ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΧΕΔΙΟ ΠΑΡΟΥΣΙΑΣΗΣ
ΑΤΟΜΙΚΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Μάιος 2017

Ατομική Διπλωματική Εργασία

ΕΠΙΘΕΣΕΙΣ ΣΕ ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΓΩΓΗΣ ΛΕΞΕΩΝ ΜΕΛΙΟΥ

Αντρεάνα Κυριακίδου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2024

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΙΘΕΣΕΙΣ ΣΕ ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΓΩΓΗΣ ΛΕΞΕΩΝ ΜΕΛΙΟΥ

Αντρεάνα Κυριακίδου

Επιβλέπων Καθηγητής
Ηλίας Αθανασόπουλος

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου.

Μάιος 2024

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου, τον κύριο Ηλία Αθανασόπουλο, για τη βοήθεια που μου παρείχε στο θέμα διότι δεν είχα γνώσεις πάνω σε αυτό. Ο κύριος Ηλίας με την ευγενική του και κατανοητική προσέγγιση με βοήθησε να κατανοήσω το θέμα και με έκανε να νιώσω πιο βέβαιη για τις γνώσεις μου. Επίσης, θα ήθελα να ευχαριστήσω θερμά τον Αντρέα Διονυσίου για την καθοδήγησή του σχετικά με το πώς να προσεγγίσω επιθέσεις σε συστήματα honeyword. Η εμπειρία μου με αυτούς τους δύο ήταν εξαιρετική και δεν θα μπορούσα να είμαι πιο ευγνώμων για την υποστήριξή τους.

Περίληψη

Η παρούσα πτυχιακή εργασία επικεντρώνεται στη μελέτη των λέξεων μελιού ή αλλιώς honeywords, μιας τεχνικής εντοπισμού παραβιασμένων διαπιστευτηρίων, και στην εφαρμογή επιθέσεων σε ένα εργαλείο παραγωγής τέτοιων λέξεων, το HoneyGen, προκειμένου να αξιολογηθεί η ανθεκτικότητά του. Συγκεκριμένα, η εργασία επικεντρώνεται στην αποτελεσματικότητα του εργαλείου αυτού σε επιθέσεις στο επίπεδο συμβολοσειρών, χρησιμοποιώντας δύο ευρετικές προσεγγίσεις που βασίζονται σε τέσσερις μετρήσεις απόστασης: Edit Distance, Jaccard Distance, Cosine Distance και Euclidean Distance. Επιπλέον, παρέχεται μια ενδελεχής αξιολόγηση βάση μιας προηγούμενης έρευνας σχετικά με τις πιθανότητες για false positive και false negative επιθέσεις. Η εργασία περιλαμβάνει μια εισαγωγή σε υπάρχον γνωστικό υλικό σχετικά με τα honeywords καθώς και τις διάφορες τεχνικές που χρησιμοποιούνται για την παραγωγή τους. Στη συνέχεια, παρουσιάζεται η μεθοδολογία που ακολουθήθηκε στην εκτέλεση της εργασίας, με έμφαση στις ευρετικές επιθέσεις και συμπεριλαμβανομένων της εύρεσης/παραγωγής των dataset (σύνολα δεδομένων) και της έρευνας για την αποτελεσματικότητα του εργαλείου στα false positive. Τέλος, παρέχεται μια αξιολόγηση των αποτελεσμάτων που προέκυψαν από την εφαρμογή των ευρετικών μεθόδων επίθεσης, καταλήγοντας στο συμπέρασμα ότι η παραγωγή honeywords αποτελεί μια αξιόλογη τεχνική ασφάλειας που αξίζει περαιτέρω μελέτη.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Γενική ιδέα πτυχιακής εργασίας

Ο κεντρικός πυλώνας αυτής της εργασίας είναι η ανάλυση των βασικών χαρακτηριστικών των συστημάτων που παράγουν honeywords για την ανίχνευση παραβιασμένων διαπιστευτηρίων και το επίπεδο ανθεκτικότητάς τους.

Τα honeywords είναι ψεύτικοι κωδικοί οι οποίοι μπαίνουν στην βάση δεδομένων μιας πλατφόρμας μαζί με τον πραγματικό κωδικό ενός χρήστη με σκοπό να συγχίσουν έναν εισβολέα. Όταν ο εισβολέας χρησιμοποιήσει έναν από αυτούς τους ψεύτικους κωδικούς τότε στο σύστημα ενεργοποιείται συναγερμός για πιθανή κυβερνοεπίθεση.

Τα honeywords πρέπει να έχουν δύο κύριες απαιτήσεις ασφαλείας:

Το πρώτο, είναι το χαρακτηριστικό της δυσκολίας εντοπισμού του πραγματικού honeyword με το μάτι, κάτι που αποτελεί θεμέλιο λίθο για την ασφάλεια του συστήματος. Η δυσκολία αυτή προέρχεται από την ικανότητα των honeywords να μοιάζουν εξαιρετικά με τον πραγματικό κωδικό του χρήστη, καθώς και από την ύπαρξη πολλών από αυτούς τους ψευδής κωδικούς πρόσβασης (honeywords) που προστίθενται στο σύστημα έτσι ώστε ένας επιτιθέμενος να έχει πολλές επιλογές και να μην μπορεί να τα ξεχωρίσει.

Το δεύτερο χαρακτηριστικό που είναι και αυτό που εξετάζουμε με τις επιθέσεις μας, είναι το χαρακτηριστικό της μη αναστρεψιμότητας (non-reversibility) των honeyword. Το

χαρακτηριστικό αυτό αποτρέπει την ανάκτηση του πραγματικού κωδικού χρήστη από τα honeywords. Αυτό δημιουργεί μια αποτελεσματική ασπίδα ενάντια σε επιθέσεις που στοχεύουν στην ανακάλυψη των πραγματικών κωδικών, ενισχύοντας την ασφάλεια του συστήματος. Η εξέταση αυτή είναι κρίσιμη για την ανάλυση της αποτελεσματικότητας και της ανθεκτικότητας των συστημάτων honeyword έναντι πιθανών κυβερνοεπιθέσεων που έχουν ως αποτέλεσμα την διαρροή των διαπιστευτηρίων των χρηστών.

1.2 Κίνητρα

Τα κίνητρα που οδήγησαν στην ανάλυση και μελέτη των συστημάτων honeyword είναι πολλαπλά και σημαντικά. Καταρχάς, η αναγνώριση της σημασίας της κυβερνοασφάλειας σε σύγχρονες τεχνολογικές εφαρμογές και δίκτυα είναι ένα βασικό κίνητρο. Η αύξηση των κυβερνοεπιθέσεων και η ανάγκη για αποτελεσματικά μέσα προστασίας απαιτούν τη συνεχή ανάπτυξη και βελτίωση των μεθόδων ασφάλειας. Είναι σημαντικό να σημειωθεί ότι ο μέσος χρόνος καθυστέρησης στον εντοπισμό των διαρρευμένων διαπιστευτηρίων εκτείνεται σε αρκετούς μήνες ή ακόμα και χρόνια, επιτρέποντας στους επιτιθέμενους να εκμεταλλευτούν υπηρεσίες και να διαδώσουν ή ακόμα και να πουλήσουν τα κλοπιμαία δεδομένα στο διαδίκτυο. Αυτό υπογραμμίζει την κρίσιμη ανάγκη για την ανάπτυξη τεχνικών που επιτρέπουν τον έγκαιρο εντοπισμό των υποβιβασμένων διαπιστευτηρίων, όπως η χρήση των honeywords, οι οποίες θα καταστήσουν τις παρωχημένες παραδοσιακές μεθόδους προστασίας. Τέλος, η παρουσία ανοικτών ερευνητικών προκλήσεων στον τομέα των honeywords προσφέρει μια ευκαιρία για την επίλυση σημαντικών προβλημάτων στην κυβερνοασφάλεια και την ενίσχυση της αντίληψής για τις τρέχουσες απειλές στον ψηφιακό χώρο.

1.3 Μεθοδολογία

Έχοντας υπόψιν τα παραπάνω κίνητρα, η μεθοδολογία που ακολουθήσαμε εστιάστηκε στην εκτενή ανάλυση των συστημάτων honeywords και στη διερεύνηση πιθανών επιθέσεων εναντίον αυτών. Συγκεκριμένα, αφιέρωσα χρόνο στη μελέτη των honeywords, εστιάζοντας στο να κατανοήσω τη φύση τους, τη χρήση τους, και τη σημασία τους για την κυβερνοασφάλεια. Στη συνέχεια, με σκοπό να κατανοήσω τη διαδικασία παραγωγής τους, διάβασα ένα οδηγό σχετικά με ένα state-of-the-art σύστημα παραγωγής

honeywords, το οποίο ονομάζεται HoneyGen. Με βάση την κατανόηση αυτή, αποφάσισα να εκτελέσω το HoneyGen για τη δημιουργία και την ανάλυση κάποιων honeywords, προσπαθώντας να εντοπίσω τρόπους επίθεσης που θα μπορούσαν να αποκαλύψουν τον πραγματικό κωδικό του κάθε χρήστη. Κατά τη διάρκεια της έρευνας, διάβασα ένα άρθρο το οποίο εξέταζε και σύγκρινε τον βαθμό στον οποίο κάθε τρόπος παραγωγής honeywords θα ήταν επιρρεπής σε false positive και false negative επιθέσεις [2]. Μέσα στους τρόπους παραγωγής που εξετάστηκαν συμπεριλαμβανόταν και ο τρόπος παραγωγής που χρησιμοποιεί το HoneyGen. Με γνώμονα τα πιο πάνω και αναλύοντας τα honeywords που παράγονται από το HoneyGen, καταλήξαμε στο συμπέρασμα ότι η χρήση μετρικών σε επίπεδο αλφαριθμητικού αποτελεί μια αποτελεσματική προσέγγιση για επιθέσεις σε συστήματα παραγωγής λέξεων μελιού.

Κεφάλαιο 2

Προηγούμενη Γνώση/ Έρευνα

2.1 Τι είναι τα honeywords και ποια η χρησιμότητα τους

Τα honeywords είναι μια προηγμένη τεχνική ασφαλείας που χρησιμοποιείται για την προστασία των συστημάτων πρόσβασης και των κρίσιμων δεδομένων και ποιο συγκεκριμένα στην προστασία του κωδικού πρόσβασης ενός χρήστη. Τα honeywords είναι στην ουσία ψεύτικοι κωδικοί οι οποίοι παράγονται παράλληλα με τους πραγματικούς κωδικούς χρηστών και αποτελούν παραπλανητικά δεδομένα που δημιουργούν σύγχυση στους επιτιθέμενους. Κάθε φορά που ένας επιτιθέμενος έχει καταφέρει να διαρρεύσει την βάση με τα διαπιστευτήρια των χρηστών και προσπαθεί να προσπελάσει έναν λογαριασμό χρήστη, αντιμετωπίζει μια λίστα από πολλαπλές εικονικές εισόδους, μετατρέποντας τη διαδικασία εντοπισμού του πραγματικού κωδικού σε μια πραγματική πρόκληση. Καθώς τα honeywords δεν έχουν καμία πραγματική αξία, η επιτυχία του επιτιθέμενου στην αποκωδικοποίηση τους δεν θα οδηγήσει σε παραβίαση του συστήματος, ενώ η ανίχνευσή τους από το σύστημα αποκαλύπτει ανεπιθύμητη δραστηριότητα. Με την χρήση των honeywords, οι οργανισμοί μπορούν να ενισχύσουν την ασφάλεια των συστημάτων τους για έλεγχο

ταυτότητας με κωδικό πρόσβασης έτσι ώστε να αντιδρούν άμεσα σε επιθέσεις οι οποίες κατάφεραν να διαρρεύσουν τους κωδικούς πρόσβασης των χρηστών.

2.2 Είδη συστημάτων παραγωγής honeywords

Σε αυτήν την ενότητα θα συζητηθούν τέσσερα (4) είδη συστημάτων παραγωγής honeyword και έπειτα θα συγκριθούν.

2.2.1 Targeted password model-based generation

Αυτό το είδος συστήματος παραγωγής honeyword, χρησιμοποιεί μοντέλα πιθανοτήτων κωδικών (όπως το List, PCFG, Markov, περισσότερα για αυτά στο παράρτημα **Γ.1**) τα οποία μαθαίνουν μια κατανομή προτύπων κωδικών πρόσβασης και μετά επιστρέφουν ένα σύνολο από αυτά με βάση της πιθανότητας τους να είναι ο πραγματικός κωδικός. Ένα πρότυπο κωδικών πρόσβασης θα μπορούσε να ήταν ένα μοτίβο από τους κωδικούς που βάζει ο χρήστης σε διαφορετικούς ιστότοπους. Για παράδειγμα για τους κωδικούς "georgeTop" και "maxTop" το template θα ήταν N1Top και έτσι αλλάζοντας το tag στο template με τον user password ή με ένα άλλο προσωπικό του στοιχείο (όνομα χρήστη, όνομα, επίθετο) τότε ένας επιτιθέμενος θα δυσκολευόταν να επιλέξει τον πραγματικό κωδικό. Ένα άλλο παράδειγμα χρησιμοποιώντας τα password models θα παρήγαμε κωδικούς με tags π.χ. N1 και μετά με τον ίδιο τρόπο θα αλλάζαμε το tag με προσωπικές πληροφορίες του χρήστη.

2.2.2 LLM-based generation

Τα Large Language Models (LLMs) είναι αλγόριθμοι τεχνητής νοημοσύνης που βασίζονται στην τεχνολογία της βαθιάς μάθησης [2, 4]. Αυτοί οι αλγόριθμοι εκπαιδεύονται σε μεγάλα σύνολα δεδομένων γλωσσικού περιεχομένου και έχουν τη δυνατότητα να κατανοούν και να παράγουν φυσική γλώσσα. Τα LLMs χρησιμοποιούνται στην παράγωγή honeywords ερωτώντας τα με προτροπές βασισμένες στον πραγματικό κωδικό πρόσβασης εισόδου. Ένα παράδειγμα τέτοιου τρόπου παραγωγής honeyword είναι το Chunk-level GPT [4] το οποίο αρχικά με έναν αλγόριθμο βρίσκει όλα τα chunks του πραγματικού κωδικού και μετά δημιουργεί ένα ερώτημα (query) το οποίο περιλαμβάνει τον πραγματικό κωδικό μαζί με τα chunks του.

Παραδείγματος χάρη, αν ο πραγματικός κωδικός ήταν “bike200” τότε τα chunks του θα ήταν τα “bike” και “200” και το ερώτημα θα ήταν «Δώσε μου {n} κωδικούς παρόμοιους με το “bike200” οι οποίοι να περιέχουν (“bike”,“200”)» (**Γ.1** – περισσότερη ανάπτυξη).

2.2.3 Random replacement-based tweaking

Κατά την εφαρμογή αυτής της τεχνικής, αντικαθιστούνται τυχαίοι χαρακτήρες του αρχικού κωδικού με άλλους χαρακτήρες του ίδιου είδους, δηλαδή αν είναι αριθμός θα αντικατασταθεί με άλλον αριθμό αν είναι γράμμα με άλλο γράμμα και αν είναι ειδικός χαρακτήρας με έναν άλλο ειδικό χαρακτήρα, δημιουργώντας έτσι νέους κωδικούς που διαφέρουν ελαφρώς από τον αρχικό κωδικό πρόσβασης. Υπάρχουν δύο παραδείγματα αλγορίθμων που χρησιμοποιούν αυτή την τεχνική, ο πρώτος είναι ο CBTt [5] ο οποίος αλλάζει τους τελευταίους t χαρακτήρες και ο άλλος ο τρόπος είναι ο CBT* [1] με τον οποίο μπορεί να αλλάξουν τυχαία όλοι οι χαρακτήρες του κωδικού πρόσβασης με βάση την ακόλουθη στρατηγική: με πιθανότητα 0.3 μετατροπή κάθε κεφαλαίου γράμματος σε πεζό γράμμα, με πιθανότητα 0.03 να αντικατασταθεί στο αντίστοιχο κεφαλαίο κάθε πεζό γράμμα και η αλλαγή κάθε ψηφίου σε ένα διαφορετικό, ομοιόμορφα επιλεγμένο ψηφίο με πιθανότητα 0.05.

2.2.4 DNN-based tweaking

Στην εφαρμογή Deep Neural Network (DNN) αλγορίθμων για την παραγωγή honeyword έχουμε δύο διαφορετικά μοντέλα τα οποία αρχικά αναπτύχθηκαν για την εύρεση κωδικών πρόσβασης. Αυτά τα δύο μοντέλα είναι το PassTrans [6] και το Pass2Path (P2P) [7]. Το πρώτο μοντέλο εκπαιδεύεται με ένα σύνολο δεδομένων πραγματικών κωδικών, ώστε να μάθει τα μοτίβα και τους κανόνες που χρησιμοποιούνται στη δημιουργία και την τροποποίηση τους. Στην συνέχεια, αυτή η μέθοδος δημιουργεί προσαρμοσμένους κωδικούς πρόσβασης ως εξής: Δεδομένου ενός κωδικού πρόσβασης εισόδου p, αρχικά τον επεξεργάζεται μέσω ενός κωδικοποιητή για να καταλάβει την δομή του, και με βάση την εκπαίδευση του μαντεύει ποιος είναι ο αρχικός χαρακτήρας. Στην συνέχεια με βάση τους προηγούμενους χαρακτήρες και τις γνώσεις του μαντεύει τους επόμενους έναν, έναν. Το PassTrans χρησιμοποιώντας μια τεχνική που ονομάζεται αναζήτηση με δέσμη (beam search) παράγει τα q πιο πιθανά

μονοπάτια (passwords) τα οποία μπορούν να χρησιμοποιηθούν ως honeywords. Το δεύτερο μοντέλο (P2P) χρησιμοποιεί ένα πολύπλοκο σύστημα νευρωνικού δικτύου για να δημιουργήσει τροποποιημένους κωδικούς πρόσβασης με ασφάλεια. Το μοντέλο αρχικά λαμβάνει έναν κωδικό πρόσβασης ως είσοδο και επιστρέφει ένα μονοπάτι επεξεργασίας που δείχνει τις αλλαγές που πρέπει να γίνουν για να παραχθεί ο νέος κωδικός. Αυτές οι αλλαγές περιλαμβάνουν εισαγωγή, αντικατάσταση ή διαγραφή χαρακτήρων από τον αρχικό κωδικό.

2.3 HoneyGen

Το HoneyGen είναι ένα σύστημα παραγωγής honeywords το οποίο χρησιμοποιεί μια υβριδική μορφή των αλγορίθμων Chaffing-By-Tweaking και Chaffing-With-a-Password-Model [1]. Συγκεκριμένα, αρχικά εφαρμόζει τον αλγόριθμο Chaffing-With-a-Password-Model για εκπαίδευση του FastText [8] σε ένα σύνολο δεδομένων πραγματικών κωδικών πρόσβασης. Έπειτα, δίνοντας στο FastText τον πραγματικό κωδικό πρόσβασης, λαμβάνει ένα διάνυσμα που αντιπροσωπεύει την αναπαράσταση του συγκεκριμένου κωδικού πρόσβασης στο διανυσματικό χώρο. Στη συνέχεια, επιλέγει τους κορυφαίους k πλησιέστερους γείτονες με φθίνουσα σειρά βάσει της ομοιότητας cosine. Στην συνέχεια με την χρήση του αλγορίθμου Chaffing-By-Tweaking, και συγκεκριμένα του CBT* τον οποίο έχουμε αναλύσει, τροποποιεί κάθε έναν από τους k γείτονες που επιλέχθηκαν προηγουμένως. Η τροποποίηση αυτή παράγει ένα σύνολο μεγέθους k των τροποποιημένων κωδικών. Τέλος, η λίστα των honeywords αποτελείται από την ένωση των k γειτόνων που επιλέχθηκαν μέσω του Chaffing-With-a-Password-Model και των τροποποιημένων κωδικών που παράχθηκαν με τη χρήση του CBT* αλγορίθμου. Για την βελτιστοποίηση αυτού του συστήματος η χρήση του σε κάθε πλατφόρμα θα πρέπει να εκπαιδεύσει το FastText πάνω στο δικό τους σύνολο δεδομένων πραγματικών κωδικών πρόσβασης έτσι ώστε να μην μπορεί ο επιτιθέμενος να συγκρίνει τα αποτελέσματα (λίστα από honeywords) με αυτό και να εντοπίσει έτσι τον πραγματικό κωδικό. Τέλος, ένα μειονέκτημα αυτού του τρόπου παραγωγής είναι πως δεν αντιμετωπίζει τα “targeted attacks” δηλαδή τα attacks που γίνονται με την γνώση προσωπικών δεδομένων του χρήστη.

2.4 Επιθέσεις σε συστήματα παραγωγής Honeyword

Στην μελέτη αυτή, βρέθηκαν δύο είδη επιθέσεων σε συστήματα παραγωγής honeyword, το πρώτο είναι το λεγόμενο false positive attack και το δεύτερο το false negative attack. Παρακάτω θα αναλυθεί το κάθε ένα και πως με αυτούς τους τρόπους μπορεί κάποιος να επιτεθεί στα συστήματα αυτά.

False Positive attacks: Σε μια false positive επίθεση, το σύστημα ανίχνευσης απειλών ή ανιχνευτής επιθέσεων λαμβάνει λανθασμένα την ύπαρξη μιας απειλής ή επίθεσης, χωρίς να υπάρχει πραγματικά κάποια απειλή. Αυτό μπορεί να συμβεί επειδή το σύστημα είναι ρυθμισμένο να είναι υπερβολικά ευαίσθητο ή επειδή οι αλγόριθμοι ανίχνευσης δεδομένων είναι ελλιπείς. Το αποτέλεσμα είναι ότι αθώοι χρήστες ή κανονικές δραστηριότητες μπορεί να ερμηνευτούν λανθασμένα ως απειλές ή επιθέσεις, με αποτέλεσμα να υποστούν αρνητικές συνέπειες όπως η απόρριψη πρόσβασης σε υπηρεσίες ή ακόμη και η απώλεια εμπιστοσύνης στο σύστημα ανίχνευσης. Αυτό μπορεί να έχει σοβαρές συνέπειες για την ασφάλεια και την αξιοπιστία του συστήματος. Πιο συγκεκριμένα στα συστήματα παραγωγής honeyword, αυτό μπορεί να είναι αρκετά επικίνδυνο και συχνό φαινόμενο, λόγω του ότι τα honeywords είναι πολύ παρόμοια με τον πραγματικό κωδικό και έτσι αν κατά λάθος ένας χρήστης πληκτρολογήσει τον κωδικό του λάθος μπορεί να καταχωρίσει ένα από τα honeywords και αυτό θα ενεργοποιούσε τον συναγερμό για πιθανή επίθεση ενώ στην πραγματικότητα είναι απλά ένα λάθος του χρήστη. Όταν ένα σύστημα το οποίο χρησιμοποιεί honeywords έχει πολλά false positive είναι άχρηστο αφού ο σκοπός του είναι να εμποδίζει τους επιτιθέμενους να εισέλθουν στο σύστημα και όχι τους κανονικούς χρήστες. Ένα σύστημα με πολλά false positives θα έχει ως αποτέλεσμα μόνο την δυσφορία των χρηστών.

False Negative attacks: Σε μια false negative επίθεση, το σύστημα ανίχνευσης απειλών αποτυγχάνει να ανιχνεύσει μια πραγματική απειλή ή επίθεση που υπάρχει. Αυτό μπορεί να οφείλεται σε διάφορους παράγοντες, όπως η χρήση ανεπαρκών ή ξεπερασμένων αλγορίθμων ανίχνευσης, η έλλειψη ενημερωμένων ανιχνευτών για νέες απειλές, ή ακόμη και η εξελικτική φύση των κυβερνοεπιθέσεων. Αυτό μπορεί να οδηγήσει σε σοβαρές επιπτώσεις όπως διαρροή δεδομένων, διακοπή λειτουργίας συστημάτων ή

ακόμη και υπονόμευση της εμπιστοσύνης στο σύστημα ασφαλείας. Στο πλαίσιο των honeywords, σε ένα σενάριο false negative επίθεσης, ο επιτιθέμενος επιτυγχάνει να μαντέψει ή να αποκτήσει έναν από τους πραγματικούς κωδικούς χωρίς να συναντήσει κανένα από τα honeywords, αποφεύγοντας έτσι εντελώς τον μηχανισμό ανίχνευσης που βασίζεται στα honeywords.

Αντιμετώπιση:

Οι δύο αυτές προκλήσεις μπορεί να φανούν σχετικά εύκολες στην αντιμετώπιση μεμονωμένα ωστόσο, όταν αντιμετωπίζονται ταυτόχρονα, η πολυπλοκότητα αυξάνεται σημαντικά. Για παράδειγμα, αν δημιουργήσει κανείς τα honeywords με αρκετές διαφορές από τον πραγματικό κωδικό, προκειμένου να αποφευχθεί η πιθανότητα εισαγωγής κατά λάθος από τον χρήστη ενός από αυτά, μπορεί να δημιουργηθεί ένα περιβάλλον όπου ο επιτιθέμενος μπορεί να ξεχωρίσει εύκολα τον πραγματικό κωδικό από τα honeywords. Κατά συνέπεια, ο επιτιθέμενος θα μπορούσε να αποφύγει την ενεργοποίηση συναγερμού εισαγωγής honeyword, κάτι που θα οδηγούσε σε ένα false negative. Αυτή η δυνητική αντίφαση επιδεινώνει την πολυπλοκότητα στην ανάπτυξη αποτελεσματικών συστημάτων ανίχνευσης απειλών που βασίζονται σε honeywords και επισημαίνει την ανάγκη για προσεκτική σχεδίαση και αξιολόγηση των μέτρων ασφαλείας και της δημιουργίας των honeywords.

2.5 Μετρικές Απόστασης

Σε αυτό το υποκεφάλαιο θα αναφέρουμε και θα εξηγήσουμε πως λειτουργούν οι τέσσερεις μετρικές απόστασης που χρησιμοποιήσαμε στις ευρετικές μεθόδους, παραδίδοντας και κάποια μικρά παραδείγματα για την κάθε μία [9,10,11,12].

Edit Distance/ Levenshtein Distance (Απόσταση Levenshtein)

- Μετράει τον αριθμό των επεξεργαστικών ενεργειών (εισαγωγές, διαγραφές, αντικαταστάσεις) που απαιτούνται για να μετατραπεί μια συμβολοσειρά σε μια άλλη.
- Φόρμουλα: $D_{\text{edit}}(x,y)$=Minimum number of operations to convert x to y
- Για τον υπολογισμό της χρησιμοποίησα την βιβλιοθήκη Levenshtein στην Python

Παράδειγμα:

1. Για είσοδο τα δύο συμβολοσειρές, το str1= “cut” και το str2 = “cat”, θα έχουμε ως έξοδο την τιμή ένα (1), διότι μπορούμε να αντικαταστήσουμε στο str1 το γράμμα “u” με το γράμμα “a” έτσι ώστε να γίνει και αυτό ίσο με “cat”.

2. Για είσοδο τα δύο συμβολοσειρές, το str1= “ sunday ” και το str2 = “saturday”, θα έχουμε ως έξοδο την τιμή τρία (3), διότι βασικά αυτό που θέλουμε είναι να μετατρέψουμε στο str1 το “un” με “atur”. Άρα, αρχικά αντικαθιστούμε το “n” με το “r” και στην συνέχεια προσθέτουμε το “t” και μετά το “a”.

Jaccard Distance (Απόσταση Jaccard)

- Ορίζεται ως η αναλογία του μεγέθους της τομής των συνόλων προς το μέγεθος της ένωσής τους.
- Φόρμουλα: $D_{Jaccard}(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}$
- Για τον υπολογισμό της χρησιμοποίησα την μέθοδο jaccard_distance της βιβλιοθήκης nltk.metrics στην python

Παράδειγμα:

Έχουμε ως είσοδο δύο συμβολοσειρές, τα str1= “cut” και str2= “hut” τότε αρχικά πρέπει να τις μετατρέψουμε σε σετ, set1 = {‘c, ‘u’, ‘t’} και set2 = {‘h’, ‘u’, ‘t’} οπότε η τομή τους θα μας έδινε το σετ inter_set = {‘u’, ‘t’} που έχει μέγεθος 2. Η ένωση τους θα μας έδινε το union_set = {‘c’, ‘u’, ‘t’, ‘h } που έχει μέγεθος 4. Άρα το jaccard distance θα ήταν ίσο με $D_{Jaccard}$ =1- $\frac{2}{4}$ =0.5.

Cosine Distance (Απόσταση Cosine)

- Υπολογίζει τη γωνία μεταξύ των δύο διανυσμάτων στον n-διάστατο χώρο.
- Φόρμουλα: D_{cosine} (A,B)=1- Cosine Similarity = 1- $\frac{A \cdot B}{\|A\| \cdot \|B\|}$
- Για τον υπολογισμό της χρησιμοποίησα το distance module της βιβλιοθήκης scipy.spatial και την μέθοδο distance.cosine

Παράδειγμα:

Για είσοδο τις συμβολοσειρές str1= “cut” str2 = “hut”:

1. Μετέτρεψε τις συμβολοσειρές σε διανύσματα:

   Str1= "cat" -> Vector 1: [1, 1, 1, 0] (c:1, a:1, t:1, h:0)

   Str2: "hat" -> Vector 2: [0, 1, 1, 1] (c:0, a:1, t:1, h:1)

2. Υπολόγισε το εσωτερικό γινόμενο των διανυσμάτων:

   Dot product = (1*0) + (1*1) + (1*1) + (0*1) = 0 + 1 + 1 + 0 = 2

3. Υπολόγισε τα magnitudes:

   Magnitude of Vector 1 = $\sqrt{1^2 + 1^2 + 1^2 + 0^2} \approx 1.732$

   Magnitude of Vector 2 = $\sqrt{0^2 + 1^2 + 1^2 + 1^2} \approx 1.732$

4. Υπολόγισε το Cosine Similarity:

   Cosine Similarity $= \frac{Dot\ Product}{Magnitude\ of\ Vector\ 1 * Magnitude\ of\ Vector\ 2} = \frac{2}{1.732 * 1.732} \approx 0.667$

5. Υπολόγισε την Cosine Distance:

   Cosine Distance = 1−Cosine Similarity = 1− 0.667 ≈ 0.333

Euclidean Distance (Ευκλείδεια Απόσταση)

- Είναι η ευθεία απόσταση μεταξύ δύο σημείων σε έναν n-διάστατο χώρο.
- Υπολογίζεται ως η τετραγωνική ρίζα του αθροίσματος των τετραγώνων των διαφορών των αντίστοιχων συνιστωσών.
- Φόρμουλα: $D_{euclidean}(p, q) = \sqrt{\sum_{i=1}^{n}(qi - pi)^2}$
- Για τον υπολογισμό της χρησιμοποίησα το distance module της βιβλιοθήκης scipy.spatial και την μέθοδο distance.euclidean

  Παράδειγμα:

  Χρησιμοποιούμε και πάλι τις συμβολοσειρές και την μορφή τους σε διανύσματα όπως στο παράδειγμα για την Cosine Distance:

  Συμβολοσειρές : str1= “cut” str2 = “hut”

Διανύσματα : Str1= "cat" -> Vector 1: [1, 1, 1, 0] (c:1, a:1, t:1, h:0)

Str2: "hat" -> Vector 2: [0, 1, 1, 1] (c:0, a:1, t:1, h:1)

1. Αρχικά υπολογίζουμε την διαφορά των στοιχείων τους :

   Διαφορές = [1-0, 1-1, 1-1, 0-1] = [1, 0, 0, -1]

2. Υπολογισμός Τετραγώνων των Διαφορών:

   Τετράγωνα = [1^2, 0^2, 0^2, (-1)^2] = [1, 0, 0, 1]

3. Άθροισμα Τετραγώνων:

   Άθροισμα = 1 + 0 + 0 + 1 = 2

4. Τετραγωνική Ρίζα του Αθροίσματος:

   Ευκλείδεια Απόσταση = 2≈1.4142≈1.414

Κεφάλαιο 3

Μεθοδολογία

3.1 Εισαγωγή κεφαλαίου

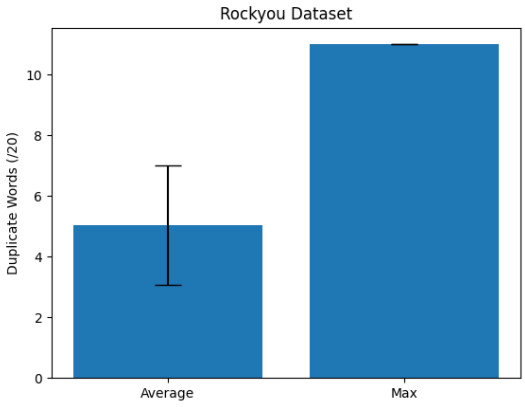
Σε αυτό το κεφάλαιο θα αναλυθεί λεπτομερώς η μεθοδολογία που χρησιμοποιήσαμε για τις επιθέσεις, καθώς και οι δύο ευρετικές μέθοδοι που εφαρμόσαμε. Αρχικά, θα εξηγήσουμε τις πολλές προσπάθειες και δοκιμές που πραγματοποιήθηκαν προτού καταλήξουμε στις δύο μεθόδους επίθεσης που θεωρήσαμε ως τις καλύτερες. Μία από αυτές τις δοκιμές βασίστηκε σε ένα άρθρο των Z. Huang, L. Bauer και M. Reiter [2] που αναφέρει διάφορες μεθόδους παραγωγής honeywords και αναλύει την ευάλωτη φύση τους σε επιθέσεις false positive και false negative.

Σε αυτό το άρθρο, μια από τις μεθόδους ήταν το Chaffing-With-a-Hybrid-Model (CHM), το οποίο χρησιμοποιεί το HoneyGen. Παρόλο που το CHM είχε καλή πιθανότητα false negative, η πιθανότητα false positive φάνηκε να είναι αρκετά υψηλή. Η εξήγηση που δόθηκε στο άρθρο ήταν ότι το CHM περιλαμβάνει ένα ντετερμινιστικό βήμα που αναζητά τους πλησιέστερους γείτονες του κωδικού πρόσβασης για τη δημιουργία των honeywords.

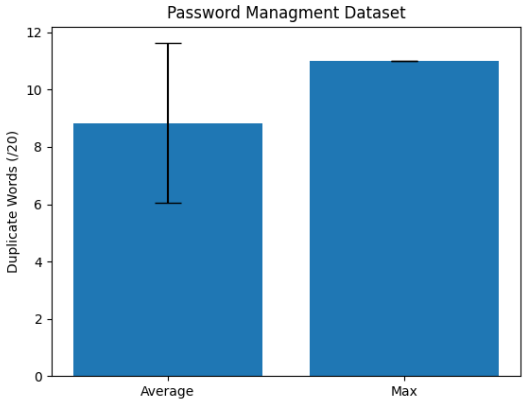
Στην εργασία αυτή, προσθέτουμε ότι επειδή οι 9 πλησιέστεροι κωδικοί δεν αλλάζουν μέσω του CBT*, υπάρχει πιθανότητα διπλότυπων κωδικών σε περίπτωση που εισαχθεί ο ίδιος κωδικός στο HoneyGen δύο φορές. Αυτό είναι ιδιαίτερα πιθανό όταν το HoneyGen δεν επανεκπαιδεύεται με έναν ανανεωμένο κατάλογο κωδικών της πλατφόρμας ή όταν δεν επανεκπαιδεύεται καθόλου ανάμεσα στις δύο εκτελέσεις του HoneyGen. Σε αυτές τις περίπτωσης, εάν ο επιτιθέμενος γνωρίζει τον κωδικό "p",

μπορεί να χρησιμοποιήσει τα διπλότυπα αυτά για να προκαλέσει false positive αποτελέσματα. Επομένως, υπάρχει μέγιστη πιθανότητα 9 στα 19 να προκαλέσει ένα ψευδές θετικό, καθώς το Fasttext είναι εκπαιδευμένο στον κατάλογο κωδικών της πλατφόρμας, επομένως αυτοί οι 9 είναι πραγματικοί κωδικοί της πλατφόρμας.

Για να επιβεβαιώσουμε αυτήν την παρατήρηση, δημιουργήσαμε ένα σύνολο δεδομένων με πραγματικούς κωδικούς πρόσβασης και τα αντίστοιχα honeywords τους. Οι κωδικοί προέρχονταν είτε από το ίδιο σύνολο δεδομένων που εκπαιδεύτηκε το μοντέλο FastText και ποιο συγκεκριμένα το σύνολο διαρρευμένων κωδικών RockYou, είτε από συστήματα διαχείρισης κωδικών που χρησιμοποιούνται σε προγράμματα περιήγησης όπως το Google Chrome, το Safari, το Firefox και το Microsoft Edge. Στη συνέχεια, επανεκπαιδεύσαμε το μοντέλο FastText και αναλύσαμε τα στατιστικά δεδομένα για τυχόν διπλότυπους κωδικούς σε δύο περιπτώσεις: όταν η πλατφόρμα εκπαιδεύει το Fasttext στον δικό της κατάλογο κωδικών (κωδικοί RockYou) και όταν η πλατφόρμα δεν εκπαιδεύει το Fasttext πάνω στον δικό της κατάλογο κωδικών (κωδικοί από συστήματα διαχείρισης κωδικών).



Σχήμα 3.1 Το average μαζί με το standard deviation και ο μέγιστος αριθμός από duplicates σε ένα σύστημα που εκπαιδεύτηκε με τους δικούς του κωδικούς.



Σχήμα 3.2 Ο μέσος όρος (average) μαζί με την τυπική απόκλιση (standard deviation) και ο μέγιστος αριθμός από duplicates σε ένα σύστημα που εκπαιδεύτηκε με κωδικούς από διαφορετική πλατφόρμα.

Όπως μπορούμε να δούμε από τις γραφικές παραστάσεις στα Σχήματα 3.1 και 3.2, ο μεγαλύτερος αριθμός από διπλότυπους κωδικούς σε και τις δύο περιπτώσεις ήταν 11, συνολικά 10/19, αν εξαιρέσουμε το πραγματικό κωδικό p. Αυτός ο αριθμός είναι μεγαλύτερος από τον αρχικό υπολογισμό των 9. Μια πιθανή εξήγηση για αυτό είναι ότι έγιναν κάποιες ίδιες τροποποιήσεις από τον CBT*.

Αν συγκρίνουμε τα δύο σχήματα, ειδικότερα τις μέσες τιμές, είναι εύκολο να καταλάβουμε ότι όταν εκπαιδεύεται το HoneyGen με το σύνολο κωδικών της ίδιας της πλατφόρμας έχει λιγότερους διπλότυπους κωδικούς εάν τρέξει το HoneyGen δύο φορές για τον ίδιο κωδικό. Το να τρέξεις το HoneyGen σε διαφορετικό κατάλογο κωδικών από αυτόν της πλατφόρμας σου, σίγουρα αφαιρεί την πιθανότητα για false positives, όμως υπάρχουν περισσότερα διπλότυπα και ο επιτιθέμενος μπορεί να συγκρίνει τα honeywords με τους υπόλοιπους κωδικούς στη βάση δεδομένων για να βρει τον πραγματικό.

Μια εξήγηση για το ότι υπάρχουν λιγότερά διπλότυπα όταν εκπαιδεύσει μια πλατφόρμα το Fasttext στον δικό της κατάλογο κωδικών, είναι ότι οι κωδικοί είναι πιο κοντά λόγω των περιορισμών ασφάλειας κάθε πλατφόρμας, με αποτέλεσμα να υπάρχουν περισσότεροι κωδικοί που μοιάζουν μεταξύ τους. Έτσι, όταν κάνουμε query το FastText για να μας δώσει τους 9 πιο κοντινούς κωδικούς, υπάρχουν περισσότερες επιλογές.

Επόμενο βήμα, μετά από αυτό το πείραμα για τα διπλότυπα, ήταν να ελέγξουμε, χρησιμοποιώντας κάποιες μετρικές απόστασης, αν θα μπορούσαμε να αναγνωρίσουμε κάποια σχέση μεταξύ του πραγματικού κωδικού και των αντίστοιχων honeywords τους. Μια τέτοια σχέση θα επέτρεπε σε έναν επιτιθέμενο να ξεχωρίσει τον πραγματικό κωδικό από τα υπόλοιπα και να συνδεθεί με αυτόν χωρίς να ενεργοποιήσει κάποιο συναγερμό. Επομένως, η επόμενη κίνηση ήταν να δημιουργήσουμε ορισμένους πίνακες 'ground truth', οι οποίοι θα περιλαμβάνουν τις αποστάσεις μεταξύ κάθε honeyword και του αντίστοιχου πραγματικού κωδικού τους, καθώς και τις αποστάσεις μεταξύ των honeywords μεταξύ τους. Χρησιμοποιώντας λοιπών τα δεδομένα από την εκτέλεση του HoneyGen πάνω σε κωδικούς από το αρχείο διαρρευμένων κωδικών του Rockyou και πάνω σε κωδικούς οι οποίοι παράχθηκαν από συστήματα διαχείρισης κωδικών δημιούργησα τους ground truth πίνακες. Οι πίνακες αυτοί έχουν σε κάθε κελί τις τιμές για τις μετρικές απόστασης Edit Distance, Jaccard Distance, Cosine Similarity και Euclidean Distance με αυτή την σειρά διαχωρισμένες με κόμμα. Ένα παράδειγμα τέτοιου πίνακα φαίνεται στον Πίνακα 3.3.

Πίνακας 3.3 – Παράδειγμα πίνακα 20x20 με τις τιμές των 4 μετρικών απόστασης για κάθε honeyword με τα υπόλοιπα. Η πρώτη στήλη/γραμμή περιέχουν τις τιμές των αποστάσεων από τον πραγματικό κωδικό.

	'iloveyou'	'ilovEyou'	'iloveyouu'	…….	'iLoVeYouu'
'iloveyou'	0, 0.0, 0.0, 1, 0.0	1, 0.25, 0.125, 0.999902, 8.84	1, 0.0, 0.125, 0.999997, 4.59	…….	4, 0.6, 0.5, 0.996828, 26.23
'ilovEyou'	1, 0.25, 0.125, 0.999902, 8.84	0, 0.0, 0.0, 1, 0.0	2, 0.25, 0.25, 0.999915, 4.26	…….	5, 0.727, 0.625, 0.997800, 17.39
'iloveyouu'	1, 0.0, 0.125, 0.999997, 4.59	2, 0.25, 0.25, 0.999915, 4.26	0, 0.0, 0.0, 1, 0.0	…….	3, 0.6, 0.33, 0.996920, 21.64
…….	…….	…….	…….	0, 0.0, 0.0, 1, 0.0	…….

'iLoVeYouu'	4, 0.6, 0.5, 0.996828, 26.23	5, 0.72, 0.625, 0.997800, 17.39	3, 0.6, 0.333, 0.996920, 21.64	…….	0, 0.0, 0.0, 1, 0.0

Με βάση αυτούς τους πίνακες (ένας για κάθε password) προσπαθήσαμε να χτίσουμε ευρετικές μεθόδους με αρχικό στόχο να μειωθεί το μέγεθος του σετ (<20 honeywords) και στην καλύτερη περίπτωση να το μειώσουνε σε 1 δηλαδή να βρεθεί ο πραγματικός κωδικός πρόσβασης.

3.2 Όριο Μέσης Απόστασης

Η ευρετική μέθοδος «Όριο Μέσης Απόστασης» είναι η πρώτη μέθοδος με την οποία θα προσπαθήσουμε να επιτεθούμε στο σύστημα HoneyGen στο επίπεδο συμβολοσειράς. Πιο συγκεκριμένα, με βάση τη μέση απόσταση μεταξύ των κωδικών που επιλέγονται από τους χρήστες και ενός συνόλου από honeywords, αυτή η μέθοδος έχει ως στόχο να μειώσει αποτελεσματικά το σετ από τα πιθανά honeywords έτσι ώστε να βελτιώσει τις πιθανότητες του επιτιθέμενου να βρει τον πραγματικό κωδικό πρόσβασης. Αρχικά, υπολογίζουμε τη μέση απόσταση (y) και την τυπική απόκλιση (σ) μεταξύ του κωδικού που επιλέγει ο χρήστης και του αντίστοιχου συνόλου από honeywords για ένα σύνολο χρηστών. Κατά την προσομοίωση μιας επίθεσης, ο επιτιθέμενος θα βρει την μέση απόσταση κάθε sweetword με τα υπόλοιπα. Όπου εξορισμού, sweetwords είναι το σύνολο των honeywords ενός κωδικού συμπεριλαμβανομένου και αυτού. Ο κωδικός του χρήστη αναγνωρίζεται ως οποιοδήποτε sweetword του οποίου η μέση απόσταση, y_s, από τα (υποτιθέμενα) honeywords του βρίσκεται εντός της υπολογισμένης τυπικής απόκλισης σ ($y - \sigma \leq y_s \leq y + \sigma$).

Η επιτυχία της μεθόδου «Όριο Μέσης Απόστασης» στην εφαρμογή επιθέσεων εξαρτάται σημαντικά από την ποιότητα των δεδομένων που χρησιμοποιούνται. Η αξιοπιστία αυτής της μεθόδου απαιτεί ένα σύνολο δεδομένων με μεγάλη ποικιλία σε κωδικούς πρόσβασης και των αντίστοιχων honeywords τους. Τα δεδομένα αυτά πρέπει να αντικατοπτρίζουν πιστά ένα πραγματικό περιβάλλον χρήστη, όπως ένα πραγματικό σύστημα που θα εφαρμόσει και θα εμπιστευτεί το HoneyGen για την κυβερνοασφάλεια του. Συνεπώς, η ποιότητα αυτού του συνόλου δεδομένων επηρεάζει την

αποτελεσματικότητα και την ακρίβεια της μεθόδου, καθώς και την αξιοπιστία των στατιστικών αποτελεσμάτων που παράγονται μέσω αυτής. Οπότε, για την εφαρμογή της μεθόδου, χρησιμοποίησα ένα σύνολο δεδομένων που περιλαμβάνει honeywords πάνω σε κωδικούς πρόσβασης οι οποίοι έχουν διαρρεύσει από διάφορες πλατφόρμες, όπως το Adult FriendFinder, το Chegg, το Dropbox, το Dubsmash, το Have I Been Pwned, το Last-fm, το LinkedIn, το MySpace, το phpBB, το Rockyou, το Yahoo, το Youku και το Zynga. Αυτά τα δεδομένα τα βρήκα έτοιμα (Πίνακα 3.4) στο main repository του HoneyGen[1].

Πίνακας 3.4 - Μορφή ενός από τα αρχεία τα οποία περιέχουν κωδικούς πρόσβασης που διέρρευσαν.

Real_password	Honeyword_0	…	Honeyword_17	Honeyword_18
shaw6844	shaW6864	…	shaqwae	sHaqwaE
whenwe10165	WHenwe10165	…	1501we	1561We
cadet2007	caDet2007	…	cali8807	cali8887
quierounauto	quieroUnauto	…	yenunahio3	yenUnahio3
jpuubgpe	jpuubgpE	…	jeyrok	jeYrok

Αυτά τα αρχεία με τους κωδικούς πρόσβασης που διέρρευσαν τα χωρίσαμε σε δεδομένα εκπαίδευσης (training file) και δεδομένα ελέγχου (testing file), όπου από τους 50,000 κωδικούς που είχε το κάθε αρχείο, οι 45,000, δηλαδή το 90%, επιλέχθηκαν τυχαία και μπήκαν στο training file, και οι υπόλοιποι 5,000 μπήκαν στο testing file. Άρα στο τέλος είχαμε 13 αρχεία για training και 13 αρχεία για testing, ένα αρχείο για κάθε μια από τις πλατφόρμες που έχουν παραβιαστεί.

Η επόμενη κίνηση ήταν να δημιουργήσουμε τους πίνακες αληθείας (Πίνακες 3.6.1 – 3.6.4) όπως κάναμε και προηγουμένως, αλλά αυτή τη φορά με τους κωδικούς που παράξαμε για κάθε training αρχείο. Όμως μετά από την πρώτη προσπάθεια να γίνει αυτό, λόγω του πολύ μεγάλου αριθμού δεδομένων, το πρόγραμμα αργούσε πάρα πολύ να κάνει τους πίνακες, ήθελε 2-3 ημέρες για το κάθε αρχείο έτσι ξαναδιαβάζοντας την ευρετική μέθοδο, μπορεί να αντιληφθεί κανείς, πως για το training δεν χρειαζόταν να

[1] https://bitbucket.org/srecgrp/honeygen-generating-honeywords-using-representation-learning/src/master/

βρούμε τις αποστάσεις από κάθε honeyword στα υπόλοιπα honeyword, αλλά μόνο τις αποστάσεις όλων των honeyword από τον πραγματικό κωδικό (ο οποίος είναι η πρώτη συμβολοσειρά σε κάθε γραμμή των αρχείων). Το αποτέλεσμα άρα ήταν τέσσερις πίνακες (έναν για κάθε μετρική απόστασης) για κάθε αρχείο. Ο κάθε πίνακας όπως μπορείτε να δείτε και στους παρακάτω πίνακες έχει στην πρώτη στήλη τον κωδικό πρόσβασης και στις υπόλοιπες στήλες τις αποστάσεις από τα honeywords του.

Τέλος, με τη χρήση αυτών των πινάκων, πραγματοποίησα τον υπολογισμό της μέσης απόστασης για κάθε γραμμή. Έπειτα, υπολόγισα τη μέση απόσταση και την τυπική απόκλιση όλων αυτών των μέσων αποστάσεων τις οποίες χρησιμοποίησα για την εφαρμογή της επίθεσης (Πίνακας 3.5).

Πίνακας 3.5 – Αποτέλεσμα υπολογισμού μέσης απόστασης και της τυπικής της απόκλισης για κάθε μετρική απόστασης.

Sheet Name	Overall Average	Standard Deviation
EditDistance	6.370478	2.251273
JaccardDistance	0.591327	0.145844
CosineDistance	0.000123	8.11E-05
EuclideanDistance	4.259335	1.443075

Παράδειγμα πινάκων για κάθε αρχείο:

Πίνακας 3.6.1 – Πίνακας για τις τιμές της Edit Distance για κάθε password σε σχέση με τα honeywords του.

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	2	6	…	11	11
agent09100	3	9	…	4	5
newj3rs3y	1	6	…	6	6
bluebirdbliss	4	5	…	5	6
carusolombardi	1	13	…	11	12
6qnefb7k	1	6	…	9	9

Πίνακας 3.6.2 – Πίνακας για τις τιμές της Jaccard Distance για κάθε password σε σχέση με τα honeywords του.

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	0.3	0.66666667	…	0.615385	0.615384615
agent09100	0.3	0.5	…	0.444444	0.6
newj3rs3y	0.222222222	0.75	…	0.727273	0.727272727
bluebirdbliss	0.363636364	0.3	…	0.3	0.454545455
carusolombardi	0.083333333	0.57142857	…	0.6	0.625
6qnefb7k	0.222222222	0.76923077	…	0.846154	0.857142857

Πίνακας 3.6.3 – Πίνακας για τις τιμές της Cosine Distance για κάθε password σε σχέση με τα honeywords του.

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	0.000205576	3.1471E-05	…	5.28E-05	7.51019E-05
agent09100	0.000141621	3.6597E-05	…	1.38E-05	3.55244E-05
newj3rs3y	1.56164E-05	3.8087E-05	…	2.39E-05	3.83854E-05
bluebirdbliss	8.73804E-05	1.0967E-05	…	1.76E-05	2.72393E-05
carusolombardi	3.52263E-05	1.5676E-05	…	3.45E-05	3.43323E-05
6qnefb7k	1.37091E-06	0.00031394	…	2.53E-05	9.21488E-05

Πίνακας 3.6.4 – Πίνακας για τις τιμές της Euclidean Distance για κάθε password σε σχέση με τα honeywords του.

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	9.864067078	6.70502615	…	2.080395	0.65640986
agent09100	4.299330711	3.282252312	…	3.909687	3.071848869
newj3rs3y	2.320716619	2.881717443	…	7.812048	3.064170122
bluebirdbliss	13.59184551	1.474627376	…	1.546087	1.612679958
carusolombardi	4.235716343	2.726480961	…	0.780517	4.242190838
6qnefb7k	0.282376707	9.163926125	…	6.876518	2.935087919

3.3 Όριο Μέσης Απόστασης προς τα κορυφαία k-πλησιέστερα honeywords

Η ευρετική μέθοδος «Όριο Μέσης Απόστασης προς τα κορυφαία k-πλησιέστερα honeywords» προσφέρει μια πιο λεπτομερή ανάλυση των δεδομένων σε σύγκριση με την προηγούμενη μέθοδο. Αντί να έχουμε μία μέση απόσταση, y, για ένα σύνολο κωδικών πρόσβασης και τα αντίστοιχα honeywords τους, τώρα έχουμε 19 μέσες τιμές, μία για κάθε honeyword, σε ένα σύνολο χρηστών.

Συγκεκριμένα, υπολογίζουμε τη μέση απόσταση, y_k (1<=k<=19), του κωδικού που επιλέγεται από τον χρήστη με κάθε ένα από τα honeywords του, πάνω σε ένα σύνολο χρηστών. Αυτό δημιουργεί έναν μονοδιάστατο (1D) πίνακα με 19 τιμές, όπου κάθε τιμή αντιστοιχεί στη μέση απόσταση προς το πλησιέστερο honeyword, το δεύτερο πλησιέστερο, το τρίτο πλησιέστερο, κ.λπ., με την αντίστοιχη απόκλιση τους, σ.

Κατά την εκτέλεση της επίθεσης, επαναλαμβάνουμε την ίδια διαδικασία, υπολογίζοντας την απόσταση του κάθε sweetword με κάθε ένα από τα υποτιθέμενα honeywords του. Επιλέγουμε ένα sweetword σαν κωδικό χρήστη, μόνο εάν η απόστασή του από τα honeywords του βρίσκεται εντός της υπολογισμένης τυπικής απόκλισης, σ, για κάθε honeyword. Με άλλα λόγια, η απόσταση από το πρώτο πλησιέστερο honeyword του sweetword πρέπει να είναι εντός της τυπικής απόκλισης, σ, που είχαμε υπολογίσει στο βήμα της εκπαίδευσης για τα πρώτα πλησιέστερα honeyword. Το ίδιο ισχύει και για τα υπόλοιπα honeywords, ανάλογα με τη σειρά της απόστασής τους από το sweetword.

Για την εφαρμογή της μεθόδου αυτής, χρησιμοποιήσαμε τα ίδια δεδομένα που είχαμε χρησιμοποιήσει και για την προηγούμενη μέθοδο. Αυτά τα δεδομένα περιλαμβάνουν μια ποικιλία κωδικών πρόσβασης και τα αντίστοιχα honeywords τους, τα οποία είχαν διαρρεύσει από διάφορες πλατφόρμες, όπως το Adult FriendFinder, το Chegg, το Dropbox, κ.λπ. Αυτά τα δεδομένα είναι πολύτιμα για τη μελέτη και την αξιολόγηση της αποτελεσματικότητας της μεθόδου, καθώς αντικατοπτρίζουν πραγματικές καταστάσεις χρήσης και πιθανές επιθέσεις. Μπορείτε να δείτε ένα παράδειγμα αυτών των δεδομένων στο Πίνακα 3.4.

Χρησιμοποιώντας αυτά τα δεδομένα, μπορέσαμε να υπολογίσουμε τις μέσες αποστάσεις και τις αντίστοιχες τυπικές αποκλίσεις για κάθε honeyword με βάση την απόσταση του από τον κωδικό του χρήστη σε ένα σύνολο χρηστών. Αυτή η προσέγγιση μας επέτρεψε να αναλύσουμε την αποτελεσματικότητα της μεθόδου και να συγκρίνουμε τα αποτελέσματα με αυτά της προηγούμενης μεθόδου. Μέσω αυτής της σύγκρισης, μπορούμε να αξιολογήσουμε την απόδοση και την ακρίβεια της κάθε μεθόδου και να εξάγουμε συμπεράσματα για την καταλληλόλητά τους ως κυβερνοεπιθέσεις.

Όπως και στην προηγούμενη ευρετική μέθοδο η επόμενη κίνηση ήταν η δημιουργία των ground-truth πινάκων για τα training data έτσι ώστε να έχουμε μία εικόνα των τιμών απόστασης και να είμαστε σίγουροι πως τα αποτελέσματα των μέσων τιμών βγήκαν πάνω σε σωστά υπολογισμένες τιμές. Σε αυτούς τους πίνακες και πάλι δεν χρειαζόταν να βρούμε τις αποστάσεις από όλα τα honeywords αλλά μόνο από τον επιλεγμένο κωδικό του χρήστη. Έτσι αντί να έχω έναν 20x20 πίνακα για κάθε σύνολο από sweetwords έχουμε απλά έναν 1D πίνακα για κάθε πραγματικό κωδικό (Πίνακες 3.8.1).

Η διαδικασία για την παραγωγή αυτών των πινάκων ήταν πολύ απλή αφού το μόνο που είχαμε να κάνουμε αφού χρησιμοποιήθηκαν τα ίδια σετ δεδομένων με την προηγούμενη ευρετική μέθοδο, ήταν να ταξινομήσουμε τους προηγούμενους πίνακες σε αύξουσα σειρά έτσι ώστε η πρώτη στήλη μετά την στήλη του password να έχει την πλησιέστερη απόσταση και η τελευταία την τιμή από το πιο «απομακρυσμένο» honeyword.

Μετά χρησιμοποιώντας τους πίνακες αυτούς υπολογίσαμε για κάθε στήλη την μέση απόσταση, y, και την τυπική απόκλιση, σ, με τελικό αποτέλεσμα ένας 2x19 πίνακας μία στήλη για κάθε honeyword, όπου η πρώτη γραμμή έχει τις μέσες αποστάσεις και η δεύτερη τις τυπικές αποκλίσεις (Πίνακες 3.7.1 – 3.7.4).

Πίνακας 3.7.1 – Μέσες αποστάσεις και τυπικές αποκλίσεις για το Edit Distance.

Honeyword1	Honeyword2	Honeyword3	...	Honeyword18	Honeyword19
1.337940843	3.82848508	4.276228	...	9.748328	10.14251428
0.62148042	2.1015184	2.084766	...	5.647097	6.06147129

Πίνακας 3.7.2 – Μέσες αποστάσεις και τυπικές αποκλίσεις για το Jaccard Distance.

Honeyword1	Honeyword2	Honeyword3	...	Honeyword18	Honeyword19
0.192701471	0.362792042	0.416709924	...	0.793576308	0.827603838
0.116487	0.187908	0.183428349	...	0.131075	0.119945

Πίνακας 3.7.3 – Μέσες αποστάσεις και τυπικές αποκλίσεις για το Cosine Distance.

Honeyword1	Honeyword2	Honeyword3	...	Honeyword18	Honeyword19
1.35E-05	1.86E-05	2.22E-05	...	0.000395	0.000738

8.08E-06	1.08E-05	1.32E-05	…	0.000378	0.00074

Πίνακας 3.7.4 – Μέσες αποστάσεις και τυπικές αποκλίσεις για το Euclidean Distance.

Honeyword1	Honeyword2	Honeyword3	…	Honeyword18	Honeyword19
0.486307	0.802265	1.114105		9.200173	11.37393
0.390318	0.60189	0.759443		3.048414	3.912827

Πίνακας 3.8.1 – Ταξινομημένος πίνακας τιμών της Edit Distance για την μέθοδο «Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords».

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	2	4	…	11	11
agent09100	3	3	…	9	9
newj3rs3y	1	5	…	9	9
bluebirdbliss	4	5	…	14	14
carusolombardi	1	7	…	13	13
6qnefb7k	1	6	…	16	16

Πίνακας 3.8.2 – Ταξινομημένος πίνακας τιμών της Jaccard Distance για την μέθοδο «Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords».

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	0.3	0.5	…	0.833333	0.923077
agent09100	0.25	0.3	…	0.833333	0.846154
newj3rs3y	0.222222	0.6	…	0.857143	1
bluebirdbliss	0.111111	0.2	…	0.857143	0.866667
carusolombardi	0.083333	0.307692	…	0.625	0.647059
6qnefb7k	0.222222	0.769231	…	1	1

Πίνακας 3.8.3 – Ταξινομημένος πίνακας τιμών της Cosine Distance για την μέθοδο «Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords».

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	2.2769E-05	3.14713E-05	…	0.000337	0.000546
agent09100	1.37687E-05	1.85966E-05	…	0.000253	0.000281
newj3rs3y	1.50204E-05	1.56164E-05	…	0.000405	0.003918
bluebirdbliss	1.09673E-05	1.21593E-05	…	0.000198	0.000234

carusolombardi	1.06692E-05	1.19805E-05	…	0.000121	0.00013
6qnefb7k	1.37091E-06	2.52724E-05	…	0.000334	0.000391

Πίνακας 3.8.4 – Ταξινομημένος πίνακας τιμών της Euclidean Distance για την μέθοδο «Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords».

Password	Honeyword1	Honeyword2	…	Honeyword18	Honeyword19
80irisnell	0.34244558	0.65640986	…	10.24719	11.92277
agent09100	0.42895618	0.468705952	…	7.37415	9.592788
newj3rs3y	0.58886784	1.153828382	…	13.04459	14.95182
bluebirdbliss	0.1843738	0.558090627	…	7.770987	13.59185
carusolombardi	0.78051704	1.353855968	…	6.092308	10.40471
6qnefb7k	0.28237671	0.685522974	…	9.163926	10.67729

Κεφάλαιο 4

Αξιολόγηση Αποτελεσμάτων

4.1 Εισαγωγή Κεφαλαίου

Η ανάλυση των αποτελεσμάτων των επιθέσεων σε ένα σύστημα που παράγει honeywords αποτελεί σημαντικό κομμάτι της συγκεκριμένης διπλωματικής εργασίας. Στόχος της μελέτης είναι να αξιολογήσουμε την απόδοση του συστήματος και να αναλύσουμε τις διάφορες τεχνικές επιθέσεων που εφαρμόστηκαν για τη μείωση του συνόλου των honeywords, προκειμένου να δούμε εάν αποτελεί απειλή για τη λειτουργία του συστήματος HoneyGen και εάν απαιτείται η λήψη μέτρων. Για την αξιολόγηση αυτή δημιουργήσαμε τέσσερα σχετικά γραφήματα για κάθε αρχείο (13 αρχεία) τα οποία αξιολογούν την κάθε μετρική απόστασης (Edit, Jaccard, Cosine, Euclidean) και τα οποία θα βοηθήσουν στο τέλος του κεφαλαίου αυτού όπου θα γίνει σύγκριση των δύο ευρετικών μεθόδων.

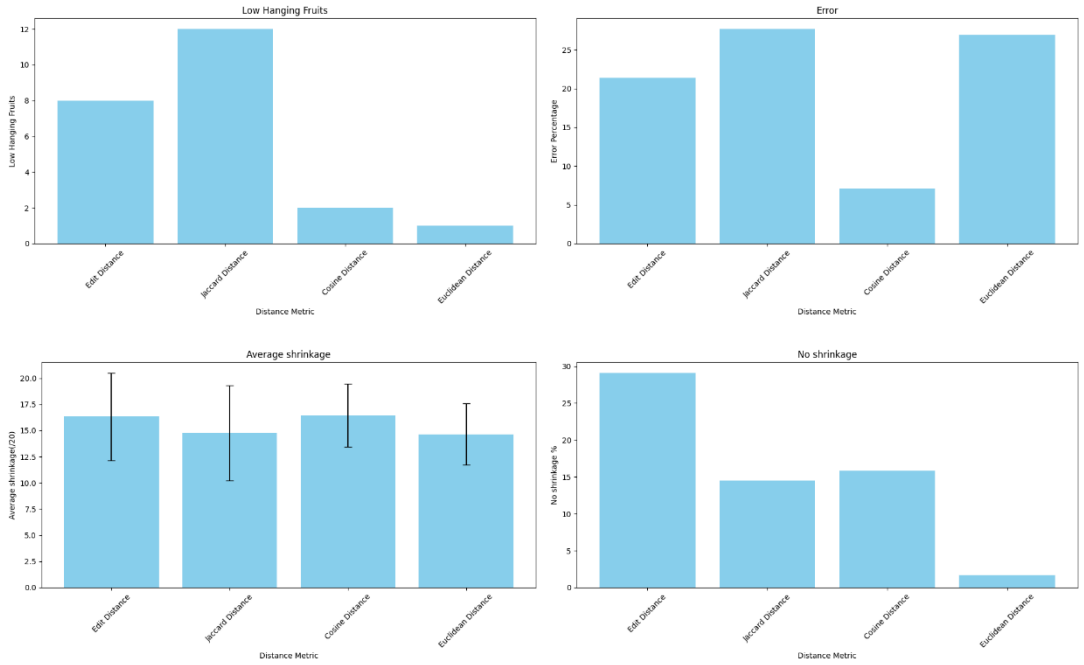
Τα γραφήματα αυτά παρουσιάζουν:

- Τη μέση μείωση του συνόλου των honeywords (Average Shrinkage) μαζί με την τυπική απόκλιση
- Το ποσοστό αποτυχίας μείωσης του σετ (No Shrinkage)
- Το ποσοστό σφάλματος (Error) : σφάλμα θεωρείτε όταν το ευρετικό αν και μείωσε το σύνολο των honeywords, στην προσπάθεια του να το μειώσει απέκλεισε και τον πραγματικό κωδικό.
- Low hanging fruits: Low hanging fruits είναι οι περιπτώσεις στις οποίες ο αλγόριθμος βρήκε τον σωστό κωδικό και είναι ένα από τα κύρια στατιστικά τα οποία πρέπει να δούμε διότι το πιο πιθανό ένας επιτιθέμενος μπορεί να δει μόνο

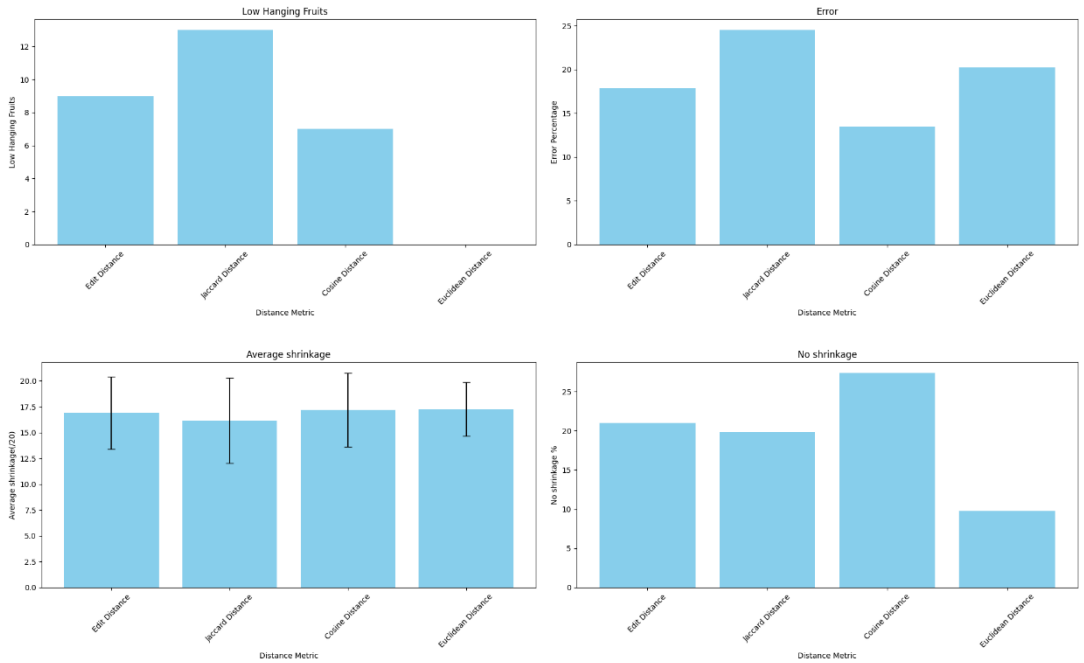
αυτές τις περιπτώσεις οι οποίες του δίνουν μία πιθανή λύση και όχι αυτές που απλά μείωσαν το αρχικό σύνολο σε ένα πιο μικρό.

4.2 Όριο Μέσης Απόστασης

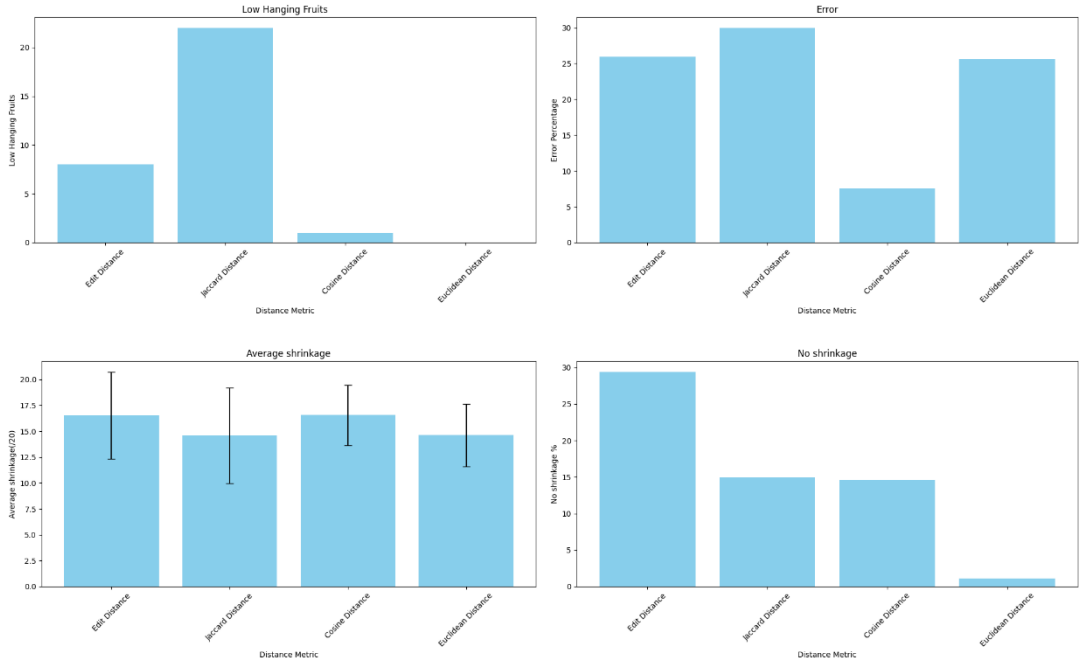
Στο παρόν υποκεφάλαιο, θα εξετάσουμε την ευρετική μέθοδο του «Ορίου Μέσης Απόστασης» ως μια από τις τεχνικές επίθεσης που χρησιμοποιούνται για τη μείωση του συνόλου των honeywords στο σύστημα HoneyGen, και θα εξετάσουμε την αποτελεσματικότητά της στην πράξη. Με την παρουσίαση των σχετικών γραφημάτων και δεδομένων, θα επιχειρήσουμε να αναδείξουμε τις δυνατότητες και τις περιορισμούς αυτής της μεθόδου στο πλαίσιο της προστασίας του συστήματος αυθεντικοποίησης.



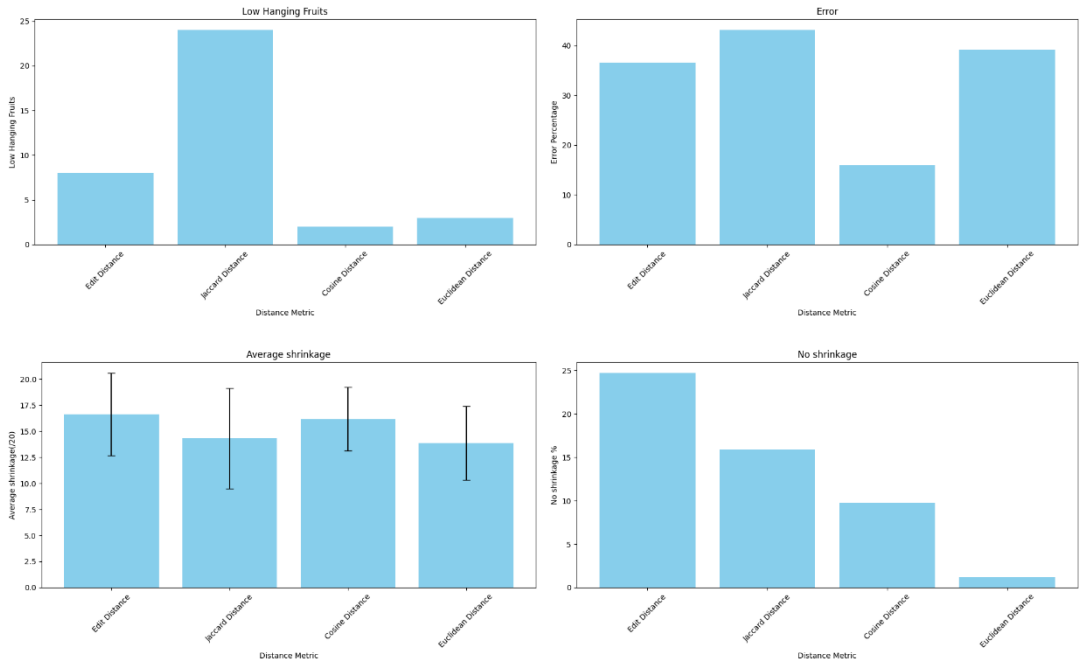
Σχήμα 4.1.1 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Adult FriendFinder.



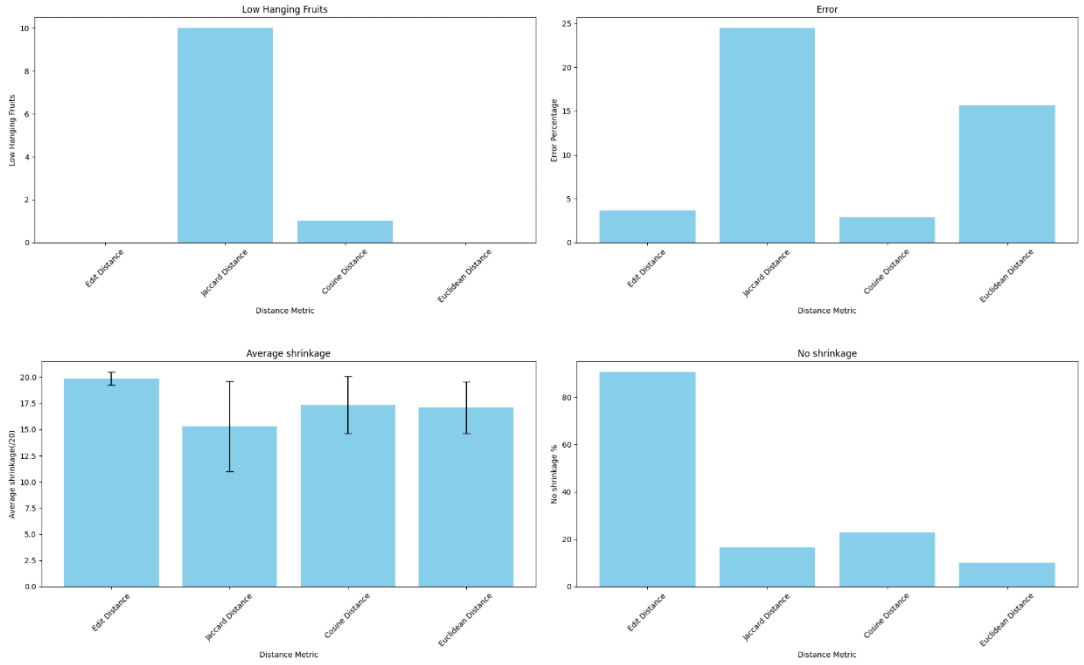
Σχήμα 4.1.2 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Chegg.



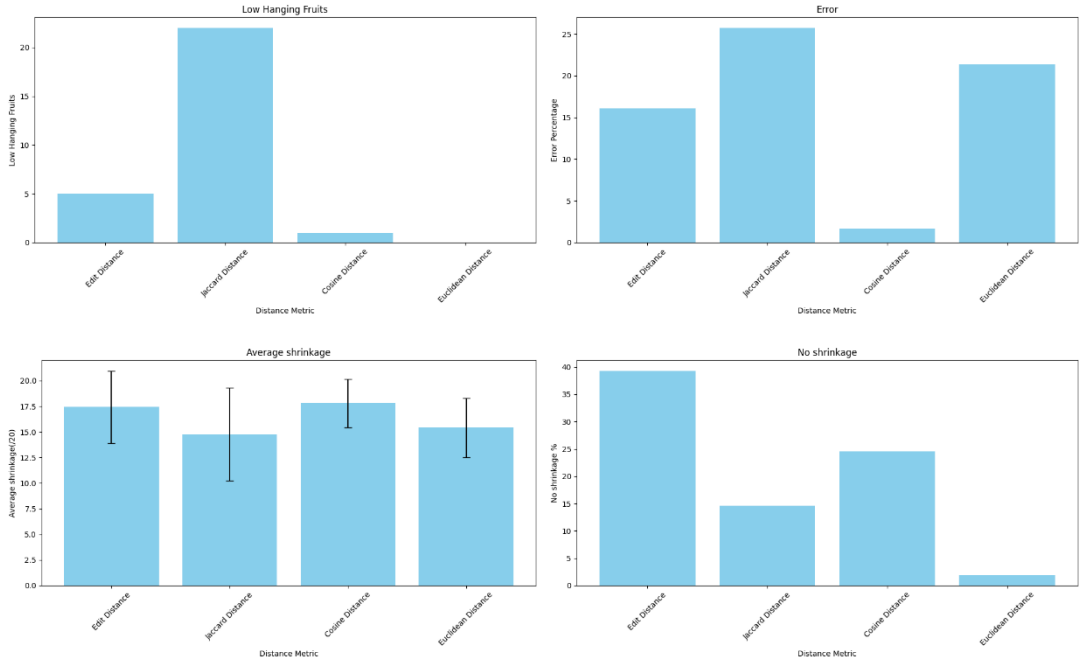
Σχήμα 4.1.3 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Dropbox.



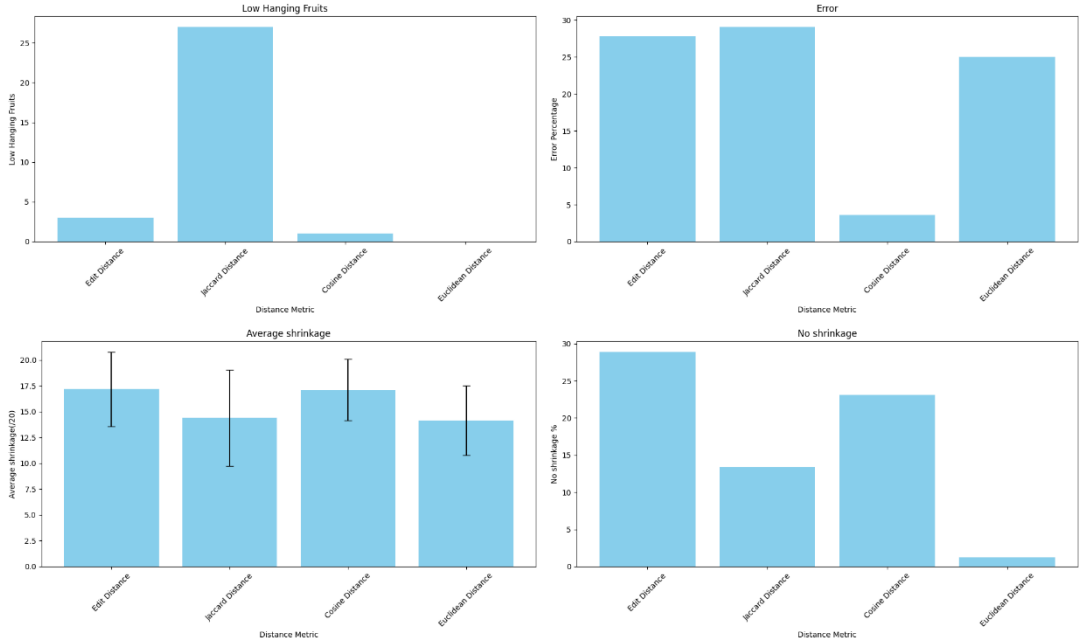
Σχήμα 4.1.4 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Dubsmash.



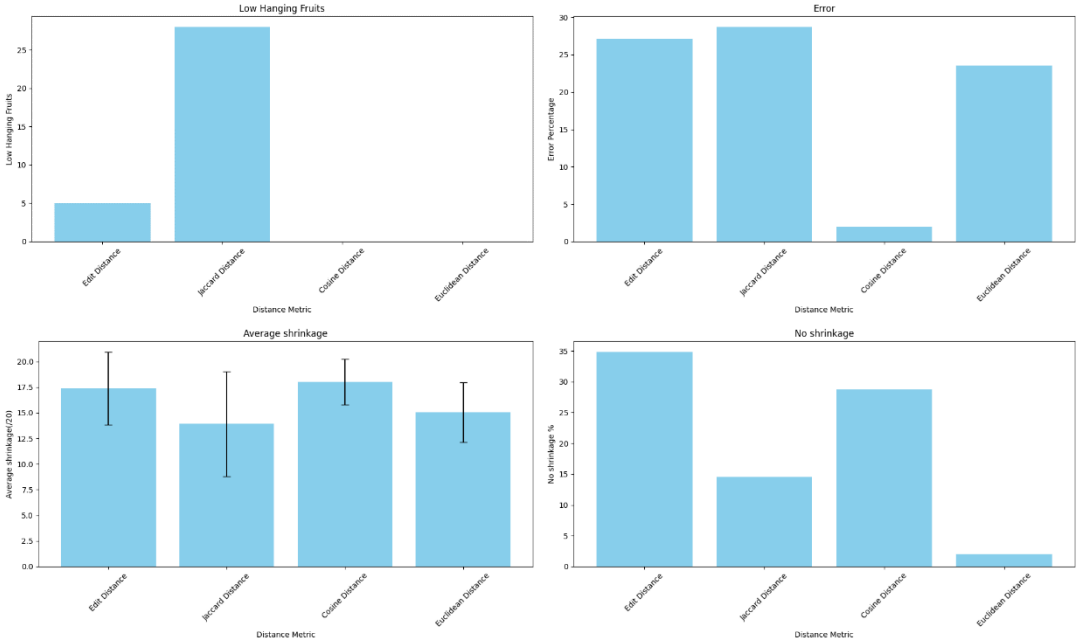
Σχήμα 4.1.5 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Have-I-Been-Pawned.



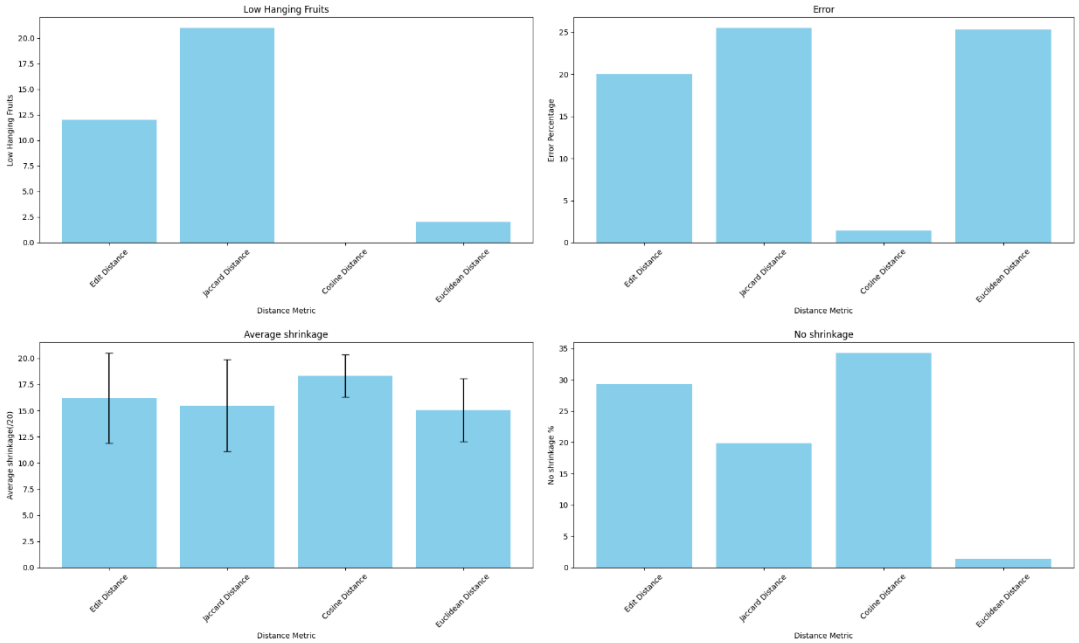
Σχήμα 4.1.6 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Last-Fm.



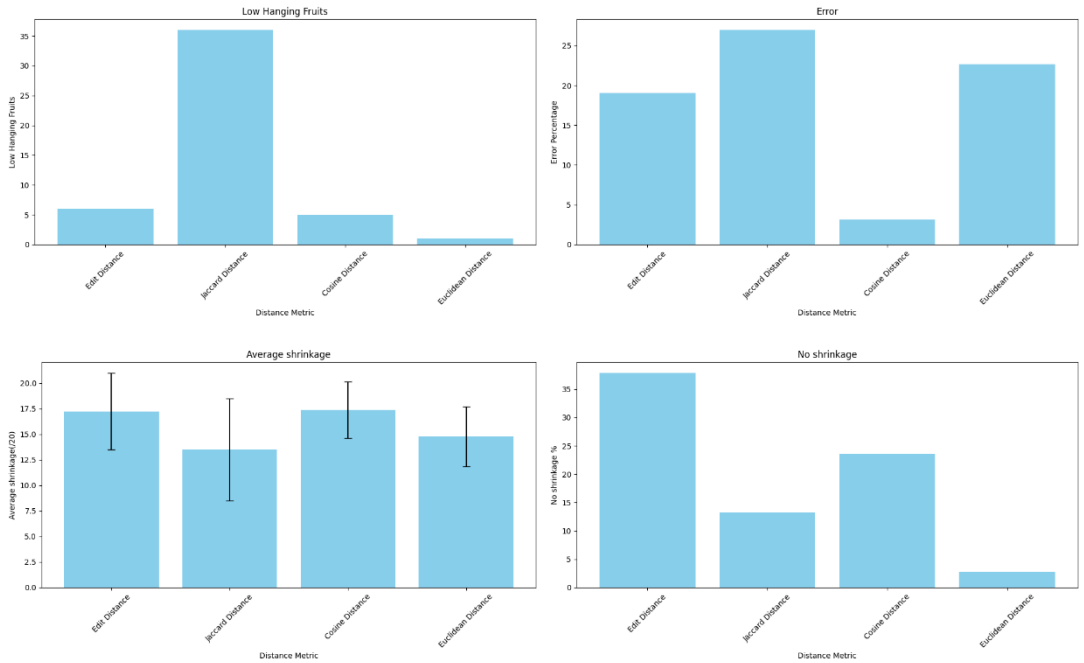
Σχήμα 4.1.7 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας LinkedIn.



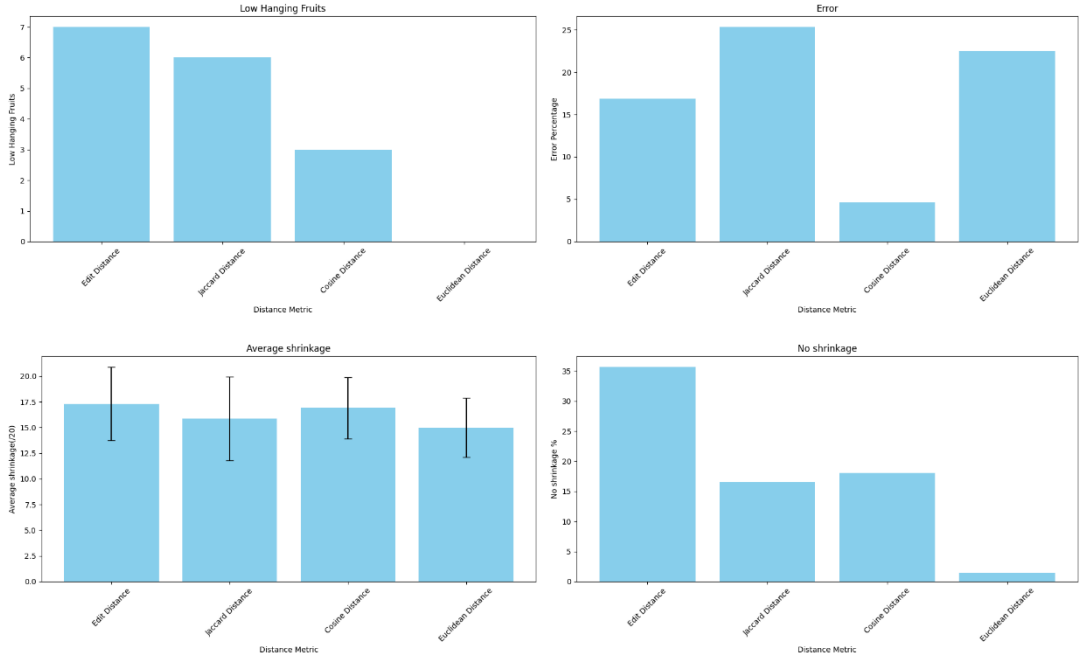
Σχήμα 4.1.8 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας MySpace.



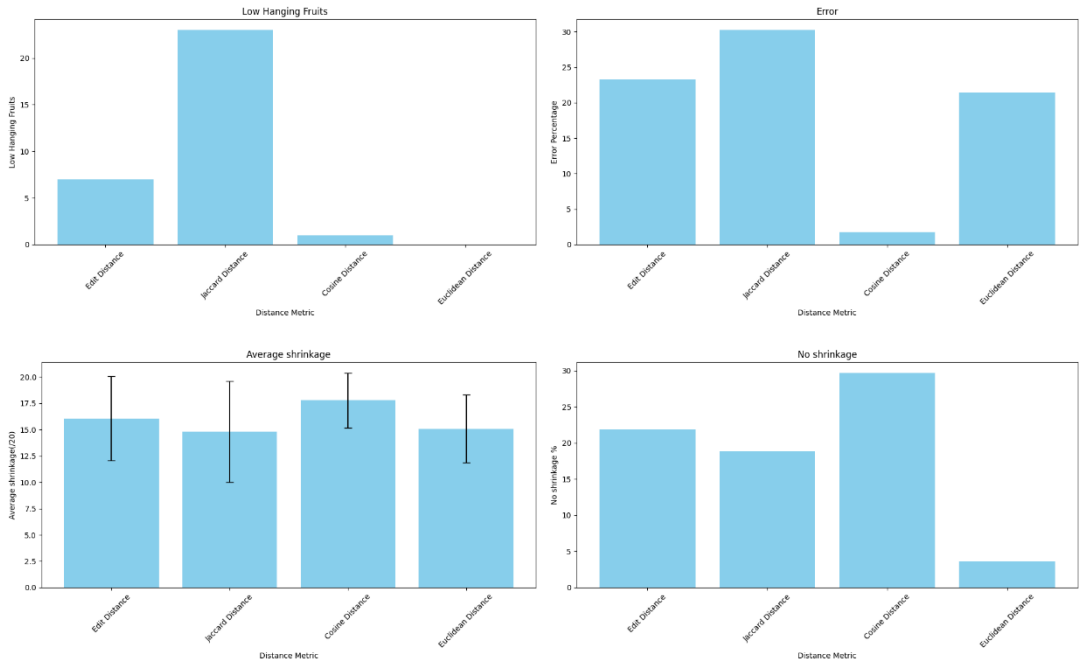
Σχήμα 4.1.9 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας phpBB.



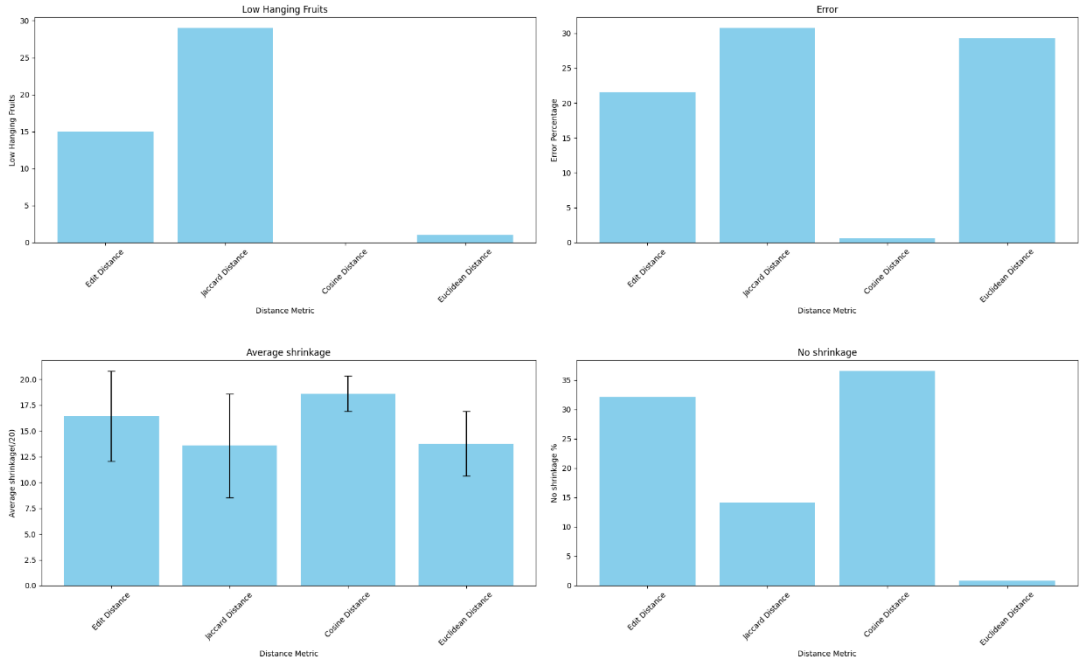
Σχήμα 4.1.10 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Rockyou.



Σχήμα 4.1.11 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Yahoo.



Σχήμα 4.1.12 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Youku.



Σχήμα 4.1.13 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Zynga.

Low hanging fruits:

Η ανάλυση των αποτελεσμάτων ξεκινά με τον σχολιασμό των ευρημάτων για τις μετρικές Edit Distance, Jaccard Distance, Cosine Distance και Euclidean Distance για όλα τα αρχεία, επικεντρώνοντας αρχικά στις απλούστερες περιπτώσεις, γνωστές και ως “low hanging fruits”.

Ας εξετάσουμε πρώτα τα αποτελέσματα για τα low hanging fruits χρησιμοποιώντας τη μετρική Edit Distance. Η καλύτερη τιμή, δηλαδή το μεγαλύτερο πλήθος σωστών κωδικών που εντοπίστηκαν, παρατηρείται όταν είχαμε dataset το αρχείο Zynga (Σχήμα 4.1.13), όπου είναι ίση με 15. Η δεύτερη καλύτερη τιμή εντοπίστηκε για το dataset phpBB (Σχήμα 4.1.9) και είναι ίση με 12. Η χειρότερη τιμή για αυτήν την μετρική είναι ίση με 0, δηλαδή δεν βρέθηκε σε καμία περίπτωση ο σωστός κωδικός και μπορούμε να την δούμε στο Σχήμα 4.1.5 το οποίο είναι για το αρχείο Have-I-Been-Pawned. Οι τιμές για τα υπόλοιπα αρχεία έχουν ως εξής: για το αρχείο LinkedIn (Σχήμα 4.1.7) η τιμή ήταν ίση με 3, για τα αρχεία Last-FM (Σχήμα 4.1.6) και Myspace (Σχήμα 4.1.8) έχουμε την τιμή 5, για το αρχείο Rockyou (Σχήμα 4.1.10) έχουμε την τιμή 6, για τα αρχεία Yahoo (Σχήμα 4.1.11) και Youku (Σχήμα 4.1.12) έχουμε την τιμή 7, για τα αρχεία Adult FriendFinder (Σχήμα 4.1.1), Dropbox (Σχήμα 4.1.3) και Dubsmash (Σχήμα 4.1.4) έχουμε την τιμή 8 και τέλος στο αρχείο Chegg (Σχήμα 4.1.2) την τιμή 9.

Η επόμενη μετρική απόστασης που θα αναλύσουμε είναι η Jaccard Distance. Για αυτήν την μετρική έχουμε ως καλύτερη τιμή, την τιμή 36 η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο RockYou (Σχήμα 4.1.10). Η δεύτερη καλύτερη τιμή εντοπίστηκε για dataset το αρχείο Zynga (Σχήμα 4.1.13) και είναι ίση με 29. Η χειρότερη τιμή για αυτήν την τεχνική είναι ίση με το 6 και εντοπίστηκε με dataset το αρχείο Yahoo (Σχήμα 4.1.11). Τέλος οι υπόλοιπες τιμές σε αύξουσα σειρά για αυτήν την μετρική είναι 10 για το αρχείο Have -I-Been-Pawned (Σχήμα 4.1.5), 12 για το αρχείο Adult FriendFinder (Σχήμα 4.1.1), 13 για το αρχείο Chegg (Σχήμα 4.1.2), 21 για το αρχείο phpBB (Σχήμα 4.1.9), 22 για τα αρχεία Dropbox (Σχήμα 4.1.3) και Last-FM (Σχήμα 4.1.6), 23 για το αρχείο Youku (Σχήμα 4.1.12), 24 για το αρχείο Dubsmash (Σχήμα 4.1.4), 27 για το αρχείο LinkedIn (Σχήμα 4.1.7) και 28 για το αρχείο Myspace(Σχήμα 4.1.8).

Η επόμενη μετρική είναι η Cosine Distance. Για αυτήν την μετρική έχουμε ως καλύτερη τιμή, την τιμή 7 η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Chegg (Σχήμα

4.1.2). Η δεύτερη καλύτερη τιμή εντοπίστηκε για dataset το αρχείο RockYou (Σχήμα 4.1.10) και είναι ίση με 5. Η χειρότερη τιμή για αυτήν την τεχνική είναι ίση με το 0 και εντοπίστηκε με dataset πολλά αρχεία και πιο συγκεκριμένα τα αρχεία MySpace (Σχήμα 4.1.8), PhpBB (Σχήμα 4.1.9) και Zynga (Σχήμα 4.1.13). Τέλος οι υπόλοιπες τιμές για αυτήν την μετρική είναι 1 για τα αρχεία Dropbox (Σχήμα 4.1.3), Have -I-Been-Pwned (Σχήμα 4.1.5), Last-FM (Σχήμα 4.1.6), LinkedIn (Σχήμα 4.1.7),και Youku (Σχήμα 4.1.12), 2 για τα αρχεία Adult FriendFinder (Σχήμα 4.1.1) και Dubsmash (Σχήμα 4.1.4) και 3 για το αρχείο Yahoo (Σχήμα 4.1.11).

Η τελευταία μετρική είναι η Euclidean Distance για την οποία έχουμε ως καλύτερη τιμή, την τιμή 3 η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Dubsmash (Σχήμα 4.1.4). Η δεύτερη καλύτερη τιμή εντοπίστηκε για dataset το αρχείο phpBB (Σχήμα 4.1.9) και είναι ίση με 2. Η χειρότερη τιμή για αυτήν την τεχνική είναι ίση με το 0 και εντοπίστηκε με dataset πολλά αρχεία και πιο συγκεκριμένα τα αρχεία Chegg (Σχήμα 4.1.2), Dropbox (Σχήμα 4.1.3), Have -I-Been-Pwned (Σχήμα 4.1.5), Last-FM (Σχήμα 4.1.6), LinkedIn (Σχήμα 4.1.7), MySpace (Σχήμα 4.1.8), Yahoo (Σχήμα 4.1.11) και Youku (Σχήμα 4.1.12). Τέλος οι υπόλοιπες τιμές για αυτήν την μετρική είναι 1 για όλα τα υπόλοιπα αρχεία, δηλαδή το Adult FriendFinder (Σχήμα 4.1.1), το RockYou (Σχήμα 4.1.10) και το Zynga (Σχήμα 4.1.13).

Κάνοντας μία απλή σύγκριση των αποτελεσμάτων για αυτό το στατιστικό μπορούμε να δούμε πως τα καλύτερα αποτελέσματα τα έχει η μετρική Jaccard Distance με την μόνη εξαίρεση να γίνεται στο αρχείο Yahoo στο οποίο η καλύτερη μετρική είναι η Edit Distance όμως για πολύ λίγο (7 το Edit Distance και 6 το Jaccard Distance). Επιπλέον, μπορούμε να δούμε πως η χειρότερη μετρική είναι το Euclidean distance το οποίο έχει τιμή 0 για τα περισσότερα αρχεία, δηλαδή δεν καταλήγουμε σχεδόν ποτέ στο σύνολο όλων των 5000 leaked κωδικών που έχουν το κάθε testing αρχείο σε ένα σετ από honeywords που να περιέχει μόνο τον σωστό κωδικό. Τέλος θα ήθελα να αναφέρω πως αν και τα αποτελέσματα μοιάζουν μικρά με την πρώτη ματιά, είναι σημαντικά στο πλαίσιο της ασφάλειας δικτύων. Η εκμετάλλευση ακόμα και λίγων "low hanging fruits" μπορεί να επιτρέψει σε επιτιθέμενους να αποκτήσουν πρόσβαση και να εκβιάσουν τους χρήστες.

Ας προχωρήσουμε στην ανάλυση των υπόλοιπων στατιστικών: ποσοστό αποτυχίας μείωσης του σετ, ποσοστό σφάλματος και μέση μείωση του σετ. Θα εξετάσουμε αυτά τα στατιστικά με τον ίδιο τρόπο όπως και προηγουμένως, ξεκινώντας με την ανάλυση των αποτελεσμάτων για κάθε μετρική απόστασης σε όλα τα αρχεία και στο τέλος, θα πραγματοποιήσουμε μια σύγκριση των αποτελεσμάτων της κάθε μετρικής για να αποκτήσουμε μια πιο ολοκληρωμένη εικόνα.

Ποσοστό σφάλματος

Ας εξετάσουμε πρώτα τα αποτελέσματα για το "ποσοστό σφάλματος" ξεκινώντας όπως και πριν με τη μετρική Edit Distance. Η καλύτερη τιμή, δηλαδή το μικρότερο ποσοστό σφάλματος που εντοπίστηκε για αυτήν την μετρική, παρατηρείται όταν είχαμε dataset το αρχείο Have-I-Been-Pwned (Σχήμα 4.1.5), όπου είναι ίσο με 3.68%. Η δεύτερη καλύτερη τιμή εντοπίστηκε όταν το dataset ήταν το αρχείο Last-FM (Σχήμα 4.1.6) και είναι ίσο με 16.08 %. Το χειρότερο ποσοστό για αυτήν την μετρική είναι ίσο με 36.64%, δηλαδή περίπου στο 1/3 των σετ αφαιρέθηκε και ο σωστός κωδικός, αυτό το ποσοστό είναι στο Σχήμα 4.1.4 το οποίο είναι για το αρχείο Dubsmash. Τα ποσοστά σφάλματος για τα υπόλοιπα αρχεία έχουν ως εξής: για τα αρχεία Yahoo (Σχήμα 4.1.11) και Chegg (Σχήμα 4.1.2) η τιμή ήταν περίπου ίση με 17%, για τα αρχεία Adult FriendFinder (Σχήμα 4.1.1), phpBB (Σχήμα 4.1.9), Rockyou (Σχήμα 4.1.10) και Zynga (Σχήμα 4.1.13) έχουμε τιμή περίπου ίση με 20%, για το αρχείο Youku (Σχήμα 4.1.12) έχουμε τιμή περίπου 23%, για το αρχείο Dropbox (Σχήμα 4.1.3) περίπου 26% και για τα αρχεία LinkedIn (Σχήμα 4.1.7) και Myspace (Σχήμα 4.1.8) έχουμε περίπου την τιμή 27.

Η επόμενη μετρική απόστασης που θα αναλύσουμε είναι η Jaccard Distance. Για αυτήν την μετρική το καλύτερο ποσοστό σφάλματος, έχει την τιμή 24.48% η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Have-I-Been-Pwned (Σχήμα 4.1.5). Η δεύτερη καλύτερη τιμή είναι πάρα πολύ κοντά στην πρώτη, εντοπίστηκε όταν είχαμε ως dataset το αρχείο Chegg (Σχήμα 4.1.2) και είναι ίση με 24.5%. Το χειρότερο ποσοστό για αυτήν την τεχνική είναι ίσο με το 43.16% και εντοπίστηκε στο dataset αρχείο Dubsmash (Σχήμα 4.1.4). Τέλος οι υπόλοιπες τιμές σε αύξουσα σειρά για αυτήν την μετρική είναι περίπου 25% για τα αρχεία Last-FM (Σχήμα 4.1.6), phpBB (Σχήμα 4.1.9), Yahoo (Σχήμα 4.1.11), για το αρχείο Adult FriendFinder (Σχήμα 4.1.1) και το αρχείο

Rockyou (Σχήμα 4.1.10) έχουμε ποσοστό σφάλματος περίπου 27% και για τα αρχεία LinkedIn (Σχήμα 4.1.7) και MySpace (Σχήμα 4.1.8) ποσοστό περίπου 29%. Τέλος για τα αρχεία Dropbox (Σχήμα 4.1.3), Youku (Σχήμα 4.1.12) και Zynga (Σχήμα 4.1.13) έχουμε ποσοστό περίπου 30%.

Η επόμενη μετρική είναι η Cosine Distance. Για αυτήν την μετρική το καλύτερο ποσοστό σφάλματος, έχει την τιμή 0.64% η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Zynga (Σχήμα 4.1.13). Η δεύτερη καλύτερη τιμή εντοπίστηκε με dataset το αρχείο PhpBB (Σχήμα 4.1.9) και είναι ίση με 1.42% με πολύ κοντινές τις τιμές να είναι και το αρχείο Last-FM (Σχήμα 4.1.6), Youku (Σχήμα 4.1.12) και MySpace (Σχήμα 4.1.8) των οποίων τα ποσοστά είναι 1.68%, 1.78% και 1.94% αντίστοιχα. Η χειρότερη τιμή για αυτήν την τεχνική είναι ίση με το 15.96% και εντοπίστηκε με dataset το αρχείο Dubsmash (Σχήμα 4.1.4). Τέλος οι υπόλοιπες τιμές για αυτήν την μετρική είναι για τα αρχεία Have -I-Been-Pwned (Σχήμα 4.1.5) και LinkedIn (Σχήμα 4.1.7) περίπου 3% και για τα αρχεία RockYou (Σχήμα 4.1.10) και Yahoo (Σχήμα 4.1.11) περίπου 5%. Τα αρχεία Adult FriendFinder (Σχήμα 4.1.1) και Dropbox (Σχήμα 4.1.3) έχουν ποσοστό περίπου 7%. Τέλος το αρχείο Chegg έχει τη δεύτερη χειρότερη τιμή η οποία είναι 13.44%.

Η τελευταία μετρική είναι η Euclidean Distance για την οποία έχουμε ως καλύτερη τιμή, το ποσοστό 15.66% η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Have-I-Been-Pawned (Σχήμα 4.1.5). Η δεύτερη καλύτερη τιμή εντοπίστηκε με dataset το αρχείο Chegg (Σχήμα 4.1.2) και είναι ίση με 20.22%. Η χειρότερη τιμή για αυτήν την τεχνική είναι ίση με το 39.28% και εντοπίστηκε με dataset το αρχείο Dubsmash (Σχήμα 4.1.4). Οι υπόλοιπες τιμές για αυτήν την μετρική είναι περίπου 21% για τα αρχεία Last-FM (Σχήμα 4.1.6) και Youku (Σχήμα 4.1.12). Πολύ κοντά βρίσκονται και τα αρχεία RockYou (Σχήμα 4.1.10) και Yahoo (Σχήμα 4.1.11) των οποίων τα ποσοστά λάθους είναι γύρω στο 22%. Επιπλέον έχουμε το MySpace (Σχήμα 4.1.8) το οποίο έχει ποσοστό 23% και τα αρχεία Dropbox (Σχήμα 4.1.3), LinkedIn (Σχήμα 4.1.7) και phpBB (Σχήμα 4.1.9) τα οποία έχουν ποσοστό περίπου ίσο με 25% και σε μικρή απόσταση το αρχείο Adult FriendFinder (Σχήμα 4.1.1) το οποίο έχει ποσοστό 26%. Τέλος το αρχείο Zynga (Σχήμα 4.1.13) έχει ποσοστό κοντά στο 29%.

Στην σύγκριση των αποτελεσμάτων για το στατιστικό του ποσοστού σφάλματος τα στατιστικά για τις μετρικές Edit Distance, Jaccard Distance και Euclidean Distance είναι πολύ κοντά. Όμως η μετρική Cosine Distance έχει τα καλύτερα αποτελέσματα με μεγάλη διαφορά από τις άλλες. Αυτό μπορεί να συμβαίνει διότι όπως είπα και προηγουμένως στην παραγωγή honeywords με το σύστημα HoneyGen υπάρχει ένα βήμα το οποίο χρησιμοποιεί το cosine similarity. Τώρα αν συγκρίνουμε με βάση τις τιμές της κάθε μετρικής απόστασης από όλα τα αρχεία τότε θα δούμε πως αυτή που έχει πάντοτε την μεγαλύτερη τιμή σφάλματος είναι η Jaccard Distance η οποία πριν είχε τα καλύτερα αποτελέσματα, και μετά ακολουθεί η Euclidean Distance η οποία είχε και πριν κακά αποτελέσματα.

Μέσο ποσοστό μείωσης του σετ

Το επόμενο στατιστικό το οποίο θα εξεταστεί είναι το "μέσο ποσοστό μείωσης του σετ" ξεκινώντας όπως και τις προηγούμενες φορές με τη μετρική Edit Distance.

Η καλύτερη τιμή σε αυτό το στατιστικό είναι το μικρότερο ποσοστό μείωσης που καταγράφηκε. Διότι, το ποσοστό αυτό, αναφέρεται στο μέγεθος των σετ από τα 20 μετά την εφαρμογή της ευρετικής μεθόδου. Για αυτήν την μετρική, η καλύτερη τιμή παρατηρήθηκε με dataset το αρχείο Youku (Σχήμα 4.1.12), όπου έμειναν κατά μέσο όρο 16.1/20 honeywords. Η δεύτερη καλύτερη τιμή εντοπίστηκε όταν το dataset ήταν το αρχείο phpBB (Σχήμα 4.1.9) και είναι ίσο με 16.2/20. Όμως γενικά υπάρχουν πολύ κοντινά ποσοστά σε αυτά τα δύο, όπως το 16.3/20 στο αρχείο Adult FriendFinder (Σχήμα 4.1.1), 16.4/20 στο Zynga (Σχήμα 4.1.13) και 16.5 στο Dropbox (Σχήμα 4.1.3). Το χειρότερο ποσοστό μείωσης για αυτήν την μετρική είναι ίσο με 19.9/20, δηλαδή τις παραπάνω φορές να μην έχει μειωθεί σχεδόν καθόλου. Αυτό το ποσοστό είναι στο Σχήμα 4.1.5 το οποίο είναι για το αρχείο Have-I-Been-Pwned. Τα υπόλοιπα ποσοστά μείωσης για τα αρχεία Chegg (Σχήμα 4.1.2), Last-Fm (Σχήμα 4.1.6), LinkedIn (Σχήμα 4.1.7), Myspace (Σχήμα 4.1.8), Rockyou (Σχήμα 4.1.10) και Yahoo (Σχήμα 4.1.11), είναι όλα πολύ κοντά στο 17/20 απομένοντα honeywords.

Η επόμενη μετρική απόστασης που θα αναλύσουμε είναι η Jaccard Distance. Για αυτήν την μετρική το καλύτερο ποσοστό μέσης μείωσης, έχει την τιμή 13.5/20 η οποία

παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Rockyou (Σχήμα 4.1.10) και πολύ κοντά στα 13.6/20 είναι και το Zynga (Σχήμα 4.1.13). Η επόμενη καλύτερη τιμή εντοπίστηκε όταν είχαμε ως dataset το αρχείο MySpace (Σχήμα 4.1.8) και είναι ίση με 13.9/20. Το χειρότερο ποσοστό για αυτήν την τεχνική είναι ίσο με το 16.2/20 και εντοπίστηκε στο dataset αρχείο Chegg (Σχήμα 4.1.2), πολύ κοντά σε αυτή την τιμή είναι και το αρχείο Yahoo με ποσοστό 15.9/20. Τέλος οι υπόλοιπες τιμές για αυτήν την μετρική είναι περίπου στο 14/20 για τα αρχεία LinkedIn (Σχήμα 4.1.7) και Dubsmash (Σχήμα 4.1.4), ενώ στα υπόλοιπα αρχεία (Adult FriendFinder (Σχήμα 4.1.1), Dropbox (Σχήμα 4.1.3), Have-I-Been-Pwned (Σχήμα 4.1.5), Last-Fm (Σχήμα 4.1.6), phpBB (Σχήμα 4.1.9) και Youku (Σχήμα 4.1.12) τα ποσοστά είναι κοντά στο 15/20.

Η επόμενη μετρική είναι η Cosine Distance. Για αυτήν την μετρική το καλύτερο ποσοστό μείωσης, έχει την τιμή 16.2/20 η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Dubsmash (Σχήμα 4.1.4). Η δεύτερη καλύτερη τιμή εντοπίστηκε με dataset το αρχείο Dropbox (Σχήμα 4.1.3) και στο αρχείο Adult FriendFinder (Σχήμα 4.1.1) είναι ίση με 16.5/20 και 16.6 αντίστοιχα. Η χειρότερες τιμές για αυτήν την τεχνική είναι αυτές που είναι περίπου στα 18/20 στα αρχεία: Last-FM (Σχήμα 4.1.6), MySpace (Σχήμα 4.1.8), phpBB (Σχήμα 4.1.9) και Zynga (Σχήμα 4.1.13) και την χειρότερη από αυτές να είναι το 18.6/20 στο αρχείο Zynga. Τα αρχεία που μένουν έχουν ποσοστό κοντά στο 17/20.

Η τελευταία μετρική είναι η Euclidean Distance για την οποία έχουμε ως καλύτερη τιμή, το ποσοστό 13.8/20 η οποία παρατηρήθηκε όταν είχαμε ως dataset το αρχείο Zynga (Σχήμα 4.1.13). Η δεύτερη καλύτερη τιμή εντοπίστηκε με dataset το αρχείο Dubsmash (Σχήμα 4.1.4) και είναι πολύ κοντινή με την καλύτερη, με ποσοστό 13.9/20. Η χειρότερη τιμή για αυτήν την μετρική είναι ίση με το 17.2/20 στο αρχείο Chegg (Σχήμα 4.1.2) και σε κοντινή απόσταση από αυτήν το αρχείο Have-I-Been-Pwned (Σχήμα 4.1.5) με το ποσοστό 17.1/20. Οι υπόλοιπες τιμές για αυτήν την μετρική είναι 14.2 και 14.6 για τα αρχεία LinkedIn (Σχήμα 4.1.7) και Dropbox (Σχήμα 4.1.3) αντίστοιχα, ενώ στα υπόλοιπα αρχεία το ποσοστό είναι περίπου 15/20.

Στην σύγκριση των αποτελεσμάτων για το στατιστικό του ποσοστού σφάλματος τα στατιστικά για την ίδια μετρική απόστασης είναι πολύ κοντινά το ένα με το άλλο και

έχουν πολύ μικρή απόκλιση. Εξαίρεση έχουμε για την μετρική Edit Distance και συγκεκριμένα στο αρχείο Have-I-Been-Pwned στο οποίο βλέπουμε μία πολύ μικρή μέση μείωση των σετ. Η μετρική η οποία έχει τα καλύτερα αποτελέσματα είναι η Jaccard Distance στην οποία τα ποσοστά στα παραπάνω αρχεία είναι περίπου 14/20. Για την μετρική με τα χειρότερα αποτελέσματα βρίσκονται ισοδύναμες οι μετρικές Edit Distance και Cosine Distance οι οποίες στα παραπάνω αρχεία έχουν μείωση του σετ μόνο στα 17/20. Εδώ θα μπορούσαμε να σχολιάσουμε επίσης και το ότι τα αποτελέσματα, φαίνονται απόλυτα λογικά αφού η Cosine Distance η οποία είχε το πιο μικρό σφάλμα όμως κακά αποτελέσματα στα "low hanging fruits" σε αυτήν την μετρική έχουμε πολύ μικρή μείωση. Ενώ στην μετρική την οποία είχε τα καλύτερα αποτελέσματα στα low hanging fruits και τα χειρότερα αποτελέσματα στο ποσοστό σφάλματος, την Jaccard Distance, σε αυτήν την μετρική έχει την μεγαλύτερη μείωση του σετ. Αυτό είναι λογικό αφού όσο πιο πολύ μειωθεί το σετ τόσο πιο πολύ ρίσκο υπάρχει να αφαιρεθεί ο πραγματικός κωδικός από το σετ.

Ποσοστό αποτυχίας μείωσης του σετ

Τελευταίο στατιστικό το οποίο θα αναλύσουμε περιεκτικά είναι το ποσοστό αποτυχίας μείωσης. Δηλαδή το ποσοστό των σετ σε κάθε αρχείο στα οποία οι μετρικές δεν κατάφεραν να μειώσουν το μέγεθος τους. Αυτό το στατιστικό μοιάζει πολύ με το προηγούμενο για αυτό και θα αναφέρω μόνο την καλύτερη και χειρότερη περίπτωση για την κάθε μετρική.

Στην Edit Distance, καλύτερη περίπτωση ήταν στο αρχείο Chegg (Σχήμα 4.1.2) και στο αρχείο Youku (Σχήμα 4.1.12) με το ποσοστό καθόλου μείωσης και στις δύο περιπτώσεις να είναι κοντά στο 21%. Στην χειρότερη περίπτωση την οποία μπορούμε να αντιληφθούμε και μέσα από το προηγούμενο στατιστικό ήταν στο αρχείο Have-I-Been-Pwned (Σχήμα 4.1.5) το οποίο είχε 90% καθόλου μείωση του σετ άρα το 90% είχε μείνει στο μέγεθος 20.

Στην μετρική απόστασης Jaccard Distance μπορούμε να δούμε πως την καλύτερη επιτυχία μείωσης την είχαμε σε δύο αρχεία το Linked-In (Σχήμα 4.1.7) και Rockyou

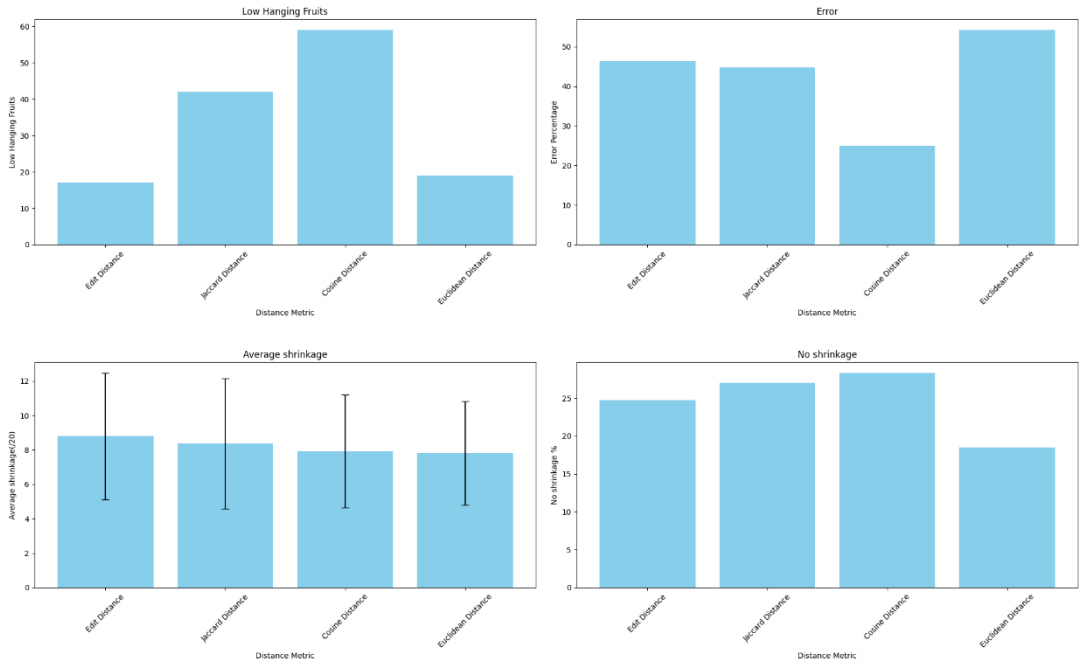
(Σχήμα 4.1.10) με ποσοστό αποτυχίας 13%. Ενώ την χειρότερη περίπτωση την έχουμε στα αρχεία Chegg (Σχήμα 4.1.2) και phpBB (Σχήμα 4.1.9) με ποσοστό αποτυχίας 20%.

Στην μετρική απόστασης Cosine Distance έχουμε την καλύτερη περίπτωση στο αρχείο Dubsmash (Σχήμα 4.1.4) με 10% όμως είναι εξαίρεση εφόσον πάνω από τα μισά αρχεία έχουν ποσοστό αποτυχίας μείωσης 23% και πάνω με το χειρότερο να είναι στο Zynga (Σχήμα 4.1.13) με ποσοστό 37%.

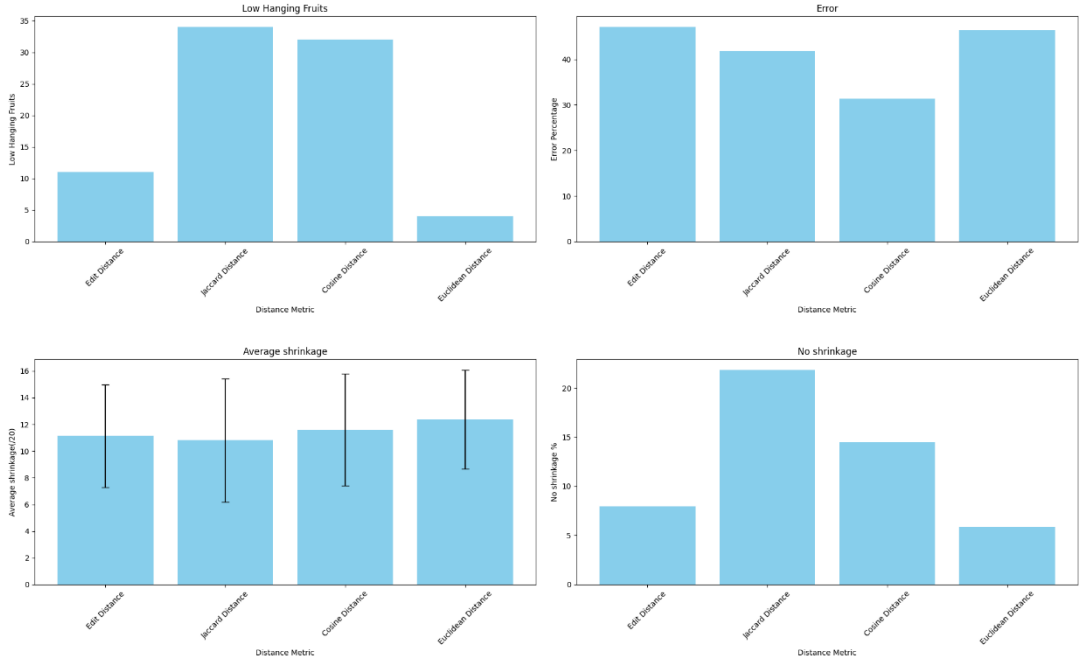
Τέλος στην μετρική Euclidean Distance είναι και η μετρική με τα καλύτερα αποτελέσματα αφού στα περισσότερα αρχεία υπάρχει μόνο 1% αποτυχίας μείωσης του σετ. Η χειρότερη περίπτωση ήταν σε δύο αρχεία τα οποία έχουν και τα δύο 10% αποτυχία, τα αρχεία αυτά είναι τα Have-I-Been-Pwned (Σχήμα 4.1.5) το οποίο είχε κακά αποτελέσματα για όλες τις μετρικές, και το Chegg (Σχήμα 4.1.2).

4.3 Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords

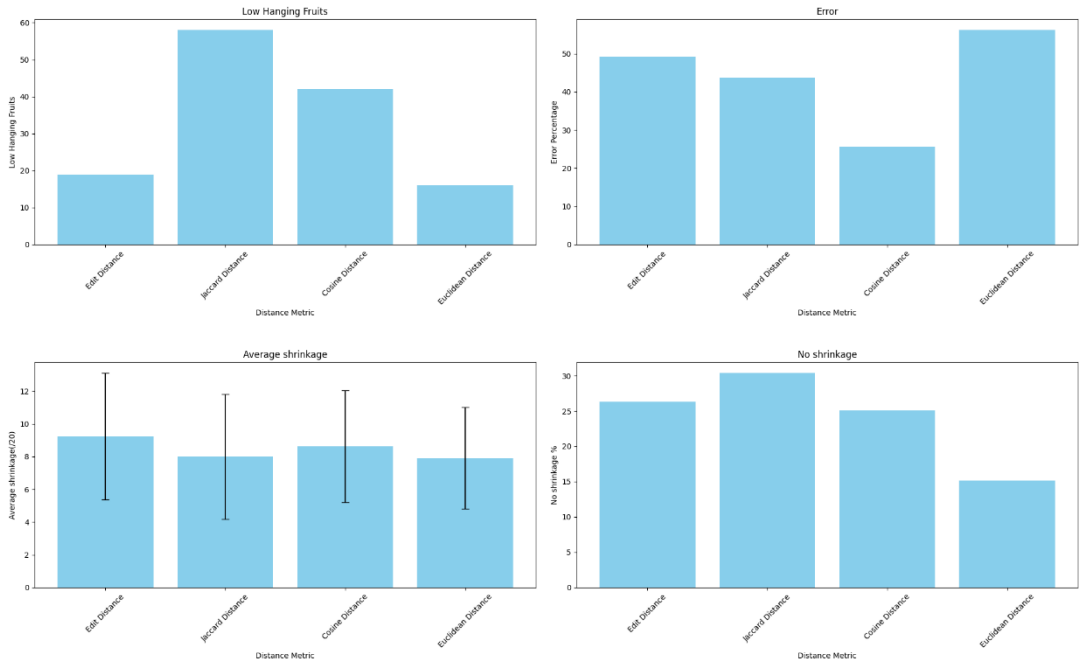
Στο αυτό το υποκεφάλαιο, θα εξετάσουμε την ευρετική μέθοδο του «Όριου Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords» ως την δεύτερη από τις τεχνικές επίθεσης που χρησιμοποίησα για τη μείωση του συνόλου των honeywords στο σύστημα HoneyGen, και θα εξετάσουμε την αποτελεσματικότητά της στην πρακτική. Με την παρουσίαση σχετικών γραφημάτων και δεδομένων παρόμοια με αυτά για την προηγούμενη τεχνική, θα επιχειρήσουμε να αναδείξουμε τις δυνατότητες και τις περιορισμούς αυτής της μεθόδου στο πλαίσιο της προστασίας του συστήματος αυθεντικοποίησης.



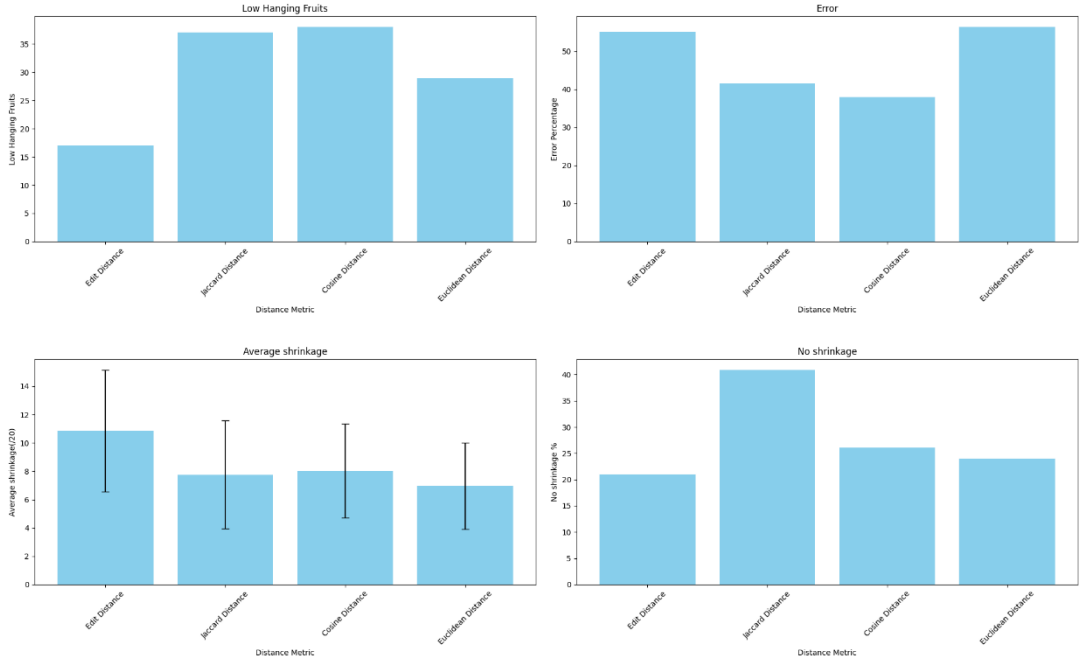
Σχήμα 4.2.1 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Adult FriendFinder.



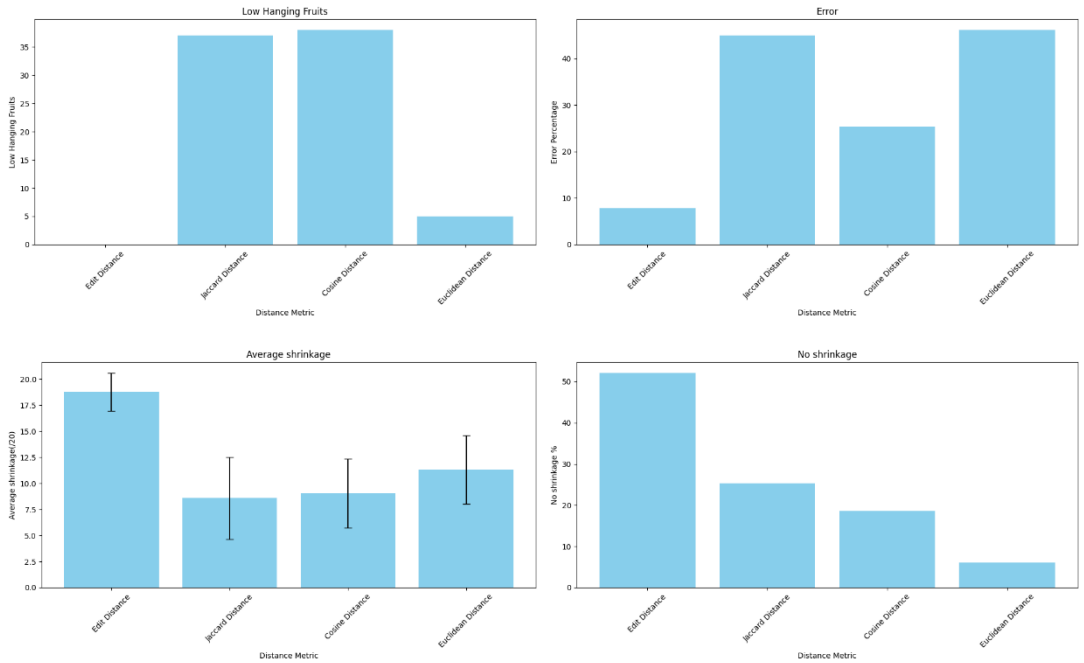
Σχήμα 4.2.2 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Chegg.



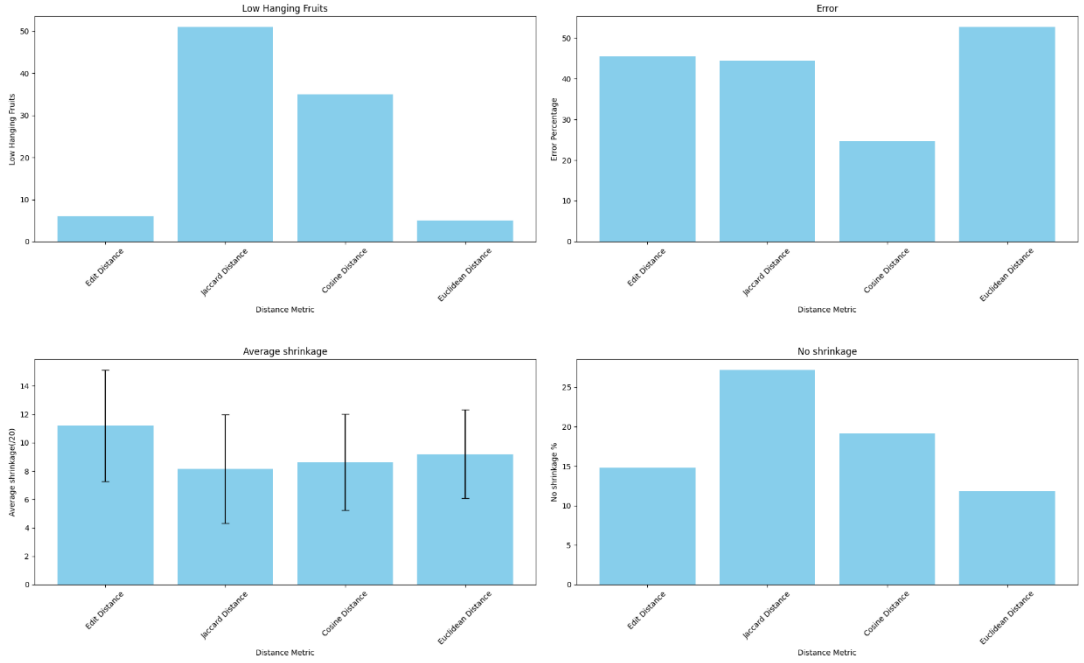
Σχήμα 4.2.3 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Dropbox.



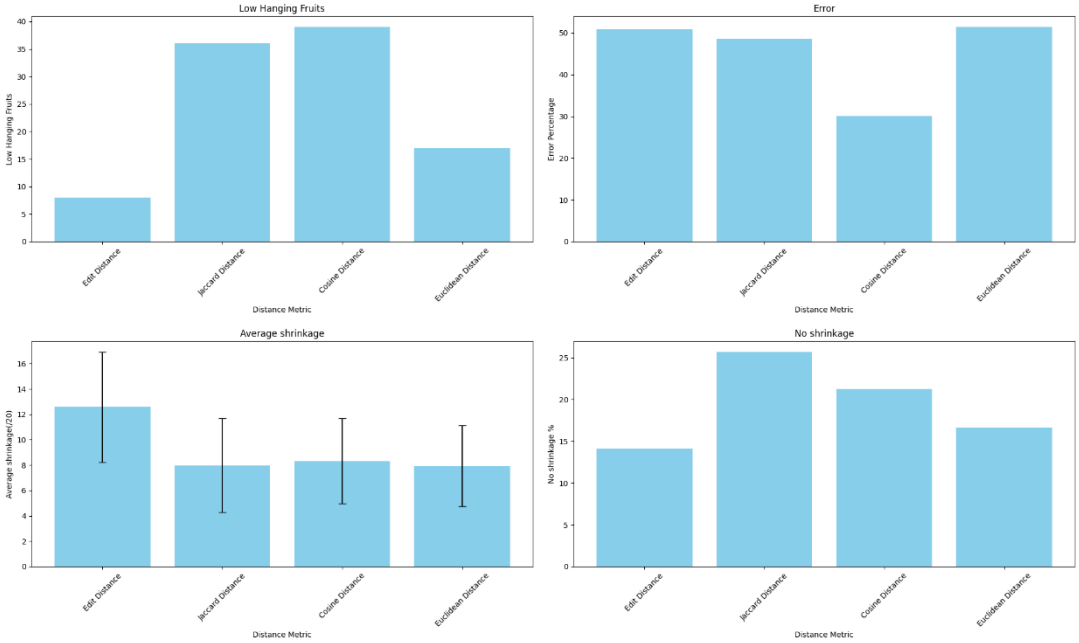
Σχήμα 4.2.4 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Dubsmash.



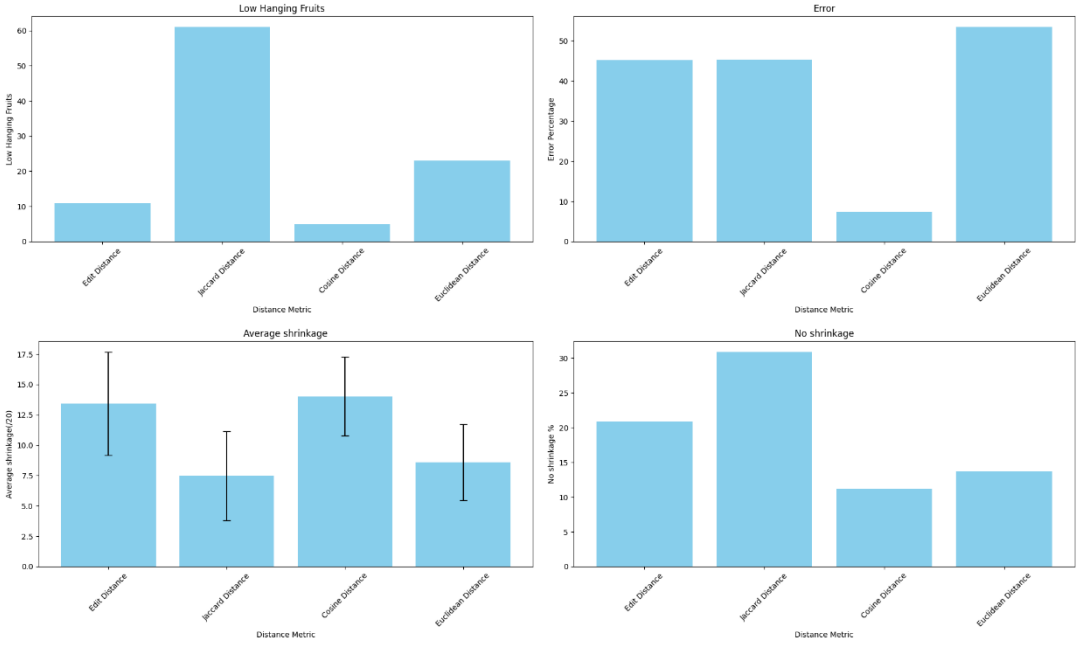
Σχήμα 4.2.5 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Have-I-Been-Pawned.



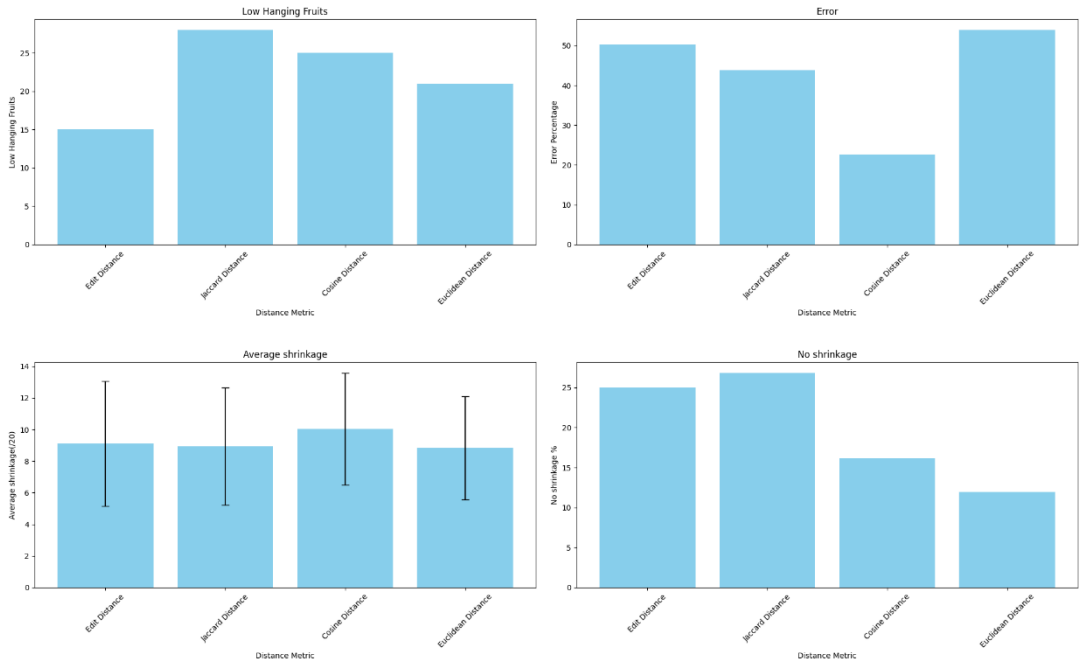
Σχήμα 4.2.6 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Last-Fm.



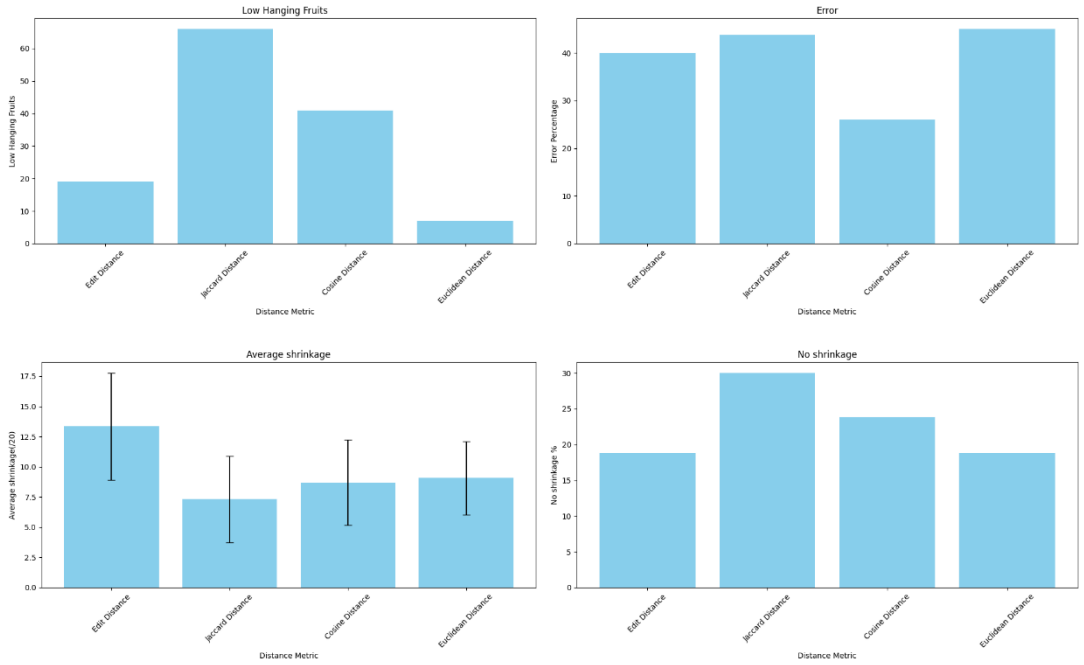
Σχήμα 4.2.7 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας LinkedIn.



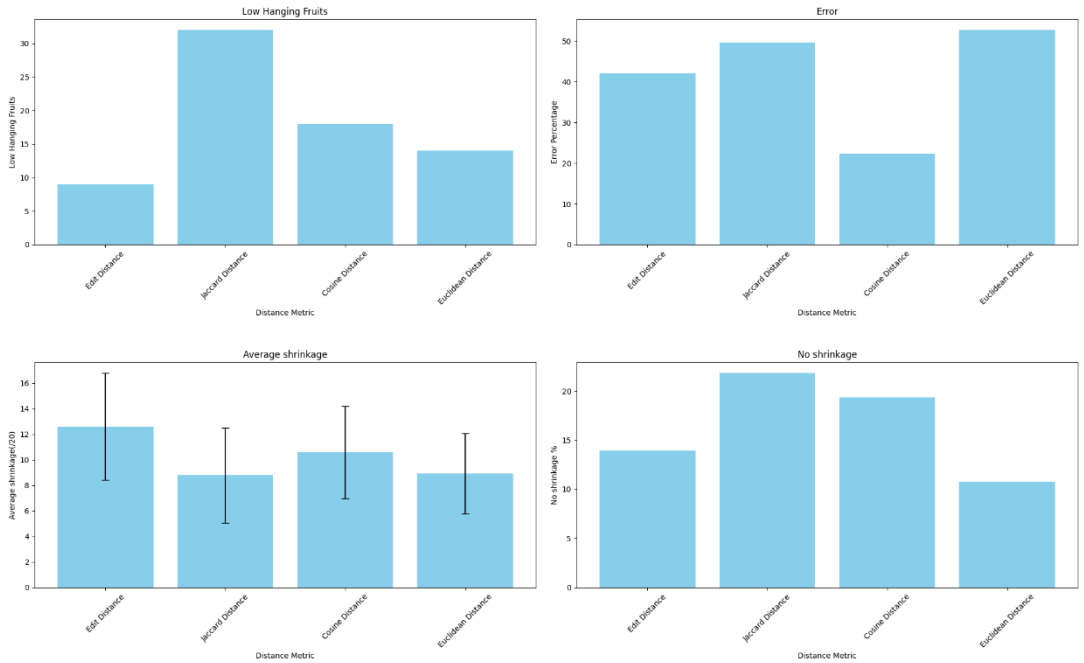
Σχήμα 4.2.8 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας MySpace.



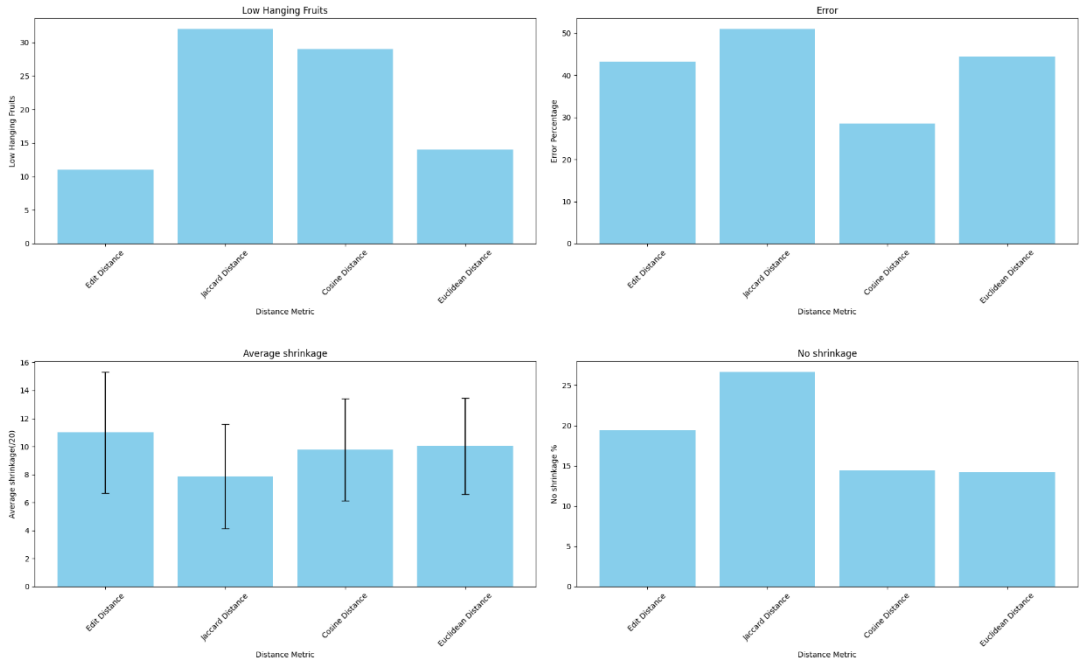
Σχήμα 4.2.9 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας phpBB.



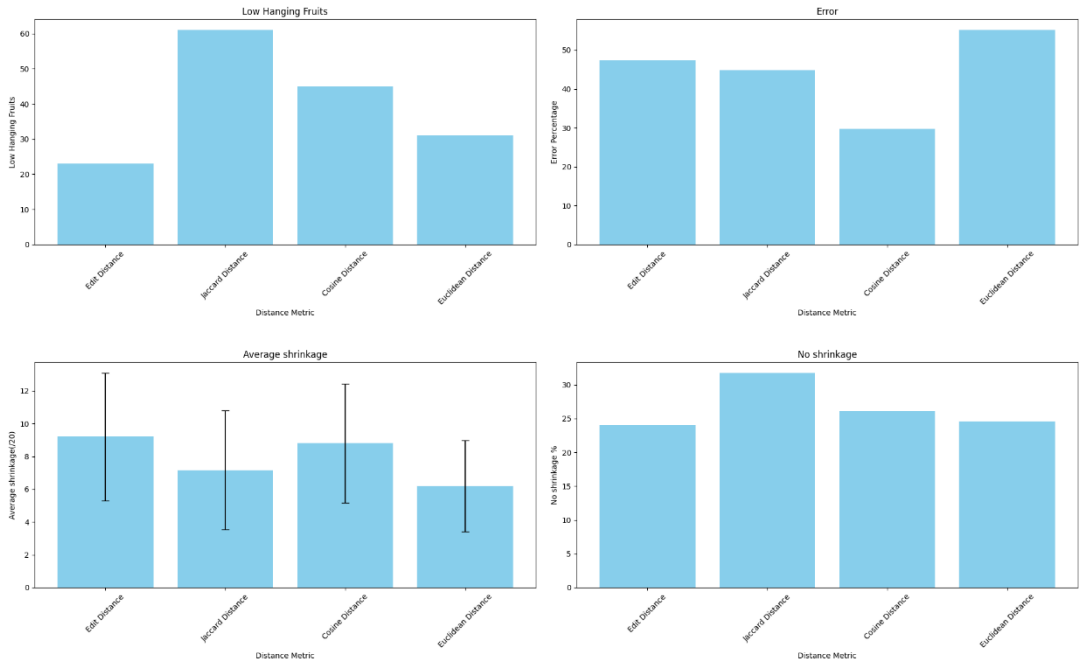
Σχήμα 4.2.10 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Rockyou.



Σχήμα 4.2.11 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Yahoo.



Σχήμα 4.2.12 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Youku.



Σχήμα 4.2.13 – Γραφικές παραστάσεις για το αρχείο της πλατφόρμας Zynga.

Στην ανάλυση δεδομένων για αυτήν την ευρετική μέθοδο, θα εξετάσουμε όπως και πριν προσεκτικά τα δεδομένα που συλλέχθηκαν για κάθε διαφορετικό στατιστικό που εφαρμόσαμε σε αυτήν την. Θα ξεκινήσουμε σχολιάζοντας τα αποτελέσματα της κάθε μετρικής απόστασης που χρησιμοποιήσαμε σε κάθε περίπτωση. Έπειτα, θα προχωρήσουμε σε μια σύντομη σύγκριση των αποτελεσμάτων από τις διάφορες μετρικές, εστιάζοντας στις τυχόν διαφορές στις επιδόσεις τους και την εφαρμοστικότητά τους στο πλαίσιο της κυβερνοασφάλειας. Με αυτόν τον τρόπο, θα διαμορφώσουμε μια πιο πλήρη και συγκριτική εικόνα για την αποτελεσματικότητα και τη χρησιμότητα κάθε μετρικής στο πλαίσιο της κυβερνοεπίθεσης ενός συστήματος αυθεντικοποίησης.

Low hanging fruits

Το πρώτο στατιστικό το οποίο θα μελετήσουμε είναι αυτό των “low hanging fruits” έτσι ώστε να δούμε εάν αυτή η ευρετική μέθοδος είναι καλή στην εύρεση των πραγματικών κωδικών των χρηστών.

Η μετρική απόστασης με την οποία θα ξεκινήσουμε την ανάλυση, είναι η Edit Distance. Για αυτή την μετρική η καλύτερη τιμή “low hanging fruits” είναι για το αρχείο

δεδομένων Zynga (Σχήμα 4.2.13), με 23 σωστούς κωδικούς. Η δεύτερη καλύτερη τιμή είναι 19 στα αρχεία δεδομένων Dropbox (Σχήμα 4.2.3) και Rockyou (Σχήμα 4.2.10). Η χειρότερη τιμή για αυτήν την μετρική είναι 0 στο αρχείο Have -I-Been-Pwned (Σχήμα 4.2.5). Στα υπόλοιπα αρχεία δεδομένων οι τιμές είναι 6, 8 και 9 στα αρχεία Last-FM (Σχήμα 4.2.6), LinkedIn (Σχήμα 4.2.7) και Yahoo (Σχήμα 4.2.11) αντίστοιχα, στα αρχεία Youku (Σχήμα 4.2.12), MySpace (Σχήμα 4.2.8) και Chegg (Σχήμα 4.2.2) έχουμε την τιμή 11, στο αρχείο phpBB (Σχήμα 4.2.9) έχουμε την τιμή των 15 και τέλος στα Dubsmash (Σχήμα 4.2.4) και Adult Friend Finder (Σχήμα 4.2.1) την τιμή 17.

Η επόμενη μετρική απόστασης είναι η Jaccard Distance. Για αυτήν την μετρική έχουμε καλύτερη τιμή "low hanging fruits" ίση με 66 για το αρχείο δεδομένων Rockyou (Σχήμα 4.2.10). Η επόμενη καλύτερη τιμή είναι η τιμή 61 για το αρχείο MySpace (Σχήμα 4.2.8) και το αρχείο Zynga (Σχήμα 4.2.13). Η χειρότερη τιμή είναι το 28 για το αρχείο δεδομένων phpBB (Σχήμα 4.2.9). Οι τιμές για τα υπόλοιπα αρχεία είναι 32 για τα αρχεία Yahoo (Σχήμα 4.2.11) και Youku (Σχήμα 4.2.12), 36 για το αρχείο LinkedIn (Σχήμα 4.2.7) και 37 για τα αρχεία Dubsmash (Σχήμα 4.2.4) και Have-I-Been-Pwned Have -I-Been-Pwned (Σχήμα 4.2.5), τέλος για τα αρχεία Adult FriendFinder (Σχήμα 4.2.1), Last-FM (Σχήμα 4.2.6) και Dropbox (Σχήμα 4.2.3) οι τιμές είναι 42, 51 και 58 αντίστοιχα.

Συνεχίζουμε με την μετρική απόστασης Cosine Distance. Για αυτήν τη μετρική έχουμε ως καλύτερη τιμή την τιμή 59 για το αρχείο Adult FriendFinder (Σχήμα 4.2.1). Δεύτερη καλύτερη τιμή είναι για το αρχείο Zynga με την τιμή 45 και ακολουθεί το Dropbox (Σχήμα 4.2.3) με την τιμή 42 ενώ πολύ κοντά είναι και η τιμή για το αρχείο Rockyou (Σχήμα 4.2.10) στο οποίο έχουμε την τιμή 41. Η χειρότερη τιμή είναι στο αρχείο MySpace (Σχήμα 4.2.8) με μόνο 5 "low hanging fruits". Οι υπόλοιπες τιμές είναι και αυτές αρκετά ψηλές με την επόμενη μικρότερη να είναι το 18 στο αρχείο Yahoo (Σχήμα 4.2.11) και ακολουθούν οι τιμές 25 για το αρχείο phpBB (Σχήμα 4.2.9), 29 για το αρχείο Youku (Σχήμα 4.2.12), 32 για το αρχείο Chegg (Σχήμα 4.2.2), 35 για το αρχείο Last-FM (Σχήμα 4.2.6), 38 για τα αρχεία Dubsmash (Σχήμα 4.2.4) και Have-I-Been-Pwned Have -I-Been-Pwned (Σχήμα 4.2.5) και τέλος το αρχείο δεδομένων LinkedIn (Σχήμα 4.2.7) με την τιμή των 39.

Τελευταία μετρική απόστασης για το στατιστικό των “low hanging fruits” είναι η Euclidean Distance. Στη Euclidean Distance η καλύτερη τιμή των “low hanging fruits” είναι η 31 στο αρχείο Zynga (Σχήμα 4.2.13). Η επόμενη καλύτερη τιμή είναι το 29 στο αρχείο Dubsmash (Σχήμα 4.2.4) και ακολουθούν οι τιμές για τα αρχεία MySpace (Σχήμα 4.2.8) και phpBB (Σχήμα 4.2.9) με τις τιμές των 23 και 21 αντίστοιχα. Η χειρότερη τιμή είναι για αρχείο δεδομένων το Chegg (Σχήμα 4.2.2) με την τιμή 4 όμως πολύ κοντά είναι και τα αρχεία Have-I-Been-Pwned Have -I-Been-Pwned (Σχήμα 4.2.5) και Last-FM (Σχήμα 4.2.6) με την τιμή 4. Τέλος οι υπόλοιπες τιμές είναι 7 για το αρχείο Rockyou (Σχήμα 4.2.10), 14 για τα αρχεία Youku (Σχήμα 4.2.12) και Yahoo, 16 για το αρχείο Dropbox (Σχήμα 4.2.3) και πολύ κοντά στα 17 “low hanging fruits” το αρχείο LinkedIn (Σχήμα 4.2.7) και τέλος το αρχείο Adult FriendFinder (Σχήμα 4.2.1) με την τιμή των 19.

Με μία γρήγορη ματιά μπορούμε να δούμε πως τις καλύτερες τιμές για το στατιστικό των “low hanging fruits” τις έχει η μετρική Jaccard Distance όπως και στην προηγούμενη ευρετική μέθοδο. Όμως σε αντίθεση με την προηγούμενη, η μετρική Cosine Distance έχει και αυτή πολύ καλές τιμές και σε ορισμένα αρχεία όπως το Adult FriendFinder (Σχήμα 4.2.1), το Dubsmash (Σχήμα 4.2.4), το Have-I-Been-Pwned (Σχήμα 4.2.5) και το LinkedIn (Σχήμα 4.2.7) μπορούμε να δούμε πως ξεπερνά τις τιμές της Jaccard Distance. Οι άλλες δύο μετρικές (Edit και Euclidean Distance) τις παραπάνω φορές έχουν καλές τιμές όμως δεν είναι κοντά στις άλλες δύο. Τέλος, σε σχέση με την προηγούμενη ευρετική μέθοδο έχουμε σε όλες τις μετρικές πολύ καλύτερα αποτελέσματα.

Ποσοστό σφάλματος

Το δεύτερο στατιστικό για το οποίο θα αξιολογήσουμε τα αποτελέσματα είναι το “ποσοστό σφάλματος”, έτσι ώστε να μπορέσουμε να συγκρίνουμε τις μετρικές ανάλογα και με αυτό και να συγκρίνουμε τα αποτελέσματα του με αυτά του “low hanging fruits”.

Η πρώτη μετρική απόστασης θα είναι και πάλι η Edit Distance. Το καλύτερο ποσοστό σφάλματος για αυτήν την μετρική είναι το 8% και το έχουμε για το αρχείο δεδομένων Have-I-Been-Pwned (Σχήμα 4.2.5) για το οποίο όμως αν προσέξουμε την τιμή των “low

hanging fruits" του είναι 0. Οπότε ναι είχαμε μικρό error όμως δεν μείναμε και καμία φορά μόνο με τον πραγματικό κωδικό. Το επόμενο καλύτερο ποσοστό είναι για το αρχείο Rockyou (Σχήμα 4.2.10) και είναι 40%, μια πολύ μεγάλη απόκλιση από το καλύτερο. Έτσι και συνεχίζει με όλα τα υπόλοιπα ποσοστά να είναι κοντά στο 40% ή ακόμη και κάποια κοντά στο 50% με το χειρότερο να είναι για το αρχείο Dubsmash (Σχήμα 4.2.4) το οποίο είχε ποσοστό λάθους 55%.

Η δεύτερη μετρική απόστασης που θα αξιολογήσουμε είναι η Jaccard Distance. Σε αυτήν την μετρική το καλύτερο ποσοστό σφάλματος το είχε στο αρχείο Dubsmash (Σχήμα 4.2.4), με το ποσοστό 41%. Σε αυτήν την μετρική, όπως και στην προηγούμενη οι τιμές κυμαίνονται γύρω στο 40% και με κάποιες να είναι κοντά στο 50%. Σε αυτήν την μετρική το χειρότερο ποσοστό σφάλματος ήταν το 51% για το αρχείο Youku.

Η τρίτη μετρική απόστασης είναι η Cosine Distance. Γεια την Cosine Distance τα πράγματα καλυτερεύουν πολύ αφού το καλύτερο ποσοστό είναι το 8% στο αρχείο MySpace (Σχήμα 4.2.8) αλλά και πάλι για αυτό το ποσοστό έχουμε πολύ μικρό αποτέλεσμα για το "low hanging fruits" σε σχέση με τα υπόλοιπα για το Cosine Distance. Το επόμενο καλύτερο είναι το 22% στο αρχείο Yahoo (Σχήμα 4.2.11) και για τις υπόλοιπες τιμές έχουμε τα παραπάνω ποσοστά να κυμαίνονται κοντά στο 30, ενώ κάποια είναι και αυτά κοντά στα 23% ενώ το χειρότερο είναι για το αρχείο Dubsmash το οποίο είχε 38%.

Η τελευταία μετρική για αυτό το στατιστικό είναι η Euclidean Distance της οποίας το καλύτερο ποσοστό σφάλματος ήταν το 44% στο αρχείο Youku. Αν εξαιρέσουμε τα αρχεία Rockyou (Σχήμα 4.2.10), Have-I-Been-Pwned (Σχήμα 4.2.5) και Chegg(Σχήμα 4.1.2) τα οποία έχουν τα ποσοστά 45% το Rockyou και 46% τα άλλα δύο, οι τιμές των υπόλοιπων αρχείων είναι πάνω από το 50% και το χειρότερο ποσοστό να είναι το 56% το οποίο ισχύει για δυο αρχεία δεδομένων, το Dropbox (Σχήμα 4.2.3) και το Dubsmash (Σχήμα 4.2.4).

Μετά από την ανάλυση και αυτού του στατιστικού μπορούμε να συγκρίνουμε ακόμη καλύτερα τις μετρικές απόστασης μεταξύ τους. Αρχικά όμως πρέπει να αναφέρουμε πως με αυτήν την ευρετική μέθοδο αν και τα αποτελέσματα των "low hanging fruits" ήταν καλύτερα, τα αποτελέσματα του ποσοστού σφάλματος είναι πολύ χειρότερα για

όλες τις μετρικές. Όμως όπως και στην προηγούμενη μέθοδο μπορούμε εύκολα να εντοπίσουμε την μετρική με τα καλύτερα αποτελέσματα σε αυτό το στατιστικό η οποία είναι η Cosine Distance. Τέλος μια άλλη παρατήρηση είναι πως η Cosine Distance έχει αποτελέσματα σφάλματος πολύ κοντά στα αποτελέσματα σφάλματος των υπόλοιπων μετρικών απόστασης οι οποίες είχαν τα καλύτερα αποτελέσματα “low hanging fruits” στην προηγούμενη ευρετική μέθοδο, όμως έχει καλύτερα αποτελέσματα από αυτές στο στατιστικό εκείνο.

Μέσο ποσοστό μείωσης του σετ

Το τρίτο στατιστικό το οποίο θα εξετάσουμε είναι το "μέσο ποσοστό μείωσης του σετ". Αναλύοντας αυτό το στατιστικό θα μπορέσουμε να αξιολογήσουμε τον αρχικό στόχο των επιθέσεων ο οποίο ήταν να μειώσουμε το μέγεθος των σετ από Honeywords.

Αρχίζουμε ξανά με την ανάλυση των αποτελεσμάτων για την μετρική Edit Distance. Η καλύτερη μέση μείωση είναι στο αρχείο Adult FriendFinder (Σχήμα 4.2.1), στο οποίο η μέση μείωση του σετ είναι στα 8.8/20. Οι επόμενες καλύτερες τιμές είναι για αρχεία δεδομένων phpBB (Σχήμα 4.2.9) το οποίο είχε μέση μείωση 9.1/20 και τα αρχεία Dropbox (Σχήμα 4.2.3) και Zynga (Σχήμα 4.2.13), όπου και στα δύο υπήρχε μέση μείωση 9.2/20. Η χειρότερη μείωση είναι για το αρχείο Have-I-Been-Pwned (Σχήμα 4.2.5) στο οποίο υπήρχε μέση μείωση μόνο στα 18.8/20.

Η επόμενη μετρική απόστασης είναι η Jaccard Distance. Σε αυτή την μετρική η καλύτερη μείωση ήταν στο αρχείο Zynga (Σχήμα 4.2.13) με την τιμή 7.2/20. Πολύ κοντά σε αυτή την τιμή ήταν και το αρχείο Rockyou (Σχήμα 4.2.10)το οποίο είχε τιμή 7.3/20 και ακολουθεί το αρχείο MySpace (Σχήμα 4.2.8) το οποίο είχε μείωση 7.5/20. Για χειρότερη τιμή σε αυτήν την μετρική την έχει το αρχείο Chegg (Σχήμα 4.2.2) το οποίο έχει μέση μείωση των σετ στο 10.8/20 και το δεύτερο χειρότερο είναι το αρχείο phpBB (Σχήμα 4.2.9) με μέση μείωση 8.9/20.

Συνεχίζουμε με την μετρική Cosine Distance. Τα αρχεία τα οποία έχουν μέση τιμή, 9/20 και κάτω, είναι τα αρχεία Adult FriendFinder (Σχήμα 4.2.1), Dropbox (Σχήμα 4.2.3), Dubsmash (Σχήμα 4.2.4), Have -I-Been-Pwned (Σχήμα 4.2.5), Last-FM (Σχήμα 4.2.6),

LinkedIn (Σχήμα 4.2.7), Rockyou (Σχήμα 4.2.10) και Zynga (Σχήμα 4.2.13). Η καλύτερη μείωση είναι για το αρχείο δεδομένων Adult FriendFinder με την τιμή 7.9/20. Η επόμενη καλύτερη τιμή είναι στο αρχείο Dubsmash στο οποίο η μέση τιμή ήταν το 8/20 και ακολουθεί το LinkedIn με την τιμή 8.3. Την χειρότερη μείωση για αυτήν την μετρική την είχε το αρχείο MySpace (Σχήμα 4.2.8) το οποίο είχε μείωση μόνο 14/20.

Τελευταία μετρική και για αυτό το στατιστικό είναι η Euclidean Distance. Για αυτή την μετρική η καλύτερη τιμή είναι στο αρχείο Zynga (Σχήμα 4.2.13), το οποίο είχε μέση μείωση 6.2/20 και η επόμενη καλύτερη είναι στο αρχείο Dubsmash (Σχήμα 4.2.4) στο οποίο έχουμε 7/20 μείωση. Το αρχείο το οποίο είχε την χειρότερη μέση μείωση σε αυτήν την περίπτωση ήταν το αρχείο Zynga με μείωση 12.4/20.

Τα αποτελέσματα για αυτό το στατιστικό είναι σίγουρα καλύτερα από αυτά τις προηγούμενης ευρετικής μεθόδου. Επιπλέον μπορούμε και πάλι συγκρίνοντας τα αποτελέσματα αυτού του στατιστικού με αυτά του “low hanging fruits” και του ποσοστού σφάλματος να παρατηρήσουμε το πως συνδέονται αυτές οι τιμές. Για παράδειγμα το αρχείο Zynga (Σχήμα 4.2.13), στην Euclidean Distance έχει την καλύτερη τιμή μέσης μείωσης και την καλύτερη τιμή “low hanging fruits” αλλά έχει την χειρότερη τιμή σφάλματος. Την ίδια παρατήρηση μπορούμε να την κάνουμε και για τις υπόλοιπες μετρικές, για παράδειγμα στην μετρική Cosine Distance το αρχείο Dubsmash (Σχήμα 4.2.4) το οποίο έχει από τις καλύτερες τιμές “low hanging fruits” για την Cosine Distance, έχει την δεύτερη καλύτερη τιμή μείωσης και την χειρότερη τιμή σφάλματος. Γενικά τις περισσότερες φορές, εκτός από κάποιες εξαιρέσεις, όσο πιο πολύ μειώνεται το σετ τόσο πιο πολλά θα είναι τα “low hanging fruits” και τόσο πιο μεγάλο το σφάλμα. Η καλύτερη μετρική για αυτό το στατιστικό είναι ανάμεσα στην Jaccard Distance και Euclidean Distance, όμως και οι δύο έχουν πολύ μεγάλο ποσοστό σφάλματος.

Ποσοστό αποτυχίας μείωσης του σετ

Τα αποτελέσματα για το τελευταίο στατιστικό, το ποσοστό αποτυχίας μείωσης του σετ, δεν έχουν τόσο μεγάλη σημασία για την κυβερνοεπίθεση διότι ο επιτιθέμενος μπορεί

απλά να τα αγνοήσει όμως σίγουρα παίζουν και αυτά ρόλο στην αξιολόγηση της κάθε ευρετικής μεθόδου.

Στην μετρική Edit Distance το καλύτερο ποσοστό αποτυχίας είναι το 8% για το αρχείο Chegg (Σχήμα 4.2.2) και το χειρότερο ποσοστό είναι το 52% στο αρχείο Have-I-Been-Pwned (Σχήμα 4.2.5) το οποίο είχε γενικά τις χειρότερες τιμές σε όλα τα στατιστικά για αυτήν την μετρική. Τα υπόλοιπα αρχεία κυμαίνονται ανάμεσα στις τιμές 14% με 26%.

Στην μετρική Jaccard Distance, το καλύτερο ποσοστό αποτυχίας είναι το 22% το οποίο έχουν 2 αρχεία, το αρχείο Chegg (Σχήμα 4.2.2) και το αρχείο Yahoo (Σχήμα 4.2.11). Το χειρότερο ποσοστό αποτυχίας μείωσης του σετ είναι το 41% στο αρχείο Dubsmash (Σχήμα 4.2.4) το οποίο όμως έχει αρκετά καλή μέση μείωση του σετ, αυτό σημαίνει πως στα υπόλοιπα είχε πολύ καλή μείωση. Τα υπόλοιπα αρχεία κυμαίνονται στις τιμές 25- 32%.

Στη μετρική Cosine Distance, το καλύτερο ποσοστό αποτυχίας είναι το 11% στο αρχείο MySpace (Σχήμα 4.2.8). Το επόμενο καλύτερο ποσοστό είναι το 15% στο αρχείο Chegg (Σχήμα 4.2.2). Το χειρότερο ποσοστό αποτυχίας μείωσης του σετ είναι το 28% στο αρχείο Adult FriendFinder (Σχήμα 4.2.1), το οποίο όμως είχε την καλύτερη μέση μείωση του σετ, αυτό σημαίνει πως στα υπόλοιπα σετ είχε πολύ καλή μείωση. Τα υπόλοιπα αρχεία κυμαίνονται στις τιμές 16-26%.

Τέλος, στη μετρική Euclidean Distance, το καλύτερο ποσοστό αποτυχίας είναι το 6% το οποίο έχουν 2 αρχεία, το αρχείο Chegg (Σχήμα 4.2.2) και το αρχείο Have-I-Been-Pwned (Σχήμα 4.2.5). Το χειρότερο ποσοστό αποτυχίας μείωσης του σετ είναι το 56% στο αρχείο Dropbox (Σχήμα 4.2.3) το οποίο όμως και πάλι έχει αρκετά καλή μέση μείωση του σετ. Τα υπόλοιπα αρχεία κυμαίνονται στις τιμές 12- 25%.

Με μια μικρή σύγκριση αυτού του στατιστικού της ευρετικής μεθόδου αυτής με την προηγούμενη ευρετική μέθοδο καταλήγουμε πως τα αποτελέσματα της Edit Distance και της Cosine Distance είναι λίγο καλύτερα για αυτήν την μέθοδο, ενώ τα αποτελέσματα για τις Jaccard και Euclidean Distance είναι καλύτερα στην μέθοδο «Όριο Μέσης Απόστασης», ειδικά για την δεύτερη. Επιπλέον, είναι σημαντικό να

αναφέρουμε ότι τα αποτελέσματα αυτού του στατιστικού μπορεί να φαίνονται σε ασυμφωνία με τα αποτελέσματα των υπόλοιπων στατιστικών. Για παράδειγμα, στο αρχείο Dropbox, για την μετρική Euclidean Distance, δεν είναι λογικό να έχουμε ένα ποσοστό σφάλματος 56% και ένα ποσοστό αποτυχίας μείωσης επίσης 56%, καθώς η αποτυχία μείωσης υποδηλώνει ότι δεν έγινε μείωση στο μέγεθος του συνόλου των sweetwords (παρέμεινε στα 20), συνεπώς ο πραγματικός κωδικός πρόσβασης εξακολουθεί να είναι μέσα στο σύνολο, και έτσι δεν υπάρχει σφάλμα. Ο λόγος για αυτή την ασυμφωνία είναι πως υπήρχαν περιπτώσεις όπου κανένα από τα sweetwords σε ένα σετ δεν ήταν στο εύρος της μέσης τιμής και της τυπικής απόκλισης ανάλογα με την κάθε ευρετική μέθοδο. Αυτές οι περιπτώσεις μετριόνταν στο ποσοστό της αποτυχίας μείωσης του σετ, όμως δεν μετριόνταν στα υπόλοιπα στατιστικά.

Κεφάλαιο 5

Συμπεράσματα

5.1 Επίλογος

Συνοψίζοντας, οι δύο ευρετικές μέθοδοι που εξετάσαμε στο Κεφάλαιο 4 καλύπτουν διαφορετικούς τομείς. Η μέθοδος «Όριο Μέσης Απόστασης» έχει χειρότερα στατιστικά στα low hanging fruits από την μέθοδο «Όριο Μέσης Απόστασης προς τα κορυφαία k-πλησιέστερα honeywords», αλλά έχει χαμηλότερο ποσοστό σφάλματος. Αντίστοιχα, η δεύτερη μέθοδος έχει υψηλότερο ποσοστό σφάλματος, αλλά καλύτερα αποτελέσματα εκτός από τα low hanging fruits, και στο ποσοστό μείωσης του μεγέθους των σετ από honeywords, το οποίο ήταν και ο πρωταρχικός στόχος.

Αν εξετάζαμε την κατάσταση από τη σκοπιά του επιτιθέμενου, πιθανότατα θα προτιμούσε τη δεύτερη μέθοδο για δύο λόγους. Πρώτον, διότι θα έδινε περισσότερη έμφαση στα στατιστικά της μείωσης και ακόμη περισσότερο στα στατιστικά των low hanging fruits αφού είναι αυτά που δίνουν μια λύση. Ο δεύτερος λόγος είναι ότι με τη χρήση ενός αλγορίθμου μηχανικής μάθησης θα μπορούσαμε να μειώσουμε το ποσοστό σφάλματος και να αυξήσουμε την ακρίβεια των αποτελεσμάτων. Έτσι, μέσω ενός τέτοιου αλγορίθμου, το υψηλό ποσοστό σφάλματος που έχουμε στη δεύτερη μέθοδο θα μειωνόταν και θα είχαμε έναν πολύ καλύτερο αλγόριθμο κυβερνοεπίθεσης, ο οποίος θα έχει και υψηλά low hanging fruits, αλλά και χαμηλό ποσοστό σφάλματος.

Η μετρική η οποία έδειξε την υπερίσχυση της δεύτερης μεθόδου είναι η απόσταση Cosine Distance. Αρχικά, αυτή η μετρική είχε τα καλύτερα αποτελέσματα για την δεύτερη μέθοδο, διότι εκτός από το ότι είχε πολύ καλά αποτελέσματα στο στατιστικό

των low hanging fruits, σχεδόν ίσα με αυτά της καλύτερης μετρικής για αυτό το στατιστικό, είχε ταυτόχρονα και το καλύτερο ποσοστό σφάλματος από όλες τις μετρικές. Επιπλέον, εάν συγκρίνουμε τα αποτελέσματα της με αυτά της πρώτης μεθόδου, θα δούμε πως έχει πολύ καλύτερα αποτελέσματα low hanging fruits από όλες τις μετρικές και περίπου ίσα αποτελέσματα στο ποσοστό σφάλματος από τις άλλες.

Συγκλίνοντας, παρόλο που τα αποτελέσματα και για τις δύο μεθόδους δεν είναι ικανοποιητικά, υπάρχει δυνατότητα βελτίωσής τους, ιδίως για την εφαρμογή τους σε κυβερνοεπιθέσεις. Η συστηματική μελέτη των μεθόδων δημιουργίας honeywords από το σύστημα HoneyGen θα είναι κρίσιμης σημασίας για την εξέλιξη και την αξιοπιστία του σε τέτοιες καταστάσεις. Συμπερασματικά, τα honeywords είναι μια ενδιαφέρουσα τεχνική κυβερνοασφάλειας που όμως χρίζει περαιτέρω μελέτης έναντι επιτιθέμενων οι οποίοι μπορεί να χρησιμοποιήσουν διάφορες ευρετικές και μη επιθέσεις για να ξεχωρίσουν τον πραγματικό κωδικό πρόσβασης από τα honeywords. Με τη σωστή προσέγγιση, μπορεί να γίνει μια από τις πιο αξιόπιστες τεχνικές σε αυτόν τον τομέα. Παρουσιάζοντας μια εκτενή ανάλυση ασφαλείας σχετικά με τα honeywords, ενισχύουμε σημαντικά την πιθανότητα υιοθέτησης τους από μεγάλους οργανισμούς συστημάτων ελέγχου ταυτότητας με κωδικούς πρόσβασης.

5.2 Μελλοντική Εργασία

Για μελλοντική εργασία υπάρχουν τρείς τρόποι με τους οποίους θα μπορούσαμε να προχωρήσουμε.

1. Ανάπτυξη αλγορίθμων μηχανικής μάθησης με τους οποίους θα βελτιώσουμε την αποτελεσματικότητα των δύο μεθόδων πιο πάνω μειώνοντας επίσης το ποσοστό σφάλματος.

2. Μελέτη των ποιο πάνω αποτελεσμάτων, αλλά αυτή την φορά από την μεριά της κυβερνοασφάλειας. Δηλαδή πώς θα μπορούσαμε να αντιμετωπίσουμε τις πιο πάνω επιθέσεις πιο αποτελεσματικά έτσι ώστε να μην αφήνουμε κανένα περιθώριο κυβερνοεπίθεσης.

3. Εξερεύνηση άλλων τρόπων επίθεσης και αξιολόγησης του συστήματος. Ένας από αυτούς είναι οι επιθέσεις μέσω συστημάτων LLM όπως για παράδειγμα το GPT3.5 και το GPT4 της OpenAI.

4. Έρευνα για το πώς θα μπορούσε να ανταποκριθεί το HoneyGen σε targeted επιθέσεις όπου ο επιτιθέμενος γνωρίζει και χρησιμοποιεί προσωπικά δεδομένα του χρήστη, οπότε τα παραγόμενα honeywords πρέπει να περιέχουν και προσωπικά στοιχεία του χρήστη έτσι ώστε να ξεγελάσουν τον επιτιθέμενο.

Βιβλιογραφία

[1] A. Dionysiou, V. Vassiliades, and E. Athanasopoulos. HoneyGen: generating honeywords using representation learning. In 16th ACM Symposium on Information, Computer and Communications Security, 2021

[2] The Impact of Exposed Passwords on Honeyword Efficacy Z. Huang, L. Bauer, M. Reiter

[3] How to attack and generate honeywords. In 43rd IEEE Symposium on Security and Privacy, May 2022. D. Wang, Y. Zou, Q. Dong, Y. Song, and X. Huang.

[4] F. Yu and M. V. Martin. Honey, I chunked the passwords: Generating semantic honeywords resistant to targeted attacks using pre-trained language models. In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, 2023.

[5] A. Juels and R. L. Rivest. Honeywords: Making password-cracking detectable. In 20th ACM Conference on Computer and Communications Security, November 2013.

[6] X. He, H. Cheng, J. Xie, P. Wang, and K. Liang. Passtrans: An improved password reuse model based on transformer. In 47th IEEE International Conference on Acoustics, Speech, and Signal Processing, 2022.

[7] . Pal, T. D`aniel, R. Chatterjee, and T. Ristenpart. Beyond credential stuffing: Password similarity models using neural networks. In 40th IEEE Symposium on Security and Privacy, 2019.

[8] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016).

[9] GeeksforGeeks: https://www.geeksforgeeks.org/edit-distance-dp-5/

[10] GeeksforGeeks: https://www.geeksforgeeks.org/cosine-similarity/

[11] GeeksforGeeks: https://www.geeksforgeeks.org/find-the-jaccard-index-and-jaccard-distance-between-the-two-given-sets/

[12] GeeksforGeeks: https://www.geeksforgeeks.org/euclidean-distance/

Παράρτημα Α

Α.1 Οι μέθοδοι για τον υπολογισμό των μετρικών απόστασης

```
def edit_distance(str1, str_list):
    distances = np.array([Levenshtein.distance(str1, s) for s in str_list])
    return np.mean(distances)

def jaccard_distance_avg(str1, str_list):
    distances = np.array([jaccard_distance(set(str1), set(s)) for s in str_list])
    return np.mean(distances)

def cosine_distance(str1, str_list):
    distances = np.array([distance.cosine(model.get_word_vector(str1), model.get_word_vector(s)) for s in str_list])
    return np.mean(distances)

def euclidean_distance(str1, str_list):
    distances = np.array([distance.euclidean(model.get_word_vector(str1), model.get_word_vector(s)) for s in str_list])
    return np.mean(distances)
```

Πιο πάνω είναι οι μέθοδοι για τον υπολογισμό των μέσων αποστάσεων για κάθε μετρική της οποίες θα χρησιμοποιήσω για την πρώτη ευρετική μέθοδο.

```
def edit_distance(str1, str_list):
    distances = np.array([Levenshtein.distance(str1, s) for s in str_list])
    return np.sort(distances)

def jaccard_distance_avg(str1, str_list):
    distances = np.array([jaccard_distance(set(str1), set(s)) for s in str_list])
    return np.sort(distances)

def cosine_distance(str1, str_list):
    distances = np.array([distance.cosine(model.get_word_vector(str1), model.get_word_vector(s)) for s in str_list])
    return np.sort(distances)

def euclidean_distance(str1, str_list):
    distances = np.array([distance.euclidean(model.get_word_vector(str1), model.get_word_vector(s)) for s in str_list])
    return np.sort(distances)
```

Ποιο πάνω είναι οι μέθοδοι για τον υπολογισμό των αποστάσεων για κάθε μετρική οι οποίες τοποθετούνται μετά σε έναν πίνακα και τις ταξινομούνται. Οι μέθοδοι αυτοί χρησιμοποιούνται στην δεύτερη ευρετική μέθοδο.

Στις cosine και euclidean distance, επειδή ο υπολογισμός είναι πάνω σε διανύσματα, χρησιμοποιήθηκε η μέθοδος get_word_vector (string) του fasttext module, για την μετατροπή των συμβολοσειρών σε διανύσματα.

Παράρτημα Β

Β.1 Ο αλγόριθμος επίθεσης με την χρήση της πρώτης ευρετικής μεθόδου, « Όριο Μέσης Απόστασης»

```
    for i in range(num_substrings):
            avg_edit_distance= edit_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))
            avg_jaccard_distance= jaccard_distance_avg(substrings[i] ,(substrings[:i] + substrings[i+1:]))
            avg_cosine_distance= cosine_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))
            avg_euclidean_distance= euclidean_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))

            if (y_edit - sigma_edit <= avg_edit_distance <= y_edit + sigma_edit):
                filtered_substrings_edit.append(substrings[i]);

            if (y_jaccard - sigma_jaccard <= avg_jaccard_distance <= y_jaccard + sigma_jaccard):
                filtered_substrings_jacc.append(substrings[i]);

            if (y_cosine - sigma_cosine <= avg_cosine_distance <= y_cosine + sigma_cosine):
                filtered_substrings_cosine.append(substrings[i]);

            if (y_euclidean - sigma_euclidean <= avg_euclidean_distance <= y_euclidean + sigma_euclidean):
                filtered_substrings_euclidean.append(substrings[i])

    return filtered_substrings_edit, filtered_substrings_jacc, filtered_substrings_cosine, filtered_substrings_euclidean
```

Για κάθε σετ από honeywords:

1. Υπολογίζουμε για κάθε honeyword την μέση τιμή απόστασης με όλα τα υπόλοιπα στο σετ του. (**Α.2** για τις μεθόδους υπολογισμού των αποστάσεων)

2. Μετά μέσω του if υπολογίζουμε εάν είναι μέσα στο όριο της μέσης τιμής και της τυπικής απόκλισης που υπολογίσαμε στο βήμα της «εκπαίδευσης».

3. Επιστρέφουμε μια λίστα η οποία περιέχει τα honeywords του συγκεκριμένου σετ τα οποία πέρασαν τον έλεγχο.

Β.2 Ο αλγόριθμος επίθεσης με την χρήση της δεύτερης ευρετικής μεθόδου, «Όριο Μέσης Απόστασης προς τα κορυφαία k- πλησιέστερα honeywords»

```
for i in range(num_substrings):
        edit_distances = edit_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))
        jaccard_distances = jaccard_distance_avg(substrings[i] ,(substrings[:i] + substrings[i+1:]))
        cosine_distances = cosine_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))
        euclidean_distances = euclidean_distance(substrings[i] ,(substrings[:i] + substrings[i+1:]))
        ce=1
        cj=1
        cc=1
        ceu=1

        for j in range (num_substrings-1):
          if not (y_edit[j] - sigma_edit[j] <= edit_distances[j] <= y_edit[j] + sigma_edit[j]):
             ce=0

          if not (y_jaccard[j] - sigma_jaccard[j] <= jaccard_distances[j] <= y_jaccard[j] + sigma_jaccard[j]):
             cj=0

          if not (y_cosine[j] - sigma_cosine[j] <= cosine_distances[j] <= y_cosine[j] + sigma_cosine[j]):
             cc=0

          if not (y_euclidean[j] - sigma_euclidean[j] <= euclidean_distances[j] <= y_euclidean[j] + sigma_euclidean[j]):
             ceu=0

        if(ce==1):
          filtered_substrings_edit.append(substrings[i]);
        if(cj==1):
          filtered_substrings_jacc.append(substrings[i]);
        if(cc==1):
          filtered_substrings_cosine.append(substrings[i]);
        if(ceu==1):
           filtered_substrings_euclidean.append(substrings[i]);
return filtered_substrings_edit, filtered_substrings_jacc, filtered_substrings_cosine, filtered_substrings_euclidean
```

Για κάθε σετ από honeywords:

1. Υπολογίζουμε για κάθε honeyword τις τιμές της απόστασης του με όλα τα υπόλοιπα στο σετ του, τα οποία γίνονται sort και μπαίνουν στις μεταβλητές edit_distances, jaccard_distances, cosine_distances και euclidean_distances για κάθε μετρική αντίστοιχα (**A.2** για τις μεθόδους υπολογισμού των αποστάσεων).

2. Μετά μέσω του if υπολογίζουμε εάν είναι μέσα στο όριο της μέσης τιμής και της τυπικής απόκλισης ανάλογα με την θέση του, πιο κοντινό/μακρινό που υπολογίσαμε στο βήμα της «εκπαίδευσης» και αν είναι όλες οι αποστάσεις μέσα στο εύρος τότε βάζουμε το honeyword που ελέγχουμε μέσα σε μία λίστα.

3. Επιστρέφουμε την λίστα η οποία περιέχει τα honeywords του συγκεκριμένου σετ τα οποία πέρασαν τον έλεγχο.

Παράρτημα Γ

Περισσότερη ανάλυση των διάφορών τρόπων παραγωγής honeyword.

Γ.1 Μοντέλα πιθανοτήτων κωδικών

List Model

Το μοντέλο List αρχικά εκτιμά την πιθανότητα του κάθε κωδικού πρόσβασης p με την φόρμουλα:

$$\frac{m(p)}{m(\Sigma *)}$$

Και στην συνέχεια επιλέγει από την λίστα με τους υποψήφιους κωδικούς με βάση την υπολογισμένη πιθανότητα.

PCFG Model

Το μοντέλο PCFG αντιμετωπίζει έναν κωδικό πρόσβασης ως μια ακολουθία τμημάτων. Τρία είδη τμημάτων: τμήμα ψηφίου, τμήμα γράμματος ή τμήμα συμβόλου**.**
Το καταφέρνει αυτό ορίζοντας το αλφάβητο Σ = L∪D∪S όπου

L = {a...z,A...Z} $\bar{L}$= Σ \ L
D = {0,...,9} Δ = Σ \ D
S = {!,@,#,...} S = Σ\ S

Στην συνέχεια το PCFG μοντέλο προσδιορίζει την πιθανότητα ενός κωδικού πρόσβασης για παράδειγμα του "mpanana123!", ως το γινόμενο $P(L_7D_3S)$ $P(mpanana \mid L_7)$ $P(123 \mid D_3)$ $P(! \mid S)$.

Markov model

Το μοντέλο Markov, στο πλαίσιο της δημιουργίας κωδικών πρόσβασης, λειτουργεί με βάση την αρχή μιας αλυσίδας Markov τάξης s έτσι ώστε η πρόβλεψη του χαρακτήρα στην τωρινή θέση να εξαρτάται μόνο από τους προηγούμενους s χαρακτήρες.

Αλυσίδα Markov:

- Μια αλυσίδα Markov είναι ένα στοχαστικό μοντέλο που περιγράφει μια ακολουθία πιθανών γεγονότων, όπου η πιθανότητα κάθε γεγονότος εξαρτάται μόνο από την κατάσταση που επιτεύχθηκε στο προηγούμενο γεγονός.
- Η τάξη s καθορίζει πόσα προηγούμενα γεγονότα ή καταστάσεις λαμβάνονται υπόψη.

Παράδειγμα:

Ας υποθέσουμε ότι έχουμε ένα μοντέλο Markov με τάξη s=2. Εάν δημιουργούμε έναν κωδικό και οι προηγούμενοι δύο χαρακτήρες είναι "pa", τότε το μοντέλο προβλέπει τον

επόμενο χαρακτήρα με βάση την κατανομή πιθανοτήτων των χαρακτήρων που συνήθως ακολουθούν το "pa" στα δεδομένα εκπαίδευσης.
Δηλαδή, εάν στα δεδομένα εκπαίδευσης, το "pa" ακολουθείται συχνά από "s" ή "t", τότε το μοντέλο θα προβλέψει ότι ο επόμενος χαρακτήρας θα είναι ένας από αυτούς βάσει των πιθανοτήτων τους.

Γ.2 Chunk-level GPT3

Η τεχνική Chunk-level GPT-3 (CGPT3) ακολουθεί τρία στάδια.

1. Αρχικά, λαμβάνει τον κωδικό πρόσβασης του λογαριασμού ως είσοδο και εφαρμόζει μια τεχνική διαίρεσης του κωδικού που ονομάζεται PwdSegment, ώστε να επιστρέψει τα κομμάτια που αποτελούν τον κωδικό.
   Για παράδειγμα, αν ο κωδικός είναι "bike2000", το PwdSegment επιστρέφει τα "bike" και "2000" ως δύο ξεχωριστά κομμάτια.

2. Στη συνέχεια, δημιουργείται ένα prompt που περιέχει πληροφορίες για τον κωδικό και τα κομμάτια του, το οποίο δίνεται ως είσοδος στο μοντέλο GPT-3.
   Για παράδειγμα:
   “Πάραξε n κωδικούς παρόμοιους με το "bike2000", που να περιέχουν τα "bike" και "2000", με μήκος έως 8 χαρακτήρες, χωρίς να προστίθενται ψηφία στο τέλος”.

3. Το GPT-3 επιστρέφει μια λίστα κωδικών, οι οποίοι χρησιμοποιούνται ως honeywords. Η θερμοκρασία του GPT-3 ορίζεται σε 1 για να εξασφαλίσει την ποικιλομορφία των honeywords που παράγονται. Ωστόσο, αν ο αριθμός των επιστραφέντων honeywords είναι μεγαλύτερος από το n, τότε επιλέγονται οι κορυφαίοι n κωδικοί. Εάν ο αριθμός είναι λιγότερος από το n, χρησιμοποιείται η τεχνική CBT* για να προσαρμοστούν οι κωδικοί μέχρι να φτάσουν τον επιθυμητό αριθμό.