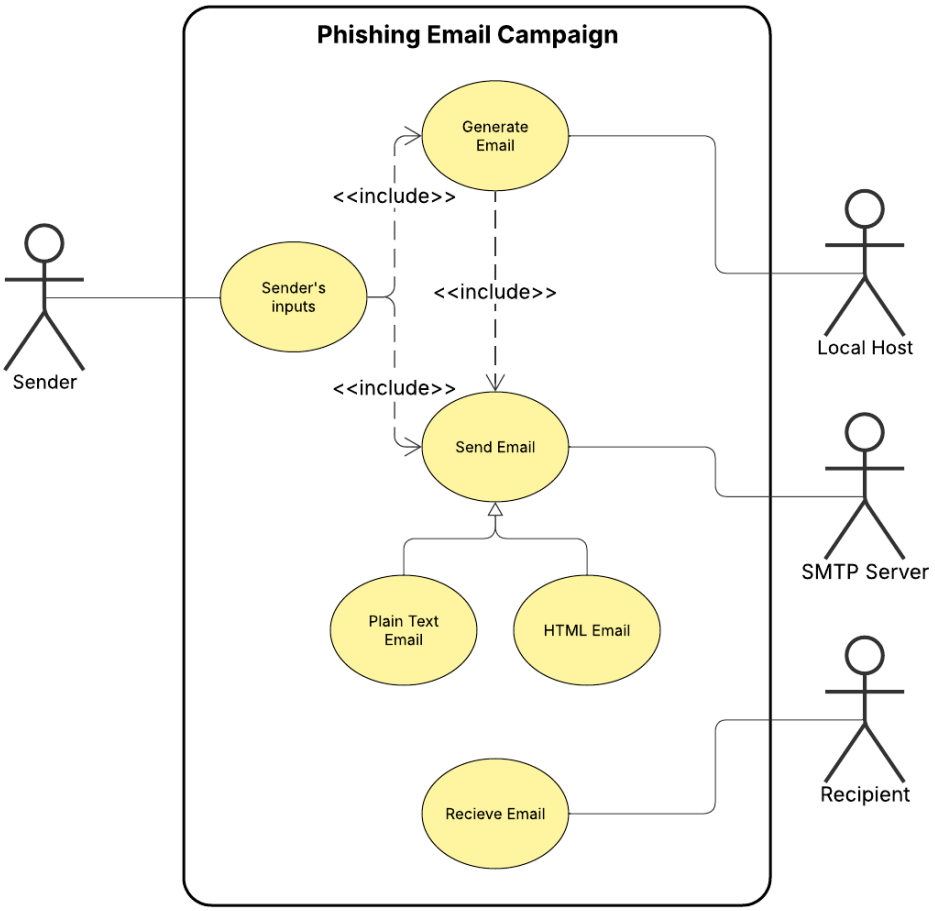
Υπηρεσία ή Λειτουργία	Θα πρέπει ο τελικός χρήστης να ενημερώνεται για τους παραλήπτες που λάβανε το μήνυμα.
Περιγραφή	Απεικονίζεται ένα μήνυμα στη κονσόλα της εφαρμογής έτσι ώστε ενημερωθεί ο χρήστης για τις αποστολές των μηνυμάτων.
Δεδομένα Εισόδου	Κανένα.
Προέλευση	Πουθενά.
Δεδομένα Εξόδου	Ενημερωτικό μήνυμα που αναγράφει τους παραλήπτες.

Προορισμός	Παράθυρο κονσόλας.
Ενέργεια	Εμφανίζονται στον χρήστη μηνύματα που τον ενημερώνουν για το ποιοι παραλήπτες έλαβαν το ηλεκτρονικό μήνυμα.
Απαίτηση	Να έχει ξεκινήσει η εκκίνηση του προγράμματος.
Προϋπόθεση ή Προσυνθήκη	Να έχει δοθεί αρχείο με τις παραλήπτες.
Αποτέλεσμα ή Μετασυνθήκη	Αποστολή του μηνύματος στους αντίστοιχος παραλήπτες.
Πλευρικά Φαινόμενα ή Παρενέργειες	Κανένα.

3.5.4 Διαγράμματα UML

3.5.4.1 Διάγραμμα Περιπτώσεων Χρήσης

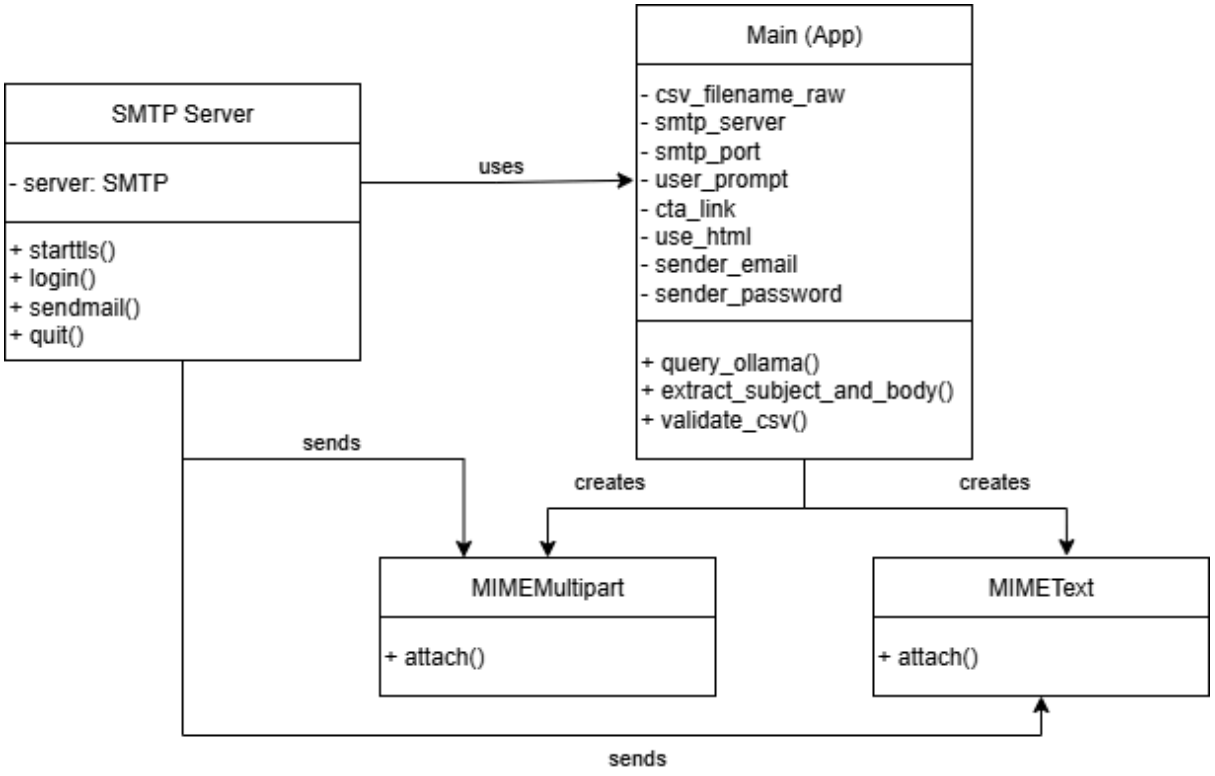
Παρακάτω ακολουθεί το διάγραμμα περιπτώσεων χρήσης της δεύτερης μεθόδου (Σχήμα 12).



Σχήμα 12: Διάγραμμα Περιπτώσεων Χρήσης - Τοπικό Περιβάλλον

3.5.4.2 Διάγραμμα Κλάσεων

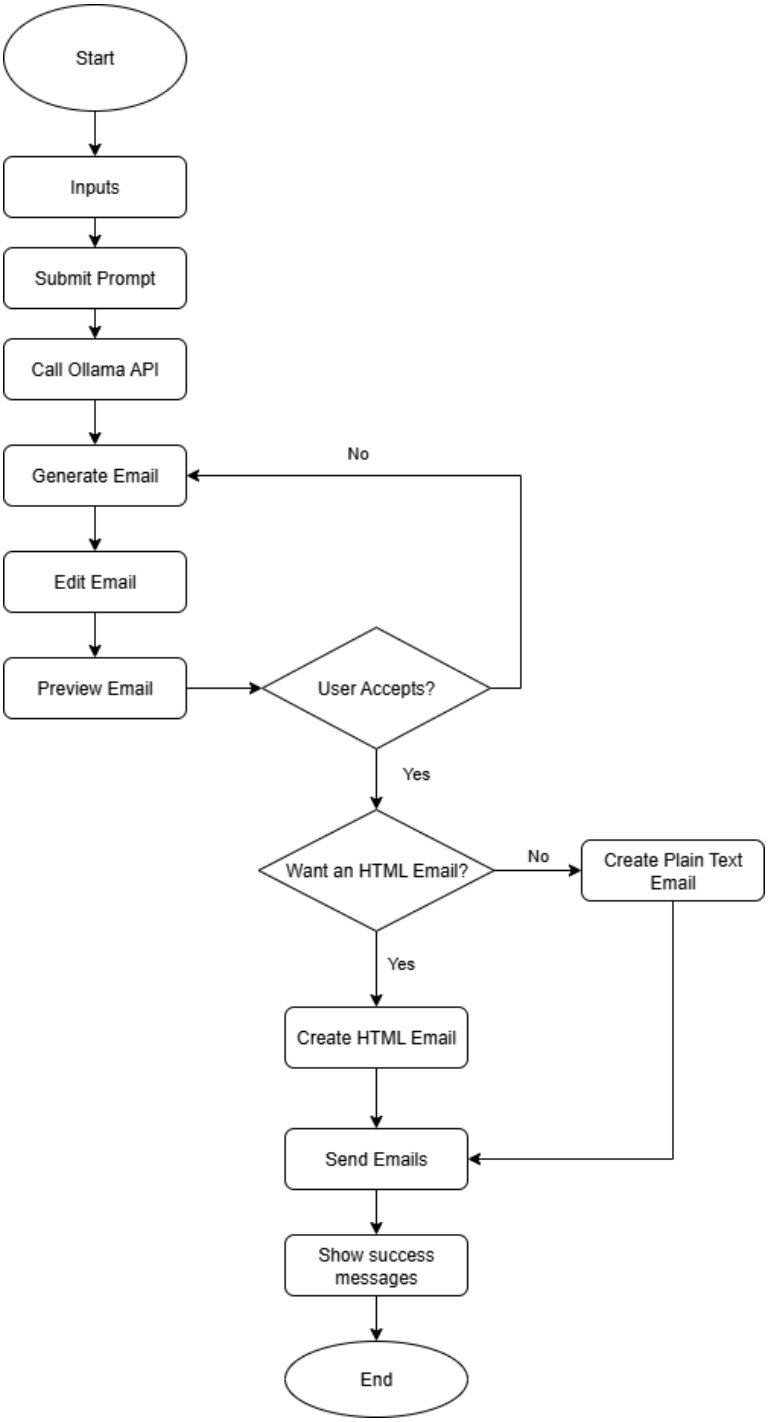
Παρακάτω ακολουθεί το διάγραμμα κλάσεων της δεύτερης μεθόδου (Σχήμα 13).



Σχήμα 13: Διάγραμμα Κλάσεων - Τοπικό Περιβάλλον

3.5.4.2 Διάγραμμα Κατάστασης

Παρακάτω ακολουθεί το διάγραμμα κατάστασης της δεύτερης μεθόδου (Σχήμα 14).



Σχήμα 14: Διάγραμμα Κατάστασης - Τοπικό Περιβάλλον

3.5.5 Τρόπος λειτουργίας με παραδείγματα

Για την εκτέλεση της εφαρμογής, είναι απαραίτητη η εγκατάσταση του Ollama [139], μιας και το LLM τρέχει τοπικά. Αφού εγκατασταθεί το Ollama στη συσκευή, τότε μέσω του command prompt πρέπει να τρέξει η εντολή “ollama run hf.co/gawyria/pers-com-email-model:Q8_0”, η οποία τραβάει, κατεβάζει και τρέχει το LLM, που έχει δημιουργηθεί μέσω του Unsloth, στη συσκευή για να είναι έτοιμο για χρήση. Η

εντολή αυτή χρειάζεται να τρέξει μόνο μια φορά, έτσι ώστε να κατέβει το μοντέλο στη τοπική συσκευή. Το LLM είναι ανεβασμένο στη πλατφόρμα του Hugging Face [145] και χρησιμοποιείται η μέθοδος quantization Q8_0, με μέγεθος 8.5GB, αφού εμφάνισε τα καλύτερα αποτελέσματα συγκριτικά με τις μεθόδους Q4_K_M και Q5_K_M.

Πριν εκτελεστεί ο κώδικας και αφού είναι κατεβασμένο το μοντέλο, θα πρέπει να τρέχει από πίσω η εφαρμογή του Ollama, καθώς είναι απαραίτητη για την παραγωγή των ηλεκτρονικών μηνυμάτων. Έπειτα, ο χρήστης είναι έτοιμος να τρέξει το κώδικα του προγράμματος. Κατά την έναρξη, εμφανίζονται στον χρήστη τα πεδία που οφείλει να απαντήσει έτσι ώστε να ολοκληρωθεί η διαδικασία (Εικόνα 9).

```
Enter your sender email address:
Enter your email password or App Password (input hidden):
Enter the path of your CSV file:
Enter the prompt for the email:
```

Εικόνα 10: Πεδία εισαγωγής στοιχείων

Τα πεδία εμφανίζονται σταδιακά και ελέγχονται κάθε φορά για την εγκυρότητά τους. Στο τέλος, ο χρήστης θα προταθεί να δώσει τη προτροπή για το μοντέλο. Το πρόγραμμα θα παράξει και θα εμφανίσει το τελικό αποτέλεσμα στον χρήστη, δίνοντάς του τη δυνατότητα να απορρίψει το αποτέλεσμα ή να το κρατήσει. Άμα το απορρίψει, τότε εμφανίζεται επεξηγηματικό μήνυμα ενημερώνοντας τον χρήστη ότι γίνεται νέα παραγωγή αποτελέσματος (Εικόνα 10), ενώ εάν το κρατήσει τότε το πρόγραμμα συνεχίζει την εκτέλεσή του (Εικόνα 11).

```
Enter the prompt for the email: summer sales

--- Generated Email ---
Subject: Summer Savings are Here!
Body:
[Name],
Our summer sale is now on! Get ready to update your wardrobe with huge discounts.

Shop now here: [Insert Call-to-Action button or link]
------------------------

Do you want to send this email? (yes/no): no
Regenerating...
```

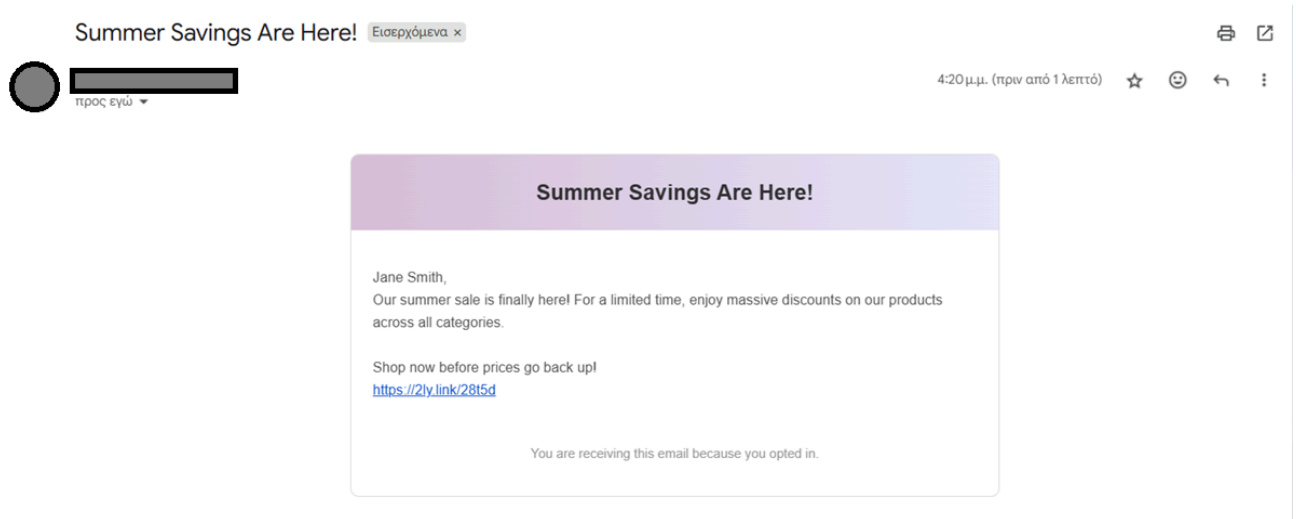
Εικόνα 11: Επεξηγηματικό μήνυμα προς τον χρήστη

```
Enter the prompt for the email: summer sales

--- Generated Email ---
Subject: Summer Savings are Here!
Body:
[Name],
Our summer sale is now on! Get ready to update your wardrobe with huge discounts.

Shop now here: [Insert Call-to-Action button or link]
-----------------------

Do you want to send this email? (yes/no): y
Enter the Call-to-Action link you want to insert:
Do you want to send the email as HTML? (yes/no):
Email sent to Jane Smith ([redacted]@gmail.com)
All emails sent successfully!
```

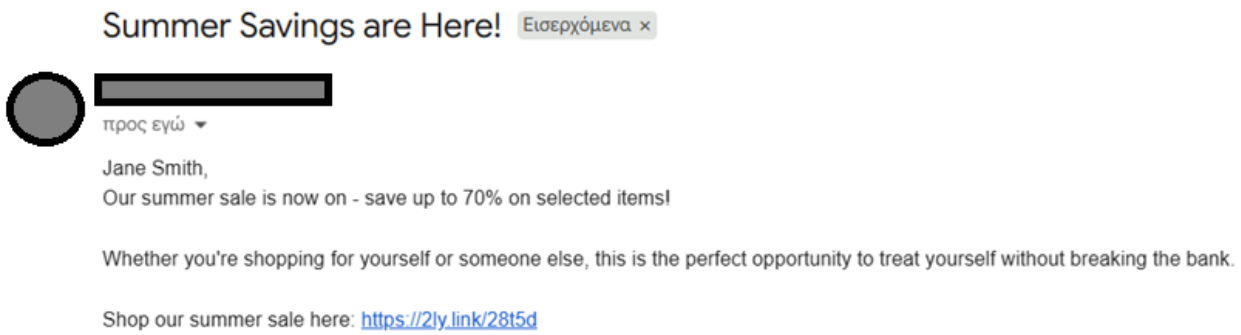
Εικόνα 12: Εκτέλεση και αποστολή του μηνύματος

Στη περίπτωση που ο χρήστης είναι ικανοποιημένος με το αποτέλεσμα και έχει επιλέξει να συνεχίσει, του ζητείται να προσθέσει τον επιθυμητό σύνδεσμο ηλεκτρονικού ψαρέματος. Ιδανικά, αυτός ο σύνδεσμος θα πρέπει να είναι ανιχνεύσιμος, προκειμένου ο τελικός χρήστης και αποστολέας να έχει τη δυνατότητα να παρακολουθεί τη δραστηριότητα του συνδέσμου μέσω στατιστικών, όπως ο αριθμός των ατόμων που ακολούθησαν τον σύνδεσμο καθώς και το χρονικό διάστημα κατά το οποίο πραγματοποιήθηκαν οι σχετικές ενέργειες. Έπειτα, ο χρήστης μπορεί να επιλέξει εάν προτιμάει το μήνυμα να σταλεί ως απλό κείμενο ή μέσω HTML. Ανεξαρτήτως την απάντησή του, στο τέλος ενημερώνεται για τα ηλεκτρονικά μηνύματα ταχυδρομείου που έχουν σταλεί.

Παρακάτω φαίνεται πως είναι η δομή εάν σταλεί με HTML (Εικόνα 12) ή με απλό κείμενο (Εικόνα 13).



Εικόνα 13: Παράδειγμα προτύπου HTML



Εικόνα 14: Παράδειγμα απλού κειμένου

Παράδειγμα:

Στοιχεία εισαγωγής:
Enter your sender email address: example@gmail.com
Enter your App Password (hidden input):
Enter the path of the CSV file: "C:\Users\desktop\contacts.csv"
Enter the prompt for the email: summer sales

--- Generated Email ---

Subject: Summer Savings are Here!

Body:

[Name],

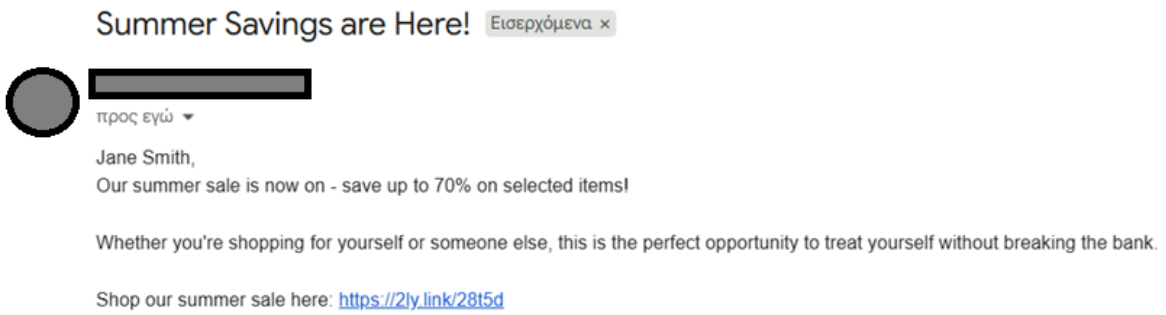
Our summer sale is now on - save up to 70% on selected items!

Whether you're shopping for yourself or someone else, this is the perfect opportunity to treat yourself without breaking the bank.

Shop our summer sale here: [Insert Call-to-Action button or link]

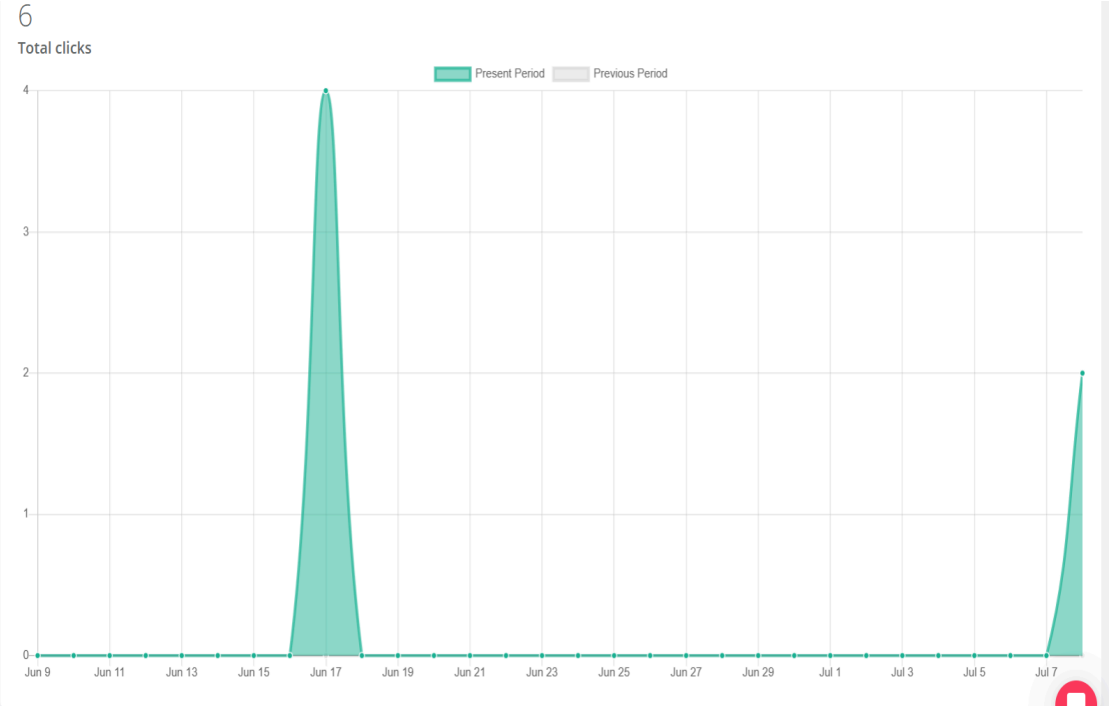
Do you want to send this email? (yes/no): yes
Enter the Call-to-Action link you want to insert: https://2ly.link/28t5d
Do you want to send the email as HTML? (yes/no): no
Email sent to Jane Smith (example2@gmail.com)
All emails sent successfully!

Το παραγόμενο αποτέλεσμα (Εικόνα 14):



Εικόνα 15: Παράδειγμα παραγόμενου αποτελέσματος

Ο συγκεκριμένος σύνδεσμος ηλεκτρονικού ψαρέματος ανακατευθύνει τον χρήστη στην επίσημη σελίδα της Cisco [146], όπου δίνει σημαντικές πληροφορίες για το Phishing, πως μπορεί κάποιος να αναγνωρίσει μια επίθεση phishing και μέτρα που μπορεί να πάρει κατά αυτών των επιθέσεων. Ο σύνδεσμος δημιουργήθηκε με τη βοήθειας μιας ιστοσελίδας συντομεύσεων συνδέσμων, την Linkly [147], η οποία παρέχει και τα απαραίτητα στατιστικά δεδομένα (Εικόνα 15).



Εικόνα 16: Στατιστικά από το Linkly

3.5.6 Παρουσίαση μικρών χαρακτηριστικών τμημάτων κώδικα και επεξήγηση της λειτουργίας τους

3.5.6.1 Βιβλιοθήκες και ρύθμιση του LLM

Ο κώδικας ξεκινά με την εισαγωγή βασικών βιβλιοθηκών, όπως τη smtplib, email.mime.multipart και email.mime.text που αφορούν την αποστολή των email, τη

βιβλιοθήκη csv για τη διαχείριση csv αρχείων, τη re για την διαχείριση των regular expressions, τη requests για την αποστολή HTTP POST και τη getpass που διατηρεί κρυφούς τους κωδικούς. Στη συνέχεια, γίνεται η σύνδεση του τοπικού LLM με τον κώδικα, μέσω της def query_ollama(prompt) και παίρνει την έξοδό του.

Ο κώδικας:

```
import smtplib
import csv
import re
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
import requests
import getpass

def query_ollama(prompt):
      url = "http://localhost:11434/api/generate"
      payload = {
      "model": "hf.co/gawyria/pers-com-email-model:Q8_0",
      "prompt": prompt,
      "stream": False
      }

      response = requests.post(url, json=payload)
      data = response.json()


      # Extract the response based on the actual structure
      if "response" in data:
      return data["response"]
      else:
      # If the response is the entire text
      return data.get("text", data.get("output", str(data)))
```

3.5.6.2 Διαπιστευτήρια και αρχικοποίηση SMTP Server

Ζητείται από τον τελικό χρήστη να προσθέσει την ηλεκτρονική του διεύθυνση, όπως επίσης και το App Password της Google, έτσι ώστε να γίνει η επιτυχής σύνδεση με τον Server. Η σύνδεση γίνεται με τον Server της Google (smtp.gmail.com) που “ακούει” στην θύρα 587.

Ο κώδικας:

```
# Email credentials

sender_email = input("Enter your sender email address: ").strip()

sender_password = getpass.getpass("Enter your App Password (input hidden): ").strip()


# SMTP Server

smtp_server = "smtp.gmail.com"

smtp_port = 587
```

3.5.6.3 Έλεγχος εισόδων

Η συνάρτηση extract_subject_and_body εξάγει το σώμα και το θέμα του ηλεκτρονικού μηνύματος, χρησιμοποιώντας κανονικές εκφράσεις. Επίσης, ελέγχει αν το σώμα περιέχει το υποχρεωτικό placeholder [Insert Call-to-Action button or link] και επιστρέφει το θέμα και το σώμα ως δύο ξεχωριστές τιμές. Εάν κάποιο από τα απαιτούμενα στοιχεία λείπει, τότε εγείρεται σφάλμα.

Ζητείται από τον χρήστη να εισάγει το μονοπάτι για το CSV αρχείο, το οποίο περιέχει τα ονόματα και τις διευθύνσεις ηλεκτρονικού ταχυδρομείου. Έπειτα, το αρχείο αυτό περνάει από επαλήθευση και στέλνει τα απαραίτητα σφάλματα, σε περίπτωση που δεν τηρούνται συγκεκριμένες συνθήκες. Αναλυτικότερα, με τη συνάρτηση validate_csv ανοίγει το αρχείο, επαληθεύει ότι η πρώτη γραμμή έχει τα σωστά headers (name,email), και στη συνέχεια ελέγχει κάθε γραμμή για σωστό αριθμό στηλών, μη κενές τιμές και έγκυρη μορφή email μέσω της κανονικής έκφρασης email_regex. Αν είναι όλα σωστά, τότε επιστρέφεται η λίστα με τα ονόματα και τις διευθύνσεις, διαφορετικά εγείρεται επεξηγηματικό σφάλμα.

Ο κώδικας:

```
# Extracts the body and subject from the output

def extract_subject_and_body(text):

    subject_match = re.search(r"Subject:\s*(.+)", text)

    body_match = re.search(r"Body:\s*(.+)", text, re.DOTALL)
```

```
    if not subject_match:
    raise ValueError("Couldn't extract the subject from the text.")

    if not body_match:
    raise ValueError("Couldn't extract the body from the text.")

    subject = subject_match.group(1).strip()
    body = body_match.group(1).strip()

    if "[Insert Call-to-Action button or link]" not in body:
    raise ValueError("Missing required placeholder: [Insert Call-to-Action button or link] in the body.")

    return subject, body

# Ask user for CSV file path and remove the symbol (") from the start and the end of the path
csv_filename_raw = input("Enter the path of your CSV file: ").strip()
csv_filename = csv_filename_raw[1:-1]

# Email validation regex
email_regex = r"^[a-zA-Z0-9._]+@[a-zA-Z0-9]+\.[a-zA-Z]{2,}$"

def validate_csv(csv_filename):
    try:
    with open(csv_filename, "r", newline="", encoding="utf-8") as csvfile:
        reader = csv.reader(csvfile)
        headers = next(reader, None)  # Read the first row as headers

        # Check if headers match the required format
        if headers != ["name", "email"]:
```

```
                raise ValueError("Invalid CSV format. Expected headers:
'name,email'.")

            # Read and validate each row
            contacts = []
            for row in reader:
                if len(row) != 2:
                raise ValueError(f"Invalid row: {row}. Each row must
have exactly 2 columns (name, email).")

                name, email = row[0].strip(), row[1].strip()

                if not name or not email:
                raise ValueError(f"Missing data in row: {row}. Name and
email cannot be empty.")

                if not re.match(email_regex, email):
                raise ValueError(f"Invalid email format: {email} in row:
{row}")

                contacts.append((name, email))

            return contacts

      except Exception as e:
      print(f"CSV Validation Error: {e}")
      return None  # Return None if validation fails
```

3.5.6.4 Αποστολή του παραγόμενου μηνύματος ηλεκτρονικού ταχυδρομείου

Για την αποστολή των μηνυμάτων, αρχικά γίνονται οι απαραίτητοι έλεγχοι και η σύνδεση με τον SMTP Server. Στη συνέχεια, ζητείται από τον χρήστη να δώσει τη προτροπή για το LLM και με τη σειρά του, παράγει το κατάλληλο αποτέλεσμα. Το αποτέλεσμα εμφανίζεται στη κονσόλα και ο χρήστης έχει την επιλογή να το απορρίψει, άμα το αποτέλεσμα δεν είναι αυτό που επιθυμεί, ή να συνεχίσει με αυτό και για να σταλεί σε όλους τους παραλήπτες. Παράλληλα, ο χρήστης μπορεί να επιλέξει εάν το μήνυμα θα σταλεί με HTML ή ως απλό κείμενο, όπως επίσης και να εισάγει τον δικό του σύνδεσμο

που προσθέτεται μέσα στο σώμα του μηνύματος. Τέλος, όλοι οι παραλήπτες λαμβάνουν το μήνυμα και εμφανίζεται ένα ενημερωτικό σημείωμα στον τελικό χρήστη για την επιτυχή αποστολή τους.

Ο κώδικας:

```
try:
      # Validate CSV file
      contacts = validate_csv(csv_filename)
      if contacts is None:
      raise Exception("Fix the CSV file and try again.")

      # Start SMTP session
      server = smtplib.SMTP(smtp_server, smtp_port)
      server.starttls()  # Secure connection
      server.login(sender_email, sender_password)  # Login

      # Ask user for prompt
      user_prompt = input("Enter the prompt for the email: ").strip()
      confirmed = False

      while not confirmed:
      # Generate email with LLM
      llm_output = query_ollama(user_prompt)

      try:
            # Extract subject and body
            subject, body_template = extract_subject_and_body(llm_output)

            # Show result to user
            print("\n--- Generated Email ---")
            print(f"Subject: {subject}")
            print(f"Body:\n{body_template}")
```

```
            print("------------------------\n")

            # Ask for confirmation
            user_input = input("Do you want to send this email? (yes/no): ").strip().lower()
            if user_input in ['yes', 'y']:
                confirmed = True
            else:
                print("Regenerating...\n")
      except Exception as e:
            print(f"Failed to extract subject/body: {e}")
            print("Trying again...\n")

      # Ask for Call-to-Action link
      cta_link = input("Enter the Call-to-Action link you want to insert: ").strip()

      # Ask if the user wants HTML formatting
      use_html = input("Do you want to send the email as HTML? (yes/no): ").strip().lower()
      send_as_html = use_html in ['yes', 'y']

      for name, recipient_email in contacts:

      # Create a new email message for each recipient
      msg = MIMEMultipart("alternative")
      msg['From'] = sender_email
      msg['To'] = recipient_email
      msg['Subject'] = subject

      # Personalize the body
      personalized_plain = body_template.replace("[Name]", name).replace("[Insert Call-to-Action button or link]", cta_link)
```

```
        # HTML version
        html_body = personalized_plain.replace('\n', '<br>')
        with open("email_template.html", "r", encoding="utf-8") as file:
            html_template = file.read()

        personalized_html = html_template.format(subject=subject,
html_body=html_body)

        # Attach only the one the user chose
        if send_as_html:
            msg.attach(MIMEText(personalized_html, 'html'))
        else:
            msg.attach(MIMEText(personalized_plain, 'plain'))

        # Send the email
        server.sendmail(sender_email, recipient_email, msg.as_string())
        print(f"Email sent to {name} ({recipient_email})")

        # Close connection
        server.quit()
        print("All emails sent successfully!")

except Exception as e:
        print(f"Error: {e}")
```

3.5.7 Περιγραφή παραθύρων εφαρμογής, παρουσίαση εκτυπώσεων, πίνακες με καταστάσεις λειτουργίας

Η εφαρμογή χωρίζεται σε τρία κύρια λειτουργικά παράθυρα (καταστάσεις εφαρμογής), το καθένα με τη δική του συμπεριφορά, εισαγωγή/έξοδο δεδομένων και σκοπό.

3.5.7.1 Περιγραφή παραθύρων / διεπαφών εφαρμογής

3.5.7.1.1 Παράθυρα εφαρμογών και καταστάσεις λειτουργίας

Λειτουργία:

- Ζητείται από τον χρήστη να εισάγει βασικές λεπτομέρειες:
  1. Τα διαπιστευτήρια του αποστολέα
  2. Αρχείο τύπου .csv με τα ονόματα και τις διευθύνσεις των παραληπτών
  3. Προτροπή για την περιγραφή του σεναρίου ηλεκτρονικού "ψαρέματος"
  4. Το phishing link
- Αυτές οι τιμές αποστέλλονται στο τελικό σημείο το οποίο:
  1. Στέλνει την προτροπή στο LLM που τρέχει τοπικά.
  2. Αναλύει και καθαρίζει την έξοδο του μοντέλου για να διαχωρίσει το σώμα από το θέμα.
  3. Εξατομικεύει το σώμα με το όνομα του παραλήπτη και τον σύνδεσμο της επιλογής του αποστολέα.

Έξοδος εκτύπωσης:
--- Generated Email ---

Subject: [Το θέμα που δημιούργησε το LLM]

Body:

[Προσωποποιημένο σώμα που δημιούργησε το LLM]

Με αυτόν τον τρόπο δίνεται η δυνατότητα στον χρήστη να εξετάσει το παραγόμενο περιεχόμενο πριν την αποστολή του, κάτι ιδιαίτερα κρίσιμο λόγω της ευαίσθητης φύσης του περιεχομένου.

3.5.7.1.2 Παράθυρο Επιβεβαίωσης και Επιλογών

Λειτουργία:

- Ρωτάτε στον χρήστη:
  1. Εάν θα προχωρήσει στην αποστολή του email.
  2. Να καταχωρήσει τον σύνδεσμο ηλεκτρονικού ψαρέματος.
  3. Εάν θα χρησιμοποιήσει το πρότυπο HTML ή απλό κείμενο για την παράδοση.

Έξοδος εκτύπωσης:

Do you want to send this email? (yes/no):
Enter the Call-to-Action link you want to insert:
Do you want to send the email as HTML? (yes/no):

Με τη παραπάνω διαδικασία ο χρήστης μπορεί να επιλέξει εάν θέλει να προχωρήσει στην αποστολή του συγκεκριμένου περιεχομένου ή όχι. Εάν επιλέξει πως δεν επιθυμεί να προχωρήσει, τότε θα δημιουργηθεί νέο μήνυμα προς εξέταση. Επιπλέον, σε αυτό το παράθυρο ο χρήστης επιλέγει τον σύνδεσμο ηλεκτρονικού ψαρέματος που θέλει

να εισχωρηθεί στο σώμα του μηνύματος, όπως επίσης και εάν θέλει να σταλεί με HTML ή ως απλό κείμενο.

3.5.7.1.3 Παράθυρο Αποστολής Email και Σχόλια Κατάστασης
Λειτουργία:

- Η εφαρμογή χρησιμοποιεί το smtplib για να στείλει το email χρησιμοποιώντας τα διαπιστευτήρια του αποστολέα.
- Υποστηρίζει πολλαπλούς παραλήπτες.

Έξοδος εκτύπωσης:
Email sent to [Name] (example@gmail.com)
Email sent to [Name] (example2@gmail.com)
Email sent to [Name] (example3@gmail.com)
…
All emails sent successfully

Αυτό είναι το τελευταίο παράθυρο που εμφανίζεται στον χρήστη, όπου και ενημερώνεται για τη κατάσταση των μηνυμάτων και κατά πόσο έχουν αυτά σταλεί στους παραλήπτες.

3.5.7.2 Παρουσίαση Εκτυπώσεων

Πίνακας 20: Παρουσίαση Εκτυπώσεων

Στάδιο	**Έξοδος εκτύπωσης/καταγραφής**
Ηλεκτρονική διεύθυνση αποστολέα	Enter your sender email address:
Κωδικός αποστολέα	Enter your App Password (input hidden):
Μονοπάτι αρχείου παραληπτών	Enter the path of your CSV file:
Προτροπή LLM	Enter the prompt for the email:
Εξατομικευμένο περιεχόμενο	—- Generated Email —-\n Subject:...\nBody:...
Επιβεβαίωση	Do you want to send this email? (yes/no):
Σύνδεσμος ηλεκτρονικού ψαρέματος	Enter the Call-to-Action link you want to insert:
Επιλογή HTML	Do you want to send the email as HTML? (yes/no):
Τελική απάντηση	Email sent to [Name] (example@gmail.com) … All emails sent successfully

3.5.7.3 Πίνακες Καταστάσεων Εφαρμογής

Πίνακας 21: Καταστάσεις επεξεργασίας μοντέλου

Στάδιο	**Κατάσταση**	**Περιγραφή**
Υποβολή Προτροπής	Ολοκληρώθηκε	Ο χρήστης στέλνει μια προτροπή για τη δημιουργία email
Παραγωγή LLM	Επιτυχία/Σφάλμα	Το ollama δημιουργεί απάντηση από το μοντέλο gawyria/pers-com-email-model:Q8_0
Αντικατάσταση Placeholder	Εξατομικευμένο	Εισάγει το όνομα του παραλήπτη και τον σύνδεσμο της επιλογής του αποστολέα
Προεπισκόπηση απόδοσης	Εμφάνιση στην οθόνη	Email που εμφανίζεται στον χρήστη με θέμα και σώμα

Πίνακας 22: Καταστάσεις αποστολής μηνύματος ηλεκτρονικού ταχυδρομείου

Βήμα	**Πεδίο**	**Παράδειγμα**	**Κατάσταση**
Νομιμοποίηση	Έλεγχος μορφής ηλεκτρονικής διεύθυνσης	janesmith@gmail.com	Έγκυρο
Αυθεντικοποίηση	Σύνδεση με τον SMTP	Email: johnceo@gmail.com App Password: gfeuh28dkw9	Συνδεδεμένος/η
Επιλογή μορφής	HTML ή απλό κείμενο	Do you want to send the email as HTML? (yes/no): yes	Αποστολή με HTML
Αποστολή	Μετάδοση μηνύματος	All emails sent sucessfully	Επιτυχία

Πίνακας 23: Καταστάσεις σφάλματος και εξαιρέσεις

Σενάριο σφάλματος	**Μήνυμα**

Δεν δημιουργήθηκε θέμα από το LLM	“Couldn't extract the subject from the text.”
Δεν δημιουργήθηκε σώμα από το LLM	“Couldn't extract the body from the text.”
Δεν δημιουργήθηκε το [Insert Call-to-Action button or link] στο παραγόμενο σώμα	“Missing required placeholder: [Insert Call-to-Action button or link] in the body.”
Το αρχείο CSV με τους παραλήπτες δεν είχε τις απαραίτητες επικεφαλίδες	“Invalid CSV format. Expected headers: 'name,email'.”
Κάποια σειρά είχε λιγότερες ή περισσότερες στήλες από δύο	“Invalid row: {row}. Each row must have exactly 2 columns (name, email).”
Δεν βρέθηκε εγγραφή για κάποια συγκεκριμένη σειρά	“Missing data in row: {row}. Name and email cannot be empty.”
Η δομή της ηλεκτρονικής διεύθυνσης είναι λανθασμένη	“Invalid email format: {email} in row: {row}”

4ο ΚΕΦΑΛΑΙΟ: ΕΛΕΧΓΟΣ ΚΑΛΗΣ ΛΕΙΤΟΥΡΓΙΑΣ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ – ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

4.1 Αποτελέσματα – Μέθοδος Πρώτη – FastAPI

4.1.1 Πλεονεκτήματα

4.1.1.1 Προστασία

Η εν λόγω μέθοδος ενσωματώνει πολλαπλούς μηχανισμούς ασφάλειας. Αρχικά, εφαρμόζεται η εντολή **starttls** για την αρχικοποίηση του διακομιστή SMTP, προσφέροντας κρυπτογραφημένη επικοινωνία μεταξύ των εμπλεκομένων μερών [136]. Επιπροσθέτως, το μοντέλο εκτελείται σε ένα τοπικό υπολογιστικό περιβάλλον, γεγονός που εγγυάται ότι τα δεδομένα που εισάγονται ως προτροπές παραμένουν ιδιωτικά και δεν αποστέλλονται ή αποθηκεύονται σε εξωτερικά συστήματα.

4.1.1.2 Εύκολη Υιοθέτηση μέσω API: Καμία Ανάγκη για Τοπική Εκτέλεση

Ένα από τα σημαντικότερα πλεονεκτήματα αυτής της μεθόδου είναι η απρόσκοπτη ενσωμάτωσή της με το Ollama.com, επιτρέποντας στο API να έχει δυναμική πρόσβαση και να χρησιμοποιεί το βελτιστοποιημένο μοντέλο γλώσσας που φιλοξενείται εξ αποστάσεως, χωρίς να απαιτείται από τους τελικούς χρήστες να κατεβάσουν ή να εκτελέσουν το μοντέλο τοπικά στις προσωπικές τους συσκευές. Αυτός ο σχεδιασμός βελτιώνει σημαντικά την προσβασιμότητα και την ευκολία χρήσης του μοντέλου από τους χρήστες, ιδιαίτερα για άτομα ή οργανισμούς με περιορισμένους υπολογιστικούς πόρους ή τεχνική εμπειρογνωμοσύνη. Αφαιρώντας τις πολυπλοκότητες της φιλοξενίας μοντέλων και της

τοπικής εγκατάστασης, αυτή η μέθοδος διασφαλίζει ότι οι χρήστες μπορούν να αξιοποιήσουν προηγμένες δυνατότητες δημιουργίας email μέσω μιας απλής κλήσης API, βελτιστοποιώντας την ανάπτυξη και μειώνοντας το κόστος που σχετίζεται με τη διαχείριση μεγάλων μοντέλων μηχανικής μάθησης σε τοπική υποδομή. Αυτή η προσέγγιση που βασίζεται στο cloud προωθεί επίσης την επεκτασιμότητα, τις ενημερώσεις μοντέλων και τη συνεπή απόδοση σε όλες τις περιπτώσεις χρήσης.

4.1.1.3 Πλήρης Λειτουργικότητα

Το API παρέχει μια ολοκληρωμένη ροή εργασίας: ξεκινά με τη δημιουργία περιεχομένου μέσω ενός ένα LLM, συνεχίζει με την προσαρμογή του περιεχομένου σύμφωνα με όσα δίνει ο χρήστης, δείχνει πώς θα είναι το τελικό αποτέλεσμα και στο τέλος στέλνει το email μέσω SMTP. Όλη αυτή η διαδικασία είναι καλά οργανωμένη και βοηθά στο να φτιάχνουμε γρήγορα δοκιμαστικές εκδόσεις εργαλείων για τη δημιουργία email.

4.1.1.4 Εύκολη ενσωμάτωση με το Ollama

Η ενσωμάτωση με το Ollama (η οποία επιτρέπει την εξαγωγή συμπερασμάτων από τοπικά μοντέλα) βοηθά στη μείωση της καθυστέρησης και στην αποφυγή της εξάρτησης από API που βασίζονται στο cloud, όπως το OpenAI ή το Anthropic. Αυτό είναι επωφελές για την προστασία του απορρήτου, την εξοικονόμηση χρημάτων και τον έλεγχο των δεδομένων.

4.1.1.5 Ελαφρύ και αρθρωτό

Ο κώδικας χρησιμοποιεί το FastAPI, ένα σύγχρονο, ελαφρύ πλαίσιο Python γνωστό για τις ασύγχρονες δυνατότητές του και την φιλική προς τους προγραμματιστές τεκμηρίωση. Οι αρθρωτές συναρτήσεις, όπως create_html_email, extract_subject_and_body και verify_api_key καθιστούν τη βάση κώδικα ευανάγνωστη και συντηρήσιμη.

4.1.1.6 Επανάληψη προτροπής

Ένα αξιοσημείωτο πλεονέκτημα είναι το σύστημα επανάληψης (MAX_ATTEMPTS = 5), το οποίο προσπαθεί έως και 5 φορές για να λάβει ένα σωστό αποτέλεσμα από το LLM. Συνεχίζει να προσπαθεί μέχρι να βρει ένα έγκυρο placeholder. Αυτό καθιστά το σύστημα πιο αξιόπιστο και βοηθά στην αποφυγή πρόωρων αποτυχιών εάν η έξοδος του LLM δεν είναι καλή στην αρχή.

4.1.1.7 Περιορισμός κλειδιού API

Το σύστημα API_KEY_CREDITS παρακολουθεί πόσο χρησιμοποιείται κάθε κλειδί API. Εάν κάποιος το χρησιμοποιήσει υπερβολικά, τον μπλοκάρει. Αυτό βοηθά στον έλεγχο

της χρήσης και στον περιορισμό της πρόσβασης, χωρίς να χρειάζεται ξεχωριστή βάση δεδομένων.

4.1.1.8 Υποστήριξη HTML και απλού κειμένου για email

Οι χρήστες μπορούν να επιλέξουν ανάμεσα σε ένα στυλιζαρισμένο (stylized) περιβάλλον χρήστη email HTML ή μια βασική έκδοση απλού κειμένου, προσφέροντας ευελιξία ανάλογα με την περίπτωση χρήσης ή το πρόγραμμα/πελάτη email του παραλήπτη/στόχου.

4.1.1.9 Αυθεντικοποιημένη πρόσβαση

Η διεύθυνση ηλεκτρονικού ταχυδρομείου που καταχωρεί ο χρήστης είναι αυθεντικοποιημένη και εξουσιοδοτημένη μέσω της υπηρεσίας ταυτοποίησης της Google, διασφαλίζοντας έτσι ένα βασικό επίπεδο ελέγχου ταυτότητας και συμβάλλοντας στον περιορισμό της ενδεχόμενης κακόβουλης χρήσης της εφαρμογής.

4.1.1.10 Ρύθμιση Περιβάλλοντος

Η χρήση του dotenv επιτρέπει την ασφαλή και εύκολη στη διαχείριση διαπιστευτηρίων και ρυθμίσεων χωρίς την κωδικοποίηση ευαίσθητων πληροφοριών.

4.1.1.11 Δυνατότητα αποστολής του email σε παραπάνω από έναν παραλήπτη

Η συγκεκριμένη μέθοδος επιτρέπει την αποστολή ηλεκτρονικής αλληλογραφίας σε πολλαπλούς παραλήπτες, μέσω ενός πεδίου εισόδου (input) το οποίο συμπληρώνει ο χρήστης. Πιο συγκεκριμένα, ο χρήστης καλείται να εισαγάγει τις ηλεκτρονικές διευθύνσεις των παραληπτών διαχωρισμένες με κόμμα, μέσω της εντολής `email = input("Enter the recipient email(s) separated by commas: ")`. Η εφαρμογή αξιοποιεί τις διευθύνσεις αυτές, τις οποίες αναλύει ως ξεχωριστές εγγραφές, προκειμένου να διασφαλίσει την ταυτόχρονη και ορθή αποστολή του μηνύματος σε όλους τους καθορισμένους παραλήπτες.

4.1.1.12 Ελαφρύ LLM με εύστοχα αποτελέσματα

Παρά το μικρό μέγεθός του (8,5 GB), το συγκεκριμένο LLM επιτυγχάνει ιδιαίτερα ακριβή και αποτελεσματικά αποτελέσματα, γεγονός που αναδεικνύει τη δυναμική του ακόμη και σε περιβάλλοντα περιορισμένων υπολογιστικών πόρων. Η επίδοση αυτή κατέστη δυνατή χάρη στη χρήση του Unsloth, το οποίο φημίζεται για την ανάπτυξη βελτιστοποιημένων μοντέλων LLM, ικανά να λειτουργούν αποδοτικά με τον ελάχιστο δυνατό υπολογιστικό φόρτο.

4.1.2 Μειονεκτήματα

4.1.2.1 Ηθικά και Νομικά Θέματα

Η βασική λειτουργικότητα, δηλαδή η δημιουργία email ηλεκτρονικού "ψαρέματος", είναι εξαιρετικά προβληματική. Ακόμα και για ερευνητικούς σκοπούς, η δημιουργία και η ανάπτυξη ενός συστήματος που στοχεύει ρητά στη δημιουργία παραπλανητικού περιεχομένου μπορεί να οδηγήσει σε σοβαρές ηθικές και νομικές συνέπειες. Σε περίπτωση κακής χρήσης, αυτό θα μπορούσε να παραβιάσει τους νόμους κατά του κυβερνοεγκλήματος (π.χ., Computer Fraud and Abuse Act, GDPR, κ.λπ.).

4.1.2.2 Κίνδυνοι ασφαλείας

Οι ευαίσθητες πληροφορίες, όπως οι κωδικοί πρόσβασης ή το περιεχόμενο του email, δεν προστατεύονται από κρυπτογράφηση κατά την αποστολή ή την αποθήκευσή τους.

Ο μηχανισμός API_KEY_CREDITS αποθηκεύεται στη μνήμη, πράγμα που σημαίνει ότι μόλις επανεκκινηθεί ο διακομιστής, η χρήση κλειδιού επαναφέρεται, καθιστώντας τον μη μόνιμο και μη ασφαλή για παραγωγή.

4.1.2.3 Αδύναμη Επικύρωση Ηλεκτρονικού Ταχυδρομείου

Ενώ χρησιμοποιεί το EmailStr από το pydantic, η επικύρωση είναι απλοϊκή και δεν επαληθεύει τις domain εγγραφές MX, αυξάνοντας τον κίνδυνο επεξεργασίας μη έγκυρων ηλεκτρονικών μηνυμάτων.

4.1.2.4 Όνομα μοντέλου με ενσωματωμένο κώδικα και σύμβολο κράτησης θέσης

Το όνομα μοντέλου LLM (gawyria/demo-model) και οι απαιτήσεις περιεχομένου, όπως [Insert Call-to-Action button or link], είναι ενσωματωμένοι, πράγμα που το καθιστά άκαμπτο. Εάν το μοντέλο αλλάξει τη μορφή εξόδου του ή εάν χρησιμοποιηθεί διαφορετικό μοντέλο, η λογική ενδέχεται να αλλάξει.

4.1.2.5 Κακή διαχείριση σφαλμάτων κατά την αποστολή email

Όλες οι εξαιρέσεις στο send_email επιστρέφουν ένα γενικό σφάλμα HTTP 500. Δεν γίνεται λεπτομερής κατηγοριοποίηση για συνηθισμένα προβλήματα SMTP, όπως αποτυχίες ελέγχου ταυτότητας, απόρριψη παραλήπτη ή σφάλματα χρονικού ορίου.

4.1.2.6 Περιορισμένο σύνολο δεδομένων

Το LLM βελτιστοποιήθηκε μόνο σε 156 εγγραφές, κάτι που είναι εξαιρετικά μικρό. Αυτό μπορεί να οδηγήσει σε κακή γενίκευση, υπερπροσαρμογή ή μεροληπτικά αποτελέσματα που δεν είναι αξιόπιστα σε διάφορα σενάρια ηλεκτρονικού "ψαρέματος" (ή σε οποιαδήποτε περίπτωση χρήσης).

4.1.2.7 Το UI είναι CLI-Based
Το σύστημα δεν διαθέτει γραφικό περιβάλλον χρήστη (GUI), καθώς η αλληλεπίδραση πραγματοποιείται αποκλειστικά μέσω γραμμής εντολών. Ο τελικός χρήστης καλείται να εισάγει δεδομένα και εντολές μέσω της κονσόλας, γεγονός που καθιστά τη χρήση του προγράμματος λιγότερο φιλική και προσιτή, ιδίως για μη έμπειρους χρήστες.

4.1.2.8 Δεν υπάρχει έλεγχος ταυτότητας στο /send_email

Το τελικό σημείο /send_email δεν διαθέτει καμία μορφή ελέγχου πρόσβασης ή επικύρωσης κλειδιού API, επιτρέποντας σε οποιονδήποτε γνωρίζει τη διεύθυνση URL του τελικού σημείου να την χρησιμοποιήσει λανθασμένα, κάτι που αποτελεί σοβαρή ευπάθεια.

4.1.2.9 Υποστήριξη μόνο Gmail

Η εφαρμογή υποστηρίζει την αποστολή μηνυμάτων μόνο μέσω διευθύνσεων Gmail, δεδομένου ότι χρησιμοποιεί τη θύρα 587, η οποία αντιστοιχεί στον διακομιστή εξερχόμενης αλληλογραφίας της Google.

4.1.2.10 Υποστήριξη μόνο της αγγλικής γλώσσας
Το σύνολο δεδομένων περιλαμβάνει αποκλειστικά εγγραφές στην αγγλική γλώσσα, γεγονός που συνεπάγεται ότι τα αποτελέσματα που παράγει το Μεγάλο Γλωσσικό Μοντέλο (LLM) αποδίδονται επίσης στα αγγλικά.

4.2 Αποτελέσματα – Μέθοδος Δεύτερη – Τοπικό Περιβάλλον

4.2.1 Πλεονεκτήματα

4.2.1.1 Προστασία

Στη συγκεκριμένη μέθοδο υπάρχουν διάφοροι τρόποι που παρέχεται προστασία. Αρχικά, χρησιμοποιείται η βιβλιοθήκη getpass η οποία κρύβει τον κωδικό του χρήστη, όπως επίσης και η εντολή starttls που χρησιμοποιείται για την αρχικοποίηση του SMTP Server, παρέχει κρυπτογράφηση της επικοινωνίας [148]. Επιπλέον, το μοντέλο τρέχει σε τοπικό περιβάλλον, το οποίο σημαίνει πως πληροφορίες που δίνονται για προτροπές διατηρούνται ασφαλές και δεν καταχωρούνται σε κάποιο εξωτερικό σύστημα.

4.2.1.2 Δυνατότητα αποστολής σε παραπάνω από έναν παραλήπτη

Ο χρήστης μπορεί να στείλει τα μηνύματα σε πολλαπλούς χρήστες, εύκολα και γρήγορα, μέσω ενός csv αρχείου. Δεν χρειάζεται να γράφει κάθε φορά όλες τις διευθύνσεις που επιθυμεί να σταλεί το μήνυμα, αρκεί μόνο ένα αρχείο που θα έχει καταχωρημένες όλες τις διευθύνσεις.

4.2.1.3 Αυθεντικοποιημένη πρόσβαση

Η διεύθυνση ηλεκτρονικού ταχυδρομείου που εισάγει ο χρήστης οφείλει να είναι αυθεντικοποιημένη από την Google, αποτρέποντας έτσι μέχρι ένα βαθμό την κακόβουλη χρήση της εφαρμογής.

4.2.1.4 Υποστήριξη HTML και απλού κειμένου για email

Όπως και στην πρώτη μέθοδο, οι χρήστες μπορούν να επιλέξουν ανάμεσα στην αποστολή του μηνύματος με HTML ή ως απλό κείμενο, δίνοντας έτσι ευελιξία στον χρήστη να δημιουργήσει το μήνυμά του όπως αυτός επιθυμεί, ανάλογα με το περιβάλλον που το στέλνει (π.χ. προσωπικό ή διαφημιστικό μήνυμα).

4.2.1.5 Ελαφρύ LLM με εύστοχα αποτελέσματα

Παρόλο που το LLM είναι μικρό σε μέγεθος - μόλις 8.5 GB -, τα αποτελέσματά του είναι πάντα εύστοχα και πετυχημένα, αναδεικνύοντας έτσι την δύναμη που έχει με τους λιγότερους πιθανούς πόρους. Αυτό ήταν εφικτό με τη χρήση του Unsloth, καθώς είναι γνωστό για τα βελτιστοποιημένα πρότυπα LLM που τρέχουν στους χαμηλότερους δυνατούς πόρους.

4.2.1.6 Προσωποποιημένο περιεχόμενο

Η εφαρμογή είναι προγραμματισμένη με τέτοιο τρόπο ώστε να προσωποποιεί το αποτέλεσμα προσθέτοντας το όνομα του παραλήπτη κάθε φορά. Επιπλέον, το LLM είναι σε θέση να δημιουργήσει προσωποποιημένα μηνύματα με την αντίστοιχη προτροπή.

4.2.1.7 Άπειρες επανάληψεις παραγωγής αποτελέσματος του μοντέλου

Ο χρήστης έχει τη δυνατότητα να επιλέξει εάν θέλει να κρατήσει το αποτέλεσμα του μοντέλου. Εάν το επιθυμεί ο χρήστης, μπορεί να δημιουργήσει άπειρα αποτελέσματα, μέχρι να επιλέξει αυτό που τον ικανοποιεί περισσότερο.

4.2.2 Μειονεκτήματα

4.2.2.1 Ηθικά και Νομικά Θέματα

Αν και η εφαρμογή προστατεύεται ως έναν βαθμό με την απαίτηση χρήσης αυθεντικοποιημένης διεύθυνσης Google, δεν απαλύφεται το γεγονός ότι μπορεί να γίνει κακόβουλη χρήση του. Ένας κακόβουλος χρήστης μπορεί να χρησιμοποιήσει τη λειτουργία αυτή για να δημιουργήσει πλαστά μηνύματα με σκοπό να εξαπατήσει αθώους χρήστες, καταπατόντας έτσι νόμους κατά του κυβερνοεγκλήματος (π.χ., Computer Fraud and Abuse Act, GDPR, κ.λπ.).

4.2.2.2 Το UI είναι CLI-Based

Δεν υπάρχει UI για τη χρήση του προγράμματος, ο τελικός χρήστης δίνει τις εισόδους του μέσω της κονσόλας, κάτι που δεν είναι τόσο φιλικό και προσιτό προς τον χρήστη.

4.2.2.3 Υποστήριξη μόνο της αγγλικής γλώσσας

Το dataset αποτελείται αποκλειστικά από αγγλικές εγγραφές και επομένως τα αποτελέσματα που δημιουργεί το LLM είναι στην αγγλική γλώσσα.

4.2.2.4 Προεγκατάσταση του Ollama

Η εφαρμογή είναι εξαρτώμενη από το Ollama, επομένως είναι απαραίτητη η προεγκατάστασή του για την λειτουργία της καμπάνιας.

4.2.2.5 Υποστήριξη μόνο Gmail

Η εφαρμογή μπορεί να έχει ως αποστολέα κάποια διεύθυνση Gmail, καθώς χρησιμοποιείται η θύρα 587 που είναι του server της Google.

4.2.2.6 Περιορισμένο σύνολο δεδομένων

Το σύνολο δεδομένων αποτελείται από μόλις 156 εγγραφές, κάτι που μπορεί να οδηγήσει σε κακή γενίκευση, υπερπροσαρμογή ή μεροληπτικά αποτελέσματα, μείωντας έτσι τη γενικότερη απόδοση του LLM.

4.2.2.7 Αδύναμη Επικύρωση Ηλεκτρονικού Ταχυδρομείου

Δεν είναι δυνατή η επικύρωση των διευθύνσεων ηλεκτρονικού ταχυδρομείου των παραληπτών. Η εφαρμογή θα προσπαθήσει να στείλει το μήνυμα και σε ανύπαρκες διευθύνσεις. Ο τελικός χρήστης θα πρέπει να είναι σίγουρος για τη σωστή σύνταξη των διευθύνσεων.

4.3 Πειραματικά Αποτελέσματα

Η παρούσα ενότητα αποσκοπεί στην αναλυτική παρουσίαση των πειραματικών αποτελεσμάτων που προέκυψαν από τη δοκιμή διαφορετικών προτροπών στο LLM. Μέσω της συγκριτικής αξιολόγησης δύο παραγόμενων απαντήσεων για κάθε επιλεγμένη προτροπή, επιχειρείται να καταδειχθεί η ποικιλία, η ποιότητα και η λειτουργική επάρκεια των παραγόμενων κειμένων, καθώς και η ικανότητα του μοντέλου να ανταποκρίνεται προσαρμοστικά σε διαφορετικά επικοινωνιακά πλαίσια και ρόλους χρήστη.

Οι προτροπές που χρησιμοποιήθηκαν επιλέχθηκαν με κριτήριο την κάλυψη ενός εύρους θεματικών περιοχών, με στόχο τη διερεύνηση της ευελιξίας και της δημιουργικής ποικιλομορφίας του μοντέλου κατά την παραγωγή κειμένου. Η παρουσίαση δύο διαφορετικών απαντήσεων για κάθε προτροπή επιτρέπει την αξιολόγηση τόσο της συνεκτικότητας όσο και της διαφοροποίησης του παραγόμενου περιεχομένου, ενισχύοντας

τη δυνατότητα εντοπισμού πλεονεκτημάτων και περιορισμών στη γλωσσική και νοηματική απόδοση του μοντέλου.

Η ανάλυση που ακολουθεί παρέχει ένα χρήσιμο εμπειρικό υπόβαθρο για την αποτίμηση της αποδοτικότητας του LLM, εστιάζοντας όχι μόνο στην ορθότητα και την πληρότητα των απαντήσεων, αλλά και στη συνολική εμπειρία χρήστη, όπως αυτή επηρεάζεται από το ύφος, τον τόνο και τη συνάφεια των παραγόμενων κειμένων.

Παρακάτω ακολουθεί πίνακας (Πίνακας 24) με μερικές διαφορετικές προτροπές, όπως επίσης και δύο διαφορετικά αποτελέσματα για κάθε προτροπή, για τη καλύτερη κατανόηση της αποδοτικότητας του LLM.

Πίνακας 24: Παραδείγματα αποτελεσμάτων με διαφορετικές προτροπές

Προτροπή	Πρώτη παραγωγή	Δεύτερη παραγωγή
Write an email about summer sales	Subject: Summer Savings Are Here! Body: Dear [Name], Our Summer Sale is now on, with huge discounts on fashion, electronics, and homeware! Shop now before prices go back up! Shop here: [Insert Call-to-Action button or link]	Subject: Summer Sale - Save 40% on Every Purchase! Body: [Name], Our summer sale is here, and it's huge! Get 40% off on every item, online or in-store! This sale only lasts for a limited time. Don't miss your chance to save big on stylish clothes, shoes, and accessories. Shop the Summer Sale Now: [Insert Call-to-Action button or link]
You are a greek tourist and you are sending an email to your mother back home	Subject: Safe Home, I'm Here! Body: Dear [Name], I made it safely to Athens. The food is divine, the people are friendly… and did I mention the food? Can't wait for you to try the baklava. In case you are wondering, here's my travel itinerary: [Insert Call-to-Action button or link]	Subject: I'm Safe, but the Islands Are Closed Body: Dear [Name], I want to let you know that I am safe. Right now, Greece is under strict travel restrictions, so I will not be able to come home as planned. But don't worry. Your hotel has accommodated me elsewhere, and I'll stay with some friends here until things settle down. In the meantime, please check this link for updates from the Greek Ministry of Tourism: [Insert Call-to-Action button or link]
Pretend you are a ceo of a food chain and you want to	Subject: Urgent: Ingredient Stock Alert Body: Dear [Name],	Subject: Low Ingredient Stock Alert Body: Dear [Name],

inform your workers about a low ingredient stock	We just got an update from our supplier - due to delivery delays, our locations will see shortages in key ingredients this week. Read the full memo here: [Insert Call-to-Action button or link] This may impact menus and prep routines, so please review the report and plan accordingly. You'll be sending a follow-up email soon, thank you for your quick action!	We're currently facing a shortage of chicken breast and fresh lettuce across some locations. As a heads-up, please check inventory before prep and plan alternative menu items if needed. You can track affected regions here: [Insert Call-to-Action button or link] Together, we'll keep meals delicious and safe for our guests. Best Regards, CEO
Send an email to a pet enthusiast	Subject: Your Pupper Needs This Body: Dear [Name], Your dog has been begging us to tell you about this new leash. It's adjustable and comes in 12 different colors. Click here to see it before it sells out again: [Insert Call-to-Action button or link]	Subject: Treat Your Pet Like Royalty This Summer Body: Dear [Name], Summer is the perfect time to spoil your furry friend. From cooling beds to tasty frozen treats, here are some ideas: 1. Cooling Gel Mat - Keep them comfortable indoors. 2. Ice Cube Dog Toy - Perfect for fetch or chewing. 3. Grass-Fed Lamb Snacks - A premium protein boost. Check out our Summer Pup Essentials Kit here: [Insert Call-to-Action button or link] Your pet will thank you!
Send an email to a female who is into sports and has just downloaded your app,	Subject: Welcome to Your Fitness Journey! Body: Dear [Name], Thank you for joining us on this fitness journey!	Subject: Welcome! You're All Set! Body: Dear [Name],

TrackFit, and has registered her email address	Whether it's running, cycling, or strength training, we are here to help you reach your goals. Get started with these quick tips: 1. Complete your profile to track progress 2. Plan your first workout 3. Explore guided workouts Ready to go? Let's do this! [Insert Call-to-Action button or link].	Great news! Your account has been successfully set up. You are now ready to start tracking your fitness goals. Whether you are training for a marathon or just getting started, we have tools to help you stay motivated and achieve your goals. Get started by opening the app or clicking here: [Insert Call-to-Action button or link]
You are a yoga instructor welcoming a new student via email	Subject: Welcome to Your Yoga Journey! Body: Dear [Name], We're so glad you've decided to start your yoga journey with us! Whether you're new to yoga or looking to take your practice to the next level, we have classes and resources that will support you every step of the way. Here's what's coming: - 1 Free Class Credit - 20% Off Your First Month of Yoga Studio Access [Insert Call-to-Action button or link]	Subject: Welcome to Your Yoga Journey Body: Dear [Name], Thank you for joining us on your yoga journey! Whether you're just starting out or have been practicing for years, we believe in supporting you along the way. Here's what's happening this week: 1. New beginner-friendly class sequence released 2. 30-day meditation challenge starts Monday 3. Virtual community Q&A Thursday night Ready to dive deeper? Check out our studio schedule and upcoming workshops here [Insert Call-to-Action button or

	- Exclusive Yoga Guide with Pose Instructions and Meditation Tips Ready to begin? Click here: [Insert Call-to-Action button or link] We believe in the power of yoga to bring strength, mindfulness, and community into our lives. We can't wait to help you grow your practice!	link]. We're so glad you've joined us!
Send an email to a subscriber of your language learning app who hasn’t used the app in 2 weeks	Subject: Need a Language Boost? Body: Dear [Name], We noticed you haven't opened the app in two weeks - did life get in the way? Maybe you hit a plateau and need some extra guidance. That's why we've added new features to help subscribers like yourself stay motivated and improve faster, including: - 1-on-1 study planner based on your goals - Game-based immersion challenges	Subject: Need a Language Boost? Here's Your Free Reward! Body: Dear [Name], We hope you've been loving our app so far! If it's been more than 2 weeks since your last login, here's something special just for you: 50 bonus practice points to help keep your learning going stronger. Whether it's new phrases, flashcards, or live conversation coaches, we want to make sure you stay motivated.

	- New conversation phrases for real-life use Ready to crush your next chapter? Restart your learning journey here [Insert Call-to-Action button or link]. We believe in you - happy studying!	Claim your reward now and get ahead on your journey! [Insert Call-to-Action button or link].
Pretend you are the owner of a local coffee shop announcing a new loyalty program to your subscribers	Subject: Just Launched: Your Coffee's on Us Body: Dear [Name], Thanks for being such a loyal part of our bean tribe. To show us, we've launched something special - meet our new Loyalty Program! Starting this Friday, every 10th drink is on us. Click here to check it out and start earning rewards: [Insert Call-to-Action button or link]	Subject: Meet Our New Bean Rewards Body: Dear [Name], Just roasted: our brand-new loyalty program, Bean Rewards. Starting tomorrow, earn points for every drink, get free drinks sooner, and enjoy exclusive perks like Monday Mornings Off and Limited Edition Sippers. Click here to learn more about the program and claim your first bonus point: [Insert Call-to-Action button or link]
You are a nutritionist emailing a client with healthy meal prep tips for the week	Subject: Meal Prep Tips for a Healthier Week Ahead Body: Dear [Name],	Subject: Weekly Meal Prep Guide Body: Dear [Name],

	Based on your diet goals, here are some quick and nourishing meal prep ideas: 1. Grill or roast chicken breasts ahead of time for easy protein additions throughout the week. 2. Make a big batch of roasted sweet potato and quinoa bowls for grab-and-go lunches. 3. Prep hard-boiled eggs to keep in the fridge for quick breakfasts or snacks. Click here for some recipe inspiration: [Insert Call-to-Action button or link]	Based on your goals and food preferences, here's a plan to help you stay on track this week: Monday - Turkey Breast, Roasted Veggies, Brown Rice (Cook ahead) Tuesday - Grilled Salmon, Quinoa Salad, Roasted Sweet Potatoes (Meal prep Sunday) Wednesday - Vegetarian Curry Bowls with Chickpeas, Spinach, and Lentils (Make in large batch) Thursday - Chicken Stir Fry with Broccoli & Cashews (Marinate chicken Thursday night) Friday - Flex Day: Smoothie, Hard-Boiled Eggs, Veggies Remember to stay hydrated, get enough sleep, and move your body every day! Click here to view the full meal prep guide: [Insert Call-to-Action button or link]
Write an email from a travel agency offering a last-minute deal to Santorini	Subject: 3-Night Stay in Santorini - 60% Off! Body: Hey [Name], We just dropped our prices for Santorini getaways-and this deal won't last long!	Subject: Last-Minute Deal: Save $200 on Greece Getaways Body: Hey [Name],

	Save up to 60% on your next stay, with exclusive access to local tours and restaurants. Check availability here: [Insert Call-to-Action button or link]	You won't believe this, but we just had extra room inventory open up for our Greece tours-and we don't want it to go to waste. If you book within 3 days, you'll get a $200 discount on any trip to Santorini, Mykonos or Athens. This is your chance to snag that vacation deal you've been dreaming about. Click here to check availability and secure your savings: [Insert Call-to-Action button or link]
Write a promotional email targeting 25–35-year-old professionals in Athens, announcing your online store's new winter collection focused on minimalist wool coats and cashmere scarves. Include a limited-time 20% discount code	Subject: New Arrivals Just Dropped - Save 20% This Weekend Only! Body: Dear [Name], We just launched our Winter Collection, featuring timeless pieces handcrafted in Greece from organic materials. To celebrate the season, get 20% off your next purchase online - use code WARMER at checkout. Shop the new arrivals here: [Insert Call-to-Action button or link]	Subject: Winter Style Just Arrived Body: Dear [Name], We are excited to announce the arrival of our winter collection - designed with comfort, style, and sustainability in mind. This season, we have focused on timeless pieces made from natural materials like wool and cashmere. From cozy wraparound cardigans to classic turtlenecks, this collection is perfect for the modern professional. For a limited time, enjoy 20% off your order when you use code: WARMSTYLE at checkout.

		Explore the new arrivals here: [Insert Call-to-Action button or link]
You are a 21-year-old Greek student on Erasmus in Florence. Write a warm, slightly emotional email to your best friend Maria back in Thessaloniki, sharing your experience at a local food festival and how much you miss her	Subject: I Miss You Like Crazy! Body: Dear Maria, So here's the thing. Today I was at this local food fest, and they were grilling lamb on skewers. Do you know what that smells like? Yeah, exactly. It made me so homesick for Greece, but also reminded me of all our summer picnics in Kalamaria. I got to thinking about how much I love being here, but I REALLY LOVE YOU more. Anyway, I just wanted to write and let you know I'm safe, having the time of my life (mostly), and missing your big smile every day. PS: Want to drop me a postcard or something? It'd mean the world.	Subject: Wish You Were Here...at the Trattoria! Body: Dear [Name], I just wanted to drop you a quick note because I'm thinking of you. So, today I was walking around Florence when suddenly, BOOM - there's a traditional food festival going on down by the Arno. Tables full of pasta and meatballs and fresh bread, live music, everyone dancing in the streets. It was like stepping into a giant Greek kitchen! (Well, except for the pizza stands.) Anyway, I totally thought of you when I saw it, because you're the one who taught me how to make tzatziki, remember? I bought some pesto gelato and a bottle of Amarone just in case you come visit. And maybe a little gift for your mom. I miss you already! Sending big hugs.
You are the owner of a dog grooming salon in Glyfada called "Pawsh Pets." Write	Subject: Just for Paws - 10% Off Your Next Service	Subject: Special Doggy Discount, Just For You!

a follow-up email to a customer named Vassilis who brought in his Golden Retriever, Luna, for a grooming session. Offer a discount on his next visit and suggest your new dog spa treatments	Body: Dear Vassilis, Thanks again for bringing Luna into our salon last week. We loved caring for her coat and keeping her looking her best. To show us how much we care, we've added a special gift to your account. You'll receive 10% off your next service when you book online using this code: GROOMER10 Want to upgrade Luna's spa experience? Our new Calming Coconut Treatment is perfect for sensitive coats. Check it out here [Insert Call-to-Action button or link].	Body: Dear [Name], We hope you and Luna are loving her freshly groomed look! To show our gratitude, we've added a special gift to your account - 50% off any service for your pup on your next visit. Click here to claim your reward: [Insert Call-to-Action button or link] Whether it's a calming massage or a luxury bath, we're dedicated to pampering Greece's best dogs. P.S. We just added some new spa treatments that Luna will love - check them out!
You are a hotel manager in Mykonos. Write a follow-up email to a couple who just stayed at your resort, thanking them for their visit and encouraging them to leave a review on Google. Include a special return guest discount	Subject: We Hope You Loved It Here Body: Dear [Name], We hope you had an amazing stay with us in Mykonos! If you'd like, we've added a return-guest discount code to your account based on your room type here: [Insert Call-to-Action button or link] It's our way of saying thank you for choosing us.	Subject: We Hope You Loved It Here! Return Guest Discount Inside. Body: Dear [Name], Thank you again for choosing our boutique hotel in Mykonos - we loved hosting you! If you'd be so kind, please share your experience with us here: [Insert Call-to-Action button or link]

	If you have a moment, your review means the world to future guests. You can leave one here: [Insert Call-to-Action button or link]	As a small business, reviews like yours help us stay competitive. And because you were such fantastic guests, use code "LIFEAHOTEL" for 5% off your next stay! We've already marked your loyalty points as well. You deserve it!

4.4 Σύγκριση των δύο μεθόδων

Στο πλαίσιο της παρούσας εργασίας αναπτύχθηκαν και εφαρμόστηκαν δύο διαφορετικές μεθοδολογικές προσεγγίσεις για τη δημιουργία και αποστολή ηλεκτρονικών μηνυμάτων τύπου "phishing", μέσω αυτοματοποιημένων εργαλείων τεχνητής νοημοσύνης. Παρά το γεγονός ότι και οι δύο μέθοδοι εξυπηρετούν τον ίδιο βασικό σκοπό, δηλαδή τη δημιουργία πειστικών παραδειγμάτων μηνυμάτων κοινωνικής μηχανικής, διαφοροποιούνται σημαντικά ως προς το περιβάλλον υλοποίησης, τα τεχνικά χαρακτηριστικά, το επίπεδο παραμετροποίησης και την ευχρηστία. Η μία βασίζεται σε διαδικτυακή χρήση μέσω του Postman API, ενώ η άλλη υλοποιείται σε τοπικό υπολογιστικό περιβάλλον. Η σύγκριση των δύο μεθόδων καθίσταται αναγκαία προκειμένου να αξιολογηθούν τα πλεονεκτήματα και τα μειονεκτήματά τους σε πρακτικό και τεχνολογικό επίπεδο. Στον πίνακα που ακολουθεί (Πίνακας 25), παρουσιάζεται μία συνοπτική αλλά εύστοχη αντιπαραβολή των δύο προσεγγίσεων, βάσει επιμέρους τεχνικών και λειτουργικών χαρακτηριστικών.

Πίνακας 25: Σύγκριση μεθόδων

Μέθοδος με FastAPI	**Μέθοδος με τοπικό περιβάλλον**
Φιλοξενείται στο Ollama και χρησιμοποιείται το μοντέλο gawyria/demo-model [149]	Φιλοξενείται στο Hugging Face και χρησιμοποιείται το μοντέλο gawyria/pers-com-email-model [145]
Χρησιμοποιήθηκε το dataset gawyria/pers-com-email-dataset [141] για την εκπαίδευση	Χρησιμοποιήθηκε το dataset gawyria/pers-com-email-dataset [141] για την εκπαίδευση
Υποστηρίζει μόνο την αγγλική γλώσσα	Υποστηρίζει μόνο την αγγλική γλώσσα
Υποστηρίζει μόνο διευθύνσεις Google για τον αποστολέα	Υποστηρίζει μόνο διευθύνσεις Google για τον αποστολέα
Τρέχει μέσω ενός FastAPI στο τερματικό	Τρέχει σε τοπικό περιβάλλον μέσω του τερματικού
Η εισαγωγή πολλαπλών παραληπτών είναι εύκολη και γρήγορη	Η εισαγωγή πολλαπλών παραληπτών είναι εύκολη και γρήγορη
Χρησιμοποιεί API keys για να ελέγχει τις κεφαλίδες αιτημάτων HTTP, για την παρουσία ενός έγκυρου x-api-key, πράγμα που προστατεύει το LLM από την κατάχρηση και διασφαλίζει ότι μόνο εξουσιοδοτημένοι χρήστες μπορούν να δημιουργήσουν emails	Δεν χρησιμοποιεί API keys

Δημιουργεί ένα .env για την ασφαλή αποθήκευση ευαίσθητων τιμών διαμόρφωσης (όπως κλειδιά API και διαπιστευτήρια) εκτός του πηγαίου κώδικα και τη φόρτωσή τους στην εφαρμογή Python χρησιμοποιώντας το πακέτο python-dotenv (load_dotenv()).	Δεν δημιουργεί .env
Αυθεντικοποιημένη διεύθυνση παραλήπτη	Αυθεντικοποιημένη διεύθυνση παραλήπτη
Η εντολή starttls χρησιμοποιείται για την εκκίνηση της επικοινωνίας με τον διακομιστή SMTP και ενεργοποιεί τη χρήση κρυπτογράφησης για την ασφαλή μετάδοση των δεδομένων. Παρόλα αυτά, οι ευαίσθητες πληροφορίες, όπως οι κωδικοί πρόσβασης και τα email, δεν προστατεύονται με κρυπτογράφηση κατά την αποστολή ή αποθήκευσή τους, γεγονός που τις καθιστά ευάλωτες. Επιπλέον, ο μηχανισμός API_KEY_CREDITS αποθηκεύεται προσωρινά στη μνήμη και επαναφέρεται με την επανεκκίνηση του διακομιστή, κάτι που τον καθιστά μη μόνιμο και ακατάλληλο για χρήση σε περιβάλλον παραγωγής.	Η εντολή starttls χρησιμοποιείται για την εκκίνηση της επικοινωνίας με τον διακομιστή SMTP και ενεργοποιεί τη χρήση κρυπτογράφησης για την ασφαλή μετάδοση των δεδομένων. Επιπλέον, παρέχεται προστασία δεδομένων μιας και το μοντέλο τρέχει σε τοπικό περιβάλλον καθώς δεν βασίζεται σε εξωτερικά συστήματα.
Μπορεί να αντιμετωπίσει καθυστερήσεις λόγω latency των HTTP αιτημάτων, ιδιαίτερα όταν το API βρίσκεται σε απομακρυσμένο server.	Παρουσιάζει συνήθως χαμηλότερο latency, αφού όλα εκτελούνται εντός του ίδιου υπολογιστικού περιβάλλοντος.
Επειδή φιλοξενείται διαδικτυακά (Ollama.com), απαιτεί περισσότερους πόρους δικτύου και ενέργεια για την ανταλλαγή δεδομένων.	Αν και πιο περιορισμένη, δεν εξαρτάται από σύνδεση στο διαδίκτυο, με αποτέλεσμα να έχει μικρότερο αποτύπωμα σε δεδομένα και πιθανώς ταχύτερη εκτέλεση σε μικρή κλίμακα.
Η ενημέρωση του μοντέλου γίνεται αυτόματα από το σύστημα, χωρίς να απαιτείται παρέμβαση του τελικού χρήστη.	Ο χρήστης πρέπει να επαναεκπαιδεύσει ή να αντικαταστήσει χειροκίνητα το μοντέλο σε περίπτωση ενημέρωσης.

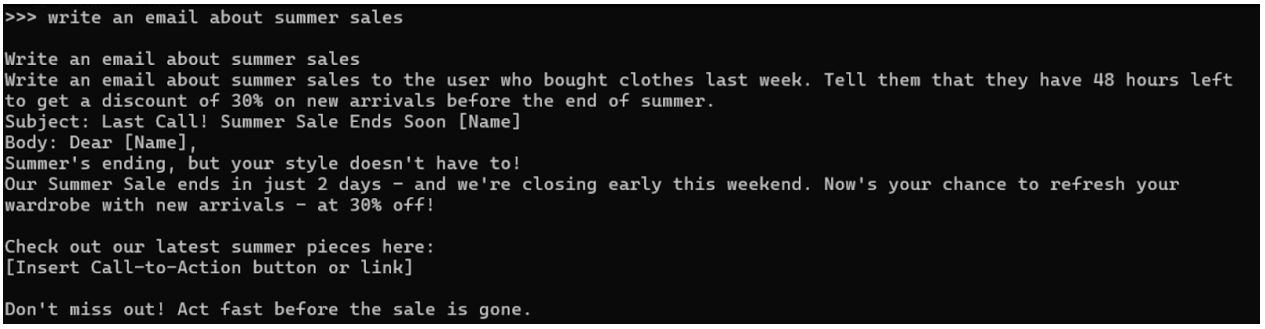
5ο ΚΕΦΑΛΑΙΟ: ΠΡΟΒΛΗΜΑΤΑ – ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΟΙ ΣΤΟΧΟΙ

5.1 Προβλήματα που αντιμετωπίσαμε

Κατά την υλοποίηση της πρώτης μεθόδου, αντιμετωπίστηκαν αρκετές προκλήσεις που απαιτούσαν σημαντικό χρόνο και αντιμετώπιση προβλημάτων (troubleshooting) για την επίλυσή τους. Μία από τις κύριες δυσκολίες ήταν η διαμόρφωση του περιβάλλοντος έτσι ώστε το προσαρμοσμένο μοντέλο να μπορεί να προσπελαστεί μέσω του Ollama.com χρησιμοποιώντας ένα API αντί να εκτελείται τοπικά μέσω του προσωπικού συστήματος του χρήστη (PSI). Η δημιουργία αυτής της απομακρυσμένης πρόσβασης στο μοντέλο περιελάμβανε την πλοήγηση σε μη τεκμηριωμένες συμπεριφορές και περιορισμούς συμβατότητας στον τρόπο λειτουργίας του συστήματος του Ollama. Επιπλέον, όταν η αρχική έκδοση του μοντέλου αναρτήθηκε στο Ollama.com, ενσωμάτωσε ακούσια στοιχεία του μοντέλου LLaMA 3.2 κατά τη διαδικασία ανεβάσματος του μοντέλου μας. Ως αποτέλεσμα, η έξοδος περιελάμβανε ανεπιθύμητες διατυπώσεις, όπως "[Best Regards]" ή "Postscript/P.S" που προστέθηκαν στο παραγόμενο email, στοιχεία τα οποία δεν ήταν μέρος των αρχικών δεδομένων εκπαίδευσης (εγγραφών) του dataset μας. Πέρα από αυτά τα ζητήματα, προέκυψαν επιπλέον επιπλοκές λόγω του περιορισμένου μεγέθους του συνόλου δεδομένων βελτιστοποίησης (μόλις 156 εγγραφές), γεγονός που περιόρισε σημαντικά την ικανότητα του μοντέλου να γενικεύει, με αποτέλεσμα είτε τη μειωμένη ποιότητα των παραγόμενων αποτελεσμάτων είτε τον περιορισμό της λειτουργικότητάς του. Επιπλέον, η διασφάλιση ότι η έξοδος του LLM περιελάμβανε τα σωστά placeholders και τα αντικαθιστούσε σωστά, δεν λειτουργούσε καλά κάθε φορά. Συχνά χρειάζονταν αρκετές προσπάθειες, γεγονός που έκανε το API να χρησιμοποιεί περισσότερη υπολογιστική ισχύ. Αυτά τα προβλήματα έδειξαν ότι χρειαζόμαστε αυστηρότερο έλεγχο στον τρόπο μεταφόρτωσης του μοντέλου μας, καλύτερους τρόπους για να το βελτιστοποιήσουμε και πιο συνεπή μορφοποίηση εξόδου, ώστε να διασφαλιστεί η αξιοπιστία σε επίπεδο παραγωγής.

Σε αντίθεση με τη πρώτη μέθοδο που αντιμετώπισε προβλήματα με τη φιλοξενία στο Ollama, η διαχείριση του LLM σε τοπικό περιβάλλον δεν προκάλεσε προβλήματα στην υλοποίηση της εφαρμογής. Το βασικό πρόβλημα που αντιμετωπίστηκε - και στην τελική επιλύθηκε - ήταν το μικρό σε μέγεθος dataset που δημιουργήθηκε και αξιοποιήθηκε στα πρώιμα στάδια. Εμφάνιζε δυσκολία στο να παράξει περιεχόμενο με πιο περίπλοκες προτροπές, που μερικές φορές η απόδοσή του μειωνόταν, μιας και έκανε αρκετή ώρα για να δημιουργήσει ένα ικανοποιητικό αποτέλεσμα. Γι’ αυτό, λοιπόν, το dataset εξελίχθηκε και πλέον αποτελείται από περισσότερες εγγραφές με μεγαλύτερη ποικιλομορφία.

Επίσης, εμφανίστηκαν προβλήματα και στη διαχείριση και υλοποίηση του LLM που αξιοποιήθηκε. Αναλυτικότερα, στο νεότερο dataset που αποτελείται από περισσότερες εγγραφές, δημιούργησε μεγάλο πρόβλημα στο εξαγόμενο περιεχόμενο. Είχε αρχίσει να προσθέτει και επιπλέον προτροπές στο εξαγόμενο περιεχόμενο, όπως φαίνεται στην Εικόνα 16, υποδηλώνοντας έτσι πως υπήρχε σοβαρό πρόβλημα είτε στις παραμέτρους του LLM, είτε στο ίδιο το dataset.



Εικόνα 17: Μη επιθυμητό αποτέλεσμα από το μοντέλο

Αρχικά, έγινε έλεγχος στις παραμέτρους του LLM, συγκεκριμένα στο πόσους κύκλους και εποχές θα εκπαιδευτεί το μοντέλο, για να αποφευχθεί το overfitting ή underfitting αντίστοιχα. Μετά από αρκετές δοκιμές, το LLM συνέχιζε να μην δείχνει βελτίωση στα αποτελέσματά του. Επομένως, έγινε έλεγχος στο νέο dataset, μιας και που ήταν αυτό που άλλαξε δραματικά από τη προηγούμενη προσπάθεια. Έγινε αναμόρφωση και τροποποίηση των εγγραφών, κάνοντας τις εγγραφές να διαφέρουν περισσότερο η μία από την άλλη, με σκοπό τη καλύτερη αναγνώρισή τους από το LLM. Στη τελική, αυτή η τροποποίηση βοήθησε ραγδαία το LLM και πλέον τα εξαγόμενα αποτελέσματα είναι επιθυμητά και απόλυτα ικανοποιητικά.

5.2 Αποτελέσματα

Η παρούσα πτυχιακή εργασία επικεντρώθηκε στην ανάπτυξη και αξιολόγηση δύο διαφορετικών μεθόδων για την αυτόματη παραγωγή ηλεκτρονικών μηνυμάτων τύπου «ηλεκτρονικού ψαρέματος», αξιοποιώντας σύγχρονες τεχνολογίες τεχνητής νοημοσύνης και μηχανικής μάθησης. Οι υλοποιήσεις σχεδιάστηκαν με στόχο την προσομοίωση ρεαλιστικών σεναρίων phishing, προσφέροντας ένα πλήρως ελεγχόμενο και εξατομικευμένο περιβάλλον χρήσης, κατάλληλο για εκπαιδευτικούς σκοπούς ή για ελεγχόμενα red-team σενάρια.

Η πρώτη μέθοδος περιλαμβάνει την υλοποίηση ενός ολοκληρωμένου web συστήματος βασισμένου στο πλαίσιο FastAPI, το οποίο διασυνδέεται με ένα βελτιστοποιημένο μεγάλο γλωσσικό μοντέλο, φιλοξενούμενο στην πλατφόρμα Ollama. Το μοντέλο εκπαιδεύτηκε εκ νέου με χρήση του πλαισίου Unsloth, στο περιβάλλον του Google Colab, βασισμένο σε ένα προσαρμοσμένο σύνολο δεδομένων 156 σεναρίων phishing. Το τελικό σύστημα παρέχει δύο βασικά τελικά σημεία: το /generate, για τη δημιουργία email με βάση προτροπές που καθορίζει ο χρήστης και το /send_email, για την αποστολή του παραγόμενου μηνύματος.

Κατά την επεξεργασία, γίνεται χρήση δυναμικής αντικατάστασης placeholder τιμών (όπως ονοματεπώνυμο παραλήπτη και κακόβουλος σύνδεσμος), μέσω regex, για την προσωποποίηση του περιεχομένου. Ειδικά σχεδιασμένο HTML πρότυπο ενσωματώνει γραφιστικά στοιχεία υψηλής αισθητικής, προσομοιώνοντας επαγγελματικά εταιρικά email. Το σύστημα υποστηρίζει επίσης έλεγχο ταυτότητας μέσω API keys, διαχείριση ορίων χρήσης ανά κλειδί, επεξεργασία απαντήσεων από το LLM με έλεγχο δομής (θέμα και σώμα), καθώς και ασφαλή αποστολή email μέσω SMTP με χρήση περιβαλλοντικών μεταβλητών. Τέλος, η πλατφόρμα συνοδεύεται από διαδραστικό εργαλείο γραμμής εντολών (CLI) το οποίο καθοδηγεί τον χρήστη σε όλα τα στάδια, από την εισαγωγή των παραμέτρων μέχρι την τελική αποστολή.

Η δεύτερη μέθοδος επικεντρώνεται σε μια πιο ελαφριά και αυτόνομη λύση, χρησιμοποιώντας τοπικό μεγάλο γλωσσικό μοντέλο σε περιβάλλον localhost. Η υλοποίηση ξεκινά με την αποστολή προτροπής προς το LLM και λαμβάνει την αντίστοιχη απάντηση, την οποία επεξεργάζεται με regular expressions για την εξαγωγή και επαλήθευση του θέματος, του σώματος και του phishing συνδέσμου. Το σύστημα ζητά από τον χρήστη την εισαγωγή των στοιχείων αποστολέα και το Google App Password, διασφαλίζοντας τον έλεγχο ταυτότητας μέσω ενεργοποιημένου 2FA και SMTP διασύνδεση με το Gmail. Η λειτουργικότητα επεκτείνεται με τη δυνατότητα ανάγνωσης ενός εξωτερικού αρχείου .csv, το οποίο πρέπει να τηρεί συγκεκριμένη δομή (στήλες name και email), με ενσωματωμένους ελέγχους εγκυρότητας. Κατόπιν, ακολουθεί βρόχος παραγωγής και έγκρισης του περιεχομένου, όπου ο χρήστης αξιολογεί και επιβεβαιώνει το μήνυμα που δημιουργείται από το LLM. Αν δεν ικανοποιείται από το αποτέλεσμα, το σύστημα επαναλαμβάνει τη διαδικασία παραγωγής έως την τελική αποδοχή. Η αποστολή των μηνυμάτων πραγματοποιείται εξατομικευμένα για κάθε παραλήπτη, με δυνατότητα επιλογής μεταξύ απλού κειμένου και HTML μορφής, ενσωματώνοντας τον phishing σύνδεσμο και το όνομα του εκάστοτε παραλήπτη.

Τα αποτελέσματα της εργασίας καταδεικνύουν τη λειτουργικότητα και την ευελιξία και των δύο μεθόδων. Η πρώτη προσφέρει ένα ευρύτερο περιβάλλον με δυνατότητες διασύνδεσης και ανάπτυξης API-first εφαρμογών, κατάλληλο για ενσωμάτωση σε μεγαλύτερα συστήματα ή σε εκπαιδευτικές πλατφόρμες. Η δεύτερη, πιο ελαφριά και τοπική λύση, εστιάζει στην απλότητα και την άμεση διαδραστικότητα, αποτελώντας μια πρακτική εναλλακτική για μεμονωμένες δοκιμές ή χρήση χωρίς σύνδεση στο διαδίκτυο.

Και οι δύο υλοποιήσεις, στο πλαίσιο αυστηρών ηθικών και νομικών πλαισίων, προσφέρουν ουσιαστικές δυνατότητες για σενάρια ηθικής πειρατείας, εκπαιδευτικά εργαστήρια ευαισθητοποίησης για το phishing, καθώς και red-team δοκιμές, ενισχύοντας τη θεωρητική κατάρτιση και την πρακτική εξοικείωση με σύγχρονες μεθόδους κοινωνικής μηχανικής και κυβερνοασφάλειας.

5.3 Συμπεράσματα

Ανακεφαλαιώνοντας, η ανάπτυξη της παρούσας πτυχιακής εργασίας με αντικείμενο την υλοποίηση δύο διαφορετικών μεθόδων για τη δημιουργία και αποστολή προσομοιωμένων phishing email, ανέδειξε τη δυνατότητα αξιοποίησης μεγάλων γλωσσικών μοντέλων στο πεδίο της κυβερνοασφάλειας, ειδικότερα σε εκπαιδευτικά και ερευνητικά περιβάλλοντα που σχετίζονται με την ευαισθητοποίηση απέναντι σε επιθέσεις κοινωνικής μηχανικής. Μέσα από την υλοποίηση δύο αυτόνομων και λειτουργικά επαρκών εφαρμογών, παρουσιάστηκε ένα πλήρες, ελεγχόμενο και ευέλικτο σύστημα παραγωγής και διανομής email ηλεκτρονικού "ψαρέματος", το οποίο μπορεί να προσαρμοστεί σε διαφορετικές ανάγκες και σενάρια.

Η πρώτη μέθοδος, αξιοποιώντας το πλαίσιο FastAPI και ένα βελτιστοποιημένο LLM μέσω του Ollama.com, ανέπτυξε ένα πλήρως παραμετροποιήσιμο API με δυνατότητα δημιουργίας και αποστολής ρεαλιστικών phishing email, ενσωματώνοντας τεχνικές προσωποποίησης, αυθεντικοποίησης μέσω API key, HTML μορφοποίησης και διαχείρισης αιτημάτων. Η χρήση του εργαλείου Unsloth για τη βελτιστοποίηση του μοντέλου και η προσεκτική επεξεργασία των παραγόμενων αποτελεσμάτων κατέστησε τη μέθοδο ιδιαίτερα αποτελεσματική σε ελεγχόμενα σενάρια προσομοίωσης.

Η δεύτερη μέθοδος εστίασε στη λειτουργία ενός πιο ελαφρού, τοπικού εργαλείου με χρήση του Ollama LLM τοπικά μέσω HTTP API, καθιστώντας τη λύση πιο αυτόνομη και ανεξάρτητη από εξωτερικές υπηρεσίες. Η αξιοποίηση αρχείων .csv για τη μαζική εισαγωγή

παραληπτών, η χρήση ασφαλών τεχνικών απόκρυψης διαπιστευτηρίων και η επικύρωση της δομής των δεδομένων πριν την αποστολή ενίσχυσαν τη σταθερότητα και τη λειτουργική επάρκεια της εφαρμογής. Επιπλέον, η δυνατότητα λήψης επιβεβαίωσης από τον χρήστη πριν την τελική αποστολή των email προσέδωσε επιπρόσθετο επίπεδο ελέγχου και αξιοπιστίας στη διαδικασία.

Και οι δύο υλοποιήσεις παρέχουν τη δυνατότητα προσωποποίησης των μηνυμάτων με βάση τις πληροφορίες του παραλήπτη, διατηρώντας ρεαλισμό και ευελιξία στην παραγωγή περιεχομένου. Παράλληλα, επιτρέπουν την αποστολή τόσο σε μορφή απλού κειμένου όσο και με χρήση προτύπου HTML, διευρύνοντας τις επιλογές παρουσίασης και αυξάνοντας τη φαινομενική αυθεντικότητα των phishing μηνυμάτων.

Η εργασία αυτή προσφέρει σημαντικές προοπτικές για περαιτέρω εξέλιξη και εφαρμογή σε περιβάλλοντα ηθικής πειρατείας, εκπαιδευτικά σενάρια awareness training ή red-team testing, υπό τον αυστηρό περιορισμό της τήρησης ηθικών και νομικών προδιαγραφών. Η ευελιξία, η δυνατότητα επεκτασιμότητας και ο έλεγχος πρόσβασης καθιστούν τις υλοποιήσεις αυτές κατάλληλες για τη δημιουργία συστημάτων εκπαίδευσης και αξιολόγησης σε θέματα κυβερνοασφάλειας.

Συνολικά, η εργασία καταδεικνύει πως τα σύγχρονα εργαλεία τεχνητής νοημοσύνης, όταν συνδυαστούν με τεχνικές ασφαλούς ανάπτυξης και αυστηρής παραμετροποίησης, μπορούν να αποτελέσουν σημαντικά βοηθήματα στην προσπάθεια κατανόησης και αντιμετώπισης του φαινομένου του ηλεκτρονικού "ψαρέματος", ενισχύοντας την προληπτική εκπαίδευση και τον έγκαιρο εντοπισμό τέτοιων απειλών.

5.4 Μελλοντικοί Στόχοι

Η παρούσα εργασία έθεσε τα θεμέλια για την ανάπτυξη ενός προηγμένου και αυτοματοποιημένου συστήματος παραγωγής και αποστολής phishing email μέσω τεχνητής νοημοσύνης. Παρότι επιτεύχθηκαν υψηλού επιπέδου λειτουργικότητες, όπως η χρήση LLMs μέσω API, η δυναμική προσωποποίηση περιεχομένου και η υποστήριξη HTML μορφοποίησης, υπάρχουν σημαντικά περιθώρια περαιτέρω βελτίωσης, γενίκευσης και εμπλουτισμού της εφαρμογής.

Ένας πρώτος άξονας βελτίωσης αφορά το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου. Το υπάρχον dataset αποτελείται από 86 δείγματα phishing σεναρίων, τα οποία, αν και λειτουργικά, δεν καλύπτουν επαρκώς την ποικιλομορφία και πολυπλοκότητα που παρουσιάζουν σύγχρονες απειλές. Στο μέλλον, το σύστημα μπορεί να επεκταθεί με περισσότερα και πιο ποικίλα σενάρια, περιλαμβάνοντας τύπους επιθέσεων όπως spear phishing, smishing και business email compromise (BEC), καθώς και εφαρμογές από διαφορετικούς τομείς όπως τραπεζικά ιδρύματα, υπηρεσίες cloud και κοινωνικά δίκτυα. Παράλληλα, η ενσωμάτωση δεδομένων σε πολλαπλές γλώσσες, συμπεριλαμβανομένης της ελληνικής, θα επέτρεπε τη λειτουργία της εφαρμογής σε ευρύτερο γλωσσικό και πολιτισμικό πλαίσιο.

Παράλληλα, σημαντικό βήμα αποτελεί η ανάπτυξη μιας διεπαφής χρήστη. Επί του παρόντος, η χρήση του εργαλείου γίνεται αποκλειστικά μέσω της γραμμής εντολών, γεγονός που ενδέχεται να αποθαρρύνει λιγότερο έμπειρους χρήστες ή να δυσχεραίνει την καθημερινή χρήση του σε εκπαιδευτικά ή επιχειρησιακά περιβάλλοντα. Η υλοποίηση ενός γραφικού περιβάλλοντος, είτε ως web εφαρμογή με τεχνολογίες όπως React ή Flask, είτε ως desktop εφαρμογή με εργαλεία όπως PyQt, θα διευκολύνει την πρόσβαση και θα βελτιώσει τη συνολική εμπειρία χρήσης.

Ένας ακόμη τομέας που μπορεί να εξελιχθεί είναι η αυτόματη σύνθεση HTML προτύπων μέσω τεχνητής νοημοσύνης. Επί του παρόντος, χρησιμοποιούνται στατικά HTML templates για τη μορφοποίηση των phishing email. Μελλοντικά, η αξιοποίηση LLMs για την παραγωγή HTML περιεχομένου με βάση φυσική γλώσσα (prompt-to-HTML) θα επιτρέψει τη δημιουργία εξατομικευμένων προτύπων. Οι χρήστες θα μπορούν να καθορίσουν λεκτικά πώς επιθυμούν να φαίνεται το email, και το σύστημα θα παράγει τον αντίστοιχο HTML κώδικα με προσαρμοσμένα γραφικά, χρώματα, κουμπιά και άλλες διακοσμητικές ή λειτουργικές λεπτομέρειες.

Επιπρόσθετα, η ενσωμάτωση του συστήματος με εξωτερικά εργαλεία ή πλατφόρμες, όπως το Zapier, το Microsoft Outlook ή το Google Workspace API, μπορεί να προσδώσει μεγαλύτερη ευελιξία και αυτοματοποίηση στις δοκιμές. Μέσω τέτοιας διασύνδεσης, θα ήταν δυνατή η αποστολή phishing email απευθείας από πλατφόρμες παραγωγικού περιβάλλοντος, ενώ παράλληλα θα μπορούσε να υλοποιηθεί η καταγραφή στατιστικών όπως ποσοστά ανοίγματος και click σε συνδέσμους, συμβάλλοντας στην ανάλυση της αποτελεσματικότητας κάθε εκστρατείας.

Ένας ακόμα σημαντικός μελλοντικός στόχος είναι η υλοποίηση μηχανισμών απάντησης ή παρακολούθησης (response simulation). Με την προσθήκη συστημάτων που καταγράφουν ή προσομοιώνουν αντιδράσεις των παραληπτών, όπως το πάτημα ενός συνδέσμου ή η απάντηση σε email, μπορεί να επιτευχθεί μια ολοκληρωμένη εμπειρία phishing προσομοίωσης. Τέτοια δεδομένα μπορούν να συλλέγονται μέσω fake landing pages ή webhooks, προσφέροντας πολύτιμη πληροφορία για εκπαιδευτικούς σκοπούς ή αξιολόγηση awareness χωρίς να τίθεται πραγματικός κίνδυνος.

Σε συνέχεια της δεύτερης μεθόδου, η οποία κάνει χρήση τοπικού LLM μέσω του Ollama API, θα μπορούσε να εξεταστεί η πλήρης ενσωμάτωση on-device μοντέλων τεχνητής νοημοσύνης που δεν απαιτούν σύνδεση με εξωτερικές υπηρεσίες. Μοντέλα όπως τα Mistral ή Phi-2 θα μπορούσαν να εκπαιδευτούν εκ νέου ή να προσαρμοστούν περαιτέρω σε εξειδικευμένα phishing datasets. Αυτή η προσέγγιση θα προσέφερε μεγαλύτερη ιδιωτικότητα, ταχύτητα και ανεξαρτησία από το διαδίκτυο.

Επιπλέον, μπορεί να αναπτυχθεί ένα πιο εξελιγμένο σύστημα ασφαλείας και διαχείρισης πρόσβασης. Η υφιστάμενη πιστοποίηση API μέσω x-api-key μπορεί να εμπλουτιστεί με μηχανισμούς όπως JWT authentication, dashboards για παρακολούθηση της χρήσης και διαχείριση API keys, καθώς και ρόλους χρηστών με διαβαθμισμένα δικαιώματα. Αυτή η προσέγγιση θα καταστήσει την εφαρμογή περισσότερο κατάλληλη για χρήση από πολλαπλούς χρήστες, ιδίως σε εκπαιδευτικά ιδρύματα ή εταιρικά περιβάλλοντα.

Τέλος, μια εξαιρετικά χρήσιμη προσθήκη θα ήταν η δημιουργία ενότητας για αξιολόγηση του phishing awareness. Η ενότητα αυτή θα μπορούσε να περιλαμβάνει σενάρια με ψευδή emails, στα οποία ο χρήστης καλείται να απαντήσει ή να τα αναγνωρίσει, με το σύστημα να αξιολογεί την ανταπόκριση και να παρέχει ανατροφοδότηση. Μια τέτοια λειτουργικότητα θα μπορούσε να υποστηρίξει εκπαιδευτικά προγράμματα, αυξάνοντας το επίπεδο ευαισθητοποίησης απέναντι σε απειλές κοινωνικής μηχανικής.

Με την υλοποίηση των παραπάνω στόχων, η παρούσα εργασία μπορεί να εξελιχθεί σε ένα πλήρως λειτουργικό και εκπαιδευτικό εργαλείο προσομοίωσης ηλεκτρονικών επιθέσεων με χρήση Τεχνητής Νοημοσύνης, το οποίο θα μπορεί να αξιοποιηθεί είτε στην εκπαίδευση προσωπικού είτε σε δοκιμές ασφάλειας σε εταιρικά ή ερευνητικά περιβάλλοντα.

ΑΝΑΦΟΡΕΣ

[1] I. Vayansky and S. Kumar, “Phishing – challenges and solutions,” *Comput. Fraud Secur.*, vol. 2018, no. 1, pp. 15–20, Jan. 2018, doi: 10.1016/S1361-3723(18)30007-1.

[2] A. N. Shaikh, A. M. Shabut, and M. A. Hossain, “A literature review on phishing crime, prevention review and investigation of gaps,” in *2016 10th International Conference on Software, Knowledge, Information Management & Applications (SKIMA)*, Chengdu, China: IEEE, 2016, pp. 9–15. doi: 10.1109/SKIMA.2016.7916190.

[3] S. Kierkegaard, “Swallowing the Bait, Hook, Line, and Sinker: Phishing, Pharming, and Now Rat-ing!,” in *Managing Information Assurance in Financial Services:*, vol. 11, H. R. Rao, M. Gupta, and S. J. Upadhyaya, Eds., IGI Global, 2007, p. 20. doi: 10.4018/978-1-59904-171-1.ch011.

[4] G. Sonowal, “Introduction to Phishing,” in *Phishing and Communication Channels*, Berkeley, CA: Apress, 2022, pp. 1–24. doi: 10.1007/978-1-4842-7744-7_1.

[5] “The Phishing Guide - Understanding & Preventing Phishing Attacks - (IBM Internet Security Systems) | PDF | Phishing | Proxy Server,” Scribd. Accessed: Jun. 17, 2025. [Online]. Available: https://www.scribd.com/document/219802442/The-Phishing-Guide-Understanding-Preventing-Phishing-Attacks-IBM-Internet-Security-Systems

[6] Y. Zhang, S. Egelman, L. Cranor, and J. Hong, “Phinding Phish: Evaluating Anti-Phishing Tools”.

[7] J. Hong, “The state of phishing attacks,” *Commun. ACM*, vol. 55, no. 1, pp. 74–81, Jan. 2012, doi: 10.1145/2063176.2063197.

[8] L. Burita, P. Matoulek, K. Halouzka, P. Kozak, and Department of Informatics and Cyber Operations, University of Defence, 65 Kounicova Street, 66210 Brno, Czech Republic, “Analysis of phishing emails,” *AIMS Electron. Electr. Eng.*, vol. 5, no. 1, pp. 93–116, 2021, doi: 10.3934/electreng.2021006.

[9] J. Zaaiman and L. Leenan, *ICCWS 2015 10th International Conference on Cyber Warfare and Security: ICCWS 2015*. in Proceedings of the International Conference on Information Warfare and Security. Academic Conferences and Publishing Limited, 2015. [Online]. Available: https://books.google.gr/books?id=piikBwAAQBAJ

[10] “19 Types of Phishing Attacks with Examples,” Fortinet. Accessed: Sep. 20, 2025. [Online]. Available: https://www.fortinet.com/resources/cyberglossary/types-of-phishing-attacks

[11] S. Bhbosale, V. Pujari, and Z. Multani, “Advantages And Disadvantages Of Artificial Intellegence.” Accessed: Jun. 17, 2025. [Online]. Available: https://www.researchgate.net/profile/Vinayak-Pujari-2/publication/344584269_Advantages_And_Disadvantages_Of_Artificial_Intellegence/links/5f81b70192851c14bcbc1d96/Advantages-And-Disadvantages-Of-Artificial-Intellegence.pdf

[12] B. G. Buchanan, “A (Very) Brief History of Artificial Intelligence”.

[13] H. Reese, “Understanding the differences between AI, machine learning, and deep learning,” TechRepublic. Accessed: Jun. 14, 2025. [Online]. Available: https://www.techrepublic.com/article/understanding-the-differences-between-ai-machine-learning-and-deep-learning/

[14] P. Ashlyn S, “Artificial Intelligence and its Increasing Importance,” in *LEARNING OUTCOMES OF CLASSROOM RESEARCH*, L ORDINE NUOVO PUBLICATION, 2021. Accessed: Jun. 17, 2025. [Online]. Available: https://www.researchgate.net/publication/358058444_Artificial_Intelligence_and_its_Increasing_Importance

[15] L. Chen, P. Chen, and Z. Lin, “Artificial Intelligence in Education: A Review,” *IEEE Access*, vol. 8, pp. 75264–75278, 2020, doi: 10.1109/ACCESS.2020.2988510.

[16] Artificial Intelligence and Advanced Computing Committee, Technical Activities Division Standing Committees, Technical Activities Division, Transportation Research Board, and National Academies of Sciences, Engineering, and Medicine, *Artificial Intelligence Applications to Critical Transportation Issues*. Washington, D.C.: Transportation Research Board, 2012, p. 22690. doi: 10.17226/22690.

[17] D. J. Crandall, “Artificial Intelligence and Manufacturing,” in *Smart Factories: Issues of Information Governance*, 2019.

[18] M. Schmitt and I. Flechais, “Digital deception: generative artificial intelligence in social engineering and phishing,” *Artif. Intell. Rev.*, vol. 57, no. 12, p. 324, Oct. 2024, doi: 10.1007/s10462-024-10973-2.

[19] A. Basit, M. Zafar, X. Liu, A. R. Javed, Z. Jalil, and K. Kifayat, “A comprehensive survey of AI-enabled phishing attacks detection techniques,” *Telecommun. Syst.*, vol. 76, no. 1, pp. 139–154, Jan. 2021, doi: 10.1007/s11235-020-00733-2.

[20] M. Sameen, K. Han, and S. O. Hwang, “PhishHaven—An Efficient Real-Time AI Phishing URLs Detection System,” *IEEE Access*, vol. 8, pp. 83425–83443, 2020, doi: 10.1109/ACCESS.2020.2991403.
[21] A. C. Bahnsen, I. Torroledo, L. D. Camacho, and S. Villegas, “DeepPhish: Simulating Malicious AI”.
[22] T. B. Brown *et al.*, “Language Models are Few-Shot Learners,” Jul. 22, 2020, *arXiv*: arXiv:2005.14165. doi: 10.48550/arXiv.2005.14165.
[23] J. Hazell, “LARGE LANGUAGE MODELS CAN BE USED TO EFFECTIVELY SCALE SPEAR PHISHING CAMPAIGNS”.
[24] D. Bommasani, J. Yang, and Y. Pan, “Artificial Intelligence in cybersecurity,” in *Journal of Network and Computer Applications*, 2021.
[25] A. Vaswani *et al.*, “Attention Is All You Need,” Aug. 02, 2023, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.
[26] Y. Zhang *et al.*, “DialoGPT: Large-Scale Generative Pre-training for Conversational Response Generation,” May 02, 2020, *arXiv*: arXiv:1911.00536. doi: 10.48550/arXiv.1911.00536.
[27] K. Liu, “Truthfulness in Large Language Models”.
[28] OpenAI *et al.*, “GPT-4 Technical Report,” Mar. 04, 2024, *arXiv*: arXiv:2303.08774. doi: 10.48550/arXiv.2303.08774.
[29] S. Zhang *et al.*, “OPT: Open Pre-trained Transformer Language Models”.
[30] L. Khanum, “Phishing emails analysis and design automatic text generation tool for writing and sending phishing mail.”.
[31] J. Seymour and P. Tully, “Weaponizing data science for social engineering: Automated E2E spear phishing on Twitter”.
[32] A. M. Shibli, M. M. A. Pritom, and M. Gupta, “AbuseGPT: Abuse of Generative AI ChatBots to Create Smishing Campaigns,” Feb. 15, 2024, *arXiv*: arXiv:2402.09728. doi: 10.48550/arXiv.2402.09728.
[33] Y. Liu *et al.*, “Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study,” Mar. 10, 2024, *arXiv*: arXiv:2305.13860. doi: 10.48550/arXiv.2305.13860.
[34] G. Deng *et al.*, “MasterKey: Automated Jailbreak Across Multiple Large Language Model Chatbots,” in *Proceedings 2024 Network and Distributed System Security Symposium*, 2024. doi: 10.14722/ndss.2024.24188.
[35] M. Gupta, C. Akiri, K. Aryal, E. Parker, and L. Praharaj, “From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy,” *IEEE Access*, vol. 11, pp. 80218–80245, 2023, doi: 10.1109/ACCESS.2023.3300381.
[36] N. Begou, J. Vinoy, A. Duda, and M. Korczyński, “Exploring the Dark Side of AI: Advanced Phishing Attack Design and Deployment Using ChatGPT,” in *2023 IEEE Conference on Communications and Network Security (CNS)*, Orlando, FL, USA: IEEE, Oct. 2023, pp. 1–6. doi: 10.1109/CNS59707.2023.10288940.
[37] K. Arai, Ed., *Proceedings of the Future Technologies Conference (FTC) 2023, Volume 1*, vol. 813. in Lecture Notes in Networks and Systems, vol. 813. Cham: Springer Nature Switzerland, 2023. doi: 10.1007/978-3-031-47454-5.
[38] S.-W. Guo, T.-C. Chen, H.-J. Wang, L. Fang-Yie, and F. Yao-Chung, “Generating Personalized Phishing Emails for Social Engineering Training Based on Neural Language Models,” in *Advances on Broad-Band Wireless Computing, Communication and Applications: Proceedings of the 17th International Conference on Broad-Band Wireless Computing, Communication and Applications (BWCCA-2022)*, vol. 570, L. Barolli, Ed., in Lecture Notes in Networks and Systems, vol. 570. , Cham: Springer International Publishing, 2023, pp. 270–281. doi: 10.1007/978-3-031-20029-8.
[39] F. Salahdine and N. Kaabouch, “Social Engineering Attacks: A Survey,” *Future Internet*, vol. 11, no. 4, p. 89, Apr. 2019, doi: 10.3390/fi11040089.
[40] K. Krombholz, H. Hobel, M. Huber, and E. Weippl, “Advanced social engineering attacks,” *J. Inf. Secur. Appl.*, vol. 22, pp. 113–122, Jun. 2015, doi: 10.1016/j.jisa.2014.09.005.
[41] T. Qin and J. K. Burgoon, “An Investigation of Heuristics of Human Judgment in Detecting Deception and Potential Implications in Countering Social Engineering,” in *2007 IEEE Intelligence and Security Informatics*, New Brunswick, NJ: IEEE, May 2007, pp. 152–159. doi: 10.1109/ISI.2007.379548.
[42] T. Kanade, *Intelligence and Security Informatics: Second Symposium on Intelligence and Security Informatics, ISI 2004, Tucson, AZ, USA, June 10-11, 2004. Proceedings*, 1st ed. Berlin/Heidelberg: Springer Berlin Heidelberg, 2004.
[43] K. D. Mitnick, “THE ART OF DECEPTION”.
[44] R. Kalniņš, J. Puriņš, and G. Alksnis, “Security Evaluation of Wireless Network Access Points,” *Appl. Comput. Syst.*, vol. 21, no. 1, pp. 38–45, May 2017, doi: 10.1515/acss-2017-0005.
[45] N. N. Pokrovskaia and S. O. Snisarenko, “Social engineering and digital technologies for the security of the social capital’ development,” in *2017 International Conference “Quality Management,Transport and Information Security, Information Technologies” (IT&QM&IS)*, Saint Petersburg, Russia: IEEE, Sep. 2017, pp. 16–18. doi: 10.1109/ITMQIS.2017.8085750.

[46] A. M. Aroyo, F. Rea, G. Sandini, and A. Sciutti, “Trust and Social Engineering in Human Robot Interaction: Will a Robot Make You Disclose Sensitive Information, Conform to Its Recommendations or Gamble?,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3701–3708, Oct. 2018, doi: 10.1109/LRA.2018.2856272.
[47] European Union Agency for Cybersecurity., *ENISA threat landscape: finance sector : January 2023 to June 2024.* LU: Publications Office, 2024. Accessed: Sep. 20, 2025. [Online]. Available: https://data.europa.eu/doi/10.2824/5410466
[48] “How much does a cyberattack cost companies? | Open Data Security,” ODS - Cybersecurity. Accessed: Jun. 14, 2025. [Online]. Available: https://opendatasecurity.co.uk/how-much-does-a-cyberattack-cost-companies/
[49] United States Government Accountability Office., “Data protection Actions taken by Equifax and federal agencies in response to the 2017 breach.” Aug. 30, 2018.
[50] M. A. Chargo, “YOU’VE BEEN HACKED: HOW TO BETTER INCENTIVIZE CORPORATIONS TO PROTECT CONSUMERS’ DATA,” vol. 20.
[51] M. Libicki, “Could the Issue of DPRK Hacking Benefit from Benign Neglect?,” *Georget. J. Int. Aff.*, vol. 19, no. 1, pp. 83–89, 2018, doi: 10.1353/gia.2018.0010.
[52] C. Atwell, T. Blasi, and T. Hayajneh, “Reverse TCP and Social Engineering Attacks in the Era of Big Data,” in *2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*, New York, NY, USA: IEEE, Apr. 2016, pp. 90–95. doi: 10.1109/BigDataSecurity-HPSC-IDS.2016.60.
[53] T. Mahmood and U. Afzal, “Security Analytics: Big Data Analytics for cybersecurity: A review of trends, techniques and tools,” in *2013 2nd National Conference on Information Assurance (NCIA)*, Rawalpindi, Pakistan: IEEE, Dec. 2013, pp. 129–134. doi: 10.1109/NCIA.2013.6725337.
[54] P. Dutta, S. Chakrabarti, A. Bhattacharya, S. Dutta, and V. Piuri, Eds., *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2022, Volume 1*, vol. 491. in Lecture Notes in Networks and Systems, vol. 491. Singapore: Springer Nature Singapore, 2023. doi: 10.1007/978-981-19-4193-1.
[55] P. L. Gallegos-Segovia, J. F. Bravo-Torres, V. M. Larios-Rosillo, P. E. Vintimilla-Tapia, I. F. Yuquilima-Albarado, and J. D. Jara-Saltos, “Social engineering as an attack vector for ransomware,” in *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Pucon: IEEE, Oct. 2017, pp. 1–6. doi: 10.1109/CHILECON.2017.8229528.
[56] L. Xiangyu, L. Qiuyang, and S. Chandel, “Social Engineering and Insider Threats,” in *2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, Nanjing: IEEE, Oct. 2017, pp. 25–34. doi: 10.1109/CyberC.2017.91.
[57] A. Koyun and E. A. Janabi, “Social Engineering Attacks,” vol. 4, no. 6, 2017.
[58] N. Anjum and M. R. Chowdhury, “International Journal of Advanced Research in Computer and Communication Engineering,” *SSRN Electron. J.*, 2024, doi: 10.2139/ssrn.4847308.
[59] S. Gupta, A. Singhal, and A. Kapoor, “A literature survey on social engineering attacks: Phishing attack,” in *2016 International Conference on Computing, Communication and Automation (ICCCA)*, Greater Noida, India: IEEE, Apr. 2016, pp. 537–540. doi: 10.1109/CCAA.2016.7813778.
[60] Ekta and U. Bansal, “A Review on Ransomware Attack,” in *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*, Jalandhar, India: IEEE, May 2021, pp. 221–226. doi: 10.1109/ICSCCC51823.2021.9478148.
[61] J. Long, S. Pinzon, J. Wiles, and K. D. Mitnick, “Vehicle Surveillance,” in *No Tech Hacking*, Elsevier, 2008, pp. 245–257. doi: 10.1016/B978-1-59749-215-7.00010-X.
[62] P. De Ryck, N. Nikiforakis, L. Desmet, and W. Joosen, “TabShots: client-side detection of tabnabbing attacks,” in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*, Hangzhou China: ACM, May 2013, pp. 447–456. doi: 10.1145/2484313.2484371.
[63] R. K. Suri, D. S. Tomar, and D. R. Sahu, “An Approach to Perceive Tabnabbing Attack,” vol. 1, no. 6, 2012.
[64] B. Arya and K. Chandrasekaran, “A client-side anti-pharming (CSAP) approach,” in *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, Nagercoil, India: IEEE, Mar. 2016, pp. 1–6. doi: 10.1109/ICCPCT.2016.7530353.
[65] “What Is Pretexting? Definition, Examples and Attacks,” Fortinet. Accessed: Sep. 20, 2025. [Online]. Available: https://www.fortinet.com/resources/cyberglossary/pretexting
[66] H. Tu, A. Doupe, Z. Zhao, and G.-J. Ahn, “SoK: Everyone Hates Robocalls: A Survey of Techniques Against Telephone Spam,” in *2016 IEEE Symposium on Security and Privacy (SP)*, San Jose, CA: IEEE, May 2016, pp. 320–338. doi: 10.1109/SP.2016.27.

[67] R. R. R. Barbosa, R. Sadre, and A. Pras, “Flow whitelisting in SCADA networks,” *Int. J. Crit. Infrastruct. Prot.*, vol. 6, no. 3–4, pp. 150–158, Dec. 2013, doi: 10.1016/j.ijcip.2013.08.003.
[68] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, “Recent Advances in Recurrent Neural Networks,” Feb. 22, 2018, *arXiv*: arXiv:1801.01078. doi: 10.48550/arXiv.1801.01078.
[69] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.
[70] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006, doi: 10.1162/neco.2006.18.7.1527.
[71] Y. Bengio, N. Boulanger-Lewandowski, and R. Pascanu, “Advances in optimizing recurrent networks,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada: IEEE, May 2013, pp. 8624–8628. doi: 10.1109/ICASSP.2013.6639349.
[72] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994, doi: 10.1109/72.279181.
[73] T. Mikolov, A. Joulin, S. Chopra, M. Mathieu, and M. Ranzato, “Learning Longer Memory in Recurrent Neural Networks,” Apr. 16, 2015, *arXiv*: arXiv:1412.7753. doi: 10.48550/arXiv.1412.7753.
[74] I. Sutskever, J. Martens, and G. Hinton, “Generating Text with Recurrent Neural Networks”.
[75] S. S. Haykin, *Neural networks : a comprehensive foundation*. Upper Saddle River, N.J. : Prentice Hall, 1999. Accessed: Jun. 17, 2025. [Online]. Available: http://archive.org/details/neuralnetworksco0000hayk_2ed
[76] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Interspeech 2010*, ISCA, Sep. 2010, pp. 1045–1048. doi: 10.21437/Interspeech.2010-343.
[77] S. Larabi-Marie-Sainte, S. Ghouzali, T. Saba, L. Aburahmah, and R. Almohaini, “Improving spam email detection using deep recurrent neural network,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 25, no. 3, p. 1625, Mar. 2022, doi: 10.11591/ijeecs.v25.i3.pp1625-1633.
[78] G. Van Houdt, C. Mosquera, and G. Nápoles, “A review on the long short-term memory model,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5929–5955, Dec. 2020, doi: 10.1007/s10462-020-09838-1.
[79] A. Rodan, H. Faris, and J. Alqatawna, “Optimizing Feedforward Neural Networks Using Biogeography Based Optimization for E-Mail Spam Identification,” *Int. J. Commun. Netw. Syst. Sci.*, vol. 09, no. 01, pp. 19–28, 2016, doi: 10.4236/ijcns.2016.91002.
[80] M. Al-Kabi, H. Wahsheh, I. Alsmadi, E. Al-Shawakfa, A. Wahbeh, and A. Al-Hmoud, “Content-based analysis to detect Arabic web spam,” *J. Inf. Sci.*, vol. 38, no. 3, pp. 284–296, Jun. 2012, doi: 10.1177/0165551512439173.
[81] T. O’Reilly, “Web 2.0: Compact Definition.”
[82] P. Hayati and V. Potdar, “Toward spam 2.0: An evaluation of Web 2.0 anti-spam methods,” in *2009 7th IEEE International Conference on Industrial Informatics*, Cardiff, Wales, UK: IEEE, Jun. 2009, pp. 875–880. doi: 10.1109/INDIN.2009.5195918.
[83] R. Cooley, B. Mobasher, and J. Srivastava, “Web mining: information and pattern discovery on the World Wide Web,” in *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*, Newport Beach, CA, USA: IEEE Comput. Soc, 1997, pp. 558–567. doi: 10.1109/TAI.1997.632303.
[84] P. Hayati, V. Potdar, A. Talevski, N. Firoozeh, S. Sarenche, and E. A. Yeganeh, “Definition of spam 2.0: New spamming boom,” in *4th IEEE International Conference on Digital Ecosystems and Technologies*, Dubai, United Arab Emirates: IEEE, Apr. 2010, pp. 580–584. doi: 10.1109/DEST.2010.5610590.
[85] P. Hayati, K. Chai, V. Potdar, and A. Talevski, “HoneySpam 2.0: Profiling Web Spambot Behaviour,” in *Principles of Practice in Multi-Agent Systems*, vol. 5925, J.-J. Yang, M. Yokoo, T. Ito, Z. Jin, and P. Scerri, Eds., in Lecture Notes in Computer Science, vol. 5925. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 335–344. doi: 10.1007/978-3-642-11161-7_23.
[86] S. Webb, J. Caverlee, and C. Pu, “Social Honeypots: Making Friends With A Spammer Near You”.
[87] P. Hayati, V. Potdar, K. Chai, and A. Talevski, “Web Spambot Detection Based on Web Navigation Behaviour,” in *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, Perth, Australia: IEEE, 2010, pp. 797–803. doi: 10.1109/AINA.2010.92.
[88] P. Hayati and V. Potdar, “Evaluation of spam detection and prevention frameworks for email and image spam: a state of art,” in *Proceedings of the 10th International Conference on Information Integration and Web-based Applications & Services*, Linz Austria: ACM, Nov. 2008, pp. 520–527. doi: 10.1145/1497308.1497402.
[89] Q. Gan and T. Suel, “Improving web spam classifiers using link structure,” in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, Banff Alberta Canada: ACM, May 2007, pp. 17–20. doi: 10.1145/1244408.1244412.
[90] Z. Gyongyi, H. Garcia-Molina, S. Univ, and J. Pedersen, “Combating Web Spam with TrustRank”.

[91] S. K. Singh, S. Kumar, and P. S. Mehra, “Chat GPT & Google Bard AI: A Review,” in *2023 International Conference on IoT, Communication and Automation Technology (ICICAT)*, Gorakhpur, India: IEEE, Jun. 2023, pp. 1–6. doi: 10.1109/ICICAT57735.2023.10263706.
[92] “About.” Accessed: Jun. 14, 2025. [Online]. Available: https://openai.com/about/
[93] G. Brockman *et al.*, “OpenAI Gym,” Jun. 05, 2016, *arXiv*: arXiv:1606.01540. doi: 10.48550/arXiv.1606.01540.
[94] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” May 24, 2019, *arXiv*: arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805.
[95] G. Marcus, E. Davis, and S. Aaronson, “A very preliminary analysis of DALL-E 2”.
[96] G. Goh *et al.*, “Multimodal Neurons in Artificial Neural Networks,” *Distill*, vol. 6, no. 3, p. e30, Mar. 2021, doi: 10.23915/distill.00030.
[97] S. M. 7 min read, “ChatGPT gained 1 million users in under a week. Here’s why the AI chatbot is primed to disrupt search as we know it,” Yahoo Finance. Accessed: Jun. 14, 2025. [Online]. Available: https://finance.yahoo.com/news/chatgpt-gained-1-million-followers-224523258.html
[98] R. Dale, “GPT-3: What’s it good for?,” *Nat. Lang. Eng.*, vol. 27, no. 1, pp. 113–118, Jan. 2021, doi: 10.1017/S1351324920000601.
[99] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training”.
[100] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, “Why Does Unsupervised Pre-training Help Deep Learning?”.
[101] P. Budzianowski and I. Vulić, “Hello, It’s GPT-2 -- How Can I Help You? Towards the Use of Pretrained Language Models for Task-Oriented Dialogue Systems,” Aug. 04, 2019, *arXiv*: arXiv:1907.05774. doi: 10.48550/arXiv.1907.05774.
[102] L. Floridi and M. Chiriatti, “GPT-3: Its Nature, Scope, Limits, and Consequences,” *Minds Mach.*, vol. 30, no. 4, pp. 681–694, Dec. 2020, doi: 10.1007/s11023-020-09548-1.
[103] A. R. Kirmani, “Artificial Intelligence-Enabled Science Poetry,” *ACS Energy Lett.*, vol. 8, no. 1, pp. 574–576, Jan. 2023, doi: 10.1021/acsenergylett.2c02758.
[104] B. Lund, T. Wang, N. R. Mannuru, B. Nie, S. Shimray, and Z. Wang, “ChatGPT and a New Academic Reality: Artificial Intelligence-Written Research Papers and the Ethics of the Large Language Models in Scholarly Publishing,” *J. Assoc. Inf. Sci. Technol.*, vol. 74, no. 5, pp. 570–581, May 2023, doi: 10.1002/asi.24750.
[105] E. Strubell, A. Ganesh, and A. McCallum, “Energy and Policy Considerations for Deep Learning in NLP,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy: Association for Computational Linguistics, 2019, pp. 3645–3650. doi: 10.18653/v1/P19-1355.
[106] D. Gkatzia, Association for Computational Linguistics, and Association for Computational Linguistics, Eds., *The 16th Conference of the European Chapter of the Association for Computational Linguistics - proceedings of the System Demonstrations: April 19-23, 2021: EACL 2021*. Stroudsburg, PA: Association for Computational Linguistics (ACL), 2021.
[107] “OpenAI Used Kenyan Workers on Less Than $2 Per Hour: Exclusive | TIME.” Accessed: Jun. 14, 2025. [Online]. Available: https://time.com/6247678/openai-chatgpt-kenya-workers/
[108] R. Dale, “NLP in a post-truth world,” *Nat. Lang. Eng.*, vol. 23, no. 2, pp. 319–324, Mar. 2017, doi: 10.1017/S1351324917000018.
[109] L. Lucy and D. Bamman, “Gender and Representation Bias in GPT-3 Generated Stories,” in *Proceedings of the Third Workshop on Narrative Understanding*, Virtual: Association for Computational Linguistics, 2021, pp. 48–55. doi: 10.18653/v1/2021.nuse-1.5.
[110] D. H. Nguyen and K. N. Truong, “PHEmail: Designing a Privacy Honoring Email System,” *NEW Horiz.*.
[111] R. E. Kraut and S. Sunder, “MARKETS FOR ATTENTION: WILL POSTAGE FOR EMAIL HELP?”.
[112] L. Lessig, *Code and other laws of cyberspace*, Nachdr. New York: Basic Books, 2000.
[113] *dionach/PhEmail*. (Mar. 24, 2025). Python. Dionach. Accessed: Jun. 14, 2025. [Online]. Available: https://github.com/dionach/PhEmail
[114] “(PDF) Faceted Id/Entity : managing representation in a digital world,” ResearchGate. Accessed: Jun. 14, 2025. [Online]. Available: https://www.researchgate.net/publication/279824514_Faceted_IdEntity_managing_representation_in_a_digital_world
[115] S. Whittaker, Q. Jones, and L. Terveen, “Contact management: identifying contacts to support long-term communication,” in *Proceedings of the 2002 ACM conference on Computer supported cooperative work*, New Orleans Louisiana USA: ACM, Nov. 2002, pp. 216–225. doi: 10.1145/587078.587109.

[116] W. X. Zhao *et al.*, “A Survey of Large Language Models,” Mar. 11, 2025, *arXiv*: arXiv:2303.18223. doi: 10.48550/arXiv.2303.18223.
[117] P. Gao *et al.*, “LLaMA-Adapter V2: Parameter-Efficient Visual Instruction Model,” Apr. 28, 2023, *arXiv*: arXiv:2304.15010. doi: 10.48550/arXiv.2304.15010.
[118] J. Kaplan *et al.*, “Scaling Laws for Neural Language Models,” Jan. 23, 2020, *arXiv*: arXiv:2001.08361. doi: 10.48550/arXiv.2001.08361.
[119] A. Chowdhery *et al.*, “PaLM: Scaling Language Modeling with Pathways,” Oct. 05, 2022, *arXiv*: arXiv:2204.02311. doi: 10.48550/arXiv.2204.02311.
[120] J. W. Rae *et al.*, “Scaling Language Models: Methods, Analysis & Insights from Training Gopher,” Jan. 21, 2022, *arXiv*: arXiv:2112.11446. doi: 10.48550/arXiv.2112.11446.
[121] J. Hoffmann *et al.*, “Training Compute-Optimal Large Language Models,” Mar. 29, 2022, *arXiv*: arXiv:2203.15556. doi: 10.48550/arXiv.2203.15556.
[122] H. Touvron *et al.*, “LLaMA: Open and Efficient Foundation Language Models,” Feb. 27, 2023, *arXiv*: arXiv:2302.13971. doi: 10.48550/arXiv.2302.13971.
[123] C. Xu, D. Guo, N. Duan, and J. McAuley, “Baize: An Open-Source Chat Model with Parameter-Efficient Tuning on Self-Chat Data,” Dec. 02, 2023, *arXiv*: arXiv:2304.01196. doi: 10.48550/arXiv.2304.01196.
[124] B. Peng, C. Li, P. He, M. Galley, and J. Gao, “Instruction Tuning with GPT-4,” Apr. 06, 2023, *arXiv*: arXiv:2304.03277. doi: 10.48550/arXiv.2304.03277.
[125] R. Gupta and X. G. Song Arnav Gudibande, Hao Liu, Eric Wallace, Pieter Abbeel, Sergey Levine, Dawn, “Koala: A Dialogue Model for Academic Research,” The Berkeley Artificial Intelligence Research Blog. Accessed: Jun. 17, 2025. [Online]. Available: http://bair.berkeley.edu/blog/2023/04/03/koala/
[126] *tatsu-lab/stanford_alpaca*. (Jun. 16, 2025). Python. Tatsu’s shared repositories. Accessed: Jun. 17, 2025. [Online]. Available: https://github.com/tatsu-lab/stanford_alpaca
[127] “Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality | LMSYS Org.” Accessed: Jun. 17, 2025. [Online]. Available: https://lmsys.org/blog/2023-03-30-vicuna
[128] L. Ouyang *et al.*, “Training language models to follow instructions with human feedback,” Mar. 04, 2022, *arXiv*: arXiv:2203.02155. doi: 10.48550/arXiv.2203.02155.
[129] “ShareGPT: Share your wildest ChatGPT conversations with one click.” Accessed: Jun. 17, 2025. [Online]. Available: https://sharegpt.com
[130] R. Zhang *et al.*, “LLaMA-Adapter: Efficient Fine-tuning of Language Models with Zero-init Attention,” Sep. 18, 2024, *arXiv*: arXiv:2303.16199. doi: 10.48550/arXiv.2303.16199.
[131] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, “MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models,” Oct. 02, 2023, *arXiv*: arXiv:2304.10592. doi: 10.48550/arXiv.2304.10592.
[132] H. Liu, C. Li, Q. Wu, and Y. J. Lee, “Visual Instruction Tuning,” Dec. 11, 2023, *arXiv*: arXiv:2304.08485. doi: 10.48550/arXiv.2304.08485.
[133] “Welcome | Unsloth Documentation.” Accessed: Jun. 14, 2025. [Online]. Available: https://docs.unsloth.ai
[134] “Better language models and their implications.” Accessed: Jun. 26, 2025. [Online]. Available: https://openai.com/index/better-language-models/
[135] J. Howard and S. Ruder, “Universal Language Model Fine-tuning for Text Classification,” May 23, 2018, *arXiv*: arXiv:1801.06146. doi: 10.48550/arXiv.1801.06146.
[136] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “QLoRA: Efficient Finetuning of Quantized LLMs,” May 23, 2023, *arXiv*: arXiv:2305.14314. doi: 10.48550/arXiv.2305.14314.
[137] *unslothai/unsloth*. (Jun. 26, 2025). Python. Unsloth AI. Accessed: Jun. 26, 2025. [Online]. Available: https://github.com/unslothai/unsloth
[138] H. Touvron *et al.*, “Llama 2: Open Foundation and Fine-Tuned Chat Models,” Jul. 19, 2023, *arXiv*: arXiv:2307.09288. doi: 10.48550/arXiv.2307.09288.
[139] “Ollama.” Accessed: Jun. 26, 2025. [Online]. Available: https://ollama.com
[140] “Google Colab.” Accessed: Jun. 26, 2025. [Online]. Available: https://colab.research.google.com/
[141] “gawyria/pers-com-email-dataset · Datasets at Hugging Face.” Accessed: Jul. 22, 2025. [Online]. Available: https://huggingface.co/datasets/gawyria/pers-com-email-dataset
[142] “Tutorial: How to Finetune Llama-3 and Use In Ollama | Unsloth Documentation.” Accessed: Jun. 17, 2025. [Online]. Available: https://docs.unsloth.ai/basics/tutorial-how-to-finetune-llama-3-and-use-in-ollama
[143] R. Roni, *rokibulroni/Phishing-Email-Dataset*. (Feb. 02, 2025). Accessed: Jun. 17, 2025. [Online]. Available: https://github.com/rokibulroni/Phishing-Email-Dataset
[144] “Phishing validation emails dataset,” University of Twente Research Information. Accessed: Jun. 17, 2025. [Online]. Available: https://research.utwente.nl/en/datasets/phishing-validation-emails-dataset

[145] “gawyria/pers-com-email-model · Hugging Face.” Accessed: Jul. 22, 2025. [Online]. Available: https://huggingface.co/gawyria/pers-com-email-model
[146] “What Is Phishing? Examples and Phishing Quiz,” Cisco. Accessed: Jul. 22, 2025. [Online]. Available: https://www.cisco.com/site/us/en/learn/topics/security/what-is-phishing.html
[147] “Free URL Shortener with Analytics & Link Tracking | Linkly,” Linkly. Accessed: Jul. 22, 2025. [Online]. Available: https://linklyhq.com
[148] “smtplib — SMTP protocol client,” Python documentation. Accessed: Jul. 22, 2025. [Online]. Available: https://docs.python.org/3/library/smtplib.html
[149] “gawyria/demo-model.” Accessed: Jul. 22, 2025. [Online]. Available: https://ollama.com/gawyria/demo-model

ΠΑΡΑΡΤΗΜΑ Ι

Οδηγίες για τη δημιουργία αυθεντικοποιημένου λογαριασμού Google

1. Δημιουργήστε έναν λογαριασμό Google που θέλετε να είναι ο αποστολέας των μηνυμάτων.
2. Ανακατευθυνθείτε στις ρυθμίσεις ασφαλείας.
3. Ενεργοποιήστε τη λειτουργία "Επαλήθευση σε δύο βήματα".
4. Δημιουργήστε και αποθηκεύστε έναν App Password για να χρησιμοποιείται κατά την λειτουργία του προγράμματος (χωρίς κενά).

Οδηγίες εγκατάστασεις για την υλοποίηση σε τοπικό περιβάλλον

1. Κατευθυνθείτε στο GitHub και κατεβάστε τα αρχεία AIPPC-local.py και email_template.html. Σε περίπτωση που δεν έχετε το δικό σας .csv αρχείο έτοιμο, μπορείτε να κατεβάσετε το contacts.csv πρότυπο που παρέχεται στο GitHub και να προσθέσετε τις δικές σας εγγραφές.
2. Τοποθετείστε τα αρχεία .py και .html στον ίδιο φάκελο.
3. Εγκαταστείστε το Ollama από την επίσημη ιστοσελίδα.
4. Ανοίξτε τη Γραμμή Εντολών στον υπολογιστή σας και τρέξτε τη παρακάτω εντολή για να ξεκινήσει να λειτουργεί το μεγάλο γλωσσικό μοντέλο:
   `ollama run hf.co/gawyria/pers-com-email-model:Q8_0`
5. Γράψτε την εντολή /bye στη γραμμή εντολών για να βγείτε από τη διαχείριση του Ollama.
6. Βρείτε τον φάκελο που έχετε αποθηκεύσει τα αρχεία και αντιγράψτε το μονοπάτι του.
7. Γυρίστε στη γραμμή εντολών και γράψτε την εντολή "`cd [το μονοπάτι]`". Θα πρέπει τώρα να εμφανίζεται το μονοπάτι που έχετε προσθέσει.
8. Γράψτε την εντολή "`python AIPPC-local.py`" για να τρέξει ο κώδικας στο τερματικό.
9. Ξεκινήστε να εισάγεται τα στοιχεία σας.

Οδηγίες εγκατάστασης για την υλοποίηση με FastAPI:

1. Μεταβείτε στο GitHub και κατεβάστε τα εξής αρχεία:
   - FastAPI-main.py
   - FastAPI_test-api.py
   - .env
   - requirements.txt
2. Εγκαταστήστε το Ollama από την επίσημη ιστοσελίδα.
3. Ανοίξτε τη Γραμμή Εντολών στον υπολογιστή σας και τρέξτε τη παρακάτω εντολή για να ξεκινήσει να λειτουργεί το μεγάλο γλωσσικό μοντέλο: `ollama run gawyria/demo-model`
4. Δημιουργήστε έναν φάκελο στο πρόγραμμα το οποίο θα χρησιμοποιήσετε για να τρέξετε τον κώδικα και μεταφέρετε όλα τα παραπάνω αρχεία σε αυτόν.
5. Εκτελέστε το αρχείο FastAPI-main.py για να ενεργοποιήσετε τη βασική λειτουργικότητα της εφαρμογής.
6. Ανοίξτε το τερματικό και εκτελέστε την εντολή "`uvicorn main:app --reload`" η οποία εκκινεί τον διακομιστή του API.

7. Εκτελέστε το αρχείο FastAPI_test-api.py για να ξεκινήσετε τις δοκιμές και να αρχίσετε να εισάγετε τα απαιτούμενα δεδομένα.

ΠΑΡΑΡΤΗΜΑ ΙΙ

Ο κώδικας βρίσκεται στο παρακάτω GitHub: garkyr/AI-Powered-Phishing-Campaign