Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Εργαστηριο Συστηματων Τεχνητης Νοημοσυνης και Μαθησης

What makes us different: Understanding the differences between human and machine-generated text

Diploma Thesis

by

Anastasios Koukas

Επιβλέπων: Γεώργιος Στάμου
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2024

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Πληροφορικής
Εργαστήριο Συστημάτων Τεχνητής Νοημοσύνης και Μάθησης

What makes us different: Understanding the differences between human and machine-generated text

Diploma Thesis

by

Anastasios Koukas

Επιβλέπων: Γεώργιος Στάμου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 17η Ιουλίου, 2024.

.....	.....	.....
Γεώργιος Στάμου	Αθανάσιος Βουλόδημος	Ανδρέας-Γεώργιος Σταφυλοπάτης
Καθηγητής Ε.Μ.Π.	Επ. Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2024

.....

Αναστασιος Κουκας
Διπλωματούχος Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Σε μια εποχή όπου τα Μεγάλα Γλωσσικά Μοντέλα (Large Language Models-LLMs) παράγουν κείμενο που μιμείται κατά πολύ την ανθρώπινη γλώσσα, η χρησιμότητα του κειμένου που παράγεται από συστήματα τεχνητής νοημοσύνης(ΤΝ) εκτείνεται σε ποικίλες εφαρμογές, από τις ειδήσεις μέχρι κρίσιμους τομείς της κοινωνίας όπως ο νομικός και ο εκπαιδευτικός. Ωστόσο, η ραγδαία ανάπτυξη των LLMs εγείρει επίσης σημαντικούς κινδύνους, όπως οι ψευδείς ειδήσεις(fake news), η ανάμειξη κειμένου παραγόμενου από συστήματα ΤΝ στην ακαδημαϊκή έρευνα και διάφορα συστήματα απάτης. Για την καταπολέμηση αυτών των απειλών, είναι ζωτικής σημασίας η διάκριση μεταξύ ανθρώπινου κειμένου και κειμένου παραγόμενου από ΤΝ. Η παρούσα μελέτη διερευνά την αποτελεσματικότητα και την ευρωστία διαφόρων ανιχνευτών κειμένου τεχνητής νοημοσύνης, εστιάζοντας στην ικανότητά τους να αντιστέκονται σε επιθέσεις παραφράσεων από χρήστες-αντιπάλους. Διερευνούμε επίσης πώς η περιπλοκότητα κειμένου (text perplexity) , ένα μέτρο του πόσο απρόβλεπτο είναι ενα κείμενο για ένα μοντέλο ΤΝ, μπορεί να χρησιμεύσει ως αξιόπιστη μετρική για την ανίχνευση κειμένων παραγόμενων από ΤΝ και παρουσιάζουμε έναν ανιχνευτή με βάση την περιπλοκότητα που ανταγωνίζεται πιο σύνθετα μοντέλα ΤΝ. Επιπλέον, εξετάζουμε τον ρόλο των μεθόδων εξηγήσιμης τεχνητής νοημοσύνης (XAI) στην κατανόηση και τη βελτίωση των μηχανισμών ανίχνευσης. Μέσω μιας έρευνας χρηστών, συγκρίνουμε τις επιδόσεις του ανθρώπου και της τεχνητής νοημοσύνης στην ανίχνευση κειμένου, κατανοούμε τις γνωστικές αποφάσεις των ανθρώπων στο πεδίο αυτό και αξιολογούμε τις δυνατότητες των τεχνικών XAI για τη βελτίωση της ανθρώπινης λήψης αποφάσεων. Αυτή η ολοκληρωμένη ανάλυση αποσκοπεί στην ενίσχυση της ανάπτυξης ισχυρών και ερμηνεύσιμων συστημάτων ανίχνευσης κειμένου ΤΝ, εξασφαλίζοντας την αξιοπιστία τους σε εφαρμογές του πραγματικού κόσμου.

Λέξεις-κλειδιά — Μεγάλα Γλωσσικά Μοντέλα , Ανίχνευση κειμένων ΤΝ, Εξηγήσιμη Τεχνητή Νοημοσύνη, Εξηγήσεις με Αντιπαράδειγμα, Περιπλοκότητα κειμένου

Abstract

In an era where Large Language Models (LLMs) generate text that closely mimics human language, the utility of machine-generated text spans diverse applications, from news composition to critical fields like law and education. However, the proliferation of LLMs also raises significant risks, such as fake news, fraudulent schemes, and academic dishonesty. To combat these threats, it is crucial to distinguish between human and AI-generated text. This study explores the efficacy and robustness of various AI text detectors, focusing on their ability to withstand adversarial paraphrasing attacks. We also investigate how text perplexity, a measure of unpredictability of text for a model, can serve as a reliable metric for detection and introduce a perplexity-based detector that competes with more complex models. Additionally, we examine the role of explainable artificial intelligence (XAI) methods in understanding and improving detection mechanisms. Through a user survey, we compare human and AI performance in text detection, understand the cognitive decisions of humans in the task of and assess the potential of XAI techniques to enhance human decision-making. This comprehensive analysis aims to bolster the development of robust and interpretable AI text detection systems, ensuring their reliability in real-world applications.

Keywords — Large Language Models, AI text detection, Explainable AI, counterfactual explanations, adversarial attacks, text perplexity

Ευχαριστίες

Αυτό το έργο δεν θα ήταν δυνατό χωρίς την υποστήριξη κάποιων ανθρώπων. Γι'αυτό λοιπόν, ευχαριστώ πολύ τον επιβλέποντα μου, κ. Στάμου Γεώργιο, για την πολύτιμη στήριξη του στην εκπόνηση της εργασίας αυτής. Ευχαριστώ επίσης τον Ορφέα Μενή-Μαστρομιχαλάκη, για την καθοδήγηση, τις ιδέες και τις συμβουλές που μου έδωσε καθ'όλη τη διάρκεια αυτής της εξερεύνησης ενός πολύ ενδιαφέροντα τομέα της Τεχνητής Νοημοσύνης. Τέλος , θα ήθελα να ευχαριστήσω όλα τα μέλη του εργαστηρίου AILS και ιδιαίτερα τους καθηγητές και βοηθούς στο μάθημα των Προχρημένων Θεμάτων Τεχνητής Νοημοσύνης, μέσα από το οποίο γεννιούνται διαρκώς νέες ιδέες για έρευνα πάνω στον τομέα.

Επίσης, δε θα μπορούσα να μην ευχαριστήσω τους γονείς μου Αντώνη και Μαρία, καθώς και την αδερφή μου Δήμητρα για τη στήριξη που μου παρείχαν όλο αυτόν τον χρόνο εκπόνησης της εργασίας, αλλά και γενικότερα προκειμένου να εκπληρώσω τα όνειρα και τους στόχους στη ζωή μου.

Κουκάς Αναστάσιος, Ιούλιος 2024

Contents

List of Figures

Chapter 0

Εκτεταμένη Περίληψη στα Ελληνικά

Στη σύγχρονη εποχή, τα κείμενα που παράγονται από μεγάλα γλωσσικά μοντέλα (LLM) γίνονται όλο και πιο δυσδιάκριτα από την ανθρώπινη γλώσσα, με αποτέλεσμα την ευρεία χρήση τους σε διάφορες εφαρμογές, όπως η σύνταξη ειδήσεων, η δημιουργία ιστοριών, η δημιουργία κώδικα και κρίσιμα πεδία όπως η νομική και η εκπαίδευση. Μια πρόσφατη έρευνα έδειξε αύξηση κατά 57,3% των ειδησεογραφικών άρθρων που δημιουργούνται από LLM σε ιστότοπους γενικού ενδιαφέροντος μεταξύ Ιανουαρίου 2022 και Μαΐου 2023, αναδεικνύοντας την αυξανόμενη επιρροή αυτών των μοντέλων σε επαγγελματικά πλαίσια αλλά και πλαίσια της καθημερινότητας. Ωστόσο, η αυξημένη πρόσβαση του κοινού στα LLM εγείρει επίσης ανησυχίες σχετικά με την κακόβουλη χρήση του κειμένου που παράγεται από μηχανές, συμπεριλαμβανομένων των ψευδών ειδήσεων, των ψευδών κριτικών, των ακαδημαϊκών εργασιών που γράφονται με τεχνητή νοημοσύνη και άλλων αμφιλεγόμενων δραστηριοτήτων.

Για την αντιμετώπιση αυτών των απειλών, η διάκριση μεταξύ ανθρώπινου και μηχανικά παραγόμενου κειμένου καθίσταται απαραίτητη. Το έργο αυτό συχνά διαμορφώνεται ως ένα πρόβλημα δυαδικής ταξινόμησης (binary classification), όπου οι ανιχνευτές αναγνωρίζουν και φιλτράρουν το κείμενο που παράγεται από μηχανές με κακόβουλη πρόθεση. Ενώ οι σημερινοί ανιχνευτές κειμένου τεχνητής νοημοσύνης μπορούν να αποδώσουν αξιοσημείωτα καλά σε μη τροποποιημένο κείμενο που παράγεται από τεχνητή νοημοσύνη, αντιμετωπίζουν σημαντικές προκλήσεις όταν έρχονται αντιμέτωποι με επιθέσεις παράφρασης. Αυτές οι επιθέσεις περιλαμβάνουν λεπτές αναδιατυπώσεις του κειμένου που παράγεται από ΤΝ για να αποφύγουν την ανίχνευση ή μικρές αλλαγές σε κείμενο γραμμένο από άνθρωπο για να προκαλέσουν ψευδώς θετικά αποτελέσματα, εκμεταλλευόμενοι τις προκαταλήψεις και τις τάσεις υπερβολικής προσαρμογής των μοντέλων ανίχνευσης.

Η αυξανόμενη πολυπλοκότητα των επιθέσεων παράφρασης αναδεικνύει την ανάγκη για πιο ανθεκτικούς και διαφοροποιημένους μηχανισμούς ανίχνευσης κειμένου ΤΝ. Τα μελλοντικά μοντέλα ανίχνευσης πρέπει να είναι ικανά να διακρίνουν τις λεπτές συγκυριακές και υφολογικές αποχρώσεις που παραμένουν σταθερές παρά την παράφραση. Επιπλέον, το ζήτημα της επεξηγηματικότητας στην ανίχνευση κειμένου ΤΝ είναι ζωτικής σημασίας. Καθώς τα μοντέλα γίνονται πιο πολύπλοκα, οι αποφάσεις τους γίνονται πιο δύσκολο να ερμηνευτούν, πράγμα που είναι προβληματικό, δεδομένων των σοβαρών επιπτώσεων της λανθασμένης ταξινόμησης κειμένου, όπως η αμφισβήτηση της ακαδημαϊκής ακεραιότητας.

Η παρούσα εργασία αποσκοπεί στην καλύτερη κατανόηση των χαρακτηριστικών που διέπουν την ανίχνευση κειμένου ΤΝ και του τρόπου με τον οποίο τα μοντέλα μηχανικής μάθησης βασίζονται σε αυτά για να επιτύχουν υψηλή ακρίβεια. Με τη διερεύνηση επιθέσεων βασισμένων σε αντιπαραδείγματα, αξιολογούμε την ανθεκτικότητα των σημερινών αλγορίθμων ανίχνευσης και εντοπίζουμε τα στοιχεία που οδηγούν στην ταξινόμηση του κειμένου ως κειμένου που έχει δημιουργηθεί από ΤΝ ή από άνθρωπο. Εξετάζουμε επίσης την περιπλοκότητα κειμένου ως βασική μετρική για τη διάκριση μεταξύ κειμένων που έχουν παραχθεί από ΤΝ και ανθρώπινων κειμένων, προτείνοντας έναν απλό ανιχνευτή που βασίζεται στην περιπλοκότητα και έχει συγκρίσιμες επιδόσεις με πιο προηγμένα μοντέλα.

Επιπλέον, διεξάγουμε μια έρευνα χρηστών για να αξιολογήσουμε τις ανθρώπινες επιδόσεις στην ανίχνευση κειμένων ΤΝ και διερευνούμε τα κριτήρια που χρησιμοποιούν οι άνθρωποι σε σύγκριση με εκείνα που χρησιμοποιούν οι ανιχνευτές που βασίζονται σε LLM. Αναλύοντας τις ευθυγραμμίσεις και τις αποκλίσεις μεταξύ των αξιολογήσεων του ανθρώπου και του μοντέλου, στοχεύουμε στη βελτίωση των επιδόσεων τόσο του αν-

θρώπου όσο και της μηχανής κατά την ανίχνευση κειμένων που δημιουργούνται από ΤΝ. Αυτή η ολοκληρωμένη προσέγγιση επιδιώκει να ενισχύσει την ευρωστία και την αξιοπιστία των συστημάτων ανίχνευσης κειμένου ΤΝ, προωθώντας τελικά την υπεύθυνη διακυβέρνηση στην εποχή της ΤΝ.

Στην ενότητα 0.1 παρουσιάζεται συνοπτικά το θεωρητικό υπόβαθρο σε ότι αφορά συστήματα παραγωγής κειμένου ΤΝ αλλά και τα συστήματα ανίχνευσης που έχουν επικρατήσει, καθώς και άλλα στοιχεία το οποία χρησιμοποιούμε στην εργασία μας. Στην ενότητα 0.2 παρουσιάζονται τα πειράματα που πραγματοποιήσαμε πάνω στα συστήματα αυτά, ενώ στην ενότητα 0.3 παρουσιάζεται η έρευνα χρηστών που διεξάγαμε καθώς και τα αποτελέσματά της.

0.1 Θεωρητικό υπόβαθρο

0.1.1 Συστήματα παραγωγής κειμένου

Η παρούσα εργασία επικεντρώνεται σε συστήματα που ταξινομούν κείμενο είτε ως ανθρώπινης παραγωγής είτε ως παραγόμενο από μεγάλα γλωσσικά μοντέλα (LLM). Τέτοια συστήματα αναφέρονται ως "ανιχνευτές κειμένου"(text detectors) ή απλώς "ανιχνευτές"(detectors). Για την αποτελεσματική διαφοροποίηση μεταξύ κειμένου που παράγεται από τον άνθρωπο και κειμένου που παράγεται από μηχανές, είναι απαραίτητο να κατανοήσουμε τον τρόπο με τον οποίο τα LLM παράγουν κείμενο. Τα πιο συχνά χρησιμοποιούμενα μοντέλα, όπως η σειρά GPT, χρησιμοποιούν αρχιτεκτονικές μετασχηματιστών μονής κατεύθυνσης που προβλέπουν το επόμενο τμήμα κειμένου με βάση τα προηγούμενα χρησιμοποιώντας αυτοεπιβλεπόμενη μάθηση. Αρχικά, η παραγωγή κειμένου βασίστηκε σε ντετερμινιστικές μεθόδους όπως η άπληστη αναζήτηση και η αναζήτηση με δέσμη(beam search), οι οποίες όμως συχνά οδηγούσαν σε επαναλαμβανόμενο κείμενο. Αντίθετα, τα σύγχρονα μοντέλα χρησιμοποιούν στοχαστικές προσεγγίσεις, η δειγματοληψία πυρήνα, για την παραγωγή πιο ρευστού και συνεκτικού κειμένου.

Οι εξελίξεις στα μοντέλα παραγωγής κειμένου έχουν εισαγάγει τεχνικές όπως η κλιμάκωση της θερμοκρασίας [23], η οποία ρυθμίζει την τυχαιότητα της εξόδου, και η προ-εκπαίδευση μεγάλης κλίμακας που ακολουθείται από λεπτομερή ρύθμιση σε συγκεκριμένες εργασίες ή τομείς. Αυτές οι μέθοδοι βελτιώνουν την ικανότητα των μοντέλων να παράγουν ποικίλο και σχετικό με το πλαίσιο κείμενο. Για παράδειγμα, μοντέλα όπως το T5 [66] επιδεικνύουν αξιοσημείωτη ευελιξία αξιοποιώντας τόσο τη γενική γνώση που παρέχεται από την προ-εκπαίδευση, καθώς και τη λεπτομερή ρύθμιση για συγκεκριμένες εργασίες. Επιπλέον, η ενισχυτική μάθηση από ανθρώπινη ανατροφοδότηση (RLHF), όπως φαίνεται στο InstructGPT [65], ευθυγραμμίζει το παραγόμενο κείμενο με τις ανθρώπινες προτιμήσεις ενσωματώνοντας δεδομένα που παρέχονται από τον άνθρωπο στη διαδικασία εκπαίδευσης.

Αυτές οι εξελίξεις στην παραγωγή κειμένου, ενώ βελτιώνουν την ποιότητα και τη συνοχή του κειμένου που παράγεται από τεχνητή νοημοσύνη, παρουσιάζουν σημαντικές προκλήσεις για τους ανιχνευτές κειμένου. Η συνεχής βελτίωση των μεθόδων δειγματοληψίας, η στρατηγική χρήση της προ-εκπαίδευσης και της τελειοποίησης και η ενσωμάτωση της ανθρώπινης ανατροφοδότησης έχουν ενισχύσει συλλογικά τις δυνατότητες παραγωγής των LLMs. Κατά συνέπεια, οι ανιχνευτές κειμένου πρέπει να εξελίσσονται ώστε να συμβαδίζουν με αυτά τα εξελιγμένα μοντέλα, εξασφαλίζοντας την ακριβή διαφοροποίηση μεταξύ κειμένου που έχει παραχθεί από τον άνθρωπο και κειμένου που έχει παραχθεί από συστήματα TN, παρά την αυξανόμενη δυσκολία που θέτουν οι προηγμένες τεχνικές παραγωγής.

0.1.2 Ανίχνευση κειμένων TN από άνθρωπο

Προτού επικεντρωθούμε αποκλειστικά σε αυτοματοποιημένες μεθόδους ανίχνευσης για τον εντοπισμό κειμένων που παράγονται από μηχανές, είναι σημαντικό να αναγνωρίσουμε το ρόλο που μπορεί να διαδραματίσει ο άνθρωπος σε αυτό το έργο. Οι άνθρωποι ως συντονιστές, για παράδειγμα, μπορούν να επιβλέπουν τα αυτοματοποιημένα συστήματα, παρέχοντας μια ανθρώπινη πινελιά σε περιβάλλοντα όπως τα μέσα κοινωνικής δικτύωσης, όπου η συνεργασία μεταξύ των ανθρώπινων κριτών και των αυτόματων ανιχνευτών είναι απαραίτητη για την εξάλειψη του κακόβουλου περιεχομένου που παράγεται από μηχανές. Μελέτες έχουν εξετάσει τις επιδόσεις των ανθρώπινων αξιολογητών στον εντοπισμό κειμένων που δημιουργούνται με τεχνητή νοημοσύνη, υπογραμμίζοντας την ανάγκη για ανθρώπινη συμμετοχή παρά τις εξελίξεις στα αυτοματοποιημένα εργαλεία.

Έχουν αναπτυχθεί διάφορα εργαλεία για την ενίσχυση της ανθρώπινης ικανότητας στον εντοπισμό κειμένου που παράγεται από TN. Ένα αξιοσημείωτο παράδειγμα είναι το εργαλείο GLTR (Giant Language Model Test Room) [20], το οποίο χρησιμοποιεί στατιστικές ανωμαλίες στο κείμενο που παράγεται από μοντέλα όπως το GPT-2 για να βοηθήσει τους ανθρώπινους αξιολογητές. Το GLTR οπτικοποιεί την πιθανότητα εμφάνισης κάθε λέξης σε ένα κείμενο, βοηθώντας τους χρήστες να εντοπίσουν μοτίβα ενδεικτικά του περιεχομένου που παράγεται από μηχανές. Ωστόσο, αυτό το εργαλείο αντιμετωπίζει προκλήσεις με πιο προηγμένα μοντέλα όπως το GPT-3 και νεότερα, τα οποία χρησιμοποιούν δειγματοληψία top-p (πυρήνα) αντί για δειγματοληψία top-k, καθιστώντας την ανίχνευση πιο δύσκολη για τους ανθρώπους παρά τη βοήθεια του εργαλείου.

Οι επιδόσεις των ανθρώπινων αξιολογητών ποικίλλουν σημαντικά. Ορισμένες μελέτες, όπως μία που αφορά κείμενο που παράγεται από το GPT-3 [10], διαπίστωσαν ότι οι μη εκπαιδευμένοι άνθρωποι δεν έχουν καλύτερη απόδοση από την τυχαία επιλογή. Αντίθετα, μια άλλη μελέτη σε φοιτητές πανεπιστημίου έδειξε ότι οι άνθρωποι μπορούν να αναγνωρίσουν κείμενο που παράγεται από μηχανές με ακρίβεια περίπου 70%. Το εργαλείο RoFT (Real or Fake Text) [13] αξιολογεί τις ανθρώπινες επιδόσεις στην αναγνώριση του ορίου όπου το κείμενο που

έχει γραφτεί από άνθρωπο μεταπίπτει σε κείμενο που έχει παραχθεί από μηχανή. Αυτό το εργαλείο δείχνει ότι οι άνθρωποι έχουν καλύτερες επιδόσεις από την τυχαία επιλογή, αν και η ευκολία της ανίχνευσης επηρεάζεται από την πολυπλοκότητα του μοντέλου, με τα μικρότερα μοντέλα να είναι ευκολότερα στην ανίχνευση.

Η πρόσφατη έρευνα δείχνει ότι ενώ οι άνθρωποι μπορούν να αποδώσουν καλύτερα από την τυχαία επιλογή στην ανίχνευση κειμένου που παράγεται από μηχανή, η ακρίβειά τους μειώνεται καθώς τα LLM γίνονται πιο περίπλοκα. Ακόμη και οι εκπαιδευμένοι ανθρώπινοι κριτές δυσκολεύονται να φτάσουν την ακρίβεια των τελευταίας τεχνολογίας αυτοματοποιημένων ανιχνευτών, οι οποίοι υπερέχουν σε καταστάσεις όπου οι άνθρωποι είναι πιο πιθανό να εξαπατηθούν από κείμενο που μοιάζει με ανθρώπινο. Επιπλέον, η επεκτασιμότητα αποτελεί σημαντική πρόκληση για την ανίχνευση ανθρώπινου περιεχομένου- καθώς ο όγκος του περιεχομένου αυξάνεται, οι αυτοματοποιημένοι ανιχνευτές είναι απαραίτητοι για την αποτελεσματική επεξεργασία μεγάλων ποσοτήτων κειμένου. Ωστόσο, ο συνδυασμός της ανθρώπινης επίβλεψης με την αυτοματοποιημένη ανίχνευση παραμένει ζωτικής σημασίας, καθώς οι ανθρώπινοι κριτές μπορούν να χειριστούν ακραίες περιπτώσεις, να επαληθεύσουν τις αυτοματοποιημένες αποφάσεις και να ελέγξουν για προκαταλήψεις στους αλγορίθμους τεχνητής νοημοσύνης.

Για να αντιμετωπιστούν αυτές οι προκλήσεις, η παρούσα μελέτη περιλαμβάνει μια έρευνα χρηστών, όπου οι συμμετέχοντες αναλαμβάνουν την ανίχνευση κειμένου τεχνητής νοημοσύνης σε διάφορους τομείς και πλαίσια. Αυτή η έρευνα αποσκοπεί στη σύγκριση των ανθρώπινων επιδόσεων με τους αυτοματοποιημένους ανιχνευτές και στη συλλογή δεδομένων σχετικά με το τι επηρεάζει τις ανθρώπινες αποφάσεις κατά την ταξινόμηση κειμένων. Τα ευρήματα αυτής της έρευνας θα βοηθήσουν στη βελτίωση του σχεδιασμού των συστημάτων ανίχνευσης και θα παράσχουν πληροφορίες σχετικά με τις ανθρώπινες γνωστικές διεργασίες στην ανίχνευση κειμένων ΤΝ. Ενσωματώνοντας τις ανθρώπινες γνώσεις με προηγμένους αλγορίθμους ανίχνευσης, ο στόχος είναι να αναπτυχθεί μια πιο ολοκληρωμένη και αποτελεσματική προσέγγιση για τον εντοπισμό περιεχομένου που παράγεται από μηχανές. Η λεπτομερής ανάλυση και τα αποτελέσματα της έρευνας χρηστών παρουσιάζονται στην ενότητα 0.3 στα ελληνικά και πιο αναλυτικά στο κεφάλαιο 5 στα αγγλικά.

0.1.3 Αυτοματοποιημένες μέθοδοι ανίχνευσης

Το έργο της διάκρισης μεταξύ ανθρώπινου και μηχανικά παραγόμενου κειμένου έχει συγκεντρώσει σημαντικό ενδιαφέρον από την κοινότητα του NLP, με αποτέλεσμα να έχει δημιουργηθεί ένα ευρύ φάσμα τεχνικών τεχνητής νοημοσύνης και μηχανικής μάθησης.

Ανίχνευση βάσει χαρακτηριστικών

Μια εξέχουσα προσέγγιση είναι η ανίχνευση βάσει χαρακτηριστικών, η οποία επικεντρώνεται στον εντοπισμό συγκεκριμένων χαρακτηριστικών που διαφοροποιούν το ανθρώπινο κείμενο από αυτό κείμενο που παράγεται από μηχανή. Αυτά τα χαρακτηριστικά συχνά αναδεικνύουν εγγενείς αδυναμίες στο περιεχόμενο που παράγεται από ΤΝ, όπως η έλλειψη συντακτικής και λεξιλογικής ποικιλομορφίας, συνοχής και σκοπού, καθώς και ζητήματα επαναληπτικότητας. Το ανθρώπινο κείμενο παρουσιάζει γενικά ένα ευρύτερο φάσμα δομών προτάσεων και λεξιλογίου, διατηρεί μια λογική ροή και αποφεύγει τις περιττές επαναλήψεις, καθιστώντας το πιο πλούσιο και πιο ελκυστικό σε σύγκριση με τα αντίστοιχα κείμενα που παράγονται από μηχανές.

Οι μέθοδοι ανίχνευσης με βάση τα χαρακτηριστικά ποσοτικοποιούν αυτές τις διαφορές για την ανάπτυξη κριτηρίων αναγνώρισης κειμένου που παράγεται από μηχανήματα. Για παράδειγμα, η μέτρηση της επαναληπτικότητας των n-grams μπορεί να αποκαλύψει την υπερβολική χρήση ορισμένων φράσεων, ένα κοινό χαρακτηριστικό στο περιεχόμενο που παράγεται από τεχνητή νοημοσύνη. Μελέτες έχουν δείξει ότι αυτή η προσέγγιση μπορεί να επιτύχει υψηλή ακρίβεια, ιδίως με παλαιότερα μοντέλα που χρησιμοποιούν στρατηγικές δειγματοληψίας top-k. Μια άλλη μέθοδος περιλαμβάνει την εξέταση της συχνότητας και της κατανομής των χαρακτήρων, η οποία μπορεί να διαφέρει ανάλογα με τη μέθοδο δειγματοληψίας που χρησιμοποιεί η ΤΝ. Στυλομετρικά χαρακτηριστικά, όπως το μέσο μήκος πρότασης και ο αριθμός των σημείων στίξης, έχουν επίσης χρησιμοποιηθεί για τον εντοπισμό διαφορών, ιδίως σε μικρότερα κείμενα όπως τα tweets.[39],[63]

Παρά την αρχική τους επιτυχία, οι παραδοσιακές μέθοδοι που βασίζονται σε χαρακτηριστικά έχουν καταστεί σε μεγάλο βαθμό παρωχημένες έναντι προηγμένων μοντέλων και εξελιγμένων τεχνικών δειγματοληψίας. Τα σύγχρονα παραγωγικά LLM, ιδίως εκείνα που χρησιμοποιούν δειγματοληψία top-p (πυρήνα), παράγουν κείμενο που είναι πιο ποικίλο και κατάλληλο για το πλαίσιο Που τίθεται, μειώνοντας την αποτελεσματικότητα αυτών των μεθόδων. Καθώς τα γλωσσικά μοντέλα έχουν γίνει μεγαλύτερα και πιο εξελιγμένα, το έργο της ανίχνευσης έχει γίνει όλο και πιο δύσκολο. Οι μέθοδοι που βασίζονται σε χαρακτηριστικά, αν και εξακολουθούν να είναι

χρήσιμες ως μέτρο σύγκρισης, έχουν ξεπεραστεί από προηγμένα μοντέλα μηχανικής μάθησης όσον αφορά την ακρίβεια και την ανθεκτικότητα σε επιθέσεις.

Παρ' όλα αυτά, οι παραδοσιακές μέθοδοι που βασίζονται σε χαρακτηριστικά διατηρούν κάποια αξία λόγω της απλότητας και της ευκολίας πειραματισμού τους. Παρέχουν ένα χρήσιμο σημείο αναφοράς για την αξιολόγηση της απόδοσης πιο σύνθετων ανιχνευτών, διασφαλίζοντας ότι οι προηγμένες μέθοδοι αποδίδουν τουλάχιστον το ίδιο καλά με τις απλούστερες προσεγγίσεις. Σε ορισμένες περιπτώσεις, ο συνδυασμός μεθόδων βασισμένων σε χαρακτηριστικά με προηγμένα εργαλεία, όπως μοντέλα μετασχηματιστών, μπορεί να βελτιώσει την ακρίβεια ανίχνευσης. Η στυλομετρική ανάλυση συνεχίζει να είναι σημαντική, ιδίως σε περιπτώσεις όπου οι μέθοδοι που βασίζονται σε LLM μπορεί να είναι λιγότερο αποτελεσματικές, όπως τα πολύ σύντομα κείμενα.

Εν κατακλείδι, ενώ οι μέθοδοι ανίχνευσης με βάση τα χαρακτηριστικά δεν αποτελούν πλέον τη λύση αιχμής, παραμένουν ένα πολύτιμο εργαλείο στη συνεχιζόμενη προσπάθεια ανίχνευσης κειμένων που παράγονται από τεχνητή νοημοσύνη. Η ευκολία χρήσης τους, η ικανότητά τους να χρησιμεύουν ως σημείο αναφοράς και η δυνατότητα ενσωμάτωσης με πιο προηγμένες τεχνικές διασφαλίζουν ότι εξακολουθούν να διαδραματίζουν ρόλο στην ανάπτυξη ισχυρών συστημάτων ανίχνευσης.

Ανίχνευση με μεθόδους μηδενικών δειγμάτων

Οι μέθοδοι ανίχνευσης μηδενικών δειγμάτων για την αναγνώριση κειμένου που παράγεται από μηχανές αξιοποιούν τα εκπαιδευμένα γλωσσικά μοντέλα (LLM) χωρίς πρόσθετη λεπτομερή εκπαίδευση με δείγματα κειμένων. Μια πρώιμη προσέγγιση τέτοιων μεθόδων χρησιμοποιούσε τη συνολική λογαριθμική πιθανότητα του κειμένου, εφαρμόζοντας ένα κατώφλι για την ταξινόμηση του κειμένου ως παραγόμενου από μηχανή, εάν η πιθανότητά του ήταν πιο κοντά στο μέσο όρο του παραγόμενου από μηχανή κειμένου από ό,τι του ανθρώπινου κειμένου. Ωστόσο, αυτή η μέθοδος δεν υπερείχε σε σχέση με τις παραδοσιακές τεχνικές που βασίζονται σε χαρακτηριστικά. Σημαντική πρόοδος στην ανίχνευση με μεθόδους μηδενικών δειγμάτων σημειώθηκε με την εισαγωγή του DetectGPT[57], το οποίο ανιχνεύει κείμενο που δημιουργήθηκε από μηχανήματα εξετάζοντας τις περιοχές αρνητικής καμπυλότητας της λογαριθμικής πιθανότητας του μοντέλου. Το DetectGPT διαταράσσει ελαφρώς το κείμενο και το βαθμολογεί ως μηχανικά παραγόμενο εάν αυτές οι διαταραχές έχουν χαμηλότερες λογαριθμικές πιθανότητες από το αρχικό κείμενο, επιτυγχάνοντας ανώτερες επιδόσεις σε σύγκριση με άλλες μεθόδους μηδενικών δειγμάτων κατά την κυκλοφορία του.

Παρά την αρχική του επιτυχία, το DetectGPT έχει αρκετούς περιορισμούς. Απαιτεί τη γνώση του συγκεκριμένου LLM που χρησιμοποιείται για τη δημιουργία κειμένου, καθώς τα διάφορα μοντέλα παρουσιάζουν διαφορετικές πιθανότητες για λέξεις με βάση τα δεδομένα εκπαίδευσής τους. Αυτό το καθιστά μη πρακτικό για σενάρια που περιλαμβάνουν πολλαπλά ή άγνωστα LLM. Επιπλέον, το DetectGPT είναι υπολογιστικά ακριβό, καθώς και ευάλωτο σε επιθέσεις παράφρασης, όπου ελαφρώς τροποποιημένο κείμενο εξαπατά τον ανιχνευτή. Παρόλο που έχουν προταθεί βελτιώσεις όπως η αποτελεσματική δειγματοληψία [3] και τα υποκατάστατα μοντέλα Bayes[54], αυτές οι βελτιώσεις αντιμετωπίζουν μόνο εν μέρει τα προβλήματα ευρωστίας και εξάρτησης του DetectGPT από το συγκεκριμένο μοντέλο, περιορίζοντας την πρακτική του χρήση. Παράλληλα ,έχουν εμφανιστεί και άλλες μέθοδοι ανίχνευσης μηδενικών δειγμάτων, όπως η αξιοποίηση της πληροφορίας κατάταξης λογαρίθμου και η εκτίμηση της εγγενούς διαστατικότητας των ενσωματώσεων κειμένου. Αυτές οι προσεγγίσεις είναι πολλά υποσχόμενες, αλλά απαιτούν περαιτέρω δοκιμές σε πρακτικά περιβάλλοντα.

Η ανάλυση της περιπλοκότητας κειμένου είναι μια άλλη αποτελεσματική μέθοδος μηδενικών δειγμάτων. Μετρά την απρόβλεπτη φύση του κειμένου, με υψηλότερη περιπλοκότητα να υποδεικνύει περιεχόμενο που έχει παραχθεί από μηχανή. Οι ανιχνευτές που βασίζονται στην περιπλοκότητα συγκρίνουν την περιπλοκότητα ενός συγκεκριμένου κειμένου με τιμές κατωφλίου που προέρχονται από γνωστά κείμενα που έχουν παραχθεί από ανθρώπους και μηχανές, αναγνωρίζοντας αξιόπιστα κείμενα που έχουν παραχθεί από τεχνητή νοημοσύνη σε πολλές περιπτώσεις.

Οι ανιχνευτές που βασίζονται στην περιπλοκότητα του κειμένου έχουν αποδειχθεί ότι είναι από τους καλύτερους ανιχνευτές μηδενικών δειγμάτων. Στην εργασία μας, παρουσιάζουμε έναν απλό ανιχνευτή βασισμένο στην περιπλοκότητα κειμένου, αποδεικνύοντας την απόδοσή του στο ίδιο επίπεδο με τα καλύτερα λεπτομερώς ρυθμισμένα μοντέλα σε διάφορους τομείς. Αυτό αναδεικνύει τη δυνατότητα των μεθόδων ανίχνευσης μηδενικών δειγμάτων να παραμείνουν αποτελεσματικές καθώς τα γλωσσικά μοντέλα συνεχίζουν να εξελίσσονται, προσφέροντας μια πρακτική και εύκολα κλιμακούμενη λύση για την ανίχνευση κειμένων τεχνητής νοημοσύνης.

Ανίχνευση με λεπτομερή ρύθμιση μοντέλων μηχανικής μάθησης

Η προσέγγιση που θεωρείται αποτελεσματικότερη στην ανίχνευση κειμένων TN είναι η λεπτομερής ρύθμιση(fine-tuning) προ-εκπαιδευμένων γλωσσικών μοντέλων, όπως το BERT ή το RoBERTa[83]. Αυτή περιλαμβάνει την εκπαίδευση αυτών των μοντέλων με παραδείγματα ανίχνευσης υπό επίβλεψη. Η προσέγγιση έχει δείξει μεγάλη αποτελεσματικότητα, ιδίως με μοντέλα που ανιχνεύουν κείμενο από τα ίδια ή παρόμοια μοντέλα. Μελέτες έχουν διαπιστώσει ότι η λεπτομερής ρύθμιση του RoBERTa σε παραδείγματα δειγματοληψίας top-p μπορεί να επιτύχει υψηλή ακρίβεια σε διάφορες μεθόδους δειγματοληψίας, ξεπερνώντας ακόμη και τη λεπτομερή ρύθμιση του μονόδρομου μοντέλου GPT-2. Ωστόσο, οι ανιχνευτές λεπτομερούς ρύθμισης απαιτούν σημαντικά παραδείγματα εκπαίδευσης ανά κλάση για βέλτιστη απόδοση και ενδέχεται να δυσκολεύονται να γενικεύσουν σε διάφορους τομείς.

Μια σημαντική πρόκληση με τους λεπτομερώς ρυθμισμένους ανιχνευτές είναι η εξειδίκευσή τους- τείνουν να αποδίδουν εξαιρετικά καλά στον τομέα στον οποίο εκπαιδεύτηκαν, αλλά λιγότερο αποτελεσματικά εκτός αυτού. Για παράδειγμα, ενώ οι ανιχνευτές με βάση το RoBERTa υπερέχουν σε εργασίες όπως η ανίχνευση ψευδών ειδήσεων ή η ανάλυση ακαδημαϊκών εγγράφων[46][47], η απόδοσή τους πέφτει σε ευρύτερα πλαίσια. Αυτός ο συμβιβασμός μεταξύ της ακρίβειας σε συγκεκριμένο τομέα και της γενίκευσης είναι ζωτικής σημασίας και στα πειράματά μας χρησιμοποιούμε τόσο το RoBERTa, όσο και μικρότερους ανιχνευτές που βασίζονται στον BERT για να καταδείξουμε αυτό το σημείο.

Επιπλέον, οι λεπτομερώς συντονισμένοι ανιχνευτές αντιμετωπίζουν ζητήματα ευρωστίας. Δεδομένης της φύσης των μοντέλων ανοικτού κώδικα όπως το BERT και το RoBERTa, μπορεί να είναι ευάλωτα σε επιθέσεις από αντιπάλους, όπου μια μικρή παράφραση κειμένου μπορεί να εξαπατήσει τον ανιχνευτή. Παρά τις προκλήσεις αυτές, η λεπτομερής ρύθμιση παραμένει μια εξέχουσα μέθοδος για την ανίχνευση κειμένου με τεχνητή νοημοσύνη, ιδίως σε συγκεκριμένους τομείς όπου η υψηλή ακρίβεια είναι ζωτικής σημασίας και υπάρχουν επαρκή δεδομένα εκπαίδευσης. Η προσαρμοστικότητα αυτής της μεθόδου στα ειδικά χαρακτηριστικά του τομέα της επιτρέπει να επιτυγχάνει ανώτερες επιδόσεις σε σύγκριση με πιο γενικές προσεγγίσεις ανίχνευσης.

Ανίχνευση με μοντέλα αντιφατικής εκπαίδευσης

Πλαίσια όπως το RADAR[27] και το OUTFOX[37] είναι πρωτοπόρα στην εξέλιξη των ανιχνευτών κειμένου τεχνητής νοημοσύνης μέσω αντιφατικών παραδειγμάτων εκπαίδευσης. Αυτά τα πλαίσια χρησιμοποιούν μια προσέγγιση διπλού μοντέλου: έναν ανιχνευτή που εκπαιδεύεται για να αναγνωρίζει κείμενο που παράγεται από TN και έναν παραφραστή που εκπαιδεύεται για να παράγει κείμενο που μπορεί να εξαπατήσει τον ανιχνευτή. Εκπαιδεύοντας επαναληπτικά αυτά τα μοντέλα σε αντιπαράθεση, προσομοιώνουν αντίστοιχα σενάρια του πραγματικού κόσμου, ενισχύοντας την ανθεκτικότητα του ανιχνευτή απέναντι σε εξελιγμένες επιθέσεις, συμπεριλαμβανομένων της παράφρασης και της χειραγώγησης βάσει προτροπής(prompt manipulation).

Οι μέθοδοι αντιφατικής εκπαίδευσης προσομοιώνουν σενάρια επίθεσης κατά τη διάρκεια της εκπαίδευσης, όπου ο παραφραστής παράγει παραδείγματα που προκαλούν τον ανιχνευτή με στόχο την εξαπάτησή του. Αυτή η αντιφατική προσέγγιση όχι μόνο βελτιώνει την ακρίβεια του ανιχνευτή σε αρχικές εργασίες ανίχνευσης, αλλά και ενισχύει σημαντικά την ανθεκτικότητά του έναντι αντιφατικών επιθέσεων σε σύγκριση με τις παραδοσιακές μεθόδους λεπτομερούς ρύθμισης. Αυτή η ανθεκτικότητα είναι ιδιαίτερα κρίσιμη σε πρακτικές εφαρμογές όπου το κείμενο που παράγεται από τεχνητή νοημοσύνη μπορεί να χειραγωγηθεί για να αποφύγει την ανίχνευση.

Στη δική μας έρευνα, δίνουμε έμφαση στην αξιολόγηση των ανιχνευτών που βασίζονται στην αντιφατική μάθηση, όπως το RADAR, για να εκτιμήσουμε την αποτελεσματικότητά τους έναντι αντιφατικών επιθέσεων που μοιάζουν με παραφράσεις. Αυτό αναδεικνύει μια στροφή προς πιο εξελιγμένα πλαίσια ανίχνευσης που μπορούν να αντέξουν τις εξελισσόμενες απειλές που βασίζονται σε κείμενο που παράγεται από τεχνητή νοημοσύνη, εξασφαλίζοντας έτσι πιο αξιόπιστους και ανθεκτικούς μηχανισμούς ανίχνευσης για ποικίλες εφαρμογές.

Ανίχνευση με υδατοσήμανση

Η υδατοσήμανση (watermarking) έχει αναδειχθεί ως μια νέα προσέγγιση για την αντιμετώπιση του ολοένα και πιο πολύπλοκου έργου της διάκρισης κειμένου που παράγεται με TN από περιεχόμενο γραμμένο από τον άνθρωπο. Η τεχνική αυτή περιλαμβάνει την ενσωμάτωση κρυφών δεικτών στο κείμενο που παράγεται από μεγάλα γλωσσικά μοντέλα (LLM), οι οποίοι είναι ανεπαίσθητοι από τους ανθρώπινους αναγνώστες αλλά ανιχνεύονται από ειδικούς αλγορίθμους. Πρωταρχικός στόχος είναι η δημιουργία μιας αξιόπιστης υπογραφής που υποδεικνύει

την προέλευση του κειμένου, βοηθώντας έτσι στην ανίχνευση και τη ρύθμιση του περιεχομένου που παράγεται από TN. Για παράδειγμα, η μέθοδος που περιγράφεται λεπτομερώς στο [35] κατηγοριοποιεί τα τμήματα ενός κειμένου σε "πράσινες" και "κόκκινες" λίστες χρησιμοποιώντας τυχαιοποίηση, διασφαλίζοντας ότι το υδατογράφημα παραμένει κρυφό για τους ανθρώπους αλλά αναγνωρίσιμο από αυτοματοποιημένους ανιχνευτές που είναι εξοικειωμένοι με τις λίστες.

Ωστόσο, η ευρεία υιοθέτηση της υδατοσήμανσης σε τρέχοντα μοντέλα παραγωγής κειμένου τεχνητής νοημοσύνης όπως το ChatGPT παραμένει περιορισμένη λόγω διαφόρων προκλήσεων. Στα τεχνικά εμπόδια περιλαμβάνεται η ανάπτυξη υδατογραφήματος που είναι τόσο ανθεκτικό απέναντι σε διάφορους χειρισμούς του κειμένου όσο και αρκετά μη ανιχνεύσιμο ώστε να μην υποβαθμίζει την ευχέρεια και την ποιότητα του παραγόμενου κειμένου. Επιπλέον, ηθικοί προβληματισμοί γύρω από την πιθανή κατάχρηση του υδατογραφημένου περιεχομένου και ανησυχίες σχετικά με τον αντίκτυπο στις επιδόσεις περιπλέκουν περαιτέρω την εφαρμογή της. Οι προσπάθειες συντονισμού για την τυποποίηση σε διαφορετικά μοντέλα και περιβάλλοντα ανάπτυξης θέτουν επίσης πρακτικά εμπόδια.

Παρά τις προκλήσεις αυτές, η υδατοσήμανση συνεχίζει να ερευνάται και να βελτιώνεται ενεργά. Πρόσφατες μελέτες διερευνούν την ανθεκτικότητα των υδατοσημάτων υπό ποικίλες συνθήκες, όπως η ανθρώπινη επαναδιατύπωση, η παράφραση από μη υδατοσημασμένα μοντέλα ή η ενσωμάτωση σε έγγραφα μεγαλύτερης διάρκειας[36]. Οι προσπάθειες για τη βελτίωση των αλγορίθμων υδατοσήμανσης αποσκοπούν στη διασφάλιση της αποτελεσματικότητάς τους σε διάφορα σενάρια και στην αντιμετώπιση ζητημάτων ασφάλειας, όπως η αποτροπή της ανίχνευσης πλαστών εγγράφων ή της μη εξουσιοδοτημένης χειραγώγησης.

Εν κατακλείδι, ενώ η υδατοσήμανση υπόσχεται την πρόοδο των δυνατοτήτων ανίχνευσης κειμένου TN, η τρέχουσα εφαρμογή της σε επικρατούντα μοντέλα παραμένει περιορισμένη λόγω τεχνικών περιπλοκών και πρακτικών εκτιμήσεων. Οι τρέχουσες ερευνητικές προσπάθειες αποσκοπούν στην αντιμετώπιση αυτών των προκλήσεων, ανοίγοντας ενδεχομένως το δρόμο για την ευρύτερη υιοθέτηση και ενσωμάτωση της υδατογράφησης ως εργαλείου για την ενίσχυση της ιχνηλασιμότητας και της ρύθμισης του κειμένου που παράγεται από TN στο μέλλον.

0.1.4 Περιπλοκότητα κειμένου

Η περιπλοκότητα κειμένου (text perplexity) χρησιμεύει ως θεμελιώδης μετρική στην επεξεργασία φυσικής γλώσσας (NLP), ιδίως για την αξιολόγηση της προβλεπτικής ικανότητας γλωσσικών μοντέλων, όπως το GPT-2 ή το OPT, τα οποία λειτουργούν με αυτοπαλινδρομικό (auto-regressive) τρόπο. Ποσοτικοποιεί την αβεβαιότητα αυτών των μοντέλων κατά την πρόβλεψη της επόμενης λέξης ή τμήματος με βάση τα προηγούμενα συμφραζόμενα. Μια χαμηλότερη βαθμολογία περιπλοκότητας υποδηλώνει ότι το μοντέλο προβλέπει την ακολουθία κειμένου με μεγαλύτερη ακρίβεια, γεγονός που υποδηλώνει υψηλότερη προβλεψιμότητα και, συνεπώς, ενδεχομένως υποδηλώνει ότι το κείμενο δημιουργήθηκε από το ίδιο το μοντέλο. Αντίθετα, ένα υψηλότερο σκορ περιπλοκότητας σημαίνει πιο απρόβλεπτη συμπεριφορά, υποδηλώνοντας ότι το κείμενο μπορεί να μοιάζει περισσότερο με ανθρώπινο, καθώς οι άνθρωποι συχνά παράγουν λιγότερο προβλέψιμα γλωσσικά πρότυπα.

Η εφαρμογή της περιπλοκότητας είναι στενά συνδεδεμένη με την αρχιτεκτονική του γλωσσικού μοντέλου. Τα μοντέλα αυτόματης παλινδρόμησης προβλέπουν τα τμήματα λέξεων (tokens) διαδοχικά, με κάθε πρόβλεψη να εξαρτάται από τα προηγούμενα τμήματα. Ο τύπος της περιπλοκότητας αντικατοπτρίζει αυτό το γεγονός υπολογίζοντας την αρνητική λογαριθμική πιθανότητα κάθε token δεδομένου του προηγούμενου πλαισίου. Αυτό καθιστά την περιπλοκότητα κατάλληλο μέτρο για την αξιολόγηση του πόσο καλά ένα μοντέλο συμμορφώνεται με τα στατιστικά πρότυπα της φυσικής γλώσσας.

Σε πρόσφατες μελέτες, η περιπλοκότητα έχει αξιοποιηθεί πέρα από την αξιολόγηση μοντέλων για να βοηθήσει στην ανίχνευση κειμένων τεχνητής νοημοσύνης. Για παράδειγμα, το HowkGPT[85] χρησιμοποιεί την ανάλυση περιπλοκότητας για να διακρίνει τις εργασίες που δημιουργούνται από το ChatGPT από εκείνες που έχουν γραφτεί από ανθρώπους. Η προσέγγιση αυτή αξιοποιεί την παραδοχή ότι το κείμενο που παράγεται από TN τείνει να εμφανίζει χαμηλότερες βαθμολογίες περιπλοκότητας λόγω της προβλέψιμης φύσης του, σε αντίθεση με την υψηλότερη περιπλοκότητα και απρόβλεπτη φύση που συνήθως συναντάται στο κείμενο που παράγεται από άνθρωπο.

Επιπλέον, εμπορικοί ανιχνευτές όπως ο GPTZero[80] έχουν ενσωματώσει σχετικές μετρικές όπως η "έκρηξη", η οποία αντικατοπτρίζει τη συχνότητα των επαναλαμβανόμενων φράσεων σε τμήματα κειμένου που παράγονται

από TN. Αυτές οι μετρικές συμπληρώνουν την περιπλοκότητα στον εντοπισμό μοτίβων που διαφοροποιούν το περιεχόμενο που παράγεται από μηχανές και το περιεχόμενο που γράφεται από τον άνθρωπο. Ακόμη πιο προηγμένες υλοποιήσεις, όπως ο ανιχνευτής Binoculars[25], προχωρούν ένα βήμα παραπέρα εισάγοντας την ανάλυση της διασταυρούμενης περιπλοκότητας. Συγκρίνοντας την περιπλοκότητα ενός δείγματος κειμένου που παράγεται από ένα μοντέλο με εκείνη ενός άλλου μοντέλου, αυτοί οι ανιχνευτές ενισχύουν την ακρίβεια στη διάκριση μεταξύ κειμένων που έχουν παραχθεί από TN και κειμένων που έχουν συνταχθεί από άνθρωπο σε διάφορα σύνολα δεδομένων.

Στην έρευνά μας, αναλύουμε την αποτελεσματικότητα της περιπλοκότητας ως βασικού χαρακτηριστικού στην ανίχνευση κειμένων TN. Υλοποιούμε έναν απλό ανιχνευτή με βάση την περιπλοκότητα που μοιάζει με τον HowkGPT και αξιολογούμε την απόδοσή του σε πολλαπλά σύνολα δεδομένων. Η αξιολόγηση αυτή περιλαμβάνει τη σύγκριση των αποτελεσμάτων του με εκείνα από εξελιγμένους ανιχνευτές όπως το Binoculars, οι οποίοι παρέχουν πληροφορίες σχετικά με την ευρωστία και τη διακριτική ικανότητα της περιπλοκότητας στην αναγνώριση κειμένου που έχει δημιουργηθεί από TN. Μέσω ολοκληρωμένων δοκιμών και συγκρίσεων, στοχεύουμε να διαφωτίσουμε τον τρόπο με τον οποίο η περιπλοκότητα συμβάλλει στο ευρύτερο τοπίο της ανίχνευσης κειμένου TN, αναδεικνύοντας τα δυνατά σημεία και τους πιθανούς περιορισμούς της σε πραγματικές εφαρμογές.

Τελικά, ενώ η περιπλοκότητα παραμένει μια βασική μετρική για την αξιολόγηση των επιδόσεων των γλωσσικών μοντέλων, η εφαρμογή της στην ανίχνευση κειμένων TN υπογραμμίζει τη χρησιμότητά της στη διαφοροποίηση μεταξύ κειμένων που έχουν παραχθεί από μηχανές και κειμένων που έχουν συνταχθεί από ανθρώπους. Καθώς η έρευνα στον τομέα της επεξεργασίας φυσικής γλώσσας συνεχίζει να εξελίσσεται, η ενσωμάτωση της περιπλοκότητας σε πλαίσια ανίχνευσης μαζί με άλλες προηγμένες τεχνικές υπόσχεται να βελτιώσει την ακρίβεια και την αξιοπιστία στον εντοπισμό περιεχομένου που έχει δημιουργηθεί από TN σε διάφορους τομείς.

0.1.5 Επιθέσεις παράφρασης

Η πρόκληση της ακριβούς ανίχνευσης κειμένου που παράγεται από τεχνητή νοημοσύνη εν μέσω ολοένα και πιο εξελιγμένων επιθέσεων, όπως η παράφραση, εξακολουθεί να αποτελεί σημαντικό πρόβλημα στην έρευνα της επεξεργασίας φυσικής γλώσσας. Οι επιθέσεις παράφρασης περιλαμβάνουν τη χρήση ελαφρών νευρωνικών δικτύων για την τροποποίηση του κειμένου που παράγεται από μεγάλα γλωσσικά μοντέλα (LLM) με τρόπους που αποφεύγουν σύγχρονους ανιχνευτές κειμένου TN. Πρόσφατες μελέτες [70] καταδεικνύουν ότι οι επιθέσεις αυτές μπορούν να παρακάμψουν με επιτυχία ποικίλες μεθόδους ανίχνευσης, συμπεριλαμβανομένης της υδατοσήμανσης και των ανιχνευτών που βασίζονται σε νευρωνικά δίκτυα όπως το DetectGPT ή οι ανιχνευτές με λεπτομερή ρύθμιση. Αυτό αναδεικνύει την ανθεκτικότητα των τεχνικών παράφρασης και τη δυσκολία επίτευξης ισχυρής ανίχνευσης απέναντι σε τέτοιου είδους χειρισμούς.

Στο [38] προτείνεται μια στρατηγική άμυνας με την ανάκτηση σημασιολογικά παρόμοιου κειμένου που παράγεται από ένα LLM, υποθέτοντας ότι το κείμενο αυτό θα παρουσιάζει χαρακτηριστικά που είναι πιο τυπικά για περιεχόμενο που παράγεται από τον άνθρωπο. Ωστόσο, αυτή η προσέγγιση είναι ειδική για το μοντέλο και εγείρει ανησυχίες για την προστασία της ιδιωτικότητας λόγω της εξάρτησής της από την πρόσβαση στις εξόδους του LLM. Εν τω μεταξύ, συνεχίζονται οι συζητήσεις σχετικά με την ταξινόμηση κειμένου που έχει παραφραστεί από ανθρώπους μετά την αρχική παραγωγή του από LLM. Τέτοιο κείμενο, που χρησιμοποιείται συχνά σε εκπαιδευτικά πλαίσια, θολώνει τη διάκριση μεταξύ περιεχομένου που έχει παραχθεί από τεχνητή νοημοσύνη και περιεχομένου που έχει συνταχθεί από άνθρωπο, περιπλέκοντας τις προσπάθειες ανίχνευσης.

Ως απάντηση σε αυτές τις προκλήσεις, η έρευνά μας διερευνά επιθέσεις με αντιπαραδείγματα που μιμούνται την παράφραση για την αξιολόγηση της ευπάθειας των ανιχνευτών. Διαπιστώνουμε ότι, ενώ οι ανιχνευτές που βασίζονται σε νευρωνικά δίκτυα μπορούν να χειραγωγηθούν με ελάχιστες διαταραχές κειμένου, τέτοιες αλλοιώσεις μπορεί να υποβαθμίσουν την ποιότητα του κειμένου, περιορίζοντας έτσι την πρακτικότητα των τακτικών αποφυγής σε πραγματικές εφαρμογές. Επιπλέον, διερευνούμε τον αντίκτυπο αυτών των επιθέσεων επίθεσης σε κατανομές περιπλοκότητας κειμένου, με στόχο να κατανοήσουμε πόσο αποτελεσματικά οι μετρικές περιπλοκότητας μπορούν να διακρίνουν μεταξύ κειμένου που έχει παραχθεί από τεχνητή νοημοσύνη και κειμένου που έχει παραχθεί από άνθρωπο σε σενάρια επίθεσης.

Συνολικά, η περιήγηση στο τοπίο της ανίχνευσης κειμένου TN και της αποφυγής παραφράσεων απαιτεί την εξισορρόπηση των τεχνικών εξελίξεων με ηθικές εκτιμήσεις και πρακτικούς περιορισμούς. Καθώς ο τομέας εξελίσσεται, ο εντοπισμός εγγενών χαρακτηριστικών που διαφοροποιούν αξιόπιστα μεταξύ κειμένου που έχει

παραχθεί από τεχνητή νοημοσύνη και κειμένου που έχει παραχθεί από άνθρωπο παραμένει ζωτικής σημασίας για την ανάπτυξη ισχυρών μεθόδων ανίχνευσης ικανών να αντέχουν στις εξελισσόμενες τακτικές των αντιπάλων.

0.1.6 Εξηγήσιμη Τεχνητή Νοημοσύνη

Η εργασία αυτή επικεντρώνεται στην κατανόηση των βαθύτερων μηχανισμών που κρύβονται πίσω από τον τρόπο με τον οποίο τα κείμενα ταξινομούνται ως παραγόμενα από τεχνητή νοημοσύνη ή ως ανθρώπινα, δίνοντας έμφαση στη σημασία της ερμηνευσιμότητας και των μεθόδων Εξηγήσιμης Τεχνητής Νοημοσύνης (Explainable AI - XAI). Η ερμηνευσιμότητα στην ΤΝ αναφέρεται στο να γίνουν οι διαδικασίες λήψης αποφάσεων των συστημάτων ΤΝ διαφανείς και κατανοητές στους ανθρώπους, παρέχοντας σαφείς, αναγνώσιμες από τον άνθρωπο εξηγήσεις για τις προβλέψεις του μοντέλου. Αυτή η διαφάνεια είναι ζωτικής σημασίας για τη βελτίωση των μοντέλων, αποκαλύπτοντας προκαταλήψεις ή αδυναμίες, ενισχύοντας την εμπιστοσύνη στα συστήματα αυτοματοποιημένης ανίχνευσης και παρέχοντας πληροφορίες για τα χαρακτηριστικά που διαφοροποιούν το ανθρώπινο κείμενο από το κείμενο που παράγεται από την ΤΝ.

Η ερμηνευσιμότητα είναι ιδιαίτερα σημαντική σε περιβάλλοντα υψηλού κινδύνου, όπως οι έλεγχοι ακαδημαϊκής ακεραιότητας, η επαλήθευση νομικών εγγράφων και οι αξιολογήσεις της αυθεντικότητας των ειδήσεων. Σε αυτά τα πλαίσια, η δυνατότητα εξήγησης του λόγου για τον οποίο ένα κείμενο ταξινομήθηκε με έναν συγκεκριμένο τρόπο είναι απαραίτητη για τη λογοδοσία και την αποδοχή από τους χρήστες. Κάνοντας τις λειτουργίες των συστημάτων ανίχνευσης τεχνητής νοημοσύνης κατανοητές, οι ενδιαφερόμενοι μπορούν να εμπιστεύονται και να επαληθεύουν τα αποτελέσματα του συστήματος, πράγμα ζωτικής σημασίας για την αποτελεσματική ανάπτυξή τους.

Οι μέθοδοι εξηγήσιμης τεχνητής νοημοσύνης μπορούν να κατηγοριοποιηθούν σε δύο κύριες προσεγγίσεις: ερμηνευσιμότητα που επιτυγχάνεται από το ίδιο το μοντέλο και ερμηνευσιμότητα που επιτυγχάνεται μέσω μεταγενέστερης ανάλυσης [58]. Η πρώτη περιλαμβάνει τον σχεδιασμό μοντέλων που είναι διαφανή και εύκολα ερμηνεύσιμα από την αρχή, όπως τα δέντρα αποφάσεων, η γραμμική παλινδρόμηση και τα συστήματα που βασίζονται σε κανόνες. Αυτά τα μοντέλα παρέχουν απλές, κατανοητές από τον άνθρωπο εξηγήσεις των διαδικασιών λήψης αποφάσεων, καθιστώντας τα εγγενώς κατανοητά.

Η μεταγενέστερη ερμηνεία (post-hoc interpretation), από την άλλη πλευρά, περιλαμβάνει την ανάλυση του μοντέλου μετά την ολοκλήρωση των διαδικασιών εκπαίδευσης και ταξινόμησης. Αυτή η προσέγγιση είναι απαραίτητη για τα μοντέλα μαύρου κουτιού, όπως είναι τα βαθιά νευρωνικά δίκτυα, τα οποία επιτυγχάνουν υψηλή ακρίβεια αλλά δεν είναι εύκολα κατανοητά. Οι post-hoc μέθοδοι περιλαμβάνουν την ανάλυση της σημασίας των χαρακτηριστικών, την οπτικοποίηση των μηχανισμών προσοχής και τη δημιουργία κειμενικών ή οπτικών εξηγήσεων της συμπεριφοράς του μοντέλου. Αυτές οι τεχνικές μπορούν επίσης να ενισχύσουν τα εγγενώς ερμηνεύσιμα μοντέλα παρέχοντας πρόσθετες γνώσεις.

Δεδομένου ότι τα μοντέλα μαύρου κουτιού προσφέρουν συνήθως την υψηλότερη ακρίβεια στην ανίχνευση κειμένων ΤΝ, η παρούσα εργασία επικεντρώνεται κυρίως στην post-hoc ερμηνεία για την επεξηγηματικότητα. Η κατανόηση και η εξήγηση των αποφάσεων αυτών των πολύπλοκων μοντέλων είναι ζωτικής σημασίας, ιδίως σε ευαίσθητες εφαρμογές όπου τα διακυβεύματα είναι υψηλά. Αξιοποιώντας τις post-hoc μεθόδους, στοχεύουμε να διασφαλίσουμε την εμπιστοσύνη στα αποτελέσματα που παράγει κάθε σύστημα και να παρέχουμε σαφείς αιτιολογήσεις για κάθε απόφαση ταξινόμησης, ενισχύοντας τόσο την αποτελεσματικότητα όσο και την αξιοπιστία των συστημάτων ανίχνευσης κειμένου ΤΝ.

Ερμηνεύσιμα Μοντέλα

Ο ευκολότερος τρόπος για να επιτευχθεί ερμηνευσιμότητα στα μοντέλα ΤΝ είναι να σχεδιαστούν με σαφείς, κατανοητούς αλγορίθμους, όπως η γραμμική ή η λογιστική παλινδρόμηση, τα δέντρα αποφάσεων και τα συστήματα κανόνων. Αυτά τα μοντέλα παρέχουν διαφανείς εξόδους που μπορούν εύκολα να αναχθούν σε χαρακτηριστικά εισόδου. Για την ανίχνευση κειμένων ΤΝ, οι μέθοδοι βασισμένες σε χαρακτηριστικά, οι οποίες αξιοποιούν κατανοητές από τον άνθρωπο ιδιότητες, όπως η λεξιλογική ποικιλομορφία και τα συντακτικά πρότυπα, είναι εγγενώς επεξηγήσιμες. Η ανάλυση με βάση την περιπλοκότητα, που χρησιμοποιείται σε ορισμένους ανιχνευτές κειμένου τεχνητής νοημοσύνης, προσφέρει μια μέση λύση μεταξύ ερμηνευσιμότητας και πολυπλοκότητας. Ενώ η αμηχανία παρέχει μια σαφή αριθμητική ένδειξη της αβεβαιότητας ενός μοντέλου, απαιτεί εγγενή γνώση του γλωσσικού μοντέλου που χρησιμοποιείται, κάτι που την καθιστά μη πλήρως εξηγήσιμη.

Ωστόσο, όπως προαναφέρθηκε στην ενότητα 0.1.3, οι μέθοδοι που βασίζονται σε χαρακτηριστικά, παρά τη διαφάνειά τους, συχνά υστερούν σε ακρίβεια σε σύγκριση με τα προηγμένα μοντέλα μηχανικής μάθησης και είναι ευάλωτες σε χειραγώγηση, όπως οι επιθέσεις παράφρασης. Τα εγγενώς ερμηνεύσιμα μοντέλα μπορούν εύκολα να εξαπατηθούν από χρήστες που κατανοούν τη λειτουργία τους. Έτσι, ενώ προσφέρουν διαφάνεια, οι περιορισμοί τους καθιστούν αναγκαία τη χρήση τεχνικών post-hoc ερμηνευσιμότητας για πιο ισχυρή και ακριβή ανίχνευση κειμένου TN, διασφαλίζοντας ότι ακόμη και οι αποφάσεις των πολύπλοκων μοντέλων μπορούν να γίνουν κατανοητές και αξιόπιστες.

Post-hoc ερμηνευσιμότητα

Οι μέθοδοι post-hoc ερμηνείας προσφέρουν σημαντικά πλεονεκτήματα, καθώς βασίζονται αποκλειστικά στην είσοδο και την έξοδο ενός μοντέλου και όχι στις εσωτερικές λειτουργίες του. Αυτό επιτρέπει τη χρήση των πιο ακριβών διαθέσιμων μοντέλων χωρίς να περιορίζονται από την ανάγκη εγγενούς ερμηνευσιμότητας. Καθώς η πολυπλοκότητα των μοντέλων μηχανικής μάθησης αυξάνεται, οι μέθοδοι που είναι ανεξάρτητες από το μοντέλο γίνονται όλο και πιο πρακτικές. Οι post-hoc μέθοδοι μπορούν να χωριστούν σε καθολικές και τοπικές εξηγήσεις: οι καθολικές μέθοδοι παρέχουν μια συνολική κατανόηση του τρόπου με τον οποίο τα χαρακτηριστικά επηρεάζουν τις προβλέψεις ενός μοντέλου κατά μέσο όρο, ενώ οι τοπικές μέθοδοι επικεντρώνονται στην εξήγηση συγκεκριμένων μεμονωμένων προβλέψεων. Παραδείγματα καθολικών μεθόδων περιλαμβάνουν τα υποκατάστατα μοντέλα και το SHAP για την παγκόσμια σημασία των χαρακτηριστικών, ενώ το LIME[68] είναι μια δημοφιλής τοπική μέθοδος που παράγει ερμηνεύσιμες εξηγήσεις προσεγγίζοντας τη συμπεριφορά του μοντέλου μαύρου κουτιού γύρω από ένα συγκεκριμένο σημείο δεδομένων.

Στην ανίχνευση κειμένων τεχνητής νοημοσύνης, οι μέθοδοι μεταγενέστερης ερμηνευσιμότητας είναι ιδιαίτερα χρήσιμες για την κατανόηση των αποφάσεων του μοντέλου. Για παράδειγμα, η ανάλυση της σημασίας των χαρακτηριστικών έχει δείξει ότι η περιπλοκότητα είναι ένας βασικός παράγοντας στην ταξινόμηση κειμένων TN, αν και δεν είναι εγγενώς ερμηνεύσιμη. Για την ενίσχυση της διαφάνειας, μέθοδοι όπως το LIME μπορούν να δημιουργήσουν γραφήματα σπουδαιότητας λέξεων, επισημαίνοντας ποιες λέξεις επηρέασαν περισσότερο την πρόβλεψη του ταξινομητή. Αυτή η προσέγγιση, την οποία χρησιμοποιούμε στην έρευνά μας, παρέχει σαφέστερη εικόνα των συγκεκριμένων χαρακτηριστικών του κειμένου που καθοδηγούν τις αποφάσεις του μοντέλου, καθιστώντας τα πολύπλοκα μοντέλα πιο κατανοητά και αξιόπιστα σε πρακτικές εφαρμογές.

Ερμηνευσιμότητα με αντιφατικές εξηγήσεις

Οι αντιφατικές εξηγήσεις (counterfactual explanations) είναι μια μεταγενέστερη μέθοδος post-hoc ερμηνευσιμότητας που επικεντρώνεται στο ποιες αλλαγές στην είσοδο θα άλλαζαν την πρόβλεψη ενός μοντέλου. Αυτές οι εξηγήσεις είναι ανεξάρτητες από το μοντέλο και ιδιαίτερα φιλικές προς τον άνθρωπο λόγω της αντιθετικής τους φύσης, καθιστώντας τις εφαρμόσιμες σε ένα ευρύ φάσμα συστημάτων, ακόμη και σε εκείνα που δεν χρησιμοποιούν μηχανική μάθηση. Στο πλαίσιο της ανίχνευσης κειμένου TN, οι αντιφατικές εξηγήσεις μπορούν να βοηθήσουν τους χρήστες να κατανοήσουν τι πρέπει να αλλάξει για να ταξινομηθεί διαφορετικά ένα κείμενο, παρέχοντας μια σαφή και διαισθητική κατανόηση της διαδικασίας λήψης αποφάσεων του μοντέλου.

Σε πρακτικές εφαρμογές, η κοινότητα NLP έχει αναπτύξει πλαίσια για την υλοποίηση αντιφατικών εξηγήσεων με τη χρήση γλωσσικών μοντέλων παραγωγής κειμένου. Έχουν προταθεί τεχνικές όπως η απόκρυψη τμημάτων του κειμένου και η βελτιστοποίηση των αντικαταστάσεων για την αλλαγή της εξόδου, όπως φαίνεται σε έργα όπως [69] και [6]. Επιπλέον, γενικές προσεγγίσεις όπως το Polyjuice [89] δημιουργούν διαταραχές που αλλάζουν τη σημασιολογία μιας πρότασης χωρίς να στοχεύουν σε συγκεκριμένο προγνωστικό παράγοντα, προσφέροντας ευελιξία στη δημιουργία αντιφατικών εξηγήσεων.

Οι αντιφατικές εξηγήσεις εξυπηρετούν διάφορους σκοπούς, συμπεριλαμβανομένης της εκπλήρωσης του αποτελέσματος, της διερεύνησης του συστήματος και της ανίχνευσης ευπαθειών. [50] Η ανίχνευση τρωτών σημείων, που συνδέεται στενά με τις αντιθετικές επιθέσεις (adversarial attacks), εντοπίζει πιθανές αδυναμίες σε ένα σύστημα. Οι αντιθετικές επιθέσεις, όπως αυτές που δημιουργούνται από το TextFooler [33], αποσκοπούν στη δημιουργία διαταραχών που εκθέτουν ζητήματα ευρωστίας στα μοντέλα. Αυτές οι επιθέσεις δεν παράγουν απαραίτητα ελάχιστες ή ρευστές αλλαγές, και μπορεί να εισάγουν θόρυβο, σε αντίθεση με τις αντιφατικές εξηγήσεις, οι οποίες επιδιώκουν ελάχιστες και ουσιαστικές αλλαγές.

Στην παρούσα μελέτη, χρησιμοποιούνται αντιθετικές επιθέσεις μέσω του πλαισίου TextFooler για τη διερεύνηση διαφορετικών περιπτώσεων χρήσης των αντιφατικών εξηγήσεων. Στόχος είναι να αναδειχθούν ζητήματα ευρ-

ωστίας, να αποκτηθούν βαθύτερες γνώσεις σχετικά με τις προβλέψεις του ανιχνευτή κειμένου της ΤΝ και να δοθούν εξηγήσεις στους ανθρώπους. Για να αξιολογηθεί η αποτελεσματικότητα αυτών των εξηγήσεων, διεξάγεται μια έρευνα χρηστών όπου οι συμμετέχοντες εκτελούν ανίχνευση κειμένου ΑΙ τόσο χωρίς βοήθεια όσο και με τη βοήθεια αντιφατικών εξηγήσεων και εξηγήσεων LIME. Η μελέτη αυτή αποσκοπεί να καθορίσει κατά πόσον αυτές οι μέθοδοι ερμηνευσιμότητας μπορούν να ενισχύσουν τις ανθρώπινες επιδόσεις στην ανίχνευση κειμένου που παράγεται από ΤΝ, προσφέροντας πολύτιμες πληροφορίες για τη βελτίωση των συστημάτων ανίχνευσης κειμένου ΤΝ.

Σύνοψη

Η πρόσφατη βιβλιογραφία έχει διερευνήσει το συνδυασμό διαφορετικών μεθόδων ερμηνευσιμότητας για να βελτιώσει τις γνώσεις μας γύρω από τα μοντέλα μηχανικής μάθησης μαύρου κουτιού. Ένα αξιοσημείωτο παράδειγμα είναι η εργασία [4], η οποία ενσωματώνει την τοπική σημασία των χαρακτηριστικών, όπως παρατηρείται στο LIME, με τις αντιφατικές εξηγήσεις. Αυτή η υβριδική προσέγγιση αξιολογεί τη σημαντικότητα της αλλαγής συμβολισμού μεταξύ μιας περίπτωσης και του αντιπαραδειγματικού δείγματος, προσφέροντας μια διαφοροποιημένη κατανόηση της συμβολής των μεμονωμένων χαρακτηριστικών στις αποφάσεις του μοντέλου. Αυτός ο συνδυασμός γεφυρώνει το χάσμα μεταξύ τοπικής και καθολικής ερμηνευσιμότητας, παρέχοντας πιο ολοκληρωμένες και φιλικές προς τον χρήστη εξηγήσεις. Συνοψίζοντας, η κατανόηση των μηχανισμών πίσω από την ταξινόμηση κειμένου που δημιουργείται από ΤΝ ή από τον άνθρωπο είναι ζωτικής σημασίας για τη βελτίωση της διαφάνειας και της αξιοπιστίας των συστημάτων τεχνητής νοημοσύνης. Οι post-hoc μέθοδοι, όπως το LIME και οι αντιφατικές εξηγήσεις, βελτιώνουν σημαντικά την ερμηνευσιμότητα του μοντέλου, εξηγώντας τις επιμέρους προβλέψεις και προτείνοντας ελάχιστες αλλαγές εισόδου για την αλλαγή των αποτελεσμάτων. Αυτό όχι μόνο βοηθά στην κατανόηση και τη βελτίωση των επιδόσεων του μοντέλου, αλλά και ενισχύει την εμπιστοσύνη των χρηστών καθιστώντας τις αποφάσεις της ΤΝ πιο διαφανείς. Με την ενσωμάτωση αυτών των μεθόδων, η εργασία μας αποσκοπεί στη βελτίωση των δυνατοτήτων ανίχνευσης και της ερμηνευσιμότητας των ανιχνευτών κειμένου ΤΝ, με μια έρευνα χρηστών στην ενότητα 0.3 (ή πιο αναλυτικά στο Κεφάλαιο 5) να διερευνά την αποτελεσματικότητά τους στην υποβοήθηση της ανθρώπινης απόδοσης στην ανίχνευση κειμένου που παράγεται από ΤΝ.

0.2 Πειραματικό Μέρος

0.2.1 Επισκόπηση των πειραμάτων

Αρχικά, θεωρούμε σημαντικό να δώσουμε ένα περίγραμμα των πειραμάτων που διεξάγονται, ώστε να αποσαφηνιστεί γύρω από ποιον άξονα εκτελούμε τα πειράματα και πώς συνδυάζουμε τις μεθοδολογίες που συζητήθηκαν στην προηγούμενη ενότητα .

Διαχωρίζουμε τα πειράματα που πραγματοποιήσαμε σε 2 κατηγορίες: στην εξερεύνηση των ανιχνευτών κειμένου ΤΝ με αντιθετικές επιθέσεις και στην ανάλυση με βάση την περιπλοκότητα κειμένου. Στην πρώτη κατηγορία, χρησιμοποιούμε ως κύριο εργαλείο το πλαίσιο TextFooler [33], το οποίο βασίζεται στην αλλαγή διαφόρων λέξεων σε μια πρόταση με συνώνυμα, έως ότου η πρόβλεψη ενός μοντέλου αλλάξει. Το εργαλείο αυτό έχει δοκιμαστεί σε πολλές εργασίες στον τομέα της ΤΝ, όπως η αλλαγή του συναισθηματικού πλαισίου κάποιου κειμένου ή η μετατροπή μιας θετικής κριτικής σε αρνητική. Εμείς αξιοποιούμε το σύστημα αυτό για να παραπλανήσουμε ανιχνευτές κειμένου ΤΝ, αλλάζοντας την πρόβλεψή τους από κείμενο ΤΝ σε ανθρώπινο και αντίστροφα.

Οι βασικοί ανιχνευτές κειμένου ΤΝ που χρησιμοποιούμε είναι το RADAR[27] και το RoBERTa, ενώ εκπαιδεύουμε και μικρότερα μοντέλα DistilBERT για να προσομοιώσουμε τα μοντέλα λεπτομερούς ρύθμισης. Επίσης, χρησιμοποιούμε 6 βασικά σύνολα δεδομένων από τη βιβλιογραφία, τα οποία αντιπροσωπεύουν δείγματα από διαφορετικούς τομείς γραπτού λόγου έτσι ώστε να έχουμε μια πληρέστερη εικόνα της απόδοσης αλλά και της ευρωστίας των ανιχνευτών κειμένου ΤΝ. Περισσότερες πληροφορίες για τα σύνολα δεδομένων υπάρχουν στην ενότητα 3.3.

Στη δεύτερη κατηγορία πειραμάτων χρησιμοποιούμε την ανάλυση περιπλοκότητας για να διερευνήσουμε την υπόθεση ότι η περιπλοκότητα κειμένου (text perplexity) είναι ενδεικτική των εγγενών διαφορών μεταξύ ανθρώπινου και μηχανικά παραγόμενου κειμένου, βοηθώντας στην ακριβή ταξινόμηση. Στη βάση αυτή σχεδιάζουμε έναν αλγόριθμο ολισθαίνοντος παραθύρου προσεγγίζοντας την πραγματική αυτοπαλινδρομική αποσύνθεση των πιθανοτήτων ακολουθίας. Τροποποιούμε αυτόν τον αλγόριθμο σε ανιχνευτή κειμένου ΤΝ προσδιορίζοντας ένα

βέλτιστο κατώφλι για τις βαθμολογίες ακαταλληλότητας χρησιμοποιώντας δεδομένα εκπαίδευσης, επιτρέποντάς μας να ταξινομήσουμε κείμενα ως παραγόμενα από TN ή ανθρώπινα.

Για να επικυρώσουμε περαιτέρω τη μέθοδό μας, τη συγκρίνουμε με πιο προηγμένες τεχνικές, όπως η σύγκριση της περιπλοκότητας με συμπληρώσεις κειμένου από LLMs, όπως μελετήθηκε στο [25]. Με τον προσδιορισμό των βέλτιστων κατωφλίων ειδικά για τους τομείς κειμένου, υπογραμμίζουμε τη σημασία των συμφραζομένων στην ταξινόμηση με βάση την περιπλοκότητα και προσφέρουμε μια θεμελιώδη μέθοδο για την αξιολόγηση ανιχνευτών κειμένου TN. Αυτή η βασική γραμμή μπορεί να βοηθήσει τους ερευνητές στην αξιολόγηση της ευρωστίας και της αποτελεσματικότητας των ανιχνευτών τους έναντι διαφόρων συνόλων δεδομένων.

0.2.2 Ποσοτικά αποτελέσματα της ανάλυσης ανιχνευτών

Αρχικά αξιολογούμε την απόδοση των διάφορων ανιχνευτών στα σύνολα δεδομένων μας, με τα αποτελέσματα να φαίνονται στον Πίνακα 1.

Table 1: Συγκεντρωτικά αποτελέσματα απόδοσης ανιχνευτών ανά μοντέλο και σύνολο δεδομένων

Model/Dataset	**1**	**2**	**3**	**4**	**5**	**6**
RADAR	56.5%	95%	77%	71.5%	84%	63%
RoBERTa	70%	83.25%	79%	46.5%	78%	51%
DB-1	92.5%	50%	72%	57.5%	61%	71%
DB-2	54%	96.5%	48%	74%	50%	38.5&
DB-3	58.5%	96%	100%	70%	58%	40%

Ολοκληρώνοντας την φάση αυτή του πειράματος, παρατηρούμε ότι ενώ οι τελευταίας τεχνολογίας ανιχνευτές μπορούν να διακρίνουν αποτελεσματικά μεταξύ των περισσότερων κειμένων που παράγονται από τεχνητή νοημοσύνη και των ανθρώπινων κειμένων, υπάρχουν διαφορετικά επίπεδα επιτυχίας ανάλογα με τον τομέα, το σύνολο δεδομένων, το μοντέλο και την τεχνική που χρησιμοποιείται. Τα μοντέλα που έχουν προ-εκπαιδευτεί σε έναν συγκεκριμένο τομέα ή σύνολο δεδομένων επιτυγχάνουν μεγαλύτερη ακρίβεια στον συγκεκριμένο τομέα, αλλά η ακρίβεια τους δεν γενικεύεται εκτός αυτού. Εν τω μεταξύ, μεγαλύτερα μοντέλα που έχουν ρυθμιστεί λεπτομερώς για την ανίχνευση κειμένων TN γενικά αλλά όχι για ένα συγκεκριμένο σύνολο δεδομένων επιτυγχάνουν καλύτερες επιδόσεις στη γενική περίπτωση αλλά στις εξειδικευμένες περιπτώσεις δεν είναι πολύ κοντά σε έναν τέλειο ταξινομητή. Ανεξάρτητα από το σύνολο δεδομένων και το μοντέλο που χρησιμοποιείται, ένας χρήστης που επιθυμεί να χρησιμοποιήσει οποιονδήποτε από αυτούς τους ανιχνευτές σε ένα ρεαλιστικό σενάριο πρέπει να προχωρήσει πολύ προσεκτικά, καθώς φαίνεται ότι εξακολουθούν να υπάρχουν σημαντικοί περιορισμοί που παρεμποδίζουν την αποτελεσματικότητά τους, όπως τα μικρά κείμενα (ειδικά για το RADAR), τα κείμενα από μη φυσικούς ομιλητές της αγγλικής γλώσσας ή τα κείμενα από διαφορετικά LLM και τομείς (όπως παρουσιάζεται στο σύνολο δεδομένων MAGE στο οποίο οι ανιχνευτές γενικής χρήσης δυσκολεύονται περισσότερο).

Στη συνέχεια, πραγματοποιούμε αντιθετική επίθεση στους ανιχνευτές μέσω του πλαισίου TextFooler. Η δραματική αλλαγή στα ποσοστά επιτυχίας των ανιχνευτών παρουσιάζεται στον πίνακα 2.

Table 2: Απόδοση ανιχνευτών μετά τις επιθέσεις TextFooler

Model/Dataset	**1**	**2**	**3**	**4**	**5**	**6**
RADAR	18%	**0%**	**0%**	**0%**	**0%**	5.5%
RoBERTa	**1%**	**0%**	**2%**	2%	**7.5%**	2%
DB-1	**20%**	0%	**3%**	0%	0%	0%
DB-2	16%	**22%**	29%	**1.5%**	14%	9.5%
DB-3	44%	**29%**	**38%**	**11.5%**	44%	22%

Όπως προκύπτει από αυτά τα αποτελέσματα, το TextFooler καταφέρνει να αντιστρέψει την πρόβλεψη των ανιχνευτών στη συντριπτική πλειοψηφία των πειραμάτων στα οποία ο προγνωστικός μηχανισμός είχε αρχικά καλή ακρίβεια. Από αυτές τις περιπτώσεις, τα μικρότερα προ-εκπαιδευμένα μοντέλα (DB-1,DB-2,DB-3) φαίνεται να είναι ελαφρώς πιο ανθεκτικά στις διαταραχές από τα μοντέλα γενικής χρήσης, φαινόμενο το οποίο μπορεί να αποδίδεται στο ότι εκπαιδεύονται σε ένα μικρό σώμα παρόμοιων δεδομένων και επομένως είναι πιο επιρρεπή σε

υπερπροσαρμογή(ovefitting). Ενώ η υπερπροσαρμογή είναι γενικά ανεπιθύμητη, θα μπορούσε να σημαίνει ότι τα πρότυπα που μαθαίνουν αυτά τα μοντέλα είναι πολύ συγκεκριμένα και, επομένως, οι αντιθετικές επιθέσεις που βασίζονται απλώς σε αλλαγές λέξη προς λέξη μπορεί να είναι λιγότερο αποτελεσματικές εναντίον τους. Ωστόσο, όπως φαίνεται, ακόμη και στη χειρότερη περίπτωση το TextFooler καταφέρνει να ρίξει την απόδοση όλων των ανιχνευτών κάτω από το 50%, πράγμα που σημαίνει ότι με απλές επιθέσεις αντικατάστασης λέξεων οι ανιχνευτές αποδίδουν χειρότερα ακόμη και από την τυχαία επιλογή.

Επιπλέον, παρατηρούμε ότι το TextFooler αλλάζει λιγότερο από το 10% των λέξεων ενός κειμένου κατά μέσο όρο για να παραπλανήσει επιτυχώς έναν ανιχνευτή. Τα προ-εκπαιδευμένα μικρά μοντέλα τείνουν να έχουν υψηλότερο μέσο ποσοστό αλλαγμένων λέξεων και ειδικά στα σύνολα δεδομένων στα οποία έχουν προ-εκπαιδευτεί, γεγονός που συσχετίζεται με τη γενική διαπίστωση ότι είναι πιο ανθεκτικά σε αντιθετικές επιθέσεις από τους ανιχνευτές γενικής χρήσης. Μεταξύ των ανιχνευτών γενικού σκοπού , το RADAR φαίνεται να απαιτεί μεγαλύτερο ποσοστό λέξεων που αλλάζουν και μεγαλύτερο αριθμό ερωτημάτων προς εκτέλεση(queries) από το RoBERTa, γεγονός που υποδηλώνει ότι είναι πιο ανθεκτικός ανιχνευτής. Αυτό μπορεί να αποδοθεί στην αντιθετική εκπαίδευσή του, η οποία πιθανώς αύξησε την ικανότητά του να μην παραπλανάται από μικρές διαταραχές του κειμένου.

Ολοκληρώνοντας την πρώτη φάση του πειράματος, είναι προφανές ότι οι τελευταίας τεχνολογίας ανιχνευτές κειμένου τεχνητής νοημοσύνης αντιμετωπίζουν ένα σημαντικό εμπόδιο σε ότι αφορά την ευρωστία σε αντιθετικές επιθέσεις, το οποίο συμφωνεί με τα ευρήματα πολλών πρόσφατων εργασιών όπως η [28]. Δείχνουμε ότι απλές επιθέσεις αντικατάστασης λέξεων μπορούν να εξαπατήσουν ανιχνευτές που αποδίδουν με ακρίβεια πάνω από 70% και μέχρι 100%, προκειμένου να μειώσουν την ακρίβειά τους σε επίπεδα μικρότερα από την τυχαία επιλογή. Επιπλέον, διερευνούμε τις διαφορές μεταξύ προ-εκπαιδευμένων μικρότερων ανιχνευτών και μεγαλύτερων ανιχνευτών γενικής χρήσης και διαπιστώνουμε ότι οι μικρότεροι ανιχνευτές είναι ελαφρώς πιο ανθεκτικοί στις αντιθετικές διαταραχές. Αυτό πιθανώς οφείλεται στην υπερβολική προσαρμογή στον μικρό όγκο δεδομένων εκπαίδευσης, η οποία καθιστά τους ανιχνευτές πιο σίγουρους για τις προβλέψεις τους και, επομένως, πιο δύσκολο να παραπλανηθούν με απλές αντικαταστάσεις λέξεων. Ωστόσο, όπως σημειώσαμε προηγουμένως, αυτό δεν σημαίνει ότι τα μικρότερα προ-εκπαιδευμένα μοντέλα είναι καλύτεροι ανιχνευτές, καθώς συχνά έχουν πολύ κακές επιδόσεις εκτός του πεδίου εκπαίδευσής τους.

0.2.3 Αποτελέσματα της ανάλυσης περιπλοκότητας κειμένου

Στη δεύτερη σειρά πειραμάτων διερευνούμε διάφορα χαρακτηριστικά με επίκεντρο την περιπλοκότητα κειμένου, όπως εξηγείται στην ενότητα 0.1.4. Πρώτον, δοκιμάζουμε την υπόθεση ότι τα κείμενα που ταξινομούνται ως παραγόμενα από TN περιέχουν συνήθως πιο προβλέψιμες λέξεις που οδηγούν σε σημαντικά χαμηλότερη περιπλοκότητα. Ως εκ τούτου, χρησιμοποιούμε το μοντέλο GPT-2 για να μετρήσουμε την περιπλοκότητα του κειμένου στα δείγματα από όλα τα σύνολα δεδομένων μας. Τα συγκεντρωτικά αποτελέσματα παρουσιάζονται στον πίνακα 3 παρακάτω.

Table 3: Μέση περιπλοκότητα κειμένων TN και ανθρώπου ανά σύνολο δεδομένων

Dataset	**PPL of Generated Texts**	**PPL of Human Texts**
Autextification	49.6	171.86
Human vs. ChatGPT	15.54	75.82
GPT Classification	11.12	40.03
ChatGPT Detector Bias	25.9	35.67
HC3	10	57.9
MAGE	31.36	43.98
Average	**23.92**	**70.88**

Όπως προκύπτει από τον παραπάνω πίνακα, η μέση περιπλοκότητα κειμένου είναι πράγματι σημαντικά υψηλότερη για τα ανθρώπινα κείμενα από ό,τι για τα κείμενα που παράγονται από τεχνητή νοημοσύνη σε όλα τα σύνολα δεδομένων. Για να κατανοήσουμε περαιτέρω τις κατανομές της περιπλοκότητας κειμένου σε κάθε δείγμα συνόλου δεδομένων, απεικονίζουμε τις περιπλοκότητες σε μορφή γραφήματος. Τα αποτελέσματα παρουσιάζονται στα Σχήματα 0.2.1-0.2.6.

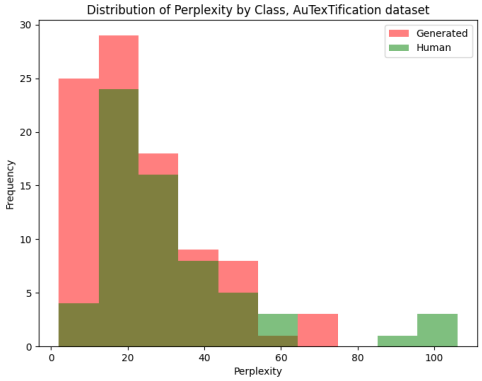

Figure 0.2.1: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων AuTexTification

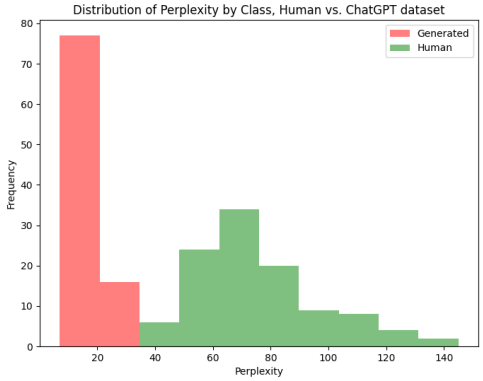

Figure 0.2.2: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων Human vs. ChatGPT

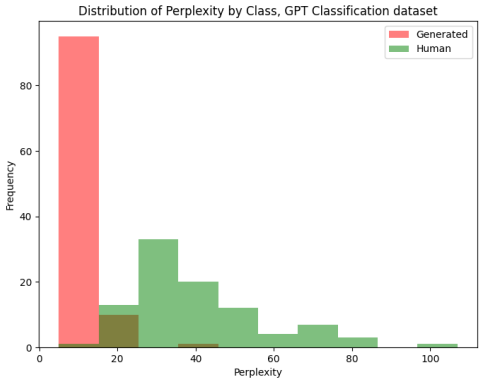

Figure 0.2.3: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων GPT Classification

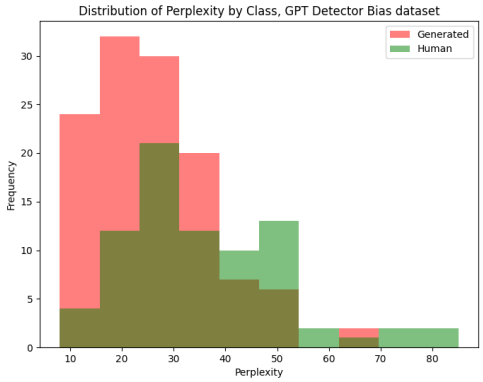

Figure 0.2.4: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων ChatGPT Detector Bias

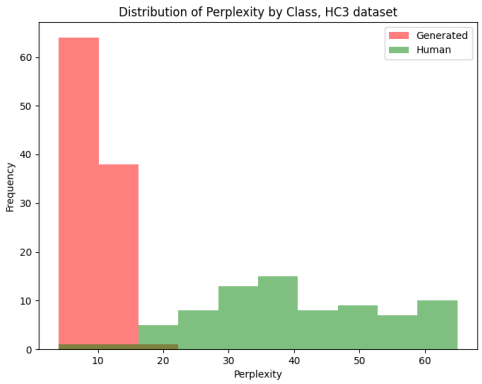

Figure 0.2.5: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων HC3

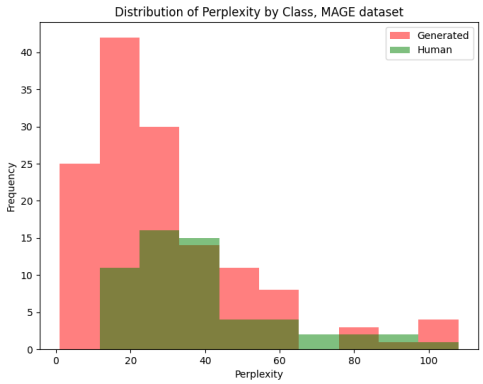

Figure 0.2.6: Κατανομή περιπλοκότητας κειμένου στο σύνολο δεδομένων MAGE

Εξετάζοντας πιο προσεκτικά αυτά τα σχήματα, μπορούμε να δούμε ότι παρόλο που τα κείμενα που δημιουργούνται από την TN (σημειωμένα με κόκκινο χρώμα στα διαγράμματα) είναι γενικά περισσότερο προς την αριστερή πλευρά (χαμηλότερη περιπλοκότητα) από τα ανθρώπινα κείμενα (σημειωμένα με πράσινο χρώμα), κάθε σύνολο δεδομένων έχει τη δική του κατανομή περιπλοκότητας που παρουσιάζει ορισμένα ενδιαφέροντα χαρακτηριστικά. Πιο συγκεκριμένα:

- Τα σύνολα δεδομένων 2, 3 και 5 (Human vs. ChatGPT , Gpt Classification και HC3) παρουσιάζουν πολύ μικρή επικάλυψη μεταξύ της περιπλοκότητας των κειμένων που παράγονται από τεχνητή νοημοσύνη και των ανθρώπινων κειμένων. Επομένως, αναμένουμε ότι ένας ανιχνευτής με βάση την περιπλοκότητα θα είναι σε θέση να διακρίνει πολύ αποτελεσματικά μεταξύ των κλάσεων, εάν επιλεγεί το σωστό κατώφλι.
- Στα σύνολα δεδομένων AuTexTification και MAGE υπάρχει σημαντική επικάλυψη μεταξύ της περιπλοκότητας των κειμένων των δύο κλάσεων. Αν και γενικά ισχύει ότι όσο πιο αριστερά κινούμαστε στα διαγράμματα (χαμηλότερη περιπλοκότητα) τόσο πιο κυρίαρχη γίνεται η κλάση AI-Generated (κόκκινη), αναμένουμε ότι θα είναι πολύ δύσκολο για έναν ανιχνευτή που βασίζεται στην περιπλοκότητα να είναι σε θέση να διακρίνει μεταξύ των κλάσεων σε αυτά τα σύνολα δεδομένων.
- Στο σύνολο δεδομένων ChatGPT Detector Bias, υπάρχει κάποια επικάλυψη μεταξύ των περιπλοκοτήτων κειμένου των δύο κλάσεων, αλλά λιγότερο από τα σύνολα δεδομένων 1 και 6. Επίσης, γίνεται εμφανές ότι τα ανθρώπινα κείμενα σε αυτό το σύνολο δεδομένων ωθούνται περισσότερο προς την αριστερή πλευρά, κάτι που είναι αναμενόμενο, δεδομένου ότι αυτό το σύνολο δεδομένων περιλαμβάνει πολλά κείμενα που προέρχονται από μη φυσικούς ομιλητές της Αγγλικής γλώσσας, τα κείμενα των παρουσιάζουν χαμηλότερη περιπλοκότητα από την κανονική.

0.2.4 Ανιχνευτής με βάση την περιπλοκότητα κειμένου

Το επόμενο βήμα είναι να κατασκευάσουμε έναν ανιχνευτή με βάση την περιπλοκότητα, ώστε να μπορέσουμε να έχουμε ένα σημείο αναφοράς που θα καθορίζει πόσο καλά μπορεί η περιπλοκότητα κειμένου να διακρίνει μεταξύ κειμένων που δημιουργούνται από τον άνθρωπο και κειμένων που δημιουργούνται από την TN. Ξεκινάμε με το αφελές πείραμα της αυθαίρετης επιλογής μιας τιμής ως κατώφλι για τον ανιχνευτή μας, ο οποίος θα ταξινομήσει κάθε κείμενο με αμηχανία κάτω από αυτό το κατώφλι ως παραγόμενο από TN και κάθε κείμενο με αμηχανία πάνω από αυτό το κατώφλι ως ανθρώπινο. Στη συνέχεια, συγκρίναμε τον αφελή ανιχνευτή μας με τα LLM γενικής χρήσης που χρησιμοποιήσαμε στην ενότητα 0.2.1. Τα αποτελέσματα αυτού του πειράματος μπορούν να παρατηρηθούν στον Πίνακα 4 παρακάτω, όπου με έντονη γραφή υποδεικνύεται ο ανιχνευτής που πέτυχε την υψηλότερη ακρίβεια.

Table 4: Απόδοση αφελών ανιχνευτών περιπλοκότητας συγκρινόμενη με τους ανιχνευτές LLM γενικής χρήσης

Model/Dataset	AuTex	Human vs GPT	GPT-Class	GPT-DB3	HC3	MAGE
RADAR-Vicuna-7B	56.5%	96.06%	77%	**71.5%**	80.5%	**63%**
ROBERTA-large	**69%**	83.2%	81%	46.5%	78%	57.5%
Naive Perplexity (t=20)	57%	88.6%	**96.5%**	51%	**96%**	53.5%
Naive Perplexity (t=40)	58%	**99.26%**	71.5%	67%	78%	61.5%

Η ανάλυση των δεδομέναν αυτών δείχνει ότι ακόμα και οι αφελείς ανιχνευτές που βασίζονται στην περιπλοκότητα υπερτερούν των ανιχνευτών LLM γενικής χρήσης σε σύνολα δεδομένων όπου υπάρχει σαφής διάκριση μεταξύ των κατανομών περιπλοκότητας που δημιουργούνται από τον άνθρωπο και την TN. Σε πιο απαιτητικά σύνολα δεδομένων με επικαλυπτόμενες κατανομές περιπλοκότητας, οι ανιχνευτές LLM παρουσιάζουν ελαφρώς καλύτερες επιδόσεις, αν και δεν ξεπερνούν την ακρίβεια του 75%. Αυτό υποστηρίζει την υπόθεσή μας ότι η περιπλοκότητα κειμένου είναι ένα σημαντικό χαρακτηριστικό για τη διάκριση κειμένου που έχει παραχθεί από TN από ανθρώπινο κείμενο. Ωστόσο, η διακύμανση της ακρίβειας των ανιχνευτών σε διαφορετικά σύνολα δεδομένων υποδηλώνει ότι οι αφελείς ανιχνευτές περιπλοκότητας είναι ακατάλληλοι για πρακτική χρήση χωρίς περαιτέρω προσαρμογή. Διαφορετικοί τομείς κειμένου έχουν διαφορετικά χαρακτηριστικά περιπλοκότητας, γεγονός που απαιτεί συγκεκριμένα όρια για ακριβή ταξινόμηση. Κατά συνέπεια, το επόμενο βήμα μας είναι να αναπτύξουμε έναν ανιχνευτή με βάση την περιπλοκότητα που καθορίζει το βέλτιστο κατώφλι με βάση το σύνολο δεδομένων, όπως η μέθοδος που χρησιμοποιείται στο HowkGPT[85] στο περιβάλλον των ακαδημαϊκών εργασιών.

Η μέθοδος που χρησιμοποιούμε για να προσεγγίσουμε αυτό το πρόβλημα είναι να "εκπαιδεύσουμε" τον ταξινομητή περιπλοκότητας με τρόπο παρόμοιο με τον τρόπο που εκπαιδεύονται τα μοντέλα τεχνητής νοημοσύνης, προκειμένου να βρούμε το ιδανικό κατώφλι για κάθε σύνολο δεδομένων. Πιο συγκεκριμένα, χωρίζουμε κάθε δείγμα συνόλου δεδομένων σε σύνολα εκπαίδευσης και δοκιμών, όπου το 80% του συνόλου γίνεται το σύνολο εκπαίδευσης και το υπόλοιπο 20% γίνεται το σύνολο δοκιμών. Στη συνέχεια, υπολογίζουμε την περιπλοκότητα για κάθε κείμενο στο σύνολο εκπαίδευσης και επαναλαμβάνουμε ένα εύρος πιθανών κατωφλίων, κρατώντας το καλύτερο κατώφλι (που παρέχει την καλύτερη ακρίβεια στο σύνολο εκπαίδευσης). Τέλος, χρησιμοποιούμε το βέλτιστο κατώφλι που βρέθηκε στο σύνολο εκπαίδευσης για να αξιολογήσουμε την ακρίβεια της μεθόδου μας στο σύνολο δοκιμής, το οποίο αποτελεί καινούρια(unseen) δεδομένα για τον ανιχνευτή.

Τα αποτελέσματα αυτής της προσέγγισης σε κάθε ένα από τα 6 σύνολα δεδομένων μας παρουσιάζονται στον Πίνακα 5.

Table 5: Μέση περιπλοκότητα κειμένων TN και ανθρώπινων ανά σύνολο δεδομένων

Dataset	**Optimal threshold**	**Training accuracy**	**Test accuracy**
AuTexTification	170	56.88%	57.5%
Human vs. ChatGPT	39.12	99.07%	100%
GPT Classification	21.48	98.12%	97.5%
ChatGPT Detector Bias	35.22	65.62	80%
HC3	16.32	98.12%	100%
MAGE	65.84	71.25%	65%

Όπως προκύπτει από τα αποτελέσματα, η ανάλυση περιπλοκότητας οδηγεί στην καλύτερη ή πολύ κοντά στην καλύτερη ακρίβεια σε κάθε σύνολο δεδομένων εκτός από το σύνολο δεδομένων 1 (AuTexTification) και το σύνολο δεδομένων 6 (MAGE). Λόγω του μικρού μήκους των κειμένων στο σύνολο δεδομένων AuTexTification και της εγγενώς περιορισμένης ποσότητας συμφραζομένων, η περιπλοκότητα δεν αποτελεί ιδανικό μέτρο για τον προσδιορισμό του κατά πόσον ένα κείμενο έχει παραχθεί από τεχνητή νοημοσύνη ή από άνθρωπο σε αυτόν τον συγκεκριμένο τομέα και, επομένως, είναι πολύ προτιμότεροι οι ειδικά ρυθμισμένοι ανιχνευτές για μικρά κείμενα, όπως μπορεί να φανεί από την υψηλή ακρίβεια που επιτυγχάνεται από το DB-1. Από την άλλη πλευρά, κανένας ανιχνευτής από αυτούς που δοκιμάστηκαν δεν επιτυγχάνει αρκετά αξιοπρεπή ακρίβεια στο σύνολο δεδομένων MAGE, καθώς το καλύτερο αποτέλεσμα του 71% ισοδυναμεί σχεδόν με την πλειοψηφική κλάση, δεδομένου ότι περίπου το 70% των κειμένων σε αυτό το σύνολο δεδομένων παράγεται από TN.

Είναι αξιοσημείωτο από όλα τα μέχρι τώρα αποτελέσματά μας ότι μεταξύ των συνόλων δεδομένων που επιλέξαμε οι ανιχνευτές δυσκολεύτηκαν πολύ περισσότερο να επιτύχουν καλή ακρίβεια στα σύνολα δεδομένων AuTexTification και MAGE. Κάνουμε την υπόθεση ότι αυτό μπορεί να οφείλεται στο γεγονός ότι αυτά τα σύνολα δεδομένων περιλαμβάνουν κείμενο από LLM που είναι εκτός της οικογένειας μοντέλων GPT, και επομένως είναι πιο δύσκολο να ανιχνευθεί από ότι κείμενο από το ChatGPT ή άλλα μοντέλα της OpenAI, στα οποία έχει επικεντρωθεί η συντριπτική πλειονότητα των εργασιών ανίχνευσης κειμένου, δεδομένου ότι κυριαρχούν στη δεξαμενή των LLM.

Για να ελέγξουμε αυτή την υπόθεση, επεκτείνουμε τον αλγόριθμο ανίχνευσης περιπλοκότητας βέλτιστου κατωφλίου μας, ο οποίος υπολογίζεται χρησιμοποιώντας τις βαθμολογίες του GPT-2, σε άλλα LLM , όπως το Palmyra (από την Writer AI), το OPT (από την Meta AI) και το GPT-NeoX (από την EleutherAI). Η απόδοση των ανιχνευτών αυτών παρουσιάζεται στα γραφήματα 0.2.7 και 0.2.8, για το σύνολο εκπαίδευσης και το σύνολο δοκιμών αντίστοιχα.

Όπως φαίνεται σε αυτά τα σχήματα, η επιλογή LLM για τον υπολογισμό των μετρικών περιπλοκότητας δεν οδηγεί σε σημαντικές διαφορές. Η ακρίβεια που επιτυγχάνεται, τόσο στα δεδομένα εκπαίδευσης όσο και στα δεδομένα δοκιμής, είναι αρκετά παρόμοια σε όλα τα μοντέλα, με το GPT-2 να έχει οριακά καλύτερες επιδόσεις από τα υπόλοιπα. Αυτό το αποτέλεσμα είναι αναμενόμενο, δεδομένου ότι η πλειονότητα των κειμένων που παράγονται από TN στα σύνολα δεδομένων μας προέρχεται από τη γενιά GPT. Συνολικά, η διαφορά μεταξύ "εύκολων" και "δύσκολων" συνόλων δεδομένων είναι εμφανής: όλα τα μοντέλα δυσκολεύονται να διαφοροποιήσουν τα σύνολα δεδομένων 1,4,και 6 χρησιμοποιώντας μόνο την περιπλοκότητα, κάτι που δικαιολογείται εύκολα αν εξετάσουμε τις σχετικές κατανομές τους μεταξύ των κλάσεων.

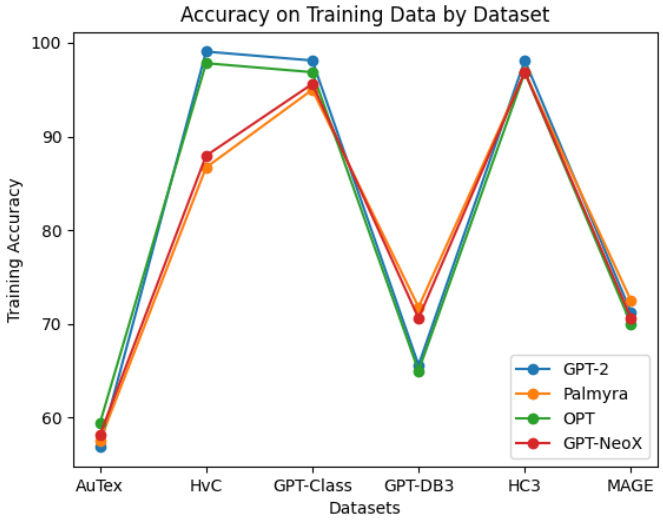

Figure 0.2.7: Ακρίβεια στο σύνολο εκπαίδευσης, ανά μοντέλο και σύνολο δεδομένων

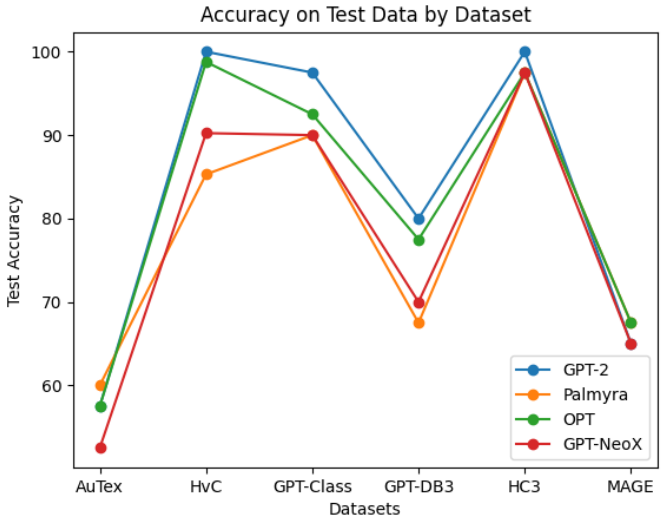

Figure 0.2.8: Ακρίβεια στο σύνολο εκπαίδευσης, ανά μοντέλο και σύνολο δεδομένων

Τέλος, πραγματοποιούμε ανάλυση με βάση τον ανιχνευτή Binoculars [25], ο οποίος χρησιμοποιεί την διασταυρούμενη περιπλοκότητα (cross perplexity) ως μέτρο του πόσο απρόβλεπτες είναι οι προβλέψεις των επόμενων τμημάτων κειμένου ενός μοντέλου για ένα άλλο μοντέλο όταν και τα δύο λειτουργούν στο ίδιο κείμενο. Συγκρίνοντας την περιπλοκότητα μιας συμβολοσειράς από ένα μοντέλο με το πώς την αντιλαμβάνεται ένα άλλο μοντέλο, η μέθοδος μπορεί να ανιχνεύσει με μεγαλύτερη ακρίβεια αν το κείμενο έχει παραχθεί από μηχανή ή έχει γραφτεί από άνθρωπο, ιδίως σε περιπτώσεις όπου προκαλείται μεταβλητότητα λόγω προτροπής. Ωστόσο, τα αποτελέσματα του συγκεκριμένου ανιχνευτή δεν διαφέρουν σημαντικά από τον απλό ανιχνευτή περιπλοκότητας, καθώς και το Binoculars δυσκολεύεται στα ίδια "δύσκολα" σύνολα δεδομένων, ενώ επιτυγχάνει σχεδόν τέλειο διαχωρισμό στα "εύκολα" σύνολα δεδομένων.

0.2.5 Σύνοψη και επιπλέον διαπιστώσεις

Μια σύνοψη της ακρίβειας όλων των ανιχνευτών με τους οποίους πειραματιστήκαμε στο πειραματικό μέρος παρουσιάζεται σε μορφή γραφήματος στο σχήμα 0.2.9 παρακάτω. Η μπλε γραμμή αντιπροσωπεύει και τους 3

ανιχνευτές DistilBERT, λαμβάνοντας το σκορ του προ-εκπαιδευμένου ανιχνευτή σε κάθε σύνολο δεδομένων.

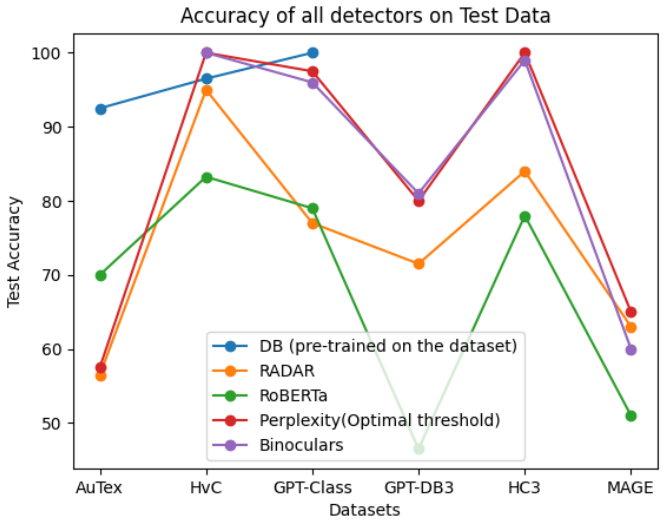

Figure 0.2.9: Απόδοση όλων των ανιχνευτών σε δοκιμαστικά δεδομένα

Όπως φαίνεται σε αυτό το σχήμα, οι ανιχνευτές που βασίζονται στην περιπλοκότητα (βέλτιστο κατώφλι και Binoculars) παρουσιάζουν καλύτερες επιδόσεις από τους ανιχνευτές που βασίζονται στα LLM στα σύνολα δεδομένων στα οποία οι κατανομές που δημιουργούνται από τον άνθρωπο και την τεχνητή νοημοσύνη είναι διακριτές. Μεταξύ τους δεν υπάρχουν σημαντικές διαφορές, γεγονός που μπορεί να αποδοθεί στο ότι δεν υπάρχουν περιπτώσεις του "προβλήματος capybara" [25], για τις οποίες ο ανιχνευτής Binoculars θα ήταν ενδεχομένως πιο κατάλληλος. Επιπλέον, αποδεικνύεται ότι η λεπτομερής ρύθμιση ενός ανιχνευτή για την εργασία ανίχνευσης κειμένου AI σε ένα συγκεκριμένο σύνολο δεδομένων οδηγεί επίσης σε πολύ ισχυρή απόδοση σε αυτό το σύνολο δεδομένων, αλλά έχει ως κόστος τη σημαντικά μικρότερη ικανότητα γενίκευσης, όπως εξηγήθηκε στην ενότητα 0.2.1.

Αποδεικνύουμε λοιπόν ότι η ανάλυση της περιπλοκότητας κειμένου, παρά την απλότητά της, μπορεί να συλλάβει αποτελεσματικά διαφορές στο κείμενο που συσχετίζονται με περιεχόμενο που έχει δημιουργηθεί από TN στην πλειονότητα των περιπτώσεων. Παρόλο που δεν ανταγωνίζονται πάντα για την υψηλότερη ακρίβεια, οι ανιχνευτές που βασίζονται στην ακαταλληλότητα μπορούν να χρησιμεύσουν ως ουσιαστικό σημείο αναφοράς ή μέτρο σύγκρισης στην εργασία ανίχνευσης κειμένου TN. Η υψηλή ακρίβεια των ανιχνευτών που βασίζονται στην περιπλοκότητα σε ένα σύνολο δεδομένων μπορεί να υποδηλώνει ότι το σύνολο δεδομένων μπορεί να είναι πολύ εύκολο για πιο εξελιγμένους ανιχνευτές, ενδεχομένως με έλλειψη ποικιλομορφίας ή πολυπλοκότητας. Αυτό μπορεί να καθοδηγήσει τη δημιουργία πιο δύσκολων συνόλων δεδομένων που αντικατοπτρίζουν καλύτερα τα σενάρια του πραγματικού κόσμου, όπου η διάκριση μεταξύ τεχνητού και ανθρώπινου κειμένου είναι πιο δύσκολη.

Επιπλέον, το TextFooler αποκαλύπτει βασικές πληροφορίες σχετικά με τους ταξινομητές κειμένου, δείχνοντας ότι οι αντιθετικές επιθέσεις μπορούν να θέσουν σε κίνδυνο την ευχέρεια του κειμένου για να παραπλανήσουν τους ταξινομητές. Αυτό υπογραμμίζει ότι τα κείμενα από ομιλητές της Αγγλικής ως δεύτερη γλώσσα είναι πιο επιρρεπή σε εσφαλμένη ταξινόμηση, ευθυγραμμιζόμενο με παρατηρήσεις από προηγούμενες μελέτες. Επιπλέον, τα κείμενα με ανθρώπινη συγγραφή συχνά διαθέτουν πιο ανεπίσημη γλώσσα, η οποία είναι λιγότερο συνηθισμένη στα παραγωγικά γλωσσικά μοντέλα (LLM). Τα LLM παράγουν συνήθως πιο επίσημη και δομημένη γλώσσα, συμβάλλοντας στην υψηλότερη απόδοση των ανιχνευτών κειμένου σε επιστημονικές περιλήψεις και άρθρα, τα οποία έχουν σταθερό ύφος. Οι παρατηρήσεις αυτές υποστηρίζονται από παραδείγματα και σχήματα που απεικονίζουν τον αντίκτυπο του γλωσσικού ύφους στην απόδοση των ταξινομητών, τα οποία βρίσκονται στην Ενότητα 4.3.

Επίσης, οι μετρήσεις με βάση την περιπλοκότητα εξετάστηκαν επίσης παράλληλα με τις προβλέψεις των ανιχνευτών LLM. Συγκρίθηκε η μέση περιπλοκότητα των αλλαγμένων και των πρωτότυπων κειμένων, δείχνοντας μια σημαντική διαφορά μεταξύ των κειμένων που δημιουργήθηκαν με τεχνητή νοημοσύνη και των κειμένων

που συντάχθηκαν από ανθρώπους σε όλα τα σύνολα δεδομένων. Αυτή η διαφορά παρέμεινε σε μεγάλο βαθμό αμετάβλητη από τις διαταραχές του TextFooler, υποδεικνύοντας ότι τα συστήματα που βασίζονται στην περιπλοκότητα, όπως το GPTZero ή το Binoculars, είναι λιγότερο πιθανό να παραπλανηθούν από αυτές τις επιθέσεις. Ωστόσο, οι ανιχνευτές μαύρου κουτιού ήταν ευάλωτοι στις διαταραχές, γεγονός που υποδηλώνει ότι χρησιμοποιούν κριτήρια πέραν της περιπλοκότητας για την ταξινόμηση. Τα ευρήματα υποδηλώνουν ότι, ενώ τόσο οι ανιχνευτές που βασίζονται στην περιπλοκότητα όσο και οι ανιχνευτές ML μπορούν να επιτύχουν υψηλή ακρίβεια, το επιτυγχάνουν αξιολογώντας διαφορετικά χαρακτηριστικά κειμένου. Συμπερασματικά, η ενσωμάτωση και των δύο μεθόδων θα μπορούσε να ενισχύσει την ανθεκτικότητα των συστημάτων ταξινόμησης κειμένου έναντι εχθρικών επιθέσεων αξιοποιώντας τα συμπληρωματικά τους πλεονεκτήματα.

0.3 Έρευνα χρηστών

Η ανθρώπινη συμμετοχή και επίβλεψη είναι κρίσιμη στη διαδικασία ανίχνευσης κειμένου ΤΝ λόγω των ηθικών επιπτώσεων και των ζητημάτων ευρωστίας που σχετίζονται με τα αυτοματοποιημένα συστήματα ανίχνευσης. Οι ανθρώπινοι κριτές παρέχουν ένα ουσιαστικό επίπεδο επαλήθευσης, μετριάζοντας τις πιθανές προκαταλήψεις, όπως αυτές κατά των μη φυσικών ομιλητών, και εξασφαλίζοντας διαφάνεια και υπευθυνότητα στις αποφάσεις της ΤΝ. Για την περαιτέρω διερεύνηση αυτού του θέματος, διεξήχθη έρευνα χρηστών με σκοπό την αξιολόγηση των ανθρώπινων επιδόσεων στον εντοπισμό κειμένων που έχουν δημιουργηθεί από ΤΝ, την κατανόηση των σχετικών γνωστικών διαδικασιών και τον εντοπισμό των χαρακτηριστικών που θεωρούνται ενδεικτικά της δημιουργίας ΤΝ. Ο σχεδιασμός και τα αποτελέσματα της έρευνας περιγράφονται παρακάτω, με στόχο την ενίσχυση της αποτελεσματικότητας και της αμεροληψίας των συστημάτων ανίχνευσης κειμένου τεχνητής νοημοσύνης.

0.3.1 Δομή έρευνας

Η έρευνα χρηστών μας διερευνά τις ανθρώπινες επιδόσεις στην ανίχνευση κειμένου που έχει δημιουργηθεί από τεχνητή νοημοσύνη και την αποτελεσματικότητα των ανιχνευτών τεχνητής νοημοσύνης, με στόχο την κατανόηση των γνωστικών διαδικασιών και των χαρακτηριστικών που χρησιμοποιούν οι άνθρωποι σε αυτό το έργο. Η μελέτη αυτή παρακινείται από την αυξανόμενη σημασία της διάκρισης μεταξύ ανθρώπινου και τεχνητής νοημοσύνης παραγόμενου περιεχομένου λόγω των ταχέων εξελίξεων στην τεχνητή νοημοσύνη. Αξιοποιώντας τα τρέχοντα σύνολα δεδομένων στην εργασία μας που έχουν δοκιμαστεί με ανιχνευτές ΤΝ, στοχεύουμε να συγκρίνουμε τη λήψη αποφάσεων από τον άνθρωπο και την ΤΝ στην ανίχνευση κειμένου, παρέχοντας μια ακριβή ανάλυση των δυνατοτήτων και των περιορισμών τους.

Η έρευνα είναι δομημένη σε τέσσερις ενότητες: εισαγωγή, δύο εργασίες ανίχνευσης κειμένου ΤΝ και μια τελική εργασία παράφρασης. Οι συμμετέχοντες ξεκινούν παρέχοντας πληροφορίες για την εμπειρία τους με τα μοντέλα ΤΝ και την αποτελεσματικότητά τους στη διαφοροποίηση μεταξύ ανθρώπινων και τεχνητών κειμένων. Στη συνέχεια προχωρούν στην πρώτη εργασία ανίχνευσης, όπου ταξινομούν 10 κείμενα είτε ως ανθρώπινα είτε ως παραγόμενα από ΤΝ και εξηγούν το σκεπτικό τους. Στη δεύτερη εργασία ανίχνευσης, οι συμμετέχοντες λαμβάνουν πρόσθετες εξηγήσεις από ανιχνευτές ΤΝ, όπως εξηγήσεις αντιφατικών παραδειγμάτων και εξηγήσεις LIME, και στη συνέχεια ταξινομούν ένα άλλο σύνολο κειμένων.

Το σώμα κειμένων για την έρευνα περιλαμβάνει μια ποικίλη εκπροσώπηση τομέων και συνόλων δεδομένων. Η έρευνα χρησιμοποιεί 50 διαλεγμένα κείμενα που χωρίζονται ομοιόμορφα σε πέντε σύνολα, συμπεριλαμβανομένων tweets, ακαδημαϊκών περιλήψεων, κειμένων από μη φυσικούς ομιλητές της αγγλικής γλώσσας, κειμένων που δημιουργήθηκαν από AI μοντέλα που δεν ανήκουν στη σειρά GPT και παραδειγμάτων που παραπλάνησαν ανιχνευτές AI. Κάθε διαχωρισμός διασφαλίζει μια ισορροπημένη εκπροσώπηση κειμένων που παράγονται από ανθρώπους και ΤΝ, διατηρώντας την ακρίβεια των ανιχνευτών γενικής χρήσης και των ανιχνευτών που βασίζονται στην περιπλοκότητα μεταξύ 80-90%.

Τέλος, οι συμμετέχοντες καλούνται να παραφράσουν σύντομες παραγράφους που παράγονται από την ΤΝ για να τις κάνουν να φαίνονται πιο ανθρώπινες, δοκιμάζοντας την κατανόησή τους για το τι διακρίνει ένα κείμενο που μοιάζει με ανθρώπινο. Τα αποτελέσματα της έρευνας, τα οποία αναλύονται παρακάτω, θα παράσχουν πληροφορίες σχετικά με τις ανθρώπινες δυνατότητες ανίχνευσης περιεχομένου που παράγεται από ΤΝ και θα ενημερώσουν για την ανάπτυξη πιο αποτελεσματικών και δίκαιων συστημάτων ανίχνευσης κειμένου ΤΝ.

0.3.2 Αποτελέσματα έρευνας χρηστών

Το εισαγωγικό μέρος της έρευνας αποκάλυψε πληροφορίες σχετικά με την εξοικείωση των συμμετεχόντων με τα μοντέλα τεχνητής νοημοσύνης και την εμπιστοσύνη τους στις δυνατότητες ανίχνευσης κειμένου με τεχνητή νοημοσύνη. Η κατανομή των απαντήσεων σχετικά με την εξοικείωση με την ΤΝ και τα μοντέλα δημιουργίας κειμένου ακολούθησε μια περίπου κανονική κατανομή, υποδεικνύοντας μια ισορροπημένη εκπροσώπηση σε διάφορα επίπεδα εξοικείωσης. Σημαντικό μέρος των συμμετεχόντων ανέφερε τακτική χρήση μοντέλων παραγωγής κειμένου όπως το ChatGPT στην εργασία ή τις σπουδές τους, υπογραμμίζοντας την ευρεία υιοθέτησή τους πέρα από εξειδικευμένους τομείς ΤΝ. Όσον αφορά την εμπιστοσύνη στην ανίχνευση κειμένου με τεχνητή νοημοσύνη, οι συμμετέχοντες έδειξαν μεγαλύτερη εμπιστοσύνη στα αυτοματοποιημένα μοντέλα (μέση βαθμολογία 3,41) σε σύγκριση με τις δικές τους ικανότητες (μέση βαθμολογία 2,78). Αυτή η διαφορά υποδηλώνει μια συγκρατημένη αισιοδοξία ως προς τις δυνατότητες των αυτοματοποιημένων συστημάτων να διακρίνουν τα κείμενα που παράγονται με τεχνητή νοημοσύνη από τα ανθρώπινα κείμενα, η οποία μετριάζεται από την επίγνωση των τρεχόντων περιορισμών και των προκλήσεων που συζητήθηκαν σε προηγούμενα κεφάλαια.

Τα αποτελέσματα της έρευνάς μας αναδεικνύουν τις σημαντικές προκλήσεις που αντιμετωπίζουν οι ανθρώπινοι κριτές στην ακριβή διάκριση μεταξύ κειμένου που έχει παραχθεί από τεχνητή νοημοσύνη και κειμένου που έχει γραφτεί από άνθρωπο. Παρά τον σχολιασμό συνολικά 540 κειμένων από 27 συμμετέχοντες, η συνολική ακρίβεια παρέμεινε στο 52,96%, μόνο οριακά πάνω από αυτό που θα επιτυγχανόταν με τυχαία μαντεψιά (50%). Η κατανομή των βαθμολογιών των χρηστών παρουσιάζεται στο Σχήμα 0.3.1.

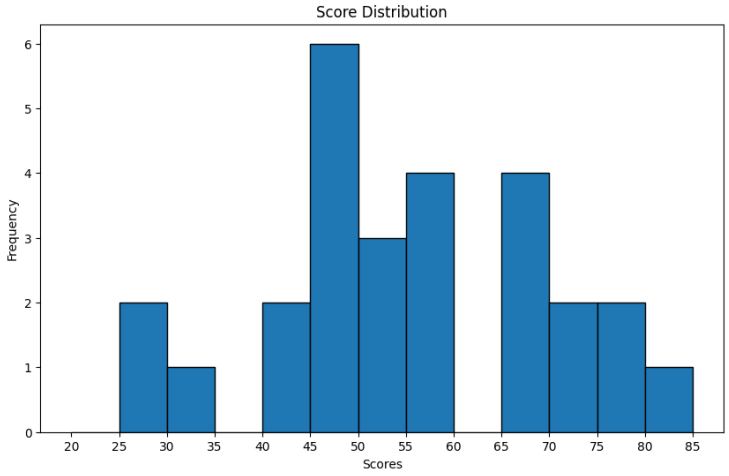

Figure 0.3.1: Κατανομή απόδοσης των συμμετεχόντων στην έρευνα χρηστών

Το εύρημα αυτό έρχεται σε αντίθεση με παλαιότερες μελέτες που διεξήχθησαν πριν από την εξάπλωση προηγμένων μοντέλων ΤΝ όπως το ChatGPT, οι οποίες έδειχναν ότι οι άνθρωποι μπορούσαν να εκτελέσουν αξιόπιστα τέτοιες εργασίες. Αντίθετα, τα αποτελέσματά μας ευθυγραμμίζονται με την πρόσφατη αντίληψη που δείχνει ότι καθώς η παραγωγή κειμένου από ΤΝ έχει εξελιχθεί, η διάκριση των αποτελεσμάτων της από την ανθρώπινη γραφή έχει γίνει όλο και πιο δύσκολη για τους ανθρώπους.

Η ανάλυση των επιδόσεων των συμμετεχόντων στις δύο ξεχωριστές εργασίες ανίχνευσης κειμένου έδειξε σταθερά ποσοστά ακρίβειας 53,7% και 52,2%, αντίστοιχα, υποδεικνύοντας ότι οι πρόσθετες εξηγήσεις που δόθηκαν κατά τη διάρκεια της δεύτερης εργασίας δεν βελτίωσαν τις επιδόσεις. Αυτό υποδηλώνει ότι, ενώ εξηγήσεις όπως τα αντιφατικά παραδείγματα και οι εξηγήσεις LIME μπορούν να διαφωτίσουν τις αποφάσεις ανίχνευσης κειμένου της ΤΝ, δεν μεταφράζονται απαραίτητα σε βελτιωμένη ακρίβεια για τους ανθρώπινους κριτές. Επιπλέον, η διερεύνηση του τρόπου με τον οποίο τα χαρακτηριστικά των συμμετεχόντων, όπως η εξοικείωση με την ΤΝ και τα μεγάλα γλωσσικά μοντέλα (LLM), επηρέασαν την απόδοση δεν αποκάλυψε καμία διακριτή συσχέτιση. Οι συμμετέχοντες σε διαφορετικά επίπεδα εξοικείωσης με τα LLM παρουσίασαν παρόμοιες κατανομές επιδόσεων, γεγονός που υποδηλώνει ότι η εξειδίκευση στις τεχνολογίες ΤΝ από μόνη της δεν προσδίδει πλεονέκτημα σε αυτή τη συγκεκριμένη εργασία.Η μόνη συσχέτιση που προέκυψε ήταν αυτή μεταξύ της εμπιστοσύνης των συμμετεχόντων στις ταξινομήσεις τους και των επιδόσεών τους, με Τα υψηλότερα επίπεδα εμπιστοσύνης να αντιστοιχούν σε καλύτερη ακρίβεια. Αυτά τα συμπεράσματα υπογραμμίζουν την πολυπλοκότητα της πρόκλησης ανίχνευσης

κειμένων TN και υποδεικνύουν δρόμους για περαιτέρω έρευνα για τη βελτίωση τόσο των ανθρώπινων όσο και των αυτοματοποιημένων μεθόδων για τη διάκριση μεταξύ κειμένων που δημιουργούνται από TN και ανθρώπινων κειμένων.

0.3.3 Αποτελέσματα ανά κατηγορία κειμένου

Η ανάλυση των ανθρώπινων επιδόσεων σε διάφορες κατηγορίες κειμένων στην έρευνά μας υπογραμμίζει τόσο τις προκλήσεις όσο και τις αποχρώσεις που είναι εγγενείς στη διάκριση κειμένων που παράγονται από τεχνητή νοημοσύνη από περιεχόμενο που έχει γραφτεί από τον άνθρωπο. Τα συγκεντρωτικά αποτελέσματα ανά κατηγορία παρουσιάζονται στο Σχήμα 0.3.2.

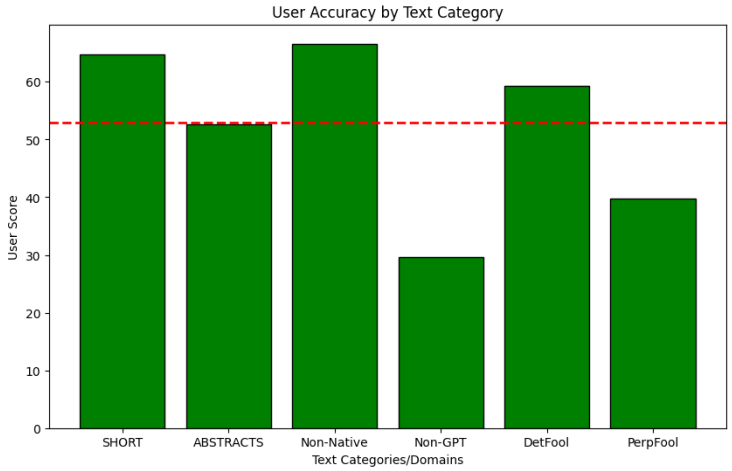

Figure 0.3.2: Κατανομή απόδοσης των συμμετεχόντων ανά κατηγορία δεδομένων

Συνολικά, οι ανθρώπινοι αξιολογητές επέδειξαν επιδόσεις οριακά καλύτερες από την τυχαία επιλογή, με μέση ακρίβεια 52,96%. Ωστόσο, οι επιδόσεις διέφεραν σημαντικά μεταξύ των διαφόρων τύπων κειμένου. Τα σύντομα κείμενα, όπως τα tweets, αποδείχθηκαν σχετικά ευκολότερα για τους ανθρώπους, επιτυγχάνοντας ακρίβεια 64,8%. Αυτή η υψηλότερη ακρίβεια πιθανόν να οφείλεται στο περιορισμένο περιεχόμενο των σύντομων κειμένων, το οποίο μπορεί να μεγεθύνει τις αποκλίσεις στη γλωσσική ποιότητα που είναι χαρακτηριστικές για το περιεχόμενο που παράγεται από TN.

Αντίθετα, οι επιστημονικές περιλήψεις αποτέλεσαν μια πιο δύσκολη πρόκληση, με τους ανθρώπινους κριτές να επιτυγχάνουν ακρίβεια 52,7%, η οποία είναι παρόμοια με τον συνολικό μέσο όρο τους. Αυτό έρχεται σε έντονη αντίθεση με τους αυτοματοποιημένους ανιχνευτές με βάση την TN και τα συστήματα που βασίζονται στην περιπλοκότητα, τα οποία υπερέχουν στον εντοπισμό μοτίβων σε δομημένα ακαδημαϊκά κείμενα. Ενδιαφέρον έχει η παρατήρηση ότι τα κείμενα που παράγονται από μοντέλα εκτός της σειράς GPT από σύνολα δεδομένων όπως το MAGE οδήγησαν σε αξιοσημείωτα κακές επιδόσεις τους αξιολογητές, οι οποίοι σημείωσαν μόλις 29,6%. Αυτό υποδηλώνει τη μη εξοικείωση των συμμετεχόντων με αυτά τα λιγότερο διαδεδομένα μοντέλα TN και υπογραμμίζει την εξελισσόμενη πολυπλοκότητα στη διαφοροποίηση μεταξύ κειμένων που παράγονται από ανθρώπους και TN σε διάφορα σύνολα δεδομένων.

Επιπλέον, η διακύμανση των επιδόσεων σε κείμενα που ξεγέλασαν τους παραδοσιακούς ανιχνευτές που βασίζονται στην TN (59,25%) έναντι των ανιχνευτών που βασίζονται στην περιπλοκότητα (39,81%) προσφέρει ενδιαφέρουσες πληροφορίες. Υποδεικνύει ότι, ενώ οι παραδοσιακοί ανιχνευτές που βασίζονται στην TN και οι άνθρωποι μπορεί να ευθυγραμμίζονται λιγότερο όταν τα κείμενα είναι ιδιαίτερα παραπλανητικά, τα συστήματα που βασίζονται στην περιπλοκότητα, τα οποία αξιολογούν την ευχέρεια και τη συνοχή των κειμένων, αποτελούν εντελώς διαφορετική πρόκληση. Αυτή η διαφορά υπογραμμίζει την ανάγκη για περαιτέρω έρευνα για τη διερεύνηση του τρόπου με τον οποίο οι διάφορες μεθοδολογίες ανίχνευσης αλληλεπιδρούν με τις ανθρώπινες γνωστικές διαδικασίες, ενισχύοντας έτσι την ανάπτυξη πιο ισχυρών συστημάτων ανίχνευσης κειμένων TN.

Συνολικά, τα ευρήματα αναδεικνύουν την περίπλοκη φύση της ανίχνευσης κειμένου TN, όπου οι ανθρώπινες επιδόσεις διαφέρουν σημαντικά ανάλογα με τα χαρακτηριστικά του κειμένου και τις μεθοδολογίες ανίχνευσης. Αυτές οι γνώσεις όχι μόνο υπογραμμίζουν τους σημερινούς περιορισμούς στις ανθρώπινες αξιολογικές ικανότητες

σε σύγκριση με τα αυτοματοποιημένα συστήματα, αλλά υπογραμμίζουν επίσης τις πολυπλοκότητες που συνεπάγεται ο σχεδιασμός αποτελεσματικών εργαλείων ανίχνευσης ΤΝ, ικανών να διακρίνουν όλο και πιο εξελιγμένο περιεχόμενο που παράγεται από ΤΝ.

0.3.4 Γενικά συμπεράσματα

Στην έρευνά μας, διερευνήσαμε τις αντιλήψεις και τις στρατηγικές των συμμετεχόντων για τη διάκριση των κειμένων που δημιουργούνται από τεχνητή νοημοσύνη από εκείνα που έχουν γραφτεί από ανθρώπους, ρίχνοντας φως τόσο στις προκλήσεις όσο και στις ιδέες για την ανίχνευση κειμένων τεχνητής νοημοσύνης. Αξίζει να σημειωθεί ότι, ενώ οι εξηγήσεις που δόθηκαν κατά τη διάρκεια της έρευνας δεν βελτίωσαν σημαντικά την ακρίβεια των συμμετεχόντων, η ανατροφοδότηση σχετικά με αυτές τις εξηγήσεις ήταν ανάμεικτη, με την πλειοψηφία να τις αξιολογεί ως μέτρια χρήσιμες. Είναι ενδιαφέρον ότι, όταν τους ζητήθηκε να επιλέξουν μεταξύ των εξηγήσεων αντιπαραδειγμάτων και LIME, ένα σημαντικό μέρος των συμμετεχόντων θεώρησε και τις δύο εξίσου ωφέλιμες, ενώ ένα άλλο σημαντικό μέρος των συμμετεχόντων θεώρησε τις εξηγήσεις με αντιπαράδειγμα ως πιο χρήσιμες. Το γεγονός αυτο υποδηλώνει μια ισορροπημένη προτίμηση για τις αντιπαραδειγματικές εξηγήσεις που αναδεικνύουν τις διαφορές μεταξύ των κειμένων που δημιουργούνται από την ΤΝ και των ανθρώπινων κειμένων. Αυτό ευθυγραμμίζεται με προηγούμενες έρευνες που δείχνουν ότι οι αντιπαραθετικές εξηγήσεις, οι οποίες απεικονίζουν τον τρόπο με τον οποίο η αλλαγή ορισμένων χαρακτηριστικών επηρεάζει το αποτέλεσμα, είναι συχνά πιο διαισθητικές για την ανθρώπινη κατανόηση σε εργασίες τεχνητής νοημοσύνης.

Ωστόσο, παρά τη διαθεσιμότητα των εξηγήσεων, η συνολική ακρίβεια των συμμετεχόντων στη διάκριση των κειμένων που δημιουργήθηκαν από ΤΝ ήταν μόνο οριακά πάνω από την τύχη. Αυτό υπογραμμίζει τη δυσκολία που αντιμετωπίζουν οι άνθρωποι να διακρίνουν μεταξύ του προηγμένου περιεχομένου που παράγεται από την ΤΝ και του αυθεντικού ανθρώπινου γραπτού λόγου. Η ανάλυση των απαντήσεων ανοικτού τύπου των συμμετεχόντων παρείχε περαιτέρω πληροφορίες σχετικά με τις στρατηγικές ταξινόμησής τους. Οι συμμετέχοντες βασίζονταν συχνά σε γλωσσικές και γραμματικές ενδείξεις, σημειώνοντας ότι τα λάθη ή οι ασυνέπειες στη χρήση της γλώσσας ήταν περισσότερο ενδεικτικά των ανθρώπινων κειμένων, καθώς τα μοντέλα ΤΝ συνήθως παράγουν πιο "γυαλισμένα" και γραμματικά σωστά αποτελέσματα. Η παρατήρηση αυτή ήταν ιδιαίτερα σημαντική στο πλαίσιο μικρότερων κειμένων όπως τα tweets, όπου το περιορισμένο πλαίσιο και η ανεπίσημη γλωσσική χρήση μπορεί να ενισχύσουν τυχόν αποκλίσεις από τη φυσική ανθρώπινη γραφή.

Επιπλέον, τα υφολογικά στοιχεία έπαιξαν καθοριστικό ρόλο στις αξιολογήσεις των συμμετεχόντων. Συχνά επεσήμαναν ότι τα κείμενα που παράγονται από την ΤΝ τείνουν να παρουσιάζουν ένα πιο τυποποιημένο ή δομημένο ύφος, υποδηλώνοντας ότι τα κείμενα που τηρούν στενά προκαθορισμένα πρότυπα ή χρησιμοποιούν εξειδικευμένο λεξιλόγιο θα μπορούσαν να σηματοδοτούν τη μηχανική παραγωγή. Αντίθετα, τα κείμενα που θεωρούνταν πιο ελεύθερα ή προσωπικά ήταν πιο πιθανό να ταξινομηθούν ως ανθρώπινα, ανταναικλώντας τις προσδοκίες των συμμετεχόντων για τα μοντέλα ΤΝ που αγωνίζονται να συλλάβουν τη λεπτή, προσωπική πινελιά που συχνά χαρακτηρίζει την ανθρώπινη επικοινωνία. Η διάκριση αυτή αναδεικνύει τις εξελισσόμενες δυνατότητες των γλωσσικών μοντέλων ΤΝ, τα οποία μπορούν να παράγουν συνεκτικές και κατάλληλες για τα συμφραζόμενα απαντήσεις σε ένα ευρύ φάσμα προτροπών, θολώνοντας ενδεχομένως τα όρια μεταξύ ανθρώπινης και μηχανικής συγγραφής.

Επιπλέον, τα συναισθηματικά και προσωπικά στοιχεία στα κείμενα αναφέρθηκαν ως σημαντικοί δείκτες της ανθρώπινης έναντι της τεχνητής νοημοσύνης. Οι συμμετέχοντες σημείωσαν ότι τα κείμενα που μεταφέρουν συναισθήματα ή προσωπικές απόψεις είναι πιο πιθανό να είναι ανθρώπινης δημιουργίας, καθώς τα μοντέλα ΤΝ συνήθως δυσκολεύονται να αναπαράγουν αυθεντικά τα ανθρώπινα συναισθήματα και τις υποκειμενικές εμπειρίες. Ωστόσο, τα ευρήματα της έρευνας αποκάλυψαν επίσης προκλήσεις στην καθολική εφαρμογή αυτών των κριτηρίων, ιδίως όταν αντιμετωπίζονται σύνολα δεδομένων όπως το MAGE, όπου τα κείμενα που δημιουργήθηκαν από την ΤΝ μιμούνταν προσωπικές αφηγήσεις αρκετά αποτελεσματικά ώστε να παραπλανούν πολλούς συμμετέχοντες.

Συμπερασματικά, ενώ οι συμμετέχοντες χρησιμοποίησαν ποικίλες στρατηγικές -γλωσσική εξέταση, υφολογική ανάλυση και συναισθηματική αξιολόγηση- για να διαφοροποιήσουν τα κείμενα που παράγονται από την ΤΝ από τα ανθρώπινα κείμενα, οι επιδόσεις τους ήταν συνολικά μέτριες. Αυτό υπογραμμίζει την πολυπλοκότητα της εργασίας ανίχνευσης κειμένων ΤΝ, η οποία επηρεάζεται από την ταχεία πρόοδο των δυνατοτήτων της ΤΝ και τη διαφοροποιημένη φύση της κατανόησης της ανθρώπινης γλώσσας. Οι μελλοντικές ερευνητικές προσπάθειες θα πρέπει να συνεχίσουν να διερευνούν αυτές τις ταξινομήσεις και να βελτιώνουν τα εργαλεία ανίχνευσης ΤΝ για να ενισχύσουν τη συνεργασία μεταξύ των ανθρώπινων κριτών και των συστημάτων ΤΝ, βελτιώνοντας τελικά την

αξιοπιστία και την αποτελεσματικότητα των αξιολογήσεων της αυθεντικότητας κειμένου σε διάφορους τομείς.

0.3.5 Αποτελέσματα εργασίας παράφρασης κειμένου

Στο τελευταίο κομμάτι της έρευνάς μας, οι συμμετέχοντες πήραν μέρος σε ένα πείραμα παράφρασης κειμένου με στόχο να αποφύγουν την ανίχνευση του ως προϊόν της τεχνητής νοημοσύνης. Τα αποτελέσματα αποκάλυψαν διαφορετικούς βαθμούς επιτυχίας στην παραπλάνηση δύο διαφορετικών μεθόδων ανίχνευσης: του γενικής χρήσης LLM ανιχνευτή RoBERTa και ενός ανιχνευτή με βάση την περιπλοκότητα, βελτιστοποιημένο για το συγκεκριμένο σύνολο δεδομένων. Για το πρώτο κείμενο, οι συμμετέχοντες κατάφεραν να εξαπατήσουν τον ανιχνευτή RoBERTa 9 στις 15 φορές και τον ανιχνευτή που βασίζεται στην περιπλοκότητα 6 στις 15 φορές. Αξίζει να σημειωθεί ότι τα κείμενα με ποσοστό ομοιότητας κάτω του 85% με το αρχικό κείμενο μπορούν να θεωρηθούν σημαντικές αναδιατυπώσεις και όχι ελαφρές παραφράσεις, υποδηλώνοντας ουσιαστικές αλλαγές που ενσωμάτωναν περισσότερα ανθρώπινα στοιχεία. Όταν επικεντρωθήκαμε στα κείμενα που διατηρούσαν υψηλότερα ποσοστά ομοιότητας, οι συμμετέχοντες παραπλάνησαν το RoBERTa 6 στις 11 φορές, ενώ ο ανιχνευτής που βασίζεται στην περιπλοκότητα παραπλανήθηκε μόνο 2 στις 11 φορές. Αυτή η αντίθεση αναδεικνύει τις προκλήσεις ανθεκτικότητας που αντιμετωπίζει το RoBERTa με τις ελαφρές παραφράσεις, υπογραμμίζοντας την αποτελεσματικότητα του ανιχνευτή που βασίζεται στην περιπλοκότητα στην ανίχνευση πιο λεπτών αλλαγών. Στη δεύτερη φάση του πειράματος, οι συμμετέχοντες αντιμετώπισαν ένα διαφορετικό κείμενο, με στόχο επίσης να αποφύγουν την ανίχνευση του ως κείμενο TN . Εδώ, η περιπλοκότητα του βασικού κειμένου ήταν υψηλότερη, υποδεικνύοντας ένα κείμενο πιο κοντά στην ανθρώπινη γραφή, το οποίο θεωρητικά θα έπρεπε να είναι πιο εύκολο να παραφραστεί. Παρόλα αυτά, οι συμμετέχοντες αντιμετώπισαν δυσκολίες, καθώς μόνο 7 από τους 11 κατάφεραν να παραφράσουν το κείμενο με επιτυχία χωρίς να το τροποποιήσουν σημαντικά και διατηρώντας υψηλό ποσοστό ομοιότητας άνω του 85%. Ακόμη και σε αυτό το σενάριο, όπου τα κείμενα ήταν ήδη πιο ανθρώπινα, οι συμμετέχοντες ήταν λιγότερο επιτυχείς στο να παραπλανούν τους ανιχνευτές με συνέπεια, με το RoBERTa και τον ανιχνευτή που βασίζεται στην περιπλοκότητα να εξαπατώνται σε λιγότερες περιπτώσεις σε σύγκριση με το πρώτο πείραμα. Αυτό το αποτέλεσμα υπογραμμίζει τις προκλήσεις που συνεπάγεται η τροποποίηση κειμένου με ταυτόχρονη διατήρηση του αρχικού νοήματος και της φυσικότητάς του. Συνολικά, το πείραμα της παράφρασης ανέδειξε τις πολυπλοκότητες που συνεπάγεται η αποφυγή της ανίχνευσης TN μέσω της τροποποίησης κειμένου. Ενώ οι συμμετέχοντες επέδειξαν την ικανότητα να αυξάνουν την περιπλοκότητα των βασικών κειμένων, υποδηλώνοντας έτσι μια πιο ανθρώπινη παραγωγή, η αποτελεσματικότητα διέφερε μεταξύ διαφορετικών μοντέλων ανίχνευσης και κειμενικών πλαισίων. Τα ευρήματα αυτά υπογραμμίζουν τη συνεχιζόμενη ανάγκη για ισχυρές μεθόδους ανίχνευσης που μπορούν να διακρίνουν λεπτές παραλλαγές σε κείμενα που παράγονται από τον άνθρωπο και την TN, καθώς και τη λεπτή κατανόηση της γλώσσας που απαιτείται για την επιτυχή παράφραση κειμένου σε ρεαλιστικά περιβάλλοντα.

0.4 Συμπεράσματα

Εν κατακλείδι, η εκτεταμένη ανάλυσή μας σχετικά με την εργασία ανίχνευσης κειμένου TN παρείχε σημαντικές πληροφορίες σχετικά με τις τρέχουσες προκλήσεις και τις πιθανές βελτιώσεις. Εξετάσαμε διεξοδικά τις αντιθετικές επιθέσεις που στοχεύουν στα σύγχρονα συστήματα αυτοματοποιημένης ανίχνευσης, διερευνώντας τη λεπτή ισορροπία μεταξύ της επίτευξης υψηλής ακρίβειας, της γενίκευσης και της ευρωστίας, εντοπίζοντας παράλληλα συγκεκριμένες αδυναμίες. Εξετάζοντας την περιπλοκότητα κειμένου ως αξιόπιστη μετρική για τη μέτρηση της μη προβλεψιμότητας ενός κειμένου στα παραγωγικά μοντέλα, αναπτύξαμε έναν απλό ανιχνευτή κειμένου TN. Αυτός ο ανιχνευτής όχι μόνο χρησιμεύει ως ένα σταθερό σημείο αναφοράς για μελλοντικούς, πιο εξελιγμένους ανιχνευτές, αλλά επίσης αναδεικνύει ότι τα τρέχοντα σύνολα δεδομένων που χρησιμοποιούνται σε αυτό το έργο μπορεί να μην αντικατοπτρίζουν με ακρίβεια ρεαλιστικά σενάρια. Επιπλέον, η έρευνά μας για τους χρήστες, που μοιάζει με ένα σύγχρονο "τεστ Turing" για προηγμένα LLM, αποκάλυψε ότι η ανθρώπινη απόδοση ανίχνευσης είναι περίπου στα επίπεδα της τυχαίας επιλογής, υποδεικνύοντας την αναξιοπιστία τους σε αυτόν τον τομέα. Τα πειράματα με μεθόδους XAI που αποσκοπούν στην ενίσχυση της ακρίβειας των ανθρώπινων ανιχνευτών δεν έδειξαν σημαντική επίδραση, αν και η βαθύτερη ανάλυση των επιδόσεων των ανθρώπινων ανιχνευτών παρείχε πολύτιμες πληροφορίες που θα μπορούσαν να καθοδηγήσουν την ανάπτυξη καλύτερων επεξηγηματικών εργαλείων στο μέλλον. Τέλος, παρατηρώντας ανθρώπους να παραφράζουν κείμενα που δημιουργούνται από TN για να φαίνονται σαν ανθρώπινα, αξιολογήσαμε την αποτελεσματικότητά τους στην παραπλάνηση διαφόρων συστημάτων ανίχνευσης, συμβάλλοντας περαιτέρω στην κατανόηση της αλληλεπίδρασης μεταξύ των ανθρώπινων και των αυτοματοποιημένων δυνατοτήτων ανίχνευσης κειμένου TN.

0.4.1 Συζήτηση

Παρά τη σημαντική πρόοδο που σημειώθηκε στην κατανόηση των περιπλοκών της εργασίας ανίχνευσης κειμένου TN, η εργασία μας έχει αρκετούς περιορισμούς. Πρώτον, επικεντρωθήκαμε στο πλαίσιο επίθεσης TextFooler λόγω της κοινής χρήσης, της χαμηλής απαίτησής του σε πόρους και της ευκολίας χρήσης του. Ωστόσο, οι υποθέσεις που έγιναν σχετικά με την ευχέρεια κειμένου ενδέχεται να είναι χαρακτηριστικά ειδικά του πλαισίου. Αυτά τα χαρακτηριστικά μπορεί να μην εμφανιστούν εάν χρησιμοποιούνταν άλλα πλαίσια αντιετθικών επιθέσεων με καλύτερη ευχέρεια, όπως το MiCE. Αυτό υποδηλώνει ότι τα συμπεράσματά μας σχετικά με την ευχέρεια και την ανιχνευσιμότητα μπορεί να εξαρτώνται από το πλαίσιο και ότι περαιτέρω έρευνα με τη χρήση ενός ευρύτερου φάσματος αντίπαλων εργαλείων είναι απαραίτητη για πιο γενικευμένα συμπεράσματα.

Επιπλέον, τα αποτελέσματα και οι υποθέσεις μας είναι εγγενώς συνδεδεμένα με τα σύνολα δεδομένων που χρησιμοποιήσαμε. Τα χαρακτηριστικά και οι λεπτομέρειες μπορεί να διαφέρουν σημαντικά ανάλογα με το πλαίσιο κάθε συνόλου δεδομένων, γι' αυτό και συμπεριλάβαμε ένα ευρύ φάσμα συνόλων δεδομένων που καλύπτουν διάφορα μήκη κειμένου και τομείς (tweets, άρθρα ειδήσεων, επιστημονικά άρθρα κ.λπ.). Αυτή η ποικιλομορφία αποσκοπούσε στην παροχή μιας ολοκληρωμένης επισκόπησης, ωστόσο τα ευρήματά μας υπογραμμίζουν τους περιορισμούς πολλών συνήθως χρησιμοποιούμενων συνόλων δεδομένων. Ειδικότερα, η μελέτη μας αποκάλυψε ότι το σύνολο δεδομένων MAGE παρουσιάζει σημαντικές προκλήσεις τόσο για τους ανιχνευτές τεχνητής νοημοσύνης όσο και για τους ανθρώπους. Η εξελιγμένη μηχανική προτροπής (prompt engineering) που εμπλέκεται στη δημιουργία του συνόλου δεδομένων MAGE εξαπατά αποτελεσματικά τόσο τους ανθρώπους όσο και τα αυτοματοποιημένα συστήματα ανίχνευσης. Ενώ η εργασία στην οποία εισάγεται το συγκεκριμένο σύνολο δεδομένων υποδηλώνει ότι η λεπτομερής ρύθμιση εξειδικευμένων ανιχνευτών για μεγάλα κείμενα μπορεί να επιτύχει επιδόσεις πάνω από 85%, πρέπει να σημειωθούν οι περιορισμοί αυτής της μεθόδου. Σε ρεαλιστικά σενάρια, η κατανομή από την οποία προέρχεται ένα κείμενο που ενδεχομένως έχει δημιουργηθεί από τεχνητή νοημοσύνη είναι συχνά άγνωστη, περιορίζοντας την πρακτικότητα τέτοιων προσεγγίσεων λεπτομερούς ρύθμισης.

Τέλος, οι προκλήσεις που θέτει το σύνολο δεδομένων MAGE δείχνουν ότι οι εργασίες ανίχνευσης κειμένου θα γίνονται όλο και πιο δύσκολες τα επόμενα χρόνια. Αυτό, σε συνδυασμό με τις αναξιόπιστες επιδόσεις των ανθρώπων στο διαχωρισμό κειμένων, υποδηλώνει ότι η πρακτική ανίχνευση μπορεί σύντομα να καταστεί ανέφικτη. Υπό το πρίσμα αυτών των προκλήσεων, ίσως ήρθε η ώρα να εξεταστούν νέες κατευθύνσεις, όπως η εφαρμογή από τους συγγραφείς του LLM μηχανισμών για την ανίχνευση των δικών τους παραγόμενων κειμένων, όπως οι τεχνικές υδατοσήμανσης. Αυτά τα προληπτικά μέτρα θα μπορούσαν να προσφέρουν μια πιο ισχυρή λύση στην εξελισσόμενη πολυπλοκότητα της ανίχνευσης κειμένων TN, εξασφαλίζοντας την ακεραιότητα και την αξιοπιστία του παραγόμενου περιεχομένου.

0.4.2 Μελλοντικές κατευθύνσεις

Η εργασία αυτή ανοίγει το δρόμο για πολυάριθμες μελλοντικές μελέτες στο πλαίσιο της ανίχνευσης κειμένων TN, οι οποίες θα μπορούσαν να έχουν σημαντικές επιπτώσεις στην κοινωνία. Οι συγκρίσεις μας μεταξύ των σύγχρονων ανιχνευτών κειμένου με βάση την TN και του δικού μας ανιχνευτή με βάση την περιπλοκότητα υποδηλώνουν ότι, ενώ επιτυγχάνουν παρόμοια υψηλή ακρίβεια, το επιτυγχάνουν μέσω διαφορετικών μηχανισμών. Οι επιθέσεις που έχουν σχεδιαστεί για να επιτεθούν σε έναν τύπο ανιχνευτή συνήθως δεν εξαπατούν τον άλλο. Οι γνώσεις από την έρευνα χρηστών μας δείχνουν ότι οι ανιχνευτές που βασίζονται στην περιπλοκότητα μπορεί να είναι πιο ανθεκτικοί στις ανθρώπινες αλλαγές και να λειτουργούν πιο παρόμοια με τον τρόπο με τον οποίο οι άνθρωποι αναγνωρίζουν και επισημαίνουν τα δυνητικά κείμενα που παράγονται από μηχανές. Ελπίζουμε αυτή η μελέτη να ενθαρρύνει την περαιτέρω έρευνα σε συστήματα βασισμένα στην περιπλοκότητα, όπως η ενσωμάτωση μιας αμυντικής "κερκόπορτας" βασισμένης στην περιπλοκότητα σε παραδοσιακούς ανιχνευτές βασισμένους στα LLM. Κάτι τέτοιο θα μπορούσε να μετριάσει τα ζητήματα ευρωστίας, καθώς οποιαδήποτε αντίπαλη επίθεση στον ανιχνευτή θα πρέπει επίσης να παρακάμψει το φίλτρο περιπλοκότητας.

Μια άλλη πολλά υποσχόμενη κατεύθυνση για μελλοντική έρευνα προκύπτει από την ανάλυση της έρευνας χρηστών. Διαπιστώσαμε ότι οι τεχνικές τοπικής ερμηνευσιμότητας, όπως η τοπική σημασία χαρακτηριστικών και οι αντιφατικές επεξηγήσεις, δεν είναι πολύ αποτελεσματικές στην υποβοήθηση των ανθρώπων σε εργασίες ανίχνευσης κειμένου TN. Ωστόσο, η ανάλυσή των ανθρώπινων αποφάσεων που κάναμε αποκάλυψε ότι οι άνθρωποι βασίζονται περισσότερο σε καθολικούς δείκτες, όπως η γλώσσα, η γραμματική, το ύφος, ο τόνος και η συνολική αίσθηση του κειμένου. Αυτοί οι παράγοντες είναι συγγενείς με τις παραδοσιακές μεθόδους που βασίζονται σε χαρακτηριστικά και στιλομετρικές μεθόδους. Αυτή η διαπίστωση υποδηλώνει ότι οι καθολικές εξ-

ηγήσεις, όπως τα συστήματα που βασίζονται σε κανόνες, θα μπορούσαν να παρέχουν καλύτερη υποστήριξη στους ανθρώπινους κριτές, ευθυγραμμιζόμενοι περισσότερο με τις φυσικές διαδικασίες αξιολόγησής τους. Επομένως, η διερεύνηση τεχνικών σφαιρικών εξηγήσεων θα μπορούσε να ενισχύσει την ανθρώπινη κατανόηση και αποτελεσματικότητα στον εντοπισμό κειμένων που δημιουργούνται από τεχνητή νοημοσύνη.

Τέλος, το τελευταίο μέρος της εργασίας μας ανέδειξε τις σημαντικές δυνατότητες κατανόησης του τρόπου με τον οποίο οι άνθρωποι παραφράζουν περιεχόμενο που παράγεται από TN. Με την αυξανόμενη επικράτηση της χρήσης των LLM για την τελειοποίηση κειμένων γραμμένων από ανθρώπους σε διάφορους τομείς, η εξέταση της αλληλεπίδρασης μεταξύ ανθρώπινης και μηχανικής συγγραφής αποτελεί έναν ενδιαφέροντα τομέα για μελλοντικό πειραματισμό. Η κατανόηση αυτής της δυναμικής μπορεί να ενημερώσει την ανάπτυξη πιο εξελιγμένων εργαλείων ανίχνευσης και να συμβάλει στη διατήρηση της ακεραιότητας του περιεχομένου που παράγεται από ανθρώπους και μηχανές. Έτσι, η μελλοντική έρευνα μπορεί να βασιστεί στα ευρήματά μας για τη δημιουργία πιο ισχυρών, αξιόπιστων και φιλικών προς τον χρήστη συστημάτων ανίχνευσης κειμένου TN.

Chapter 1

Introduction

In the current age, as the text produced by large language models (LLMs) increasingly approaches the style of human language, machine-generated text becomes more and more useful in a wide range of applications, such as news and story composition, code generation, or even domains that are essential to human society such as law [12] and education [79]. A recent survey indicated that between January 1, 2022, and May 1, 2023, the relative number of LLM-generated news articles increased by 57.3% on mainstream websites [24]. It is therefore apparent that the rapidly developing text generation capabilities of LLMs have caused them to become pivotal in many sectors of everyday life, as well as professional workflows.

However, it must also be acknowledged that as these models' availability to the public increases, so does the risk of malicious use of machine-generated text for various purposes, including but not limited to: fake news generation, fake product reviews, AI-written academic papers, online influence campaigns, online fraudulent schemes (DeepFake) [87] , spam/harassment and other potential threats. Furthermore, even if the intended use of machine-generated text might not be malicious, it has been shown that machine-generated text can be susceptible to fabrications [32], relying on outdated or wrong information, or over-relying on prompts. A comprehensive review of threat models enabled by machine-generated text is available in [11].

It is therefore essential to be able to differentiate between human and machine-generated text output, in order to be able to combat the threats posed by malicious use of LLMs. The commonly used approach is to formulate the problem of distinguishing human and machine-generated text as a (binary) classification task, where the classifier, also called a detector, will recognize and remove machine-generated text if the intent of such text is malicious. [31] This task has become a major focus point of the NLP community recently, with plenty of research put into finding the most ideal detector possible. It has been suggested that the ability of humans to effectively identify LLM-generated text is not strong enough as LLMs continue to improve, which creates a demand for AI-based machine-generated text detection. Establishing a robust mechanism to detect AI-generated text is pivotal to mitigating LLM misuse risks and fostering responsible AI governance in the LLM era [76].

The detectors currently employed for this task use a variety of methods and techniques, ranging from utilising statistical features of the input text such as word frequency, log rank or text perplexity, performing zero-shot detection in various ways, as well as fine-tuning LLMs for the classification task. In very recent literature there have been a number of different detection methodologies proposed, which, depending on the datasets and metrics used, can achieve very high performance, very close to 100% on some datasets and better than humans in almost every dataset.

However, recent research has highlighted significant robustness challenges faced by AI text classifiers, particularly in the context of paraphrasing attacks. These attacks involve a human paraphrasing text generated by an LLM in order to avoid detection, either independently or with the assistance of another system, which can also be an LLM. Research has shown that even prominent detectors, which otherwise perform well on unaltered AI-generated text, can be easily deceived by such paraphrased content, thus failing to flag it as AI-generated. [70] [28]

A less common but equally concerning type of paraphrasing attack involves making subtle edits to human-written text to trigger false positives in detection algorithms, causing them to flag the text as AI-generated. This type of attack can exploit overfitting or biases within detection models, where certain human linguistic patterns are incorrectly identified as indicative of AI-generated content. Studies have shown that these modifications can successfully deceive state-of-the-art detectors, leading to misclassification of genuine human text.

The increasing sophistication of paraphrasing attacks underscores the necessity for more nuanced and resilient AI text detection mechanisms. As the boundaries between human and AI-generated text continue to blur, the reliability of current detection models is called into question. It is imperative to develop detection strategies that are not only accurate under normal conditions but also robust against adversarial tactics such as paraphrasing. This involves enhancing the models' ability to discern subtle contextual and stylistic nuances that remain consistent despite paraphrasing.

An additional and less thoroughly explored dimension of AI-generated text detection is the issue of explainability. As previously noted, machine learning (ML) models have already surpassed human capabilities in identifying AI-generated text, and it is reasonable to anticipate that these models will continue to grow in size, complexity, and sophistication. However, this advancement comes with a significant trade-off: as models become more complex, their predictions become increasingly difficult to interpret and understand.

Given the sensitivity surrounding AI text detection, the importance of explainability cannot be overstated. For instance, consider the potential consequences if an essay or study submitted to an academic institution is flagged as AI-generated. Such a detection could carry serious implications, including questions of academic integrity and potential disciplinary actions. Therefore, it is imperative to approach the deployment of AI text detection models with caution and prioritize the development of methods to elucidate the reasoning behind model predictions.

The field dedicated to creating techniques for interpreting and explaining the behavior of complex AI models is known as eXplainable Artificial Intelligence (XAI)[59]. XAI aims to provide transparency into the decision-making processes of AI systems, enabling stakeholders to understand and trust the outcomes produced by these models.

The general-purpose goal of this work is to better understand the underlying features of the AI text detection task and how ML models rely upon them to obtain sufficient accuracy to surpass humans. Specifically, we explore adversarial perturbation-based attacks on text detection algorithms as a means to not only evaluate their robustness but also to gain insight into the specific elements that lead a detector to classify text as AI-generated or human.

Additionally, we also analyze text perplexity, a measure of the unpredictability of text, as a key metric in distinguishing between AI-generated and human texts and as an approximation of the "humanness" or "AI-ness" of a text. In particular, we examine how variations in the text perplexity distribution of the input text can affect the performance of detector models, ultimately contributing to the development of more robust and reliable AI text detection systems.

Moreover, we present a simple text perplexity-based AI text detector which performs comparably to general-purpose LLM detectors across most datasets. This finding positions text perplexity as a straightforward and interpretable benchmark for evaluating the performance of future detection methods. By establishing text perplexity as a baseline, more advanced detection methods can be evaluated more rigorously. If a method for AI text detection significantly outperforms the perplexity baseline, it demonstrates the added value of the complexity and the additional features used by the more advanced method . Conversely, high accuracy of a simple perplexity-based detector on a dataset suggests that the dataset might be too easy for more sophisticated detectors, potentially lacking in diversity or complexity. This can guide the creation of more challenging datasets that better reflect real-world scenarios where AI and human text are harder to distinguish.

To complement our technical analysis, we conduct a user survey, the main of part of which consists of the users performing the AI text detection task on various examples and explaining their thoughts. Additionally, we utilise XAI techiques such as counterfactual explanations and feature importance graphs in order to examine if such techniques can help improve human performance in the task.

The purpose of the survey is not only to assess the effectiveness of human detectors on differentiating between AI-generated and human text on various domains, but also to understand the criteria humans use to classify text as one of the two categories and to compare these criteria with those utilized by LLM-based detectors when making the same decision. By examining the alignments and discrepancies between human and model evaluations, we gain a comprehensive understanding of the detection process and the potential areas for improvement in both human and machine performance in the task.

Chapter 2

Background and related work

2.1 Text generation models

In this work we are concerned with systems that, given a textual input, attempt to classify if that input belongs to a human or a generative LLM; we will henceforth refer to those as *text detectors* or simply *detectors*. Below, we provide a general categorization of these detectors along with the related literature, and select the most suitable ones for our experimental setup.

In order to be able to effectively differentiate between human and machine-generated text, we first need to understand how LLMs generate such text. The most commonly used text generation models are unidirectional Transformer models (such as the GPT lineage, GROVER or GPT-NeoX) which perform self-supervised distribution estimation to predict the next token based on the previous ones. The probability of a given text can then be expressed as the conditional probability of the final token, given each previous token. [11]

Early attempts to generate text were using deterministic approaches, where the continuation is fully determined by the parameters given to the LLM and prefix, either by always selecting the highest probability token (called *greedy search*) or by having a fixed-size set of partially decoded sequences and selecting the one with the highest probability (called *beam search*). However, these methods depend highly on the underlying model probabilities and often result in repetitive text.[31],[26]

The state-of-the-art models, such as the GPT lineage, instead use stochastic approaches, which sample from a model-dependent distribution at each time step. There are two main strategies used for the sampling: top-k sampling, where sampling is limited to the k most probable tokens (with k fixed), or top-p (or nucleus) sampling, introduced in [26], where sampling is limited to the smallest set of tokens with a total mass above a threshold p. Thus, the number of candidate tokens varies depending on the context, which results in more fluent and coherent text. However, this sampling method can also lead to more nonfactual sentences as shown in [49].

The evolution of text generation models has seen significant advancements beyond these basic sampling methods. For instance, the introduction of temperature scaling [23] allows further control over the randomness of the generated text by adjusting the model's output distribution. Lower temperatures make the model output more deterministic, while higher temperatures increase randomness, potentially improving creativity but also increasing the risk of generating incoherent or irrelevant text.

Another significant milestone in the development of text generation models is the integration of large-scale pre-training followed by fine-tuning on specific tasks or domains. This two-step process allows models to leverage vast amounts of general knowledge during pre-training, while fine-tuning adapts the model to generate more domain-specific or task-specific text. Models like T5 (Text-to-Text Transfer Transformer) exemplify this approach, demonstrating remarkable versatility and performance across various text generation tasks. [66]

Moreover, recent developments have focused on incorporating reinforcement learning techniques to fine-tune these models based on human feedback. Reinforcement Learning from Human Feedback (RLHF) has

been applied to improve the alignment of generated text with human preferences, as demonstrated in the development of models like InstructGPT. [65] This approach utilizes human-provided data to guide the model's training, enhancing its ability to produce high-quality and contextually appropriate text.

Overall, the continuous improvement in sampling methods, the incorporation of human feedback, and the strategic combination of pre-training and fine-tuning have collectively advanced the capabilities of text generation models, making them more adept at producing coherent, contextually relevant, and high-quality text. These advancements, while enhancing the models' generative abilities, also pose significant challenges for text detectors aiming to distinguish between human and AI-generated text.

2.2 Human detection/classification of machine generated text

Before turning our attention to purely automated detection methods, it is important to note that humans can also play a role in machine-generated text detection, potentially that of overseeing an automated system and providing the human element (for example, in a social media framework where a human moderator would work together with an automatic detector to ensure malicious machine-generated content is deleted). Numerous studies and publications have reviewed the performance of human evaluators on the task of machine-generated text detection.

At the same time, several tools have been designed to enhance human capability in detecting AI-generated text. One notable example is the GLTR (Giant Language Model Test Room) tool [20]. GLTR leverages statistical irregularities in text generated by models like GPT-2 to assist human reviewers. By visualizing the likelihood of each word in a text, GLTR helps users identify patterns that are characteristic of machine-generated content, thus improving their detection accuracy. Despite its effectiveness with text generated from GPT-2, GLTR's approach faces challenges with more advanced models like GPT-3 and beyond, which utilize top-p (nucleus) sampling rather than top-k sampling. This shift in sampling methods makes it harder for human reviewers, even with tools, to detect text generated by the latest models.

The performance of human evaluators in detecting AI-generated text varies significantly. In fact, a study conducted with GPT-3 generated text [10] found that untrained humans do not perform better than random chance. However, another study where the reviewers were university students contradicts this finding and suggests that humans can still recognize machine-generated text at about 70% accuracy. [30]

The RoFT (Real or Fake text) tool [13] [14] evaluates human performance on the related task of recognizing the boundary at which a human-written text becomes machine-generated. In this tool, which is publicly available () the reviewers can also provide feedback, as to why a sentence is machine-generated. The study associated with the tool [14] also finds that reviewers do perform better than random chance on all domains. However, we have to note that machine-generated text for this tool is also provided by GPT-2, which as mentioned is significantly easier for humans to detect. This study also finds that text generated from smaller models is easier to detect for humans.

All in all, recent research has shown that while humans probably still perform somewhat better than random chance in the text detection task, as LLMs grow larger and more sophisticated, the accuracy of human reviewers is bound to decrease significantly. Human reviewers, even specifically-trained ones, still have worse accuracy than the state-of-the-art automatic detectors, which in fact perform better in the exact situations where humans are most likely to be fooled: the more "human-like" a text is to humans the more easily recognizable it is for an automated detector.[30]

Another significant challenge with human detection is scalability. As the volume of content generated by large language models (LLMs) grows, relying solely on human reviewers becomes impractical. Automated detectors offer a scalable solution, processing vast amounts of text quickly and efficiently. However, the combination of human oversight and automated detection remains essential. Human reviewers can handle edge cases, verify the decisions made by automated systems, and provide a check against potential biases in AI algorithms.

Additionally, despite the superior performance of automated detectors, human reviewers bring unique insights that can inform and improve these detectors. Humans can identify subtleties and contextual nuances that might escape purely algorithmic approaches. By analyzing the reasons behind human decisions in de-

tecting machine-generated text, researchers can gain valuable information to refine detection algorithms. For instance, patterns and anomalies that humans notice can be translated into features for machine learning models, enhancing their accuracy and robustness.

In this direction, we opted to include a user survey in this study where participants are tasked with AI text detection across various domains and contexts, with the aim to compare their performance to automated detectors, but also to extract more data on what influenced their decisions in classifying the texts.

Our aim is that the findings from this survey will not only help improve the design of detector systems in the future but also provide a deeper understanding of human cognitive processes in the context of AI text detection. Detailed analysis and results of the user survey can be found in Chapter 5. By integrating human insights with advanced detection algorithms, we aim to develop a more comprehensive and effective approach to identifying machine-generated content.

2.3 AI-based detection of machine generated text

Due to the heavy interest from the NLP community in the AI text detection task, there is a broad and diverse range of methods in which it has been attempted to use AI and machine learning techniques to differentiate between human and machine-generated text. We present the most prominent ones below.

2.3.1 Feature-based detection/classification

A feature-based approach to the detection task emphasizes identifying and leveraging specific characteristics that differentiate human from machine-generated text, often viewed as intrinsic flaws of the latter. One notable feature is the lack of syntactic and lexical diversity [18]. Machine-generated text frequently exhibits limited variety in sentence structures and vocabulary compared to human-written text. This leads to a more monotonous and predictable style, whereas human authors typically employ a broader range of sentence constructions and vocabulary, contributing to richer and more engaging content.

Another critical feature is the lack of coherence in AI-generated text. Coherence refers to the logical flow and consistency of ideas within a text, which machine-generated content can struggle to maintain, especially over longer passages. This might result in abrupt topic shifts, disjointed sentences, or contradictions that disrupt the narrative flow. In contrast, human writers generally construct coherent arguments and narratives, ensuring that their writing remains focused and logically connected. Additionally, another common issue with machine-generated text is repetitiveness. AI models might repeat phrases, ideas, or sentence structures, creating a sense of redundancy. This happens because the model may lack the nuanced understanding of context and progression that human writers possess, who tend to vary their language to avoid unnecessary repetition and maintain reader interest.

Machine-generated text can also sometimes appear aimless or generic and be characterized by a general lack of purpose. While human authors write with specific goals, intentions, and audiences in mind, AI-generated content might miss this direction, resulting in text that feels purposeless. Human-written text typically conveys a clearer sense of intention and is tailored to address specific topics or audiences.

These features—syntactic and lexical diversity, coherence, repetitiveness, and purpose—can be quantified and used to develop criteria for distinguishing human from machine-generated text without necessarily training traditional ML models. For example, the repetitiveness of n-grams (sequences of n words) can be measured to identify the overuse of certain phrases, which is more common in machine-generated text. This method used in [19] resulted in over 90% precision for top-k sampling strategies and over 80% for nucleus sampling.

Another related approach is to target the machine configuration parameters. Different modeling techniques can leave artifacts that are detectable in the generated text. [26] An example of that is token frequency: machine-generated text often does not mirror the distribution of tokens that a human-produced text has, but instead can vary depending on the chosen sampling method. Therefore, token distribution provides useful information, particularly when there is a large amount of text considered. [11]

Another method of trying to identify those artifacts is the use of stylometric features. This includes features such as average sentence length, stop word count, punctuation count and many others, which have been used

in studies to try and detect differences between human and machine-generated text especially in texts of shorter length like tweets [39], [63].

In general however, traditional feature-based methods for detecting AI-generated text have largely become obsolete, particularly when dealing with advanced models and sophisticated sampling techniques. These methods were more effective against earlier generation models that used top-k sampling, which tended to over-generate common words and thus made detection easier. However, as language models have grown larger and more advanced, detection has become significantly more challenging. Text generated by large language models (LLMs) that are fine-tuned for specific domains tends to be more human-like and harder to detect than text produced by general-purpose models. [75]

The widespread adoption of top-p (nucleus) sampling in modern generative LLMs has further diminished the effectiveness of these detection methods. Top-p sampling generates text that is more varied and contextually appropriate, reducing the likelihood of the repetitive patterns and lexical limitations that feature-based methods rely on. Consequently, these traditional methods have been surpassed in both accuracy and robustness by more sophisticated detection techniques that involve advanced machine learning models.

Despite their limitations, traditional feature-based methods still hold some value. They are relatively easy to use and experiment with, providing a useful baseline for evaluating the performance of more complex detectors. Ensuring that advanced detection methods perform at least as well as these simpler approaches helps validate their efficacy. In some cases, feature-based methods can even be combined with more advanced tools such as a transformer model to increase accuracy [55]. Moreover, stylometric analysis is still being used even in very recent studies, especially in cases where methods based on large language models might not be as effective, such as very short texts (like tweets).

Thus, while no longer the cutting-edge solution, feature-based methods remain a valuable tool in the ongoing effort to detect AI-generated text.

2.3.2 Zero-shot detection/classification

In this setting, a pretrained text generative LLM (such as GPT-2 or GROVER) is employed without fine-tuning to detect generations from itself or similar models. The original idea, as demonstrated in [75], was to use total log probability and apply a threshold based on this for classification. Text was predicted to be machine-generated if the overall likelihood of the text according to the model was closer to the mean over all machine-generated text than the mean over all human texts. However, classifiers based on this did not match the standards of even traditional feature-based methods.

A more recent study, following the release of OpenAI's ChatGPT, suggested an innovative methodology for zero-shot detection. DetectGPT [57] operates on the principle that machine-generated text often lies in negative curvature areas of the model's log probability. Therefore, it generates multiple perturbations of the considered text, and scores it as machine-generated if those minor rewrites of the text have lower log probability than the original text. At the time of its release, DetectGPT significantly outperformed the best traditional zero-shot methods and was comparable to some of the state-of-the-art fine-tuned LLM classifiers [57].

However, DetectGPT has been found to have several shortcomings. [8] First, it relies on knowledge of the specific LLM used to generate the text, as different LLMs exhibit different likelihoods for various words or tokens depending on their training data. Experiments with different LLMs have shown that DetectGPT cannot accurately detect generated text if it is unaware of the generative model that produced it. In a realistic scenario, one would need to compare with all existing LLMs, which is impractical. Other issues with DetectGPT include its intensive computational costs [3] and its inherent vulnerability to various kinds of paraphrasing attacks (explained more thouroughly in Section 2.5), where the LLM user slightly paraphrases the text in order to deceive the detector. Provided the paraphraser is good enough, the original LLM-generated phrase will show up in the algorithm's perturbations, thereby deceiving DetectGPT into classifying the input as human.

Despite these problems, DetectGPT has inspired further research into zero-shot detection. Many improvements on the base algorithm of DetectGPT were suggested, including making the perturbation step more efficient using sampling [3] and incorporating a Bayesian surrogate model in the sampling process [54]. These

methods generally lead to some improvement over the base algorithm, but key problems, such as robustness issues and reliance on knowledge of the generating LLM, remain. Consequently, the practical use of DetectGPT-based algorithms is limited, and they are now primarily used as baselines for other detectors to compare against.

Of course, zero-shot methods in general have significant advantages, like the ability to be deployed instantly without the need of training data and their generalizability to different (even unknown) models. Therefore other methods of zero-shot detection have also been explored in recent literature, including leveraging log rank information [77] and using intrinsic dimensionality estimation of the text embedding manifolds [81], which show promising performance in early experiments but have yet to be tested in practical situations.

Another subcategory of zero-shot detection methods involves utilizing text perplexity analysis. Text perplexity is a metric that measures the unpredictability of text given the context, reflecting how well a language model predicts a sample. In essence, lower perplexity indicates text that is more predictable and aligned with human writing, while higher perplexity suggests text that is less predictable and more likely to be machine-generated.

Text perplexity has shown to be a reliable indicator of whether a text is AI-generated in many cases. Perplexity-based detectors work by calculating the perplexity of a given text using a language model and comparing it against threshold values derived from known human and machine-generated texts. If the perplexity of the text falls within a range typical of machine-generated content, it is classified as such. More about text perplexity is explained in Section 2.4.

Text perplexity-based detectors are considered some of the best performing among zero-shot detectors. In this work, we introduce our own simple text perplexity based detector and show that it can perform as well as the best fine-tuned models for text detection in most domains, thereby highlighting the potential of zero-shot detection methods to remain a reliable solution for AI text detection as language models continue to evolve.

2.3.3 Fine-tuning LLMs for classification

In this setting, a pretrained language model, typically a bidirectional one such as BERT or RoBERTa, is fine-tuned to detect text generation from itself or similar models. Unlike zero-shot setups, this approach necessitates further supervised detection examples for training [31].

Early studies involving BERT and GROVER concluded that such models excel at detecting their own generated text but are less effective at identifying text generated by other models [83].However, in [75] it was demonstrated that fine-tuning the RoBERTa detector on top-p examples can also yield high performance across other sampling methods, achieving a 95% accuracy in identifying GPT-2 model generations. This performance surpassed even the fine-tuning of the unidirectional GPT-2 model itself, contradicting earlier findings. Generally, the RoBERTa-based detector performs well across various domains and was considered state-of-the-art at the time of its release. Nonetheless, a significant drawback is its requirement for a substantial amount of training examples per class to achieve optimal performance [31].

The scalability of these fine-tuned detectors to newer language models remains a pertinent issue. For instance, OpenAI fine-tuned a GPT model for text detection in 2023, but it was eventually removed due to its low accuracy [64]. However, despite the advent of newer models such as ChatGPT, RoBERTa-based detectors continue to exhibit superior accuracy in specialized sub-tasks like fake-news detection [86], and in specialized domains such as academic papers [47] or homework exercises [46].

A major concern with fine-tuned detectors is their extreme specificity. Fine-tuning a detector on a particular domain or dataset often results in compromised performance outside that domain. This trade-off between domain-specific accuracy and generalizability is crucial. In our experiments, we utilize a RoBERTa-based detector as one of our main detectors to represent the fine-tuning method, alongside smaller BERT-based detectors which we fine-tune on specific data sets for preliminary experiments. This approach demonstrates the trade-off introduced by the extent of fine-tuning: a model fine-tuned on the detection task without specific domain constraints is expected to exhibit better generalizability, whereas fine-tuning a model on both the task and a particular domain is likely to increase accuracy within that domain but decrease it outside of it.

Another critical consideration when employing these detectors in realistic scenarios is their robustness. Given

the open-source nature of the BERT/RoBERTa architecture, malicious actors can access the model weights and understand how these change with various inputs, potentially making the models susceptible to manipulation. In our study, we employ adversarial attack tools to demonstrate that even fine-tuned detectors can be easily deceived into producing incorrect outcomes if the text is paraphrased with the intention to fool the detector.

Despite these challenges, fine-tuning an LLM remains one of the most prominent methods for AI text detection, particularly when high accuracy is required within a specific domain of texts, provided that sufficient data is available to train the model effectively. This method's ability to adapt to specific characteristics of a domain enables it to achieve superior performance compared to more general approaches, making it a vital tool in the ongoing effort to detect AI-generated text accurately and efficiently.

2.3.4 Adversarial learning methods

The aforementioned robustness issues of AI-generated text detectors against paraphrasing attacks have inspired the development of novel approaches based on adversarial training. Frameworks such as RADAR [27] and OUTFOX [37] attempt to construct robust AI text detectors by employing an adversarial training paradigm. In these frameworks, two models are trained in parallel: a detector, whose objective is to identify AI-generated text, and a paraphraser, whose goal is to generate text that deceives the detector into making incorrect predictions. The detector and paraphraser are trained separately and iteratively, following classic adversarial learning practices.

Adversarial learning methods enhance the robustness of AI text detectors by simulating attack scenarios during training. The paraphraser model generates challenging examples that the detector must learn to handle, thereby improving its ability to withstand real-world adversarial attacks. These methods are designed to address not only word substitution attacks (paraphrasing) but also more sophisticated instructional prompt-based attacks [74]. For instance, [90] leverage human texts polished by ChatGPT to train a RoBERTa-based detector, enhancing its accuracy and robustness against paraphrased text.

Experimental evaluations of adversarial learning methods typically show that models trained using these techniques retain state-of-the-art accuracy on original detection tasks while demonstrating significantly improved robustness against paraphrasing attacks compared to traditional fine-tuning methods. This increased resilience is a key advantage in practical applications, where adversarial attacks are a common threat.

In our research, we conduct extensive experiments with adversarial attacks, which are similar to paraphrasing. Consequently, it is imperative to include an adversarial learning-based detector in our study to assess its effectiveness in withstanding such attacks. We therefore select the RADAR detector for inclusion in our experiments, as it represents a versatile and general-purpose adversarial-based detection framework.

This approach underscores the evolving landscape of the task, where adversarial training emerges as a promising solution to address the vulnerabilities of conventional methods. As adversarial attacks become increasingly sophisticated, the integration of adversarial learning into detection frameworks will be crucial in developing resilient and reliable AI text detectors.

2.3.5 Other attempts

As Large Language Models (LLMs) become increasingly proficient at generating human-like text, the difficulty of accurately detecting AI-generated content escalates. A proposed solution to facilitate this detection task for the community, most prominently detailed in [35] involves watermarking the text produced by LLMs. This method embeds hidden markers within the generated text. These markers are designed to be imperceptible to human readers but can be detected by specific algorithms, thus identifying whether a text was generated by a particular AI model. The primary objective of watermarking is to provide a reliable and easily detectable signature indicating the origin of the text, thereby aiding in the identification and regulation of AI-generated content.

The method proposed in the original paper categorizes tokens into "green" and "red" lists using a random number generator. Watermarked models are more likely to select tokens from the green list, resulting in a watermark that remains undetectable by human readers but is easily identifiable by an automated detector with knowledge of the list, even after some degree of paraphrasing.

This attempt opens up unexplored avenues for the AI text detection task, although it is still far from being a realistic scenario. As of now, text-generating models like ChatGPT do not typically employ watermarking. There are several reasons for this, including the technical challenges presented by the need of an undetectable and robust watermark, concerns that watermarking will decrease the fluency and performance of text generation models, and ethical implications involving the potential misuse of watermarked texts. Additionally, the practical deployment of watermarking in widely used models requires significant coordination and standardization efforts.

While watermarking is not widely employed in current AI text generation models, it remains an active area of research, with ongoing efforts aimed at developing more sophisticated algorithms for embedding and detecting watermarks that are robust against a wide range of text manipulations. For example, a follow-up study to the original watermarking paper examines the robustness of watermarked text after being rewritten by humans, paraphrased by a non-watermarked LLM, or incorporated into a longer handwritten document [36]. Additionally, [45] proposes a publicly verifiable algorithm using two different neural networks for watermark generation and detection, addressing concerns about counterfeiting the watermark during public detection and potential security breaches from having only one key.

In summary, watermarking is a promising technique for AI text detection, offering a potential solution to the growing challenge of distinguishing human-generated text from AI-generated content. However, it faces significant technical and practical challenges. Current models like ChatGPT do not typically employ watermarking, but ongoing research and development may lead to more widespread adoption in the future. Such advancements could provide an additional layer of security and traceability in AI-generated text, ultimately contributing to more effective regulation and identification of AI-generated content.

2.4 Text perplexity

A commonly used metric in NLP for evaluating language models is text perplexity (PPL). This metric quantifies how effectively a language model can predict a given text sample. Originating from the field of information theory, perplexity measures the uncertainty of a prediction model when provided with the actual outcome [85]. In this context, a lower perplexity score indicates that the prediction model can predict the text sample well, whereas a higher perplexity score suggests that the sample is more "unpredictable" and thus more challenging for the model to anticipate.

When applied to Large Language Models (LLMs), the concept of text perplexity is adapted to measure the average uncertainty of an LLM when predicting the next word or token based on the preceding sequence of words or tokens. This metric is particularly relevant to classical (auto-regressive) language models such as GPT-2 or OPT, where the prediction of future tokens is based on the context provided by the preceding tokens. In this framework, a lower perplexity score signifies that the text sample is more "predictable" to the LLM, suggesting that it is more likely to have been generated by the LLM (or a similar model). Conversely, a higher perplexity score implies that the text is more "unpredictable" to the LLM, indicating a higher likelihood of being generated by a human.

It is important to note that the applicability of perplexity as a metric is inherently tied to the architecture of the language model in question. For auto-regressive models, perplexity provides a clear measure of how well the model can predict the continuation of a text sequence. However, for bidirectional models such as BERT or RoBERTa, the concept of perplexity is potentially less well-defined [29]. Bidirectional models do not predict the next token in a sequence in the same manner as auto-regressive models; instead, they consider the entire context of the sentence or paragraph simultaneously, making the direct application of perplexity less straightforward.

By quantifying the predictability of a text sample, perplexity offers valuable insights into the model's capability to generate human-like text and aids in distinguishing between AI-generated and human-generated content. This metric's relevance and applicability are contingent on the architectural nuances of the language models under consideration.

Formally, perplexity is defined as the exponentiated average negative log-likelihood of a sequence. In the NLP context we can assume a sentence as a tokenized sequence X of length t, consisting of t tokens or words.

$$X = (x_0, x_1, \ldots, x_t)$$

Then, we can define the perplexity of X with the following formula :

$$\mathrm{PPL}(X) = \exp\left\{-\frac{1}{t}\sum_{i=1}^{t}\log p_\theta(x_i \mid x_{<i})\right\}$$

where

$$\log p_\theta(x_i \mid x_{<i})\}$$

is the conditional log likelihood of the token i given all previous tokens in the sentence. This is also equivalent to the exponentation of the cross-entropy between the data and model predictions. [29] Of course, using this formula means that the tokenization process has a big impact on perplexity calculations and should always be taken into consideration.

Perplexity has been frequently employed as a comparative measure for evaluating model perfromance across various tasks. More recently, it has also been suggested that it could be a good indicator of whether an AI has generated a particular text segment, on the premise that a lower perplexity score implies a higher likelihood of the model generating the text[20]. There have been some studies in the recent literature that utilise this in order to create an AI text detector. A good example is HowkGPT [85], which utilises perplexity analysis to identify homework assignments generated by ChatGPT. Perplexity is also being used by commercially available, closed-source detectors such as GPTZero [80], which also uses the related metric they call *burstiness*, which is a metric referring to AI-generated text displaying a higher frequency of clusters or bursts of similar words or phrases within shorter sections of the text.

A more advanced implementation of perplexity in the AI text detection task is the Binoculars detector [25], which computes the log perplexity of a text using an LLM and then compares it to the perplexity of another LLM's choices when completing a text sample (called *cross-perplexity* in their study) . If the text is written by a machine the two perplexities are expected to be similar, whereas if it is from a human source then the two metrics should differ significantly. Using this principle to build a zero-shot detector, they achieve good accuracy on a number of datasets and a very high TPR (true positive rate) when set to a really low FPR (false positive rate).

In this work we do attempt to look into the possibility that text perplexity could be an underlying feature that makes models, which do not directly employ perplexity analysis, classify a text as human or AI-generated, and therefore serve as an indicator of what makes a text intrinsically human or generated. We do construct a perplexity-based detector (similar to HowkGPT), and evaluate its performance across a number of datasets, comparing and contrasting with detector performance to determine the usefulness of perplexity in the AI text detection task. We also do run a number of datasets through the Binoculars detector [25] to compare it both to the simpler perplexity baseline and to other detectors tested on the same dataset.

2.5 Paraphrasing attacks

It is important to note that even state-of-the-art AI text detectors can be vulnerable to various attacks, where a malicious actor manipulates the input or the detector to misclassify AI-generated text as human-generated (or, less commonly, vice versa). The most prevalent of these attacks is the classic paraphrasing attack. In this scenario, a lightweight neural network model, known as a paraphraser, is applied on top of a Language Model (LLM) with the specific goal of evading detection. Recent research, such as [70], demonstrates that such a setting can evade a broad range of detectors, including watermarking schemes, neural network-based detectors, and zero-shot detectors like DetectGPT. They also mathematically prove the impossibility of the detection task when the total variation norm between human and machine-generated text distributions is small, thus providing an upper bound for the detector's efficiency.

In [38] it is also argued that a lightweight paraphraser can bypass various detection schemes. Therefore, they propose retrieving semantically similar text generation by an LLM as a means of defense against paraphrasing attacks. However, this method has limitations, such as being specific to each LLM and raising privacy

concerns. In [9] it is argued that while the impossibility result of [70] holds,it is insignificant in the broader context. They assert that text detection will always be possible given sufficient data samples, although another study [8] argues that methods of text detection will always be susceptible to tampering.

It should be noted that paraphrasing can also be performed by humans. For example, a student might generate an essay for homework using an LLM like ChatGPT, which would likely be flagged as AI-generated by a detector, and then edit it to add context, improve style, and clarity. There is ongoing debate about whether this kind of text should be flagged as AI-generated. Such text occupies a hybrid space between purely AI-generated text and human text and can potentially be viewed as either. Most studies on AI text detection consider such text to be human-generated, so the primary focus of research on paraphrasing is on LLMs paraphrasing content to appear as human text. Cases where human text is paraphrased to appear AI-generated are less common and have fewer practical implications. However, it is crucial to distinguish between human paraphrasing of human text (which ideally should not deceive a detector) and an LLM editing human text (which might occupy the hybrid space).

In our work, we explore adversarial attacks against detectors, which function similarly to paraphrasing. Our experimental results indicate that state-of-the-art neural network-based detectors are vulnerable to paraphrasing, as their classification labels can be altered with minimal text perturbation. However, for some detectors, such perturbations may degrade the text quality, making them impractical in real-world scenarios and thus rendering the detector robust in practice. We also experiment with perplexity analysis to examine how paraphrasing (simulated through adversarial attacks) impacts text perplexity distribution, which could represent the distribution of texts discussed in [70].

In practice, AI text detection and AI text paraphrasing to avoid detection are two sides of the same coin, with many studies and commercially available software dedicated to one task or the other. In this work, we emphasize the importance of identifying the intrinsic qualities that differentiate AI-generated text from human-generated text without necessarily focusing on developing the most accurate detector or paraphraser.

2.6 Explainable AI methods

The main focus of this work is understanding the deeper mechanisms beyond how texts are classified as AI-generated or human. As such, the study is also related to the concept of interpretability and Explainable AI (XAI) methods. Explainability in AI refers to the ability to make the decision-making processes of AI systems transparent and understandable to humans. This involves providing clear, human-readable explanations for why a model made a particular prediction or classification, making it easier to trust and verify the system's outputs.

In the context of AI text detection, explainability is crucial. It allows researchers and users to understand why a certain text was classified as AI-generated or human. This transparency is essential for several reasons: it can help improve the model by revealing biases or weaknesses, it fosters trust in automated detection systems by making their operations comprehensible, and it provides valuable insights into the features and patterns that differentiate human text from AI-generated text.

Moreover, explainability is particularly important in cases where the detection system is used in high-stakes environments, such as academic integrity checks, legal document verification, or news authenticity assessments. Here, being able to provide a clear rationale for each classification decision can be crucial for accountability and user acceptance.

Explainable AI methods can be classified according to various criteria. The most commonly used taxonomy, as outlined in sources such as [58] and [21], distinguishes between two primary approaches: interpretability achieved by the model itself, and interpretability achieved through post-hoc analysis.

Interpretability achieved by the model itself involves restricting the complexity of the model to ensure that its operations are inherently understandable. These models are designed to be transparent and easily interpretable from the outset. Examples include decision trees, linear regression, and rule-based systems, which provide straightforward, human-readable explanations of their decision-making processes.

On the other hand, post-hoc interpretation involves analyzing the model after the training and classification processes are complete. This approach is particularly important for black-box models, such as deep neural

networks, which achieve state-of-the-art results in many machine learning tasks, including AI text detection, but whose internal workings are not easily understood. Post-hoc methods can include techniques like feature importance analysis, visualization of attention mechanisms, and generating textual or visual explanations that describe the model's behavior, including counterfactual explanations. These methods can also be applied to inherently interpretable models to provide additional insights.

Given that black-box models typically the highest accuracy and effectiveness in the Ai text detection task, the main focus of this work regarding explainability is on post-hoc interpretation. Understanding and explaining the decisions of these complex models is crucial, especially in sensitive applications like our task, where the stakes are high, and trust in the system's outputs is essential.

2.6.1 Interpretable Models

Naturally, the easiest way to achieve interpretability is to implement it intrinsically into the model algorithms. Some of the most common interpretable machine learning algorithms include linear or logistic regression, decision trees, and rule-based systems. These models are designed to provide clear, understandable outputs that can be easily traced back to the input features. When applied to the AI text detection task, as examined in the first part of this study, basic feature-based detection methods, explained in Section 2.3.1, are usually inherently explainable. These methods leverage straightforward, human-understandable features such as lexical diversity, syntactic patterns, and text coherence, making their decision-making processes transparent.

Perplexity-based analysis, which underpins some AI text detectors such as [80], [85], and [25], occupies a middle ground between fully interpretable models and black-box models. Perplexity is a mathematical, quantitative metric that measures how well a language model can predict a text sample. In this sense, a perplexity-based detector is more interpretable than black-box classifiers like a fine-tuned RoBERTa model, as it provides a clear numerical indication of the model's uncertainty regarding a text. However, calculating perplexity requires intrinsic knowledge of the LLM used to compute it, as it depends on the probabilities assigned to tokens by the model. While any LLM, including older open-source models like GPT-2, can be used to compute perplexity, the metric itself is not fully interpretable. The threshold for classifying a text as human or AI-generated can vary depending on the style and context of the text, making it less straightforward.

As noted in Section 2.3.1, although feature-based methods might be sufficient to accurately detect text generated by smaller or older LLMs, they usually lag behind "non-interpretable" methods that utilize advanced ML models in terms of accuracy. Additionally, there is a distinct weakness in using interpretable models for the AI text detection task concerning paraphrasing. If a malicious user fully understands how a detection model works, they can manipulate the text to deceive the model and evade detection.

In conclusion, while intrinsically interpretable models and methods offer clear advantages in terms of transparency, their limitations in accuracy and vulnerability to manipulation necessitate the use of post-hoc interpretability techniques for robust AI text detection.

2.6.2 Post-hoc interpretation

Post-hoc interpretation aims to separate the explanation from the model itself, which can have significant advantages [68]. The primary benefit of post-hoc methods is that they are model-agnostic; they are independent of the model used to make a classification and rely solely on the input and output of said model. This independence allows us to use the most accurate methods available for a given task, without being constrained by the need for inherent interpretability. As machine learning models become larger and more diverse, relying on white-box methods that require intrinsic knowledge of specific models will become increasingly impractical. The flexibility offered by model-agnostic methods is particularly advantageous in an industry dominated by black-box models.

Post-hoc interpretations can be further categorized into global and local explanations. Global methods describe how certain features affect a model's predictions on average, providing a broad understanding of the model's behavior. Local methods, on the other hand, focus on specific predictions, offering detailed insights into individual decisions [58]. An example of a global method is using an interpretable model, often referred to as a surrogate model, to approximate the predictions of a black-box model. This approach helps

to provide an overall understanding of the model's decision-making process. Another example of a global explanation method is the utilization of knowledge graphs to generate rule-based explanations [42],[43],[51] where the rules are derived from the model's decision logic and contextualized using domain knowledge. Additionally, global feature importance methods, such as SHAP [48] can offer insights by showing the average impact of each feature on the model's predictions, further enhancing the interpretability of complex models. Another approach to global explanations is prototype explanations, where recent works [53] utilize semantic descriptions of data, to select prototypical data points that are representative of each class, offering a better understanding of the distribution of both the data and the model's output.

Local methods aim to explain individual predictions of the black-box model. One popular local method is LIME (Local Interpretable Model-agnostic Explanations) [67], which generates a dataset of perturbed samples around a specific data point and trains an interpretable model on this dataset. The trained model should approximate the black-box model's behavior locally, producing feature importance graphs that can be interpreted by human researchers. However, the effectiveness of this method is contingent on the accuracy of the surrogate model; if the surrogate model does not accurately mimic the black-box model, the local explanation may not be reliable. Despite this limitation, LIME and other feature importance techniques remain valuable tools for enhancing the interpretability of complex models.

In the context of AI text detection, post-hoc interpretability methods can be particularly useful. For example, [1] used feature importance analysis to identify the key factors that AI text detectors use to classify texts. The study found that perplexity was largely the most significant feature the classifiers used. However, as previously discussed, perplexity is not inherently an interpretable feature, necessitating additional methods to enhance transparency. In our work, we employ LIME [67] to generate word importance graphs, which highlight the words in a text that most influenced the classifier's prediction. This approach aims to provide clearer insights into the specific parts of a text that drive the model's decisions.

Counterfactual explanations and adversarial attacks

Another family of post-hoc interpretations that can be used to explain a model is counterfactual explanations. These explanations emphasize what should be different in the input to change the model's output (the prediction). Formally, a counterfactual explanation of a prediction describes a change to the feature values that alters the prediction to a predefined output.

Counterfactual explanations have numerous advantages as an explainability method. Firstly, they are completely model-agnostic and can be employed in any system that has an input and an output, even systems that do not use ML methods at all. Additionally, counterfactual explanations are very human-friendly due to their contrastive nature.[44] ,[56]. Applying this to our task, humans potentially seeking an explanation of why a text is AI-generated (or not) will think contrastively, considering "what should have been different for this text to have been classified as human (or AI-generated)".

Given these advantages, it is natural that the NLP community has attempted to employ these methods in practice using text-generating LLMs. Frameworks such as [69] and [6] attempt to perform counterfactual edits by masking parts of the text and optimizing the proposed replacements to change the output of a given predictor, which in our case is an AI text detector. A more general approach to counterfactual edits in text-to-text models is Polyjuice [89], which identifies perturbations that can change the general semantics of a sentence without necessarily targeting a specific predictor.

Counterfactual explanations can have various use cases. In [50], three basic use cases are examined: outcome fulfillment, where the end-user seeks advice on how to modify an input to achieve a desired output; system investigation, where the user seeks to understand model behavior; and vulnerability detection, where the end-user seeks to identify potential weaknesses in a system.

The use case of vulnerability detection is closely related to adversarial attacks. There is a family of editors that aim to generate adversarial examples to improve ML models. These perturbations of text are usually used to identify potential robustness issues with the models and/or inherent vulnerabilities. The difference between adversarial attacks and counterfactual explanations is that adversarial attacks do not necessarily edit inputs minimally or fluently, which can result in unwanted features in the perturbations (noise) [16]. An example of an adversarial attack framework is TextFooler [33], which is considered the most effective (in

terms of flip-rate) and utility-preserving (in terms of semantic content preservation) adversarial editor, and is also the one we choose for this work.

Specifically in the AI text detection task, there have been recent papers leveraging adversarial attacks, running concurrently with this work. [5] uses a simple adversarial attack of inserting a space in a text before a random comma and successfully manages to flip the prediction of many AI text detectors. While this highlights the robustness issues some AI text detectors may have, it does not provide any explanations as to the underlying mechanisms that make a detector classify a text as AI-generated or human. A more recent study [28] examines adversarial attacks on AI text detectors more deeply and proposes a reconstruction network for additional robustness against them.

Our work utilizes adversarial attacks via the TextFooler framework, as explained in Section 3.1. Using them in different circumstances, we attempt to fulfill different use cases for these explanations: highlighting potential robustness issues, understanding the way detectors make predictions on a deeper level, and providing explanations to humans. This is why, in the final part of the study, a user survey is conducted where users are asked to perform the AI text detection task, both unaided and aided by counterfactual explanations as well as LIME explanations. One purpose of this work is to identify whether such methods could improve human performance in detecting AI-generated text.

2.6.3 Hybrid attempts

In addition to the methods mentioned earlier, there have been recent attempts in the literature to combine different explainability methods to provide better insights into black-box ML models. One such example is found in a recent work, [4], which enhances interpretability by combining local feature importance, as used in LIME [67], with textual counterfactual explanations. This hybrid approach aims to make the explanations more intelligible and comprehensive.

The method involves assessing token change importance between the instance to be explained and its counterfactual sample. By evaluating which token changes are most influential in altering the prediction, this approach provides a nuanced understanding of how individual features contribute to the model's decisions. This combination of LIME and counterfactual explanations helps bridge the gap between local and global interpretability, offering a more detailed and user-friendly explanation of the model's behavior.

In summary, understanding the deeper mechanisms behind how texts are classified as AI-generated or human is vital for improving the transparency and reliability of AI systems. The integration of interpretability and Explainable AI (XAI) methods plays a crucial role in achieving this goal.

Post-hoc methods like LIME and counterfactual explanations provide significant insights into model behavior by explaining individual predictions and suggesting minimal changes to inputs that could alter outcomes. This not only aids in understanding and improving model performance but also enhances user trust by making AI decisions more transparent and comprehensible.

By incorporating these methods in our work, we aim to enhance both the detection capabilities and the interpretability of AI text detectors. The user survey conducted in this study, analyzed in Chapter 5, further explores the effectiveness of these methods in aiding human performance in detecting AI-generated text, underscoring the practical benefits of explainable AI techniques.

Chapter 3

Methodology

As outlined in previous sections, the task of AI text detection remains an open problem for the NLP community. This challenge involves achieving high accuracy in detecting AI-generated texts, a necessity given the significant societal impacts of this task. Effective AI text detection has applications in various areas such as academic integrity, legal document verification, and news authenticity, making precision and reliability essential.

To address this multifaceted problem, we conducted a series of experiments aimed at testing the accuracy and robustness of various detection techniques. Our goal was to explore both traditional and state-of-the-art methods, evaluating their effectiveness under different conditions. Additionally, we sought to understand the intrinsic differences between human and machine-generated texts, aiming to uncover how models identify and interpret these distinctions.

In this chapter, we provide a comprehensive outline of the methodologies, frameworks, detectors, and datasets we utilized in our experiments. In the following chapter (Chapter 4), we present the results of our experiments, highlighting key findings and insights. This includes an analysis of the performance of different detectors and the impact of various text features on detection accuracy. Through this detailed examination, we aim to contribute valuable knowledge to the field, advancing the understanding of AI text detection and providing a foundation for future research.

3.1 Adversarial attack framework-TextFooler

A substantial part of our experiments is centered around an adversarial attack framework, specifically TextFooler [33]. Adversarial attacks, as an explainability method, can be linked to counterfactual explanations (see Chapter 2.6). In our specific task, they provide valuable insights into why a text is classified as human-written instead of AI-generated, and vice versa.

The adversarial attack framework we use in the experiments is TextFooler, implemented through the TextAttack Python module [60]. We employ this framework with various datasets and language models, which are outlined in later chapters.

The basic premise of TextFooler involves perturbing various words in a sentence with synonyms using a word embeddings list until the label flips. Originally, this framework was tested in sentiment analysis tasks, where it could, for instance, turn a positive review into a negative one, providing the user with insights into what words would need to change for the "feeling" of the sentence to differ. However, as the authors of TextFooler note, the framework can be adapted to "fool" any text-to-text classifier. We leverage this capability to fool AI text detectors.

The parameters for our TextFooler experiments are the default settings used in the TextAttack module, which can be found in their documentation here. Specifically, the word embeddings used are counter-fitted PARAGRAM SL999 vectors, and the words are swapped with their 50 closest neighbors in those embeddings, provided they are the same part of speech (or nouns with verbs).[1] The TextAttack module employs the Universal Sentence Encoder [7] with a minimum angular similarity of ϵ =0.5. The goal is set to "Untargeted Classification," and the search method is "GreedySwapWordWIR," which uses word importance ranking to swap words greedily.

Since TextFooler was not designed with explainability as its primary purpose, it might generate perturbations that lead the detectors to out-of-distribution texts to fool them. We acknowledge that a similar result could have been achieved using a more sophisticated counterfactual framework, such as MiCE [69], or an LLM-based paraphraser like DIPPER [38]. However, we chose TextFooler due to its simplicity, open-source availability, compatibility with custom classifiers, and its light resource requirements, given our limited computing resources.

Despite its limitations, we believe TextFooler provides a solid baseline to simulate more advanced adversarial or counterfactual frameworks for our purposes. The primary goal of this experiment is not to fool detectors in the most precise manner possible but to demonstrate that state-of-the-art detectors can be fooled and to gain insights into how this can be achieved. Therefore, this should be considered more of a proof-of-concept baseline experiment, upon which future work and studies can expand.

3.2 Models used

Our research is focused on explainability of black-box AI text detectors. For the reason, we opted to use the most commonly referenced state-of-the-art models for the task, based on our review of the relevant literature (see Chapter 2). We choose RoBERTa [75] as it has been the detectors that has gotten the highest accuracy scores consistently among fine-tuned AI text detectors, and the recent RADAR [27] as it seems to achieve the highest robustness among all AI text detectors, while also representing a novel method in the task (adversarial training). The version of RoBERTa we use is roberta-large-openai-detector, which has been fine-tuned on GPT-2 text. The version of RADAR we use is RADAR-Vicuna-7B, as it is the only available open-source version.

In addition to those two models, for some of our experiments we also fine-tune a language model on some of our specific datasets each time, in order to attain higher accuracy in case the specifics of the dataset prevent the general purpose classifiers from that. Since the BERT lineage has proven to be the best when it comes to AI text detection, we use a lighter model of this lineage, DistilBERT[71], in order to create 3 fine-tuned models. We denote those DB-1 (trained on the AuTexTification dataset [72]), DB-2 (trained on the Human

[1] We initially attempted to modify the constraints to disallow word-noun substitutions in hopes of increasing the coherence of the perturbed text. However, the difference was negligible, and we preferred to keep the default settings for simplicity and reproducibility.

vs. ChatGPT dataset [1], and DB-3 (trained on the GPT Classification dataset, sentence level [62]). In order to train the models, we use TextAttack [60]'s train module, with a max sequence length of 300 (so that all the data fit into this) ,a batch size of 128, and 3 clean epochs for the model to be trained. Despite keeping both the size of the model and training time to a minimum there is noticeable difference in performance between models fine-tuned on these specific datasets and general-purpose models, which was what we attempted to showcase by training these models.

Furthermore, we also run some samples of our datasets on the Binoculars detector [25] in order to test its performance against the fine-tuned models. While not being a LLM classifier per se, the Binoculars detector does utilise two other state-of-the-art LLMs and their perplexity calculations to make its prediction. The version of Binoculars we use is the one available on HuggingFace Spaces, and we choose the "high accuracy" mode (designed to maximise the F1 score on their study and therefore the highest accuracy). Finally , we also create a simple perplexity-based detector, which is described more in depth in Section 3.4.

3.3 Datasets used

The majority of works related to AI text detection use piles of human text from datasets originally designed for other tasks such as Wikipedia[17] or bookcorpus[92] and then use prompting with an LLM of their choice (usually of the GPT lineage) to modify them, thereby generating the machine-generated text [88]. That essentially means that each study on AI text detection generates a new dataset, making it very hard to compare the performance of detectors in the task in general rather than on the specifics of each dataset. In our work we have chosen between existing and easily available datasets as a way for our experiments to be reproducible, since there are endless possibilities of adversarial attacks combining various datasets and models. We therefore attempt to use open-source datasets from HuggingFace or Kaggle. We use the datasets outlined below for our experiment: For all datasets, we use a 80%-10%-10% allocation for train, validation and test splits, where applicable.

1. **AuTexTification dataset:** The AuTexTification [73] (Automated text Identification) task was part of IberLEF 2023, the 5th Workshop on Iberian Languages Evaluation Forum. The task was organized by Symanto and Universitat Politecnica de Valencia as a competition for AI text detection with two subtasks, one being the binary classification task we examine and the second being an author attribution task, where each participant had to assign a label to a text corresponding to either human-generated text or a particular LLM from a choice of six. Both tasks were available in English and Spanish, and the data was made available following the conclusion of the competition[72]. We utilize the detection_en/train subset of the dataset, which contains about 33.800 texts, with 50.4% of them being human and the rest being AI- generated. The texts come from various sources such as tweets, legal documents and wiki articles. The average length of a text in this dataset is 54.3 words, with a standard deviation of 29.

2. **Human vs. ChatGPT (HvC) dataset:** This dataset was created for the research article Detecting AI Authorship: Analyzing Descriptive Features for AI Detection [1], with the human text taken from the arXiv database managed by Cornell University, which is available as a dataset on Kaggle[78]. The AI-generated text was created by OpenAI's ChatCompletion GPT-3.5-turbo v. 0.27.6 model, with a temperature of 0.7. The texts come from academic papers' abstracts. The full raw dataset, with 4051 texts (51.8% of them being human and the rest AI-generated) is available on Kaggle. [2] The average length of a text in this dataset is 126.6 words, with a standard deviation of 43.5.

3. **GPT Classification (GPT-Class) Dataset:** This dataset, created for [62] consists of textual articles including terminology, concepts and definitions in the broader computer science field. Human-generated text was collected from different computer science dictionaries and encyclopedias including "The Encyclopedia of Computer Science and Technology" and "Encyclopedia of Human-Computer Interaction". AI-generated text was then produced by prompting OpenAI's ChatGPT and manually documenting the resulting responses. Then, in order to evaluate the performance of detectors in shorter, sentence-level data points, each article was divided into its sentences and was labeled accordingly. The sentence level dataset, as well as the full article level dataset, are available on Kaggle [61]. We utilize both datasets. The sentence-level dataset consists of 7266 texts (sentences) , with 45.4% of them being human and the rest being AI-generated. The average length of a text in this dataset is 21.7 words, with a standard deviation of 9.72. The article-level dataset consists of 1014 texts (articles) , with 50% of them being

human and the rest being AI-generated. The average length of a text in this dataset is 155.8 words, with a standard deviation of 69.

4. **ChatGPT Detector Bias (CDB) Dataset:** This dataset was created for the study [41]. The study authors carried out a series of experiments passing a number of essays to different GPT detection models. Juxtaposing detector predictions for papers written by native and non-native English writers, the authors argue that GPT detectors disproportionately classify real writing from non-native English writers as AI-generated. The dataset is available on Kaggle and HuggingFace[41] and consists of 607 texts, with 41.1% of them being human and the rest being AI-generated. The average length of a text in this dataset is 177.2 words, with a standard deviation of 145.9.

5. **HC3 Dataset:** The Human-ChatGPT Comparison Corpus (HC3) was introduced in [22]. In this paper, the authors collect thousands of comparison responses from human experts and ChatGPT, with domains ranging from open-domain, financial, medical and psychological areas. The dataset has an English and a Chinese subset, and we utilise the English version. The meta-information for the English given in [22] show that the human answers come from a number of other datasets for other tasks, such as question answering with the ELI5 dataset [15] or the WikiQA dataset [91]. The ChatGPT answers in this dataset come from prompting ChatGPT to answer the questions the humans have answered in their text, often aligning with the concept of the human dataset to blend in more (for instance, the ChatGPT counterparts to the ELI5 dataset are generated adding "Explain like I'm five" to the end of the ChatGPT prompt). The HC3 dataset has been mentioned and used in a number of studies and surveys and will provide a good benchmark for all experiments. The version we use contains 24322 human texts and 24322 generated text for a 50%=50% balance.

6. **MAGE Dataset:** The MAGE dataset, introduced in [40], is a comprehensive testbed designed specifically for AI text detection, by collecting human-written texts from 7 distinct writing tasks (opinion statements, reviews, news articles, question answering, story generation, common sense reasoning and scientific writing) as well as machine-generated texts from 27 different LLMs including LLMs from different families than the commonly used GPT lineage. This dataset differs from other datasets as it uses a variety of domains and LLMs and therefore it is expected that the detector will have a more difficult time differentiating between human and machine texts. The MAGE dataset contains over 400000 texts, with the majority of them (70.8%) being generated and the rest being human text.

3.4 Perplexity analysis

In addition to experiments performed on the adversarial attack framework we also want to run a text perplexity-based analysis on our data. As explained in Section 2.4, we want to examine the hypothesis that text perplexity is indicative of an underlying intrinsic difference between human and machine-generated text, that guides the detectors in classifying texts with high accuracy.

To compute text perplexity we utilise various LLMs that can be accessed in an open source way, most notably GPT2-large, since most of the machine-generated text in our datasets comes from the GPT (OpenAI) lineage. We also compute perplexity stats with other LLMs such as Palmyra (from WriterAI), OPT (from Facebook/Meta) and GPT-NeoX (from EleutherAI) in order to see if choosing a particular model has any impact on the results and calculations.

For the computing algorithm we utilise the algorithm given by HuggingFace in [29] given the limitations by the models' context size. More specifically, we use a strided sliding window strategy, moving the context for the perplexity calculation by a fixed stride in order for the model to always have a large context to make predictions at each step. This algorithm, as reported in HuggingFace's work, is calculating perplexity in a way that is close to the true auto-regressive decomposition of a sequence likelihood.

In addition, we modify the algorithm in order to make it into a classifier that can compare to an AI text detector. This can be done in two ways: the simpler way is to arbitrarily choose a number as a threshold and classify every text with a perplexity score below that number as generated and every text with a perplexity score above that number as human. However, although in some datasets even arbitrarily choosing the threshold could yield surprisingly good results, the ideal threshold is most likely different for each domain of texts: scientific articles will have a different perplexity distribution than tweets or question answering texts.

Therefore, we adapt the algorithm to determine the optimal threshold based on training data which we extract from the dataset. We split the dataset into train and test splits and use the accuracy of the perplexity detector in the test split, with the optimal threshold defined on the training split. This is comparable to perplexity-based detectors such as HowkGPT [85], which used a similar method to obtain a threshold for homework essays.

We believe such a method is a good baseline to determine how "easy" it should be for a detector to make a correct prediction in a dataset. Of course, finding the optimal threshold in a realistic scenario would require knowledge of the dataset the text came from, which is not always available. A more advanced method that should work regardless of dataset is comparing the perplexity of the given text to a similar text from a text completion LLM, which is extensively studied in [25]. We also compare our results with this detector, but we should note that our aim is not to compete for the highest accuracy in the detection task, but rather to provide a simple baseline accuracy for future researchers to consider when reviewing their detectors against particular datasets.

Chapter 4

Experimental results & insights

In this chapter, we describe and present our experiments one by one, record our results and findings and try to delve into their deeper meaning. All experiments described here were conducted using Google Colab's T4 GPU module and therefore memory limitations as well as disk space limitations should be taken into account.

4.1 Adversarial attack experiment

The first series of experiments is conducted on the TextFooler adversarial attack framework that is described in Section 3.1. Before conducting the experiments, we train the 3 DistilBERT models on datasets 1,2 and 3 as described in Section 3.2.

4.1.1 Detector accuracy experiments

In the first phase of the experiment we evaluate the detection accuracy of our LLMs in each one of our datasets. This is conducted on 200 examples of the test split of each dataset and we measure the **overall accuracy** (percentage of correct predictions) as well as various other commonly used metrics for the task listed below.

Metrics used in the experiment

In calculating those metrics, we assumed that the positive class always represent the machine-generated texts and the negative class always represents human texts. Of course, this can always be changed by flipping the labels on our datasets.

1. **Machine Precision (MPrec)**:
$$\text{MPrec} = \frac{TP}{TP + FP}$$
where TP (True Positives) are the correctly predicted machine-generated texts, and FP (False Positives) are the human texts classified as machine-generated. Machine precision measures the accuracy of the detector in identifying AI-generated content.

2. **Machine Recall (MRec)**:
$$\text{MRec} = \frac{TP}{TP + FN}$$
where FN (False Negatives) are the machine-generated texts that were incorrectly classified as human. Machine recall measures the ability of the classifier to identify *all* AI-Generated texts.

3. **Human Precision (HPrec)**:
$$\text{HPrec} = \frac{TN}{TN + FN}$$

where TN (True Negatives) are the correctly predicted human texts, and FN are the machine-generated texts that were incorrectly classified as human . Human precision measures the detector's accuracy in identifying human texts.

4. **Human Recall (HRec)**:
$$\text{HRec} = \frac{TN}{TN + FP}$$
where FP (False Positives) are the human texts that were incorrectly classified as machine-generated. Human recall measures the ability of the classifier to identify *all* human texts.

5. **F1 Score**:
$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$
The F1 Score is the harmonic mean of precision and recall, providing a single metric that keeps the balance between them. It is particularly useful when there is an uneven class distribution. The harmonic mean is chosen over the arithmetic mean because it punished extreme values more (so a detector classifying everything as human text or everything as machine-generated text is punished) , thus making the F1 Score a robust measure of a classifier's accuracy.

6. **AUROC (Area Under the Receiver Operating Characteristic Curve)**:
AUROC represents the probability that a randomly chosen positive (in our case, machine-generated) text is ranked higher than a randomly chosen negative (in our case, human) text by the classifier. The ROC Curve is a plot of the true positive rate (recall) against the false positive rate at various threshold settings. AUROC quantifies the overall ability of the classifier to discriminate between the classes. An AUROC of 0.5 suggests no discriminative power (equivalent to a random classifier), whereas an AUROC of 1 indicates a perfect classifier. AUROC, in addition to the F1 score and accuracy, are the metrics that are used in most studies to compare between detectors and thus are the metrics that we turn most of our attention to.

DistilBERT results

The results of our three DistilBERT models are shown in Tables 4.1-4.3. DB-1 is the model trained on the AuTexTification dataset (comprising mostly of tweets, legal documents and wiki articles), DB-2 is the model trained on the Human vs. ChatGPT dataset (comprising of academic papers' abstracts) and DB-3 is the model trained on the sentence-level GPT Classification dataset (comprising of sentences taken from scientific articles).

Table 4.1: DB1 Evaluation Metrics

Dataset/Metric	**MPrec**	**MRec**	**HPrec**	**HRec**	**F1 Score**	**AUROC**	**Accuracy**
Autextification	0.8654	0.9890	0.9896	0.8716	**0.9268**	**0.9303**	**92.5%**
Human vs. ChatGPT	0.4819	1.0000	1.0000	0.0654	0.1228	0.5327	50%
GPT Classification	0.6627	1.0000	1.0000	0.3778	0.5484	0.6889	72%
GPT Detector Bias	0.6000	0.8926	0.3500	0.0886	0.1414	0.4906	57.5%
HC3	0.5740	0.9417	0.8065	0.2577	0.3906	0.5997	61%
MAGE	0.7356	0.9143	0.5385	0.2333	0.3256	0.5738	71%

As it is evident from these results, the small DistilBERT models are very efficient in classifying texts of the domain in which they have been trained on, but in most other domains they are not very effective, since they seem to perform similarly to a random classifier in most datasets.

In particular, the first model (DB-1) has very high Machine Recall (MRec) and Human Precision (HPrec). This combined with the low accuracy scores indicates that this detector is skewed towards predicting most of the out-of-domain texts as machine generated. We have to note here that the 71% accuracy score on the MAGE dataset should not be considered a good one (as also suggested by the low F1 and AUROC scores), since as mentioned in Section 3.3 about 70% of the texts in that dataset are machine-generated, therefore a detector that classifies every text as machine-generated would have a similar score.

Table 4.2: DB2 Evaluation Metrics

Dataset/Metric	MPrec	MRec	HPrec	HRec	F1 Score	AUROC	Accuracy
Autextification	0.4762	0.1099	0.5475	0.8991	0.6806	0.5045	54%
Human vs. ChatGPT	0.9388	0.9892	0.9902	0.9439	**0.9665**	**0.9666**	**96.5%**
GPT Classification	1.0000	0.0545	0.4639	1.0000	0.6338	0.5273	48%
GPT Detector Bias	0.9268	0.6230	0.6102	0.9231	0.7347	0.7730	74%
HC3	1.0000	0.0309	0.4748	1.0000	0.5441	0.5008	50%
MAGE	0.7931	0.1643	0.3158	0.9000	0.4675	0.5321	38.5%

Table 4.3: DB3 Evaluation Metrics

Dataset/Metric	MPrec	MRec	HPrec	HRec	F1 Score	AUROC	Accuracy
Autextification	0.9000	0.0989	0.5684	0.9908	0.7224	0.5449	58.5%
Human vs. ChatGPT	0.9464	0.9815	0.9773	0.9348	0.9556	0.9581	96%
GPT Classification	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	100%
GPT Detector Bias	0.8780	0.5902	0.5763	0.8718	0.6939	0.7310	70%
HC3	1.0000	0.2000	0.5056	1.0000	0.6716	0.6000	58%
MAGE	0.8000	0.2222	0.3000	0.8571	0.4444	0.5397	40%

On the other hand, the second model (DB-2) has high Machine Precision (MPrec) and high Human Recall (HRec). This suggests that it behaves in a way opposite to DB-1 and classifies most texts as human-generated, when they are out of its training domain. Despite that, it manages to perform distinctively better than random chance on the GPT Detector Bias dataset (F1, AUROC and accuracy scores all higher than 0.7). This can be attributed to the domain of the dataset (essays) being similar enough to the training domain of the classifier (scientific abstracts).

Looking at the results of the third model, we can see it behaves pretty similarly to the second model, classifying most texts as human when they are out of its training domain. Despite being trained only on a sentence-level dataset. it achieves the best scores out of the small models in many datasets. This suggests that the training data used for this model (sentences from articles) is close enough to many of the domains of our other datasets, in comparison to tweets for DB-1 (which differ a lot both gramatically and in style than most other datasets) and scientific abstracts for DB-2, where their very rigid structure can be something that a small model can overfit to (and therefore be easily driven out of its domain). A good example for this can be noticed in the DB-3 model performing well on the Human vs. ChatGPT dataset (the training domain of the DB-2 model), while the reverse does not hold true.

Concluding, we can see that training small sequence classification models on the text detection task and on a specific domain can lead to very good accuracy on that domain, but also usually comes with a very significant decrease in efficiency when continuing to perform the task outside of the domain. This renders such an approach ineffective unless we have a lot of very similar data to what we want to classify, which might not always be practical in a realistic situation. We therefore turn our attention to larger models that have already been fine-tuned on the text detection task, but their larger training corpus allows for more general-purpose use, which is why they are widely regarded as the state-of-the-art AI text detectors among pretrained LLMs.

RoBERTa results

The results of using RoBERTa-large-openai-detector on our datasets are shown in table 4.4. We note that the model was not fine-tuned on any of our datasets, as we want to measure accuracy levels when a large model has just been fine-tuned on the text detection task without any particular domain specification.

Table 4.4: RoBERTa Evaluation Metrics

Dataset/Metric	MPrec	MRec	HPrec	HRec	F1 Score	AUROC	Accuracy
Autextification	0.8286	0.5421	0.6231	0.8710	**0.7265**	**0.7065**	**70%**
Human vs. ChatGPT [1]	0.9778	0.6701	0.7601	0.9856	**0.8583**	**0.8278**	**83.25%**
GPT Classification	0.9706	0.6226	0.6970	0.9787	**0.8142**	**0.8007**	**79%**
GPT Detector Bias	0.7917	0.1570	0.4205	0.9367	0.5804	0.5469	46.5%
HC3	1.0000	0.5728	0.6879	1.0000	**0.8151**	**0.7864**	**78%**
MAGE	0.8966	0.3611	0.3521	0.8929	0.5051	0.6270	51%

As is evident in these results, the RoBERTa classifier achieves significantly better than random results in the majority of the datasets, although it performs slightly worse than the pretrained small models in the specific domain they were trained on. This suggests a trade-off between higher accuracy in specific domains and improved generalizability among all domains. Since RoBERTa has a much larger training corpus than the small models it will maintain its performance across more domains and datasets.

However, we notice that RoBERTa does not perform better than random chance on the GPT Detector Bias dataset, which is in agreement with [41] where it was suggested that text written by non-native English speakers might confuse AI text detectors. However, although in this study it is argued that such text might be flagged as machine-generated by detectors, our experiments with the RoBERTa detector suggests that most of the texts on this dataset are flagged as human, and in fact the reason RoBERTa does not obtain a good score in this dataset is that it classifies machine-generated texts as human texts. Regardless, it seems to be necessary to proceed with caution as even the state-of-the art detector fine-tuned on the text detection task can struggle with unseen domains, in which fine-tuning a detector for the specific domain might be the only way to achieve substantial results.

In addition to unseen domains, another reason that AI text detectors might struggle to classify texts correctly is unseen models, as is evident from the MAGE dataset results. RoBERTa is fine-tuned to identify text coming from the GPT lineage (originally from GPT-2, but it does not seem to struggle with more modern GPT models) while the MAGE dataset contains texts from many different Non-GPT models. Despite RoBERTA's accuracy being 20 points lower than DB-1 on the MAGE dataset it has better F1 and AUROC scores than it, suggesting that RoBERTA does a slightly better job at classifications than DB-1, but its accuracy score drops because it has a slight bias for classifying text as human, whereas 70% of texts in the MAGE dataset are machine-generated.

RADAR results

The results of using RADAR-Vicuna-7B on our datasets are shown in table 4.5. Again, as with RoBERTa, we take the raw model that has been fine-tuned on the models authors' datasets, and not on our own.

Table 4.5: RADAR Evaluation Metrics

Dataset/Metric	MPrec	MRec	HPrec	HRec	F1 Score	AUROC	Accuracy
Autextification	0.5521	0.9907	0.8750	0.0753	0.1386	0.5330	56.5%
Human vs. ChatGPT	0.9289	0.9949	0.9949	0.9282	**0.9604**	**0.9616**	**95%**
GPT Classification	0.7206	0.9245	0.8750	0.5957	**0.7089**	**0.7681**	**77%**
GPT Detector Bias	0.7500	0.7934	0.6528	0.5949	0.6225	0.6942	71.5%
HC3	0.7746	1.0000	1.0000	0.6444	**0.7838**	**0.8222**	**84%**
MAGE	0.7661	0.6786	0.4079	0.5167	0.4559	0.5976	63%

As we can see from these results, the RADAR classifier also performs significantly better than random on the majority of datasets, including achieving close to perfect accuracy on the Human vs. ChatGPT dataset, very close to the pretrained model on that dataset. Additionally, in contrast to the RoBERTa classifier it does perform better than random on the GPT Detector Bias dataset, and also slightly outperforms RoBERTa in both the HC3 and MAGE datasets. However, it exhibits noticeably poor performance on the AutexTificiation

[1] Test was done on 400 instead of 200 examples

dataset, classifying almost every text as AI-generated and thereby performing only marginally better than random chance.

This can be attributed to this dataset consisting of mostly short texts (e.g. tweets). During the study it became apparent that **this classifier has an inherent bias to classify short texts disproportionately as AI-generated**, both in its RADAR-Vicuna-7B version as well as the other versions available in the online demo. To confirm that, we also tested this classifier on the sentence-level GPT Classification dataset, where it achieves only a 56% accuracy (compared to 77% on its article-level counterpart), classifying texts disproportionately as AI-generated. To further confirm the inherent bias we also gave RADAR short paragraphs from this work that have not been in any way produced by an LLM; we notice that it flags all short paragraphs as AI-generated, but when the same detector is given a long text comprising of all the paragraphs put together, it correctly classifies it as human text. An example of that can be given in Figure 4.1.1 below. Out of the long-text datasets, the only one to give significant difficulties to the RADAR detector is the MAGE dataset, in which it achieves an accuracy of just 63%, which is less than the majority class baseline of 70% (which is the accuracy a detector that classified every text as AI-generated would get on this dataset).

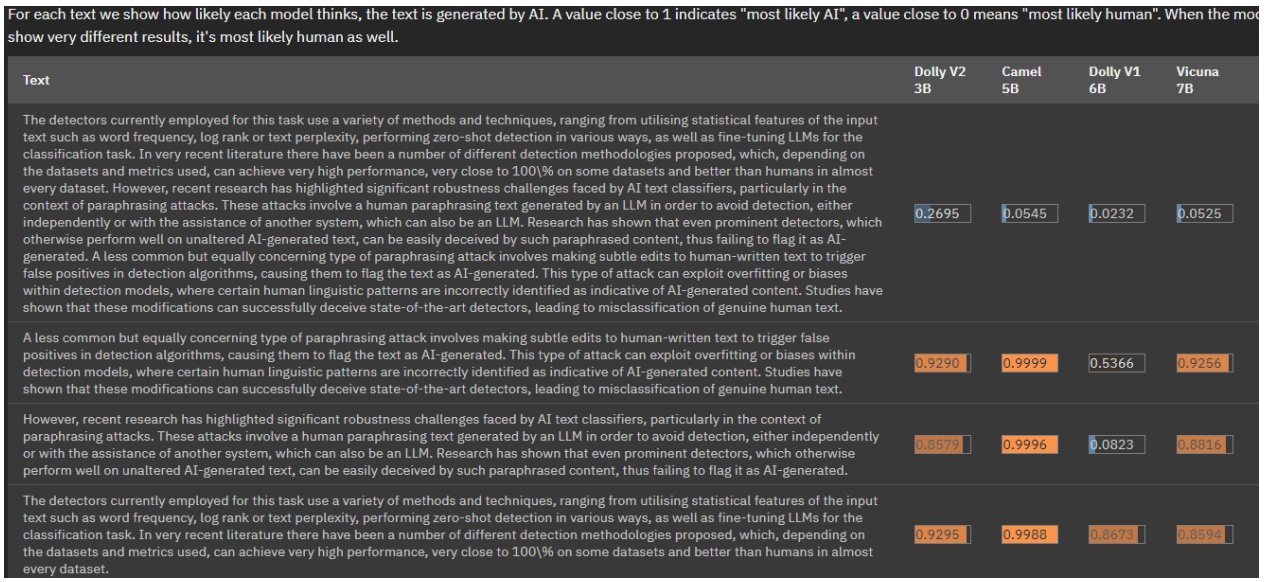

Figure 4.1.1: Examples of RADAR's bias to classify short texts as AI-generated. The larger human-written paragraph is split into 3 smaller parts, and each part of them is classified as AI generated by RADAR.

Experiment summary

The overview of the overall accuracy results, per model and dataset can be viewed on Table 4.6.

Table 4.6: Overall accuracy results, per model and dataset

Model/Dataset	1	2	3	4	5	6
RADAR	56.5%	95%	77%	71.5%	84%	63%
RoBERTa	70%	83.25%	79%	46.5%	78%	51%
DB-1	92.5%	50%	72%	57.5%	61%	71%
DB-2	54%	96.5%	48%	74%	50%	38.5%
DB-3	58.5%	96%	100%	70%	58%	40%

Concluding the first phase of the experiment, we notice that while the state-of-the-art detectors can effectively discriminate between most AI-generated and human texts, there are varying levels of success depending on the domain, dataset, model and technique used. Models that are pre-trained on a specific domain or dataset achieve higher accuracy in the specific domain but suffer from poor generalization outside of it. Meanwhile, bigger models that have been fine-tuned on the task in general but not on a particular dataset achieve better

performance across all datasets but in most cases are not very close to a perfect classifier. Regardless of the dataset and model used, a user wishing to use any of these detectors in a realistic scenario must proceed very carefully, since it appears that there are still major limitations that hinder their efficiency, such as short texts (for RADAR in particular), text from non-native English speakers or text from varying LLMs and domains (as exhibited in the MAGE dataset in which general-purpose detectors struggle the most with).

4.1.2 Accuracy after TextFooler attacks

Accuracy on perturbed texts

In this phase of the experiment we perform adversarial attacks on the models used, with the help of TextFooler from the TextAttack module as described in Section 3.1. TextFooler's interface splits attack results into three categories: successful attacks (texts in which TextFooler successfully managed to flip the prediction label with adversarial attacks), failed attacks (texts in which TextFooler failed to flip the prediction label, despite trying all possible adversarial attacks within the constraints used), and skipped attacks (texts in which the original decision of the predictor is wrong, and TextFooler does not try to attack a wrong prediction). For our purposes, the accuracy of the models after the attack corresponds to the perecentage of failed attacks.

The accuracy of the models on the perturbed texts (percentage of failed attacks) is presented in Table 4.7. Bold numbers indicate the model/dataset combinations where the original detector achieves an accuracy exceeding 70% and an AUROC of 0.67 or higher. Those are the only relevant numbers in the table, since it is beyond the scope of this analysis to evaluate the robustness of a detector that does not have an adequate accuracy from the outset.

Table 4.7: Accuracy of text detectors on perturbed text from TextFooler

Model/Dataset	**1**	**2**	**3**	**4**	**5**	**6**
RADAR	18%	**0%**	**0%**	**0%**	**0%**	5.5%
RoBERTa	**1%**	**0%**	**2%**	2%	**7.5%**	2%
DB-1	**20%**	0%	**3%**	0%	0%	0%
DB-2	16%	**22%**	29%	**1.5%**	14%	9.5%
DB-3	44%	**29%**	**38%**	**11.5%**	44%	22%

As can be seen in these results, TextFooler manages to flip the prediction label in the vast majority of experiments in which the predictor originally had a good accuracy. From those cases, the smaller pre-trained models seem to be slightly more robust in perturbations than the general-purpose models, which can be attributed to them being trained on a small corpus of very similar data and therefore being more prone to overfitting. While overfitting is generally undesirable, it could mean that the patterns these models learn are highly specific and therefore adversarial attacks that simply rely on word-for-word perturbations might be less effective against them. However, as can been seen even in the worst case TextFooler manages to drop the performance of all detectors to below 50%, which means that with simple word substitution perturbations the detectors perform worse than even random chance.

Average perturbed word percentage / Queries

Two more metrics that can be used to determine the detectors' robustness to adversarial attacks are average perturbed word percentage and the number of queries executed.

Average perturbed word percentage (APWP) denotes the proportion of words in a text that were perturbed (replaced with synonyms) during a successful adversarial attack. As the number of perturbations increases, the text inevitably undergoes some degree of semantic alteration. Therefore, a high average perturbed word percentage suggests a more robust detector, and vice versa, since this implies that in order to successfully deceive the detector, an attacker has to alter the text semantically.

Average number of queries (ANQ) represents the average number of adversarial attack quries for a sample of the dataset. A higher number of queries indicates that the model is more robust, since it requires more effort by an attacker to find a successful perturbation that alters the model's prediction.

On table 4.8 below you can find detailed statistics on these two metrics for every model/dataset combination that results in an accuracy above 70% and an AUROC of 0.67 or greater.

Table 4.8: Average perturbed word percentage and queries executed

Model/Dataset Combination	APWP	ANQ
DB1 / AuTexTification	**10.59%**	**260**
DB1 / GPT Classification	5.12%	401
DB2 / Human vs. ChatGPT	**20.15%**	**981**
DB2 / ChatGPT Detector Bias	8.85%	989
DB3 / Human vs. ChatGPT	4.35%	709
DB3 / GPT Classification	**5.37%**	**1089**
DB3 / ChatGPT Detector Bias	5.56%	1402
RoBERTa / AuTexTification	4.33%	93
RoBERTa / Human vs.ChatGPT	2.67%	176
RoBERTa / GPT Classification	2.64%	203
RoBERTa / HC3	2.8%	616
RADAR / Human vs. ChatGPT	5.5%	265
RADAR / GPT Classification	7.21%	303
RADAR / ChatGPT Detector Bias	3.8%	293
RADAR / HC3	4.04%	354

As is evident in the table above, TextFooler perturbs less than 10% of words in a text on average in order to successfully deceive a detector. Pre-trained small models tend to have a higher average perturbed word percentage and especially on the datasets they have been trained on (denoted in bold) which correlates with the general finding that they are more robust to adversarial attacks than general-purpose detectors. Among the general-purpose detectors , RADAR seems to require a bigger percentage of words to be perturbed and a higher number of queries to be executed than RoBERTa, which indicates that it is a more robust detector. This can be attributed to its adversarial training, which presumably has increased its ability to not be deceived by slight text perturbations.

Concluding the second phase of the experiment, it is evident that the state-of-the art AI text detectors face a significant obstacle in the form of robustness to adversarial attacks, which is in agreement with the findings of many recent works such as [28]. We show that simple word substitution attacks can deceive detectors that perform at an accuracy above 70% and up to 100% in order to lower their accuracy to levels less than random chance. In addition, we explore the differences between pre-trained smaller detectors and general-purpose larger detectors and find that smaller detectors are slightly more robust to adversarial perturbations. This is likely due to over-fitting to the small amount of training data, which makes the detectors more confident in their predictions and therefore harder to be deceived with simple adversarial perturbations. However, as we noted before, this does not mean that smaller pre-trained models are better detectors, since they are often performing very poorly outside of their training domain.

4.2 Text perplexity experiment

4.2.1 Text perplexity distributions

In the second series of experiments we explore various features centered around text perplexity, as explained in Section 3.4. Firstly we test the hypothesis that the texts classified as AI-generated usually contain more predictable words leading to significantly lower perplexity. Therefore, we use the GPT-2 model to measure the perplexity of the text in the samples from all of our datasets. The results are presented in Table 4.9 below.

As is evident from the table above, the average text perplexity is indeed significantly higher for human texts than for AI-generated texts. We can also extract some additional insights from this data:

- As can be seen, the AuTexTification dataset has the highest average text perplexity, both for AI-

Table 4.9: Average perplexity of generated and human texts per dataset

Dataset	PPL of Generated Texts	PPL of Human Texts
Autextification	49.6	171.86
Human vs. ChatGPT	15.54	75.82
GPT Classification	11.12	40.03
ChatGPT Detector Bias	25.9	35.67
HC3	10	57.9
MAGE	31.36	43.98
Average	**23.92**	**70.88**

generated as well as for human texts. That can be attributed to a large number of texts from this dataset being very short and therefore harder for the model to predict, especially for tweets (which comprise a large part of this dataset). Tweets typically have higher text perplexity than most other text formats (e.g. scientific articles) since they use informal language, abbreviations, and a lot of non-standard phrases. Furthermore, most tweets are limited to a certain number of characters. This can lead to incomplete or missing context, which makes it much harder for a generative LLM to predict the next sequence in a sentence coming from a tweet.

- The ChatGPT Detector Bias dataset exhibits the lowest average perplexity of human texts among all datasets. This is a dataset that includes a number of human texts coming from non-native speakers. Such text is expected to have lower perplexity than text from native speakers, since non-native speakers often use simpler vocabulary and grammatical structures, making it easier for LLMs to predict their sentences. In additions, since these texts came from language exams it is expected that the authors would avoid typos, colloquialisms or idiomatic expression as much as possible, which adds to the predictability of the texts.
- The MAGE dataset exhibits (after AuTextTification) the second highest average perplexity of AI-generated texts. This can be attributed to the AI-generated texts from this dataset coming also from LLMs outside of the GPT lineage, and therefore harder to predict for our predictor model (GPT-2). This is explored more in depth later.
- The Human vs. ChatGPT, GPT Classification and HC3 datasets all exhibit a very low average perplexity for AI-generated texts. This might be due to their scientific-related domain. Scientific essays and articles tend to have lower perplexity than the average text, because they follow a particular fromal and standardized style, which can be easier for models to predict and generate. In addition, they often use specific terminology, which if the predictor model is familiar with can significantly reduce text perplexity further.

To further understand the distributions of text perplexity within each dataset sample we plot the perplexities in graph form. The results are presented in Figures 4.2.1-4.2.6.

By examining these figures more closely, we can see that although AI-generated texts (marked with red color in the plots) are generally more towards the left side (lower perplexity) than human texts (marked with green color), each dataset has its own perplexity distribution which exhibits some interesting features. More specifically:

- Datasets 2, 3 and 5 (Human vs. ChatGPT , Gpt Classification and HC3) exhibit very little overlap between the perplexity of AI-generated and human texts. Therefore, we expect a perplexity-based detector to be able to distinguish between the classes very efficiently, if the correct threshold is chosen.
- In the AuTexTification and MAGE datasets there is significant overlap between the text perplexities of the two classes. Although it generally holds true that the more we move to the left of the diagrams (lower perplexity) the more dominant the AI-Generated (red) class becomes, we expect it to be very hard for a perplexity-based detector to be able to distinguish between the classes in these datasets.
- In the ChatGPT Detector Bias dataset, there is some overlap between the text peplexities of the two classes, but less so than Datasets 1 and 6. Also it becomes apparent that the human texts in

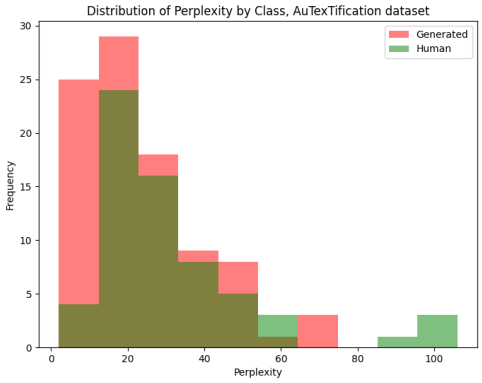

Figure 4.2.1: Text perplexity distribution in the AutexTification dataset

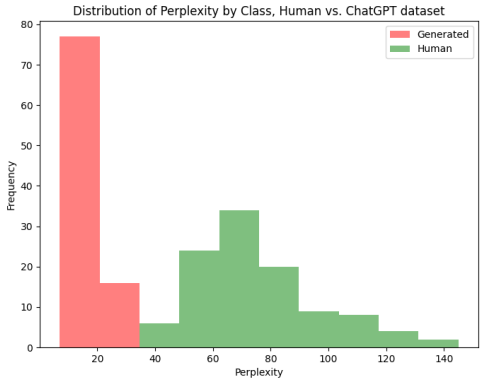

Figure 4.2.2: Text perplexity distribution in the Human vs. ChatGPT dataset

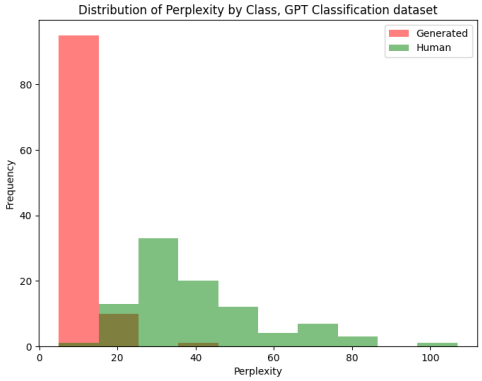

Figure 4.2.3: Text perplexity distribution in the GPT Classification dataset

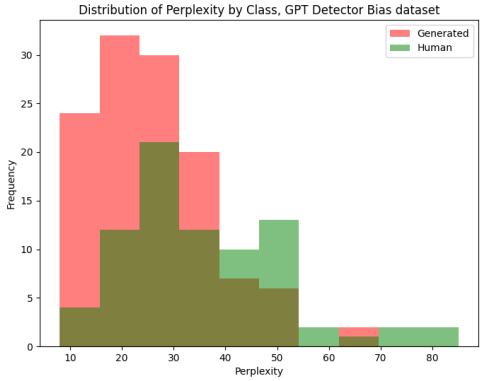

Figure 4.2.4: Text perplexity distribution in the ChatGPT Detector Bias dataset

this dataset are more pushed towards the left side ,which as described previously is expected since this dataset includes many text coming from non-native speakers which exhibit lower perplexity than normal.

4.2.2 Perplexity-based detector

Naive approach

The next step is to construct a perplexity-based detector, in order to be able to have a baseline that determines how well can text perplexity distinguish between human and AI-generated texts. We begin with the naive experiment of arbitrarily choosing a value as the threshold for our detector, which will classify every text with perplexity below that threshold as AI-generated and every text with perplexity above that threshold as human. We then compared our naive detector with the general-purpose LLMs we used in Section 4.1 . The results of this experiment can be observed in Table 4.10 below, where bold indicates the detector than achieved the highest accuracy.

As can be seen in the table above, the naive perplexity-based detectors achieve a better accuracy than the general-purpose LLMs in the datasets in which there is no significant overlap between the human and AI-generated perplexity distributions. In the datasets in which the two perplexity distributions overlap, the general-purpose LLM detector perform slightly better, but we have to note that even this performance is not very high (lower than 75% in all 3 of them). From this we can possibly come to the conclusion that:

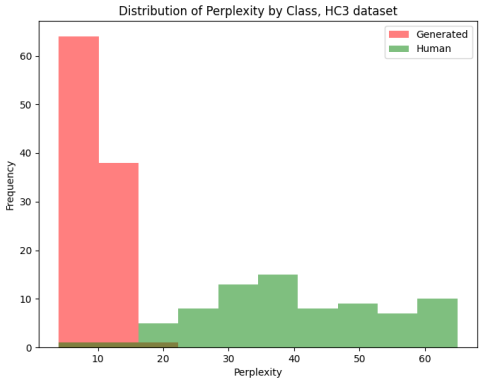

Figure 4.2.5: Text perplexity distribution in the HC3 dataset

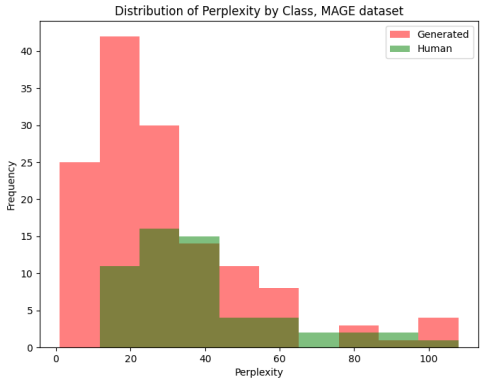

Figure 4.2.6: Text perplexity distribution in the MAGE dataset

Table 4.10: Accuracy of naive perplexity detectors compared to general-purpose LLM detectors

Model/Dataset	AuTex	Human vs GPT	GPT-Class	GPT-DB3	HC3	MAGE
RADAR-Vicuna-7B	56.5%	96.06%	77%	**71.5%**	80.5%	**63%**
ROBERTA-large	**69%**	83.2%	81%	46.5%	78%	57.5%
Naive Perplexity (t=20)	57%	88.6%	**96.5%**	51%	**96%**	53.5%
Naive Perplexity (t=40)	58%	**99.26%**	71.5%	67%	78%	61.5%

- When the dataset is "easy" (that is, both the LLM detectors and the perplexity-based detectors achieve a high score), the perplexity-based detectors perform at at least an equally high level to the LLM detectors, and most probably at a higher level.
- When the dataset is "hard"(that is, both the LLM detectors and the perplexity-based detectors achieve a lower score), the LLM detectors are slightly more accurate than the perplexity-based detectors, but in any case probably not accurate enough to make a confident prediction in a realistic scenario (for which someone would probably need to fine-tune a model on the specific domain/dataset)

Therefore, our hypothesis that text perplexity is a significant enough feature that can be used to differentiate between AI-generated and human texts is confirmed.

It is however important to recognize that the naive perplexity detectors are likely unsuitable for practical application. As demonstrated, the detectors' accuracy varies considerably across different datasets. This variation is anticipated, as each domain has different intricacies and characteristics and therefore necessitates a different text perplexity threshold. For instance, scientific articles and essays typically exhibit lower perplexity on average compared to general-purpose texts such as question-answering content, and significantly lower than tweets. Therefore, the next step in our study is to construct a perplexity-based detector that determines the ideal threshold based on the dataset provided. This is very similar to the work presented in [85] where they specifically find an optimal perplexity threshold on their dataset consisting of academic homework.

Optimization of the perplexity-based detector

The method we use to approach this problem is to "train" the perplexity classifier in a way similar to how AI models are trained, in order to find the ideal threshold for each dataset. More specifically, we split each dataset sample into train and test sets, where 80% of it becomes the training set and the remaining 20% becomes the test set. Then, we compute the perplexity for each text in the training set and iterate over a range of potential thresholds, keeping track of the best threshold (providing the best accuracy in the training set). Finally, we use the optimal threshold found on the training set to evaluate the accuracy of our method on the test set, which constitutes unseen data for the detector.

The results of this approach in each one of our 6 datasets are presented in Table 4.11.

Table 4.11: Average perplexity of generated and human texts per dataset

Dataset	Optimal threshold	Training accuracy	Test accuracy
AuTexTification	170	56.88%	57.5%
Human vs. ChatGPT	39.12	99.07%	100%
GPT Classification	21.48	98.12%	97.5%
ChatGPT Detector Bias	35.22	65.62	80%
HC3	16.32	98.12%	100%
MAGE	65.84	71.25%	65%

As anticipated, such an approach results in almost perfect accuracy on the "easy" datasets, where there is almost no overlap between the AI-generated and human text perplexity distributions. We can also see that on the "hard" datasets (in particular AuTexTification and MAGE) the algorithm finds a much higher optimal threshold than normal, which would only filter out texts that have a very high perplexity and therefore are indistinguishably human. However, since there is significant overlap between the distributions the perplexity detector cannot distinguish between the two classes effectively.

We also should note that by optimising the threshold we have indirectly given the detector access to the training data of the dataset, since the optimal threshold is obtained based on them. Therefore this method requires at least some knowledge of the dataset as opposed to the naive perplexity detectors, and can more effectively be compared to fine-tuned detectors like the DistilBERT models we used in the experiments of Section 4.1. In Table 4.12 below we present the accuracy for each dataset for each detectors, including fine-tuned detectors, general-purpose detectors and the optimal perplexity detector. For the optimal detector we obtain its accuracy on the test results, since those are unseen by the detector at the time of classifying.

Table 4.12: Overall accuracy results, per model and dataset

Model/Dataset	1	2	3	4	5	6
RADAR	56.5%	95%	77%	71.5%	84%	63%
RoBERTa	70%	83.25%	79%	46.5%	78%	51%
DB-1	**92.5%**	50%	72%	57.5%	61%	**71%**
DB-2	54%	96.5%	48%	74%	50%	38.5%
DB-3	58.5%	96%	**100%**	70%	58%	40%
Perplexity	57.5%	**100%**	97.5%	**80%**	**100%**	65%

As is evident from the results, perplexity analysis leads to the best or very close to the best accuracy in every dataset except Dataset 1 (AuTexTification) and Dataset 6 (MAGE). Due to the short length of the texts in the AuTexTification dataset, and the inherent limited amount of context, perplexity is not an ideal measure of determining whether a text is AI-generated or human in this particular domain, and therefore specifically fine-tuned detectors for short text are much more preferable, as can be evident by the high accuracy obtained by DB-1. On the other hand, no detector of those tested achieves a decent enough accuracy on the MAGE dataset, since the best result of 71% is equivalent to the majority class predictor since about 70% of the texts on this dataset are AI-generated.

4.2.3 Non-GPT LLMs as perplexity-based detectors

It is noticeable from all of our results so far that among the datasets we have picked the detectors have struggled much more to obtain good accuracy in the AuTexTification and MAGE datasets. We make the hypothesis that this might be due to those datasets including text from LLMs that are outside the GPT lineage, and therefore harder to detect than text from ChatGPT or other OpenAI models, which is what the vast majority of the text detection task has focused on since they dominate the LLM pool.

To test this hypothesis, we extend our optimal threshold perplexity detector algorithm which is calculated using GPT-2's scores to other LLMs , such as Palmyra (by Writer AI), OPT (by Meta AI) and GPT-NeoX

(by EleutherAI). The results are presented in graph form below, in figures 4.2.7- 4.2.9.

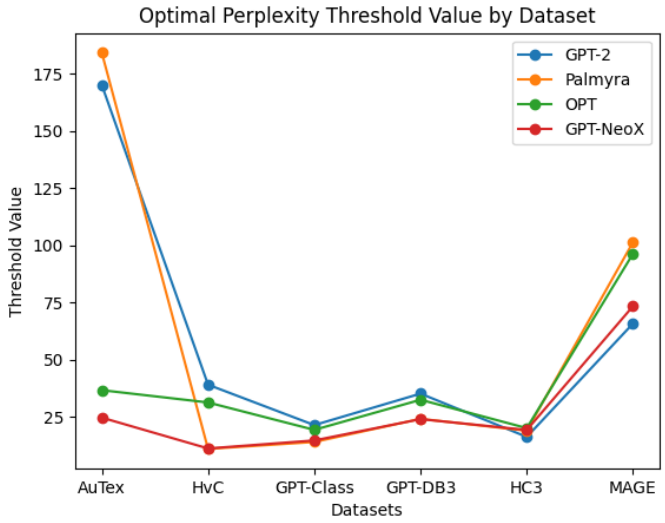

Figure 4.2.7: Optimal perplexity threshold, by model and dataset

Figure 4.2.8: Accuracy on the training set, by model and dataset

As demonstrated in these figures, the choice of LLM for computing perplexity metrics does not result in significant differences. Although there are some variations in the optimal threshold values within the AuTexTification dataset, they are largely inconsequential, as none of the models exhibit significantly high performance on this specific dataset. The accuracy obtained, both on the training and test data, is quite similar across all models, with GPT-2 performing marginally better than the others. This outcome is expected, given that the majority of AI-generated texts in our datasets originate from the GPT lineage. Overall, especially on the accuracy plots (Figures 4.2.8 and 4.2.9), the difference between 'easy' and 'hard' datasets is evident: all models struggle to differentiate datasets 1,4,and 6 using perplexity alone, something that is easily justified if we look at their relevant distributions between the classes.

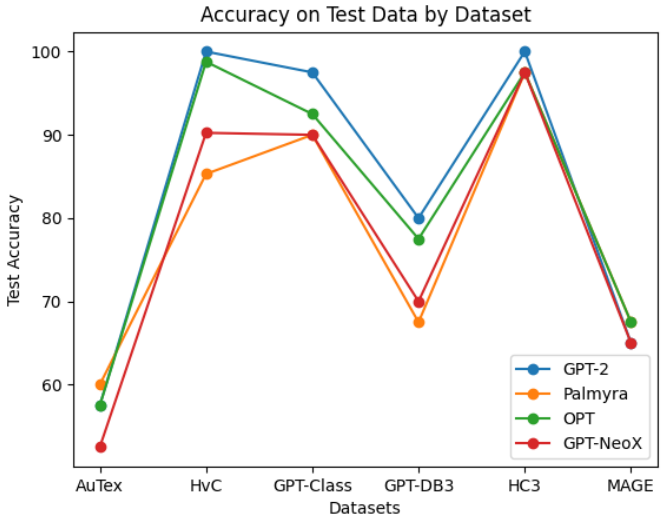

Figure 4.2.9: Accuracy on the test set, by model and dataset

4.2.4 Cross-perplexity and the Binoculars detector

The next step above in complexity is to perform cross-perplexity analysis in addition to simple perplexity analysis. As explained before, perplexity is a measure of how "surprising" a string of text is to a language model. Cross-perplexity on the other hand was introduced in [25] as a measure of how surprising the next token predictions of one model are to another model when both are operating on the same text. The approach the authors use in this study involves calculating the average per-token cross-entropy between the outputs of two models. By comparing the perplexity of a string from one model with how another model perceives it, the method can more accurately detect whether the text is machine-generated or human-written, especially in cases where prompt-induced variability is induced. This helps mitigate issues like the "capybara problem" described in the study, where the context provided by prompts can significantly alter perplexity scores, making it difficult to distinguish between human and machine-generated text based on perplexity alone.

In our work, we run our dataset samples through the Binoculars detector that is available on HuggingFace Spaces. The models used there are Falcon-7B and Falcon-7B Instruct, and we use the high-accuracy mode of the detector since we want to compare to our other detectors. The result of this experiment is shown on Table 4.13.

Table 4.13: BINOCULARS accuracy by dataset

Dataset	Accuracy of BINOCULARS detector
Autextification	- [2]
Human vs. ChatGPT	100%
GPT Classification	96%
ChatGPT Detector Bias	81%
HC3	99%
MAGE	60%

As presented on the table above, the Binoculars detector also exhibits very high accuracy in the 'easy' datasets, but struggles on the 'hard' ones as much as all the other detectors.

An overall summary of the accuracy of all detectors we experimented with in Sections 4.1 and 4.2 is presented in graph form on Figure 4.2.10 below. The blue line represents all 3 of our DistilBERT detectors, taking the

[2] Binoculars does not support text shorter than 32 tokens for perplexity calculations, and therefore we were unable to conduct the experiment in a fair way on this dataset, since a singificant portion of the texts are shorter and thus unable to be analyzed.

score of the pre-trained detector on each dataset.

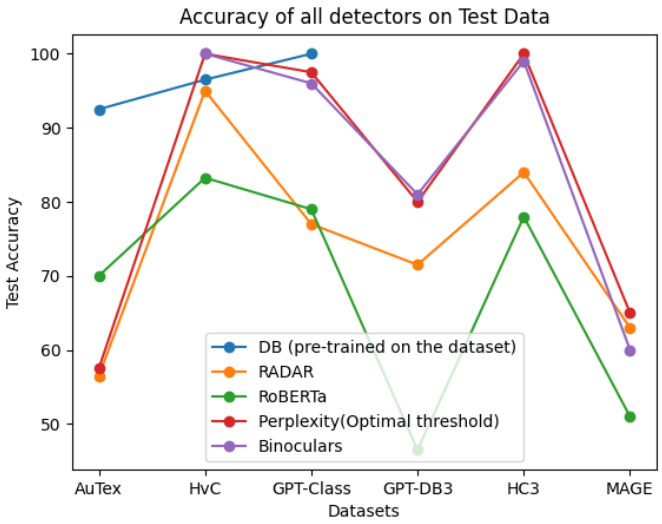

Figure 4.2.10: Accuracy of all detectors on test data

As demonstrated in this figure, the perplexity-based detectors (optimal threshold and Binoculars) exhibit better performance than LLM-based detectors in the datasets in which the human and AI-generated distributions are distinguishable. Between them there are no significant differences, which can be attributes to there being no instances of the "capybara problem" [25], for which the Binoculars detector would potentially be more suited. Moreover, it is shown that fine-tuning a detector for the AI text detection task on a particular dataset also results in very strong perfromance on this dataset, but comes at the cost of significantly less ability for generalization as explained in Section 4.1.

In summary, we demonstrate that text perplexity analysis, despite its simplicity, can effectively capture differences in text that correlate with AI-generated content on the majority of cases. While not always competing for the highest accuracy, perplexity-based detectors can serve as an essential baseline or benchmark in the AI text detection task. High accuracy of perplexity-based detectors on a dataset might suggest that the dataset might be too easy for more sophisticated detectors, potentially lacking in diversity or complexity. This can guide the creation of more challenging datasets that better reflect real-world scenarios where AI and human text are harder to distinguish.

4.3 Additional insights from the experiments

Examining the outputs produced by TextFooler reveals additional insights into the functioning and limitations of text classifiers. For instance, some examples demonstrate that TextFooler occasionally compromises text fluency to flip the predictor label. This suggests that texts with less than perfect fluency are more prone to misclassification, a phenomenon also noted by [41] The degradation in fluency caused by TextFooler indicates that adversarial attacks can exploit the fluency aspect of text to fool classifiers, making it an essential factor to consider when developing robust text classification models.

Moreover, it is evident that texts classified as human often contain oral and informal language, which is uncommon in formal written contexts. This contrasts with generative language models (LLMs), which typically avoid such expressions in their outputs. The preference of generative LLMs for more formal and structured language might be one reason why text detectors generally perform best on scientific abstracts and articles. These types of documents exhibit a consistent style of language that detectors can readily adapt to, making it easier to identify deviations from the norm. Examples supporting these observations are provided in Figure 4.3.1, illustrating the differences in language style and the resulting impact on classifier performance.

Example of syntax errors introduced, which makes the detector classify the text as human-written:

Generated (98%) to Human (81%)

(...) Meanwhile, I relish discovering the profound messages woven into dramas, uncovering the true essence behind each scene. I become deeply connected to the characters (...)

(...) Meanwhile, me relish discovering the profound messages woven into dramas, uncovering the true essence behind each scene. me become deeply connected to the characters (...)

Example of oral expressions used, which make the classifier believe the text is human-written:

Generated (78%) to Human (66%)

XDILLIGAFX13X: My car wont start, it doesnt turn over, it gives a clicking sound, keeps turning over but wont start, HELP

XDILLIGAFX13X: My car wouldnt start, it doesnt turn over, it gives a clicking sound, keeps turning over but wont start, HELP

Example of formal language used that makes the classifier believe the text is generated:

Human (73%) to Generated (68%)

@hundredreasons tho nothing could get better than The Chance, I was wrong, The Prance is the most beautiful song I ever

@hundredreasons tho nothing could get better than The Chance, I was undue, The Prance is the most beautiful song I ever

Figure 4.3.1: Examples of TextFooler perturbations

Additionally, we examine perplexity-based metrics in conjunction with the predictions made by LLM detectors. Specifically, we measure the average perplexity of the perturbed texts and compare it to the average perplexity of the original texts within the same dataset. Perplexity, which quantifies how well a probability distribution or model predicts a sample, serves as an indicator of the text's fluency and predictability. The results of this measurement are presented in Figures 4.3.2 and 4.3.3 below.

As can be seen in these figures, there is a significant difference in perplexity between AI-generated texts and human-authored texts across almost all datasets. Furthermore, this difference remains mostly unchanged by TextFooler perturbations. This implies that a perplexity-based system, such as GPTZero or Binoculars, would likely not be "fooled" by these perturbations. However, since the black-box detectors used in our experiment can be "fooled," it suggests that perplexity-based metrics do not fully capture the criteria that black-box detectors use to classify text.

These findings suggest that perplexity-based detectors and machine learning (ML) detectors achieve high accuracy through fundamentally different mechanisms. Perplexity-based detectors likely rely on the fluency and predictability of the text, capturing how well the text aligns with typical language patterns. On the other

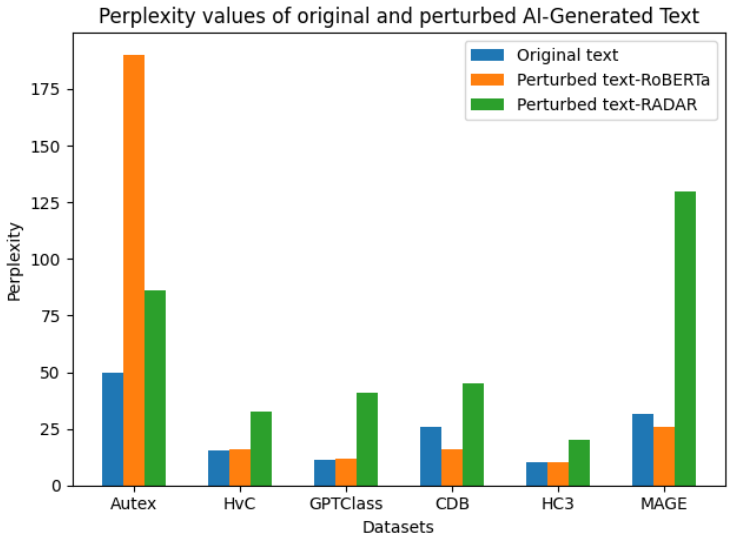

Figure 4.3.2: Perplexity values of AI-generated text

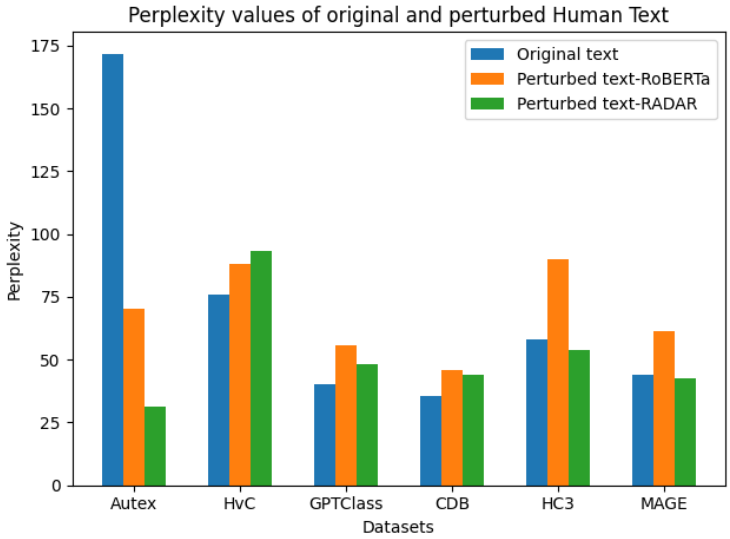

Figure 4.3.3: Perplexity values of Human text

hand, ML detectors might leverage a more complex set of features, including semantic coherence, syntactic structure, and context-specific cues. The fact that adversarial perturbations affect these systems differently indicates that while both types of detectors can achieve similar levels of accuracy, they do so by evaluating distinct aspects of the text. This divergence in their approaches provides a complementary layer of defense, suggesting that integrating both methods could enhance the robustness of text classification systems against adversarial attacks.

Chapter 5

AI text detection from a human perspective

An aspect of the AI text detection task that is less extensively studied is the role of human participation and review in the overall classification process. Given the critical implications that AI text detection can have in various circumstances, it is clear that decisions cannot be left solely to automated detector systems, especially considering the robustness issues explored in Chapter 4. Human reviewers are essential to provide a second layer of verification on a case-by-case basis, ensuring that decisions with substantial ethical implications are judged appropriately.

Human reviewing plays a crucial role in mitigating potential biases inherent in AI detection systems, such as a potential bias against non-native speakers [41]. Additionally, decisions made by AI systems must be transparent and accountable. Human involvement ensures a clear chain of responsibility for decisions made by the AI, which is vital for ethical governance.

To explore this further, we are launching a user survey aimed at investigating human performance in the AI text detection task. The survey also seeks to understand the cognitive process involved when human reviewers detect AI-generated text and to identify the features of a text that the population perceives as indicative of AI generation. Section 5.1 provides detailed information on the design of the user survey, while the results and analysis are presented in Section 5.2. We hope that this survey will not only shed light on human capabilities and perceptions in this domain but also inform the development of more effective and fair AI text detection systems.

5.1 Motivation, aim and meta of our user study

5.1.1 Introduction

Studies examining whether an AI system's output can be perceived as human stem from the original Turing test [82]. With the rapid advancements in AI in recent years, such tests have become increasingly relevant as the lines between human and machine continue to blur. In the context of the AI text detection task, recent works have evaluated human performance in specific domains, such as news articles [84] and conversations [34]. However, the constant advancement of AI systems necessitates that this topic be reviewed frequently, especially as most of the studies still leverage AI-generated content from older models such as GPT-2.

In our work, we utilise our datasets that have already been tested against current AI-based text detectors. Therefore, we can compare and differentiate how humans and AI detector models make their decisions in the text detection task, to provide a more accurate and relevant analysis of the current capabilities and limitations of both human reviewers and AI systems.

5.1.2 Survey structure

The survey is divided into four sections: an introductory part, two AI text detection tasks, and a final paraphrasing task.

Participants start by providing general information about their experience with AI and AI models, particularly text generation models like ChatGPT. They also rate how frequently they interact with such models in their work or study environment. Next, participants estimate their confidence (on a five-point scale) in the existence of an algorithm or model capable of effectively differentiating between human and AI-generated texts, as well as their confidence in their own ability to perform the same task. The introductory section concludes with an optional open-text question, where participants can suggest features they believe can help discern human from AI texts.

After the introduction, participants receive basic instructions and examples of human and AI-generated text with annotations. Then, they proceed to the first AI detection task, where they are given 10 texts and asked to identify whether each text is human or AI-generated. The survey includes 50 texts, divided into five sets. Participants choose a random number between 1 and 5 to determine which set they will annotate. After completing this task, participants are asked to rate their confidence in their answers and provide reasoning behind their decisions. This reasoning is mandatory, as participants are encouraged to explain their classifications, even if they were made randomly.

In the second AI detection task, participants again select randomly from the five sets and annotate a different set of texts. This time, they are provided with additional annotated examples by a detector, with explanations on why the detector classified a text as human or AI-generated. The explanations come from two sources: counterfactual explanations (using examples from TextFooler) and LIME explanations (providing a word importance graph for short sentences annotated by an AI text detector like RADAR or RoBERTa). These explanations remain visible at the top of the page during the task. Participants then annotate 10 different texts from the dataset. Examples of both explanations that were given to the participants are presented in Figures 5.1.1 and 5.1.2 below.

After the second AI text detection task, participants again rate their confidence and explain the features they considered in their classifications. They also evaluate the helpfulness of the provided explanations and indicate which form of explanation (counterfactual or LIME) they found more useful.

Finally, participants move on to the paraphrasing task. Here, they are given a short AI-generated paragraph flagged by AI detectors and asked to rewrite it to make it sound more human while preserving its meaning. This task tests participants' understanding of what makes a text identifiable as AI-generated and their ability to modify it to pass as human. Given that this task involves substantial writing and that most participants are non-native English speakers, it is optional.

AI-Generated Text:	Would be classified as Human text if changed like this:
This can be especially helpful if you're having trouble understanding more complex conversations. So, to summarize, learning a new language takes a lot of practice and patience. But with time and effort, you can start to understand more and more just by talking to people who are fluent in the language	This can be especially helpful if you're having trouble understanding more complex conversations. Thereby, to summarize, learning a new language takes a lot of practice and patience. But with time and effort, you can start to understand more and more just by talking to people who are fluent in the language
The vowel sound at the beginning of the word "European" is actually made by the letter "y," which is a consonant. The letter "y" makes a vowel sound when it is used as a consonant, as in the word "yes." In the word "European," the "y" makes the vowel sound "ee," which is a consonant sound. Therefore, we use the article "a" before the word "European.""]	The vowel sound at the outset of the word "European" is actually made by the letter "y," which is a consonant. The letter "y" makes a vowel sound when it is used as a consonant, as in the word "yes." In the word "European," the "y" makes the vowel sound "ee," which is a consonant sound. Alike, we use the article "a" before the word "European.""
sorry for the question i am not sure if this is a bug or something else. I have just started with Android Studio and	sorry for the question i am not sure if this is a bug or something else. me be just waging with Android Studio and

Figure 5.1.1: Example of counterfactual explanations provided in the survey

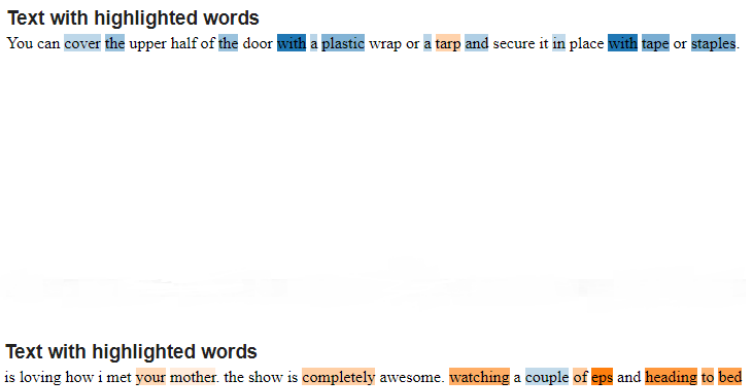

Figure 5.1.2: Example of LIME explanations provided in the survey. Blue indicates swaying the prediction towards AI-Generated, while orange indicates swaying the prediction towards Human, with darker shades indicating larger importance

5.1.3 Data used in the survey

The text corpus for our user survey was provided by our datasets which are described in Section 3.3. More specifically, the annotated examples given at the beginning were randomly generated from all datasets (being the first examples on the dataset after a data reshuffle) and are the same for all participants.

The text corpus for the survey is drawn from our datasets described in Section 3.3. The initial examples are randomly generated and the same for all participants. The 50 texts for the detection tasks are hand-picked from all datasets and uniformly split into five sets to ensure a diverse representation of domains and datasets. Each set contains at least:

- 2 tweets/short texts (taken from the AuTexTification dataset)
- 2 academic abstracts (taken from the Human vs. ChatGPT dataset)
- 1 human text written by a non-native English speaker (taken from the GDB dataset)
- 1 AI-generated text written by a non-GPT model (taken from the MAGE dataset)
- 1-2 examples that fooled a general purpose AI text detector, like RADAR or RobERTa (taken from all datasets)

- 1-2 examples that would deceive our perplexity-based detector, on the optimal threshold setting. That means AI-generated texts with perplexity higher than the optimal threshold for their dataset or human texts with perplexity lower than the optimal threshold for their dataset.

The accuracy of both general-purpose detectors and the perplexity-based detector is between 80-90% for each dataset split. Each split includes at least 4 human and 4 AI-generated texts to prevent participants from guessing a single class and achieving a good score.

The two short paragraphs for the paraphrasing task are ChatGPT-generated examples from the HC3 dataset, and they are the same for every participant to ensure consistency in comparison. They are shown in Figure 5.1.3 below.

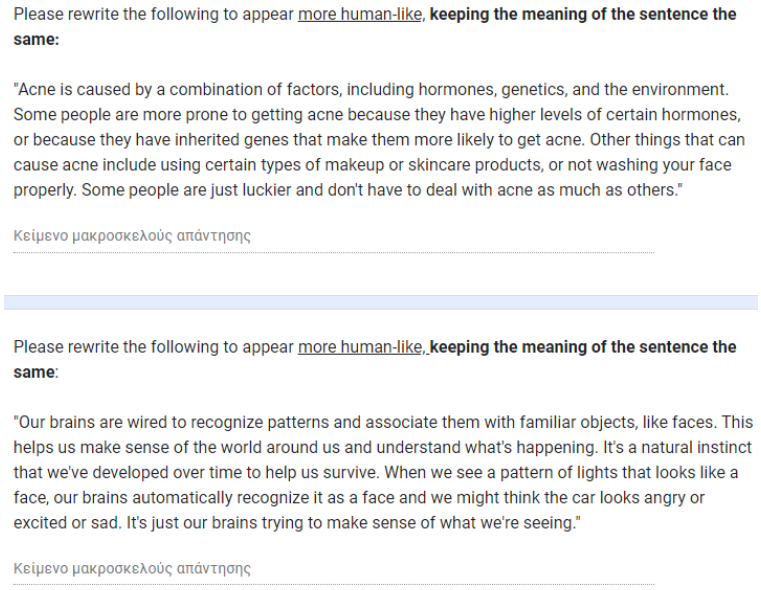

Figure 5.1.3: The short paragraphs that participants were asked to rephrase in our survey

5.2 User study results and insights

Our survey was conducted from June 29, 2024, to July 9, 2024, during which a total of 27 people participated. Each participant performed 20 text annotations, resulting in a total of 540 annotations. This sample size is considered significant for our analysis.

An aspect of our survey design was the random number selection process for text splits, ensuring that no participant annotated the same set of texts twice. This approach, combined with the requirement for participants to provide their reasoning for each classification in the open-text questions, helped ensure that all participants were attentive and engaged throughout the survey. The optional paraphrasing part was completed by 15 participants.

5.2.1 User analytics

In the introductory part of the survey, participants were asked about their familiarity with AI models and text generative models in general, using a 5-point scale. The distribution of responses is shown in Figure 5.2.1.

We observed that this distribution roughly follows a normal curve, which is expected when surveying a mixed audience. This ensures a balanced user base, incorporating responses from both individuals with no background in AI and those who are more familiar with AI and generative models. The slightly higher average response for the second question can be attributed to the widespread use of generative models like ChatGPT, which have become popular beyond AI circles and are frequently used by non-experts. This trend

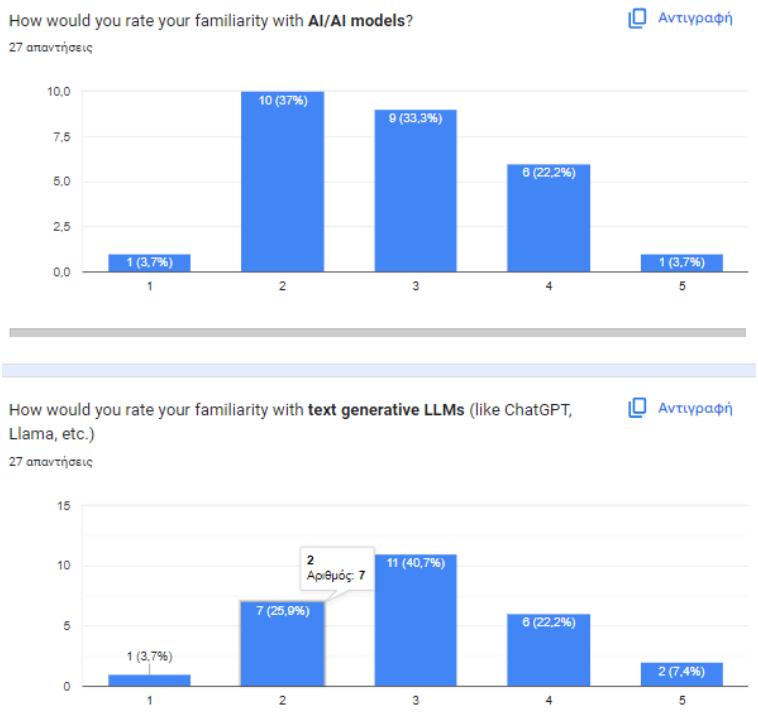

Figure 5.2.1: Users' familiarity to AI and LLMs

is also reflected in the responses to the question about the frequency of using text generative models in work or studies: 33% of participants reported using generative models at least once per week, 30% indicated occasional use, 22% mentioned rare use (once per semester or less), and only 15% stated they never use AI models in their work or studies.

Participants were also asked about their confidence in the ability of automated models to detect AI-generated text and their own confidence in performing this task. The distribution of responses is shown in Figure 5.2.2. The average confidence score on the 5-point scale for trust in automated models was 3.41, whereas the average score for trust in their own ability was 2.78. This suggests that while users have more confidence in automated systems than in themselves for AI text detection, their overall trust in AI text detection models is still far from absolute. This skepticism is understandable, given the current limitations of AI text detectors, such as issues with robustness and cross-domain performance, as discussed in Chapter 4.

5.2.2 Performance results

To measure the performance of participants in our survey during the text detection tasks, we assigned each participant a percentage score corresponding to their accuracy. Given that each participant annotated a total of 20 texts, each correct classification contributed 5 points to their score. The distribution of scores among the 27 participants is illustrated in Figure 5.2.3, which reveals that the overall performance of human reviewers was only marginally better than random chance. Specifically, out of 540 annotations, there were 286 correct classifications and 254 incorrect ones, resulting in an overall accuracy of 52.96%. Considering that a random classifier would achieve a 50% accuracy rate, our findings suggest that human performance in this task is not significantly better than chance.

These results contrast with earlier studies that were conducted before the release of ChatGPT that suggested humans might be able accurately perform AI text detection tasks. Instead, our findings align with the more recent consensus that distinguishing AI-generated text from human-written text has become nearly impossible for human reviewers. As shown in the score distribution, most participants' scores fell between 40% and 65%, with the highest score being 80% and the lowest score being 25%.

We also analyzed participants' performance separately for the first and second text detection tasks. Par-

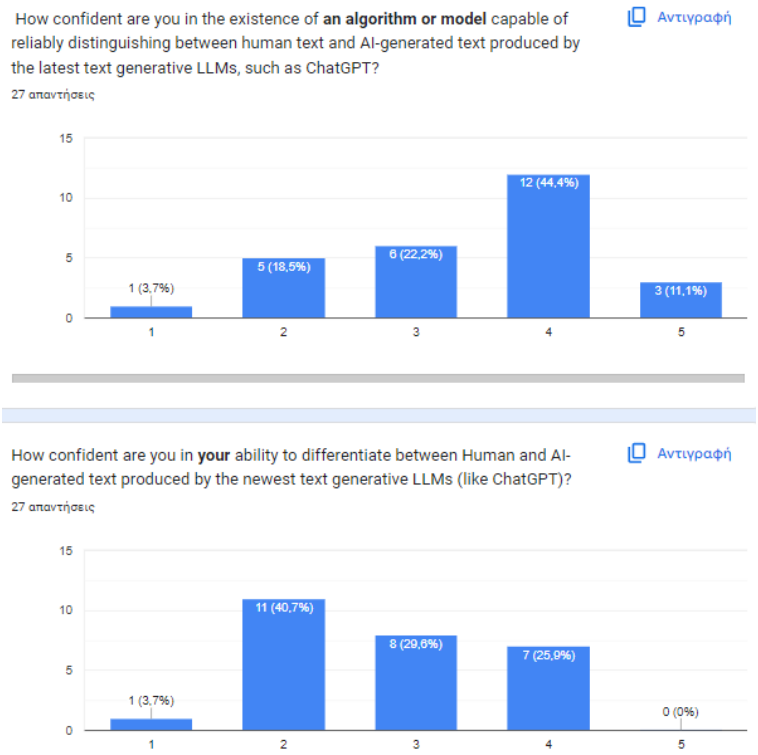

Figure 5.2.2: Users' confidence on models and themselves in the AI text detection task

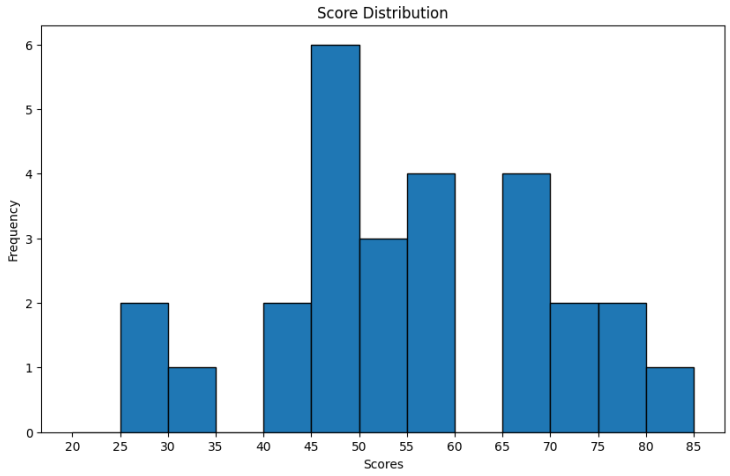

Figure 5.2.3: Distribution of percentage scores in our survey

ticipants achieved an accuracy of 53.7% on the first task and 52.2% on the second task, indicating no improvement. This suggests that the explanations provided to participants during the second task did not enhance their accuracy.

Furthermore, we attempted to correlate participant performance with their responses in the introductory part of the survey. However, filtering by familiarity with AI and large language models (LLMs) did not affect performance, as demonstrated in Figure 5.2.4. This figure shows performance scores across different levels of familiarity with text generative LLMs, with the red horizontal line indicating the overall average accuracy.

Similarly, graphs depicting the distribution of percentage scores by general AI familiarity and LLM usage appeared uncorrelated, with scores clustering randomly around the average accuracy of 52.96%. Therefore,

these graphs do not provide additional helpful information and are not shown.

The only metric that appeared to correlate somewhat with user performance was the participants' confidence in their classifications. Figure 5.2.5 illustrates that higher confidence levels were associated with better accuracy. This suggests that participants who felt more certain about their classifications tended to perform better.

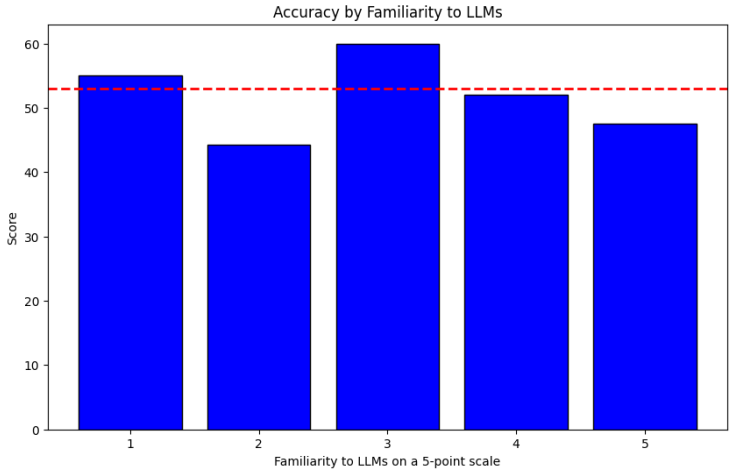

Figure 5.2.4: Distribution of percentage scores by LLM familiarity

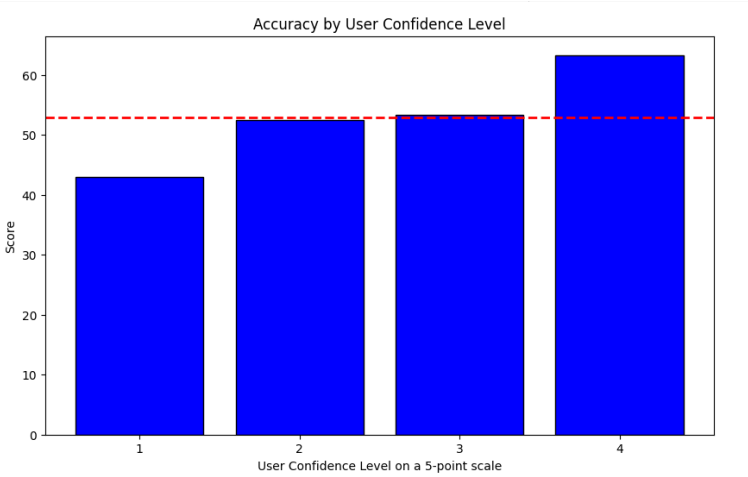

Figure 5.2.5: Distribution of percentage scores by user confidence scores

In summary, our analysis indicates that human reviewers struggle to accurately distinguish between human and AI-generated text, performing only slightly better than random chance. The provided explanations in the second task did not significantly improve accuracy, and familiarity with AI and LLMs did not impact performance. The only factor that showed a correlation with performance was participants' confidence in their answers.

5.2.3 Results by category

We also analyzed the performance of users across different text categories to examine in which domains or text formats humans perform better or worse. It's important to note, as discussed in Section 5.2.2, that overall, humans perform only marginally better than random chance.

The results, presented in Figure 5.2.6, reveal significant discrepancies in human performance across different categories, revealing further details behind the apparent randomness in their ability to detect AI-generated

texts. More specifically:

- Short Texts: Participants performed slightly better than random chance with an accuracy of 64.8%. This suggests that short texts, like tweets, are somewhat easier for humans to detect as AI-generated. The limited context in short texts often degrades the quality of AI-generated content, making it more detectable.
- Scientific Abstracts: Participants scored 52.7%, which is no better than their average performance and only marginally better than random chance. This contrasts sharply with AI-based text detectors and perplexity-based detectors, which achieve their highest performance on academic texts.
- Non-Native English Speaker Texts: Human reviewers performed slightly better than random chance with an accuracy of 66.6%. These texts, often from essays, may appear more personal and human-like to reviewers. Additionally, since the majority of our participants are non-native English speakers, they may find the writing patterns relatable and therefore be more likely to classify such texts as human.
- AI-Generated Texts from Non-GPT Models: Participants had a very poor performance, scoring only 29.6%. This suggests that participants are not familiar with these models, as ChatGPT and its derivatives dominate the text generation landscape. Additionally, these texts come from the MAGE dataset, which has been proven to be challenging for both AI-based and perplexity-based detectors. This indicates that the distinction between human and AI text in this particular dataset is significantly blurred.
- Texts that Fooled Traditional AI-Based Detectors: Participants performed slightly better than average with an accuracy of 59.25%. This supports the observation in [30] that automatic AI text detection is easier when humans are fooled and vice versa. However, this effect has largely diminished over time as AI generation and detection techniques have evolved.
- Texts that Fooled Perplexity-Based Detectors: Participants scored poorly, with an accuracy of 39.81%. This, in conjunction with the previous point, suggests that perplexity-based systems align more closely with human thinking since they utilize textual patterns rather than relying solely on training data. However, further research is needed to explore this phenomenon thoroughly.

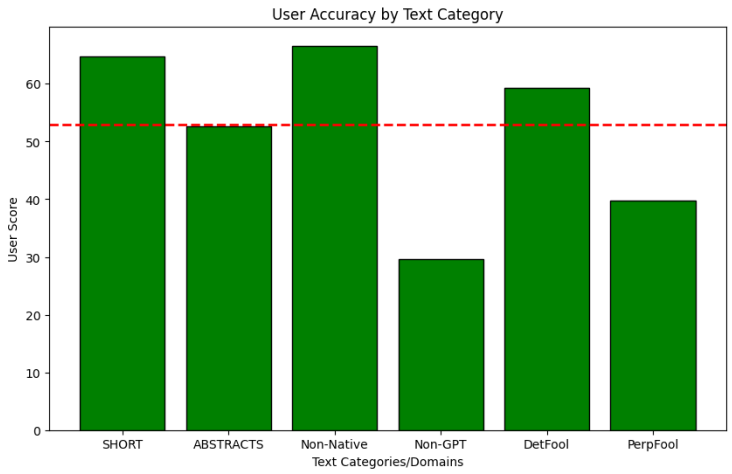

Figure 5.2.6: Distribution of percentage scores by category

Overall, although humans do not achieve the high accuracy levels of automated detector systems (80-90%) in any domain or dataset, the discrepancies in human performance across different domains highlight significant aspects of their cognitive process in performing the AI text detection task. We delve deeper into understanding this process in the next section.

5.2.4 Other insights and open question answers

In this section, we attempt to use open question answers and other insights provided by participants in the survey to further illuminate what informs their classifications.

Firstly, as observed in section 5.2.2, the explanations given to the users did not improve their accuracy. When questioned about the overall helpfulness of the explanations, participants assessed them with an average of 2.62 on a 5-point scale, indicating that the majority did not find the explanations particularly helpful. However, when asked which kind of explanations they found more helpful, 44.4% of participants responded that both counterfactual and LIME explanations were equally helpful, 37% favored counterfactual explanations, and only 18.5% noted that neither was particularly helpful. No participant indicated that LIME explanations were more helpful than counterfactual explanations.

This finding supports studies on counterfactual explanations, which suggest they are a more intuitive way of providing explainability to humans, who naturally think contrastively [44]. However, it is possible that our counterfactual explanations, provided by TextFooler and designed primarily for robustness evaluation and exploration, lacked qualities necessary for being truly helpful to humans in the context of outcome fulfillment [50] .Providing explainability in the AI text detection task remains an open question. A potential direction for future research could involve experimenting with global explanations, since the explanations provided in this survey were local. Global explanation systems might be able to elucidate the general decision-making process of AI text detectors in a way that is more interpretable and understandable to humans. Providing explainability is not a simple technical task; it may incorporate various insights from other domains such as social sciences [52] to ensure that explanations have meaningful impact for stakeholders and users.

Additionally, we can further attempt to understand the process of human detection of AI-generated texts by analyzing the open question answers, in which the survey participants were asked about what features made them classify the texts as AI-generated or human.

Based on the participants' answers, we can categorize the features they used to classify texts into three main categories: language and grammatical features, stylistic elements, and emotional and personal touch. Below are the categories with example texts and brief explanations for each:

- **Language and Grammar**: Participants often relied on grammatical and syntactical cues to determine whether a text was AI-generated or human-written. Below are three such examples of such descriptions:
  - Example 1: "A text with grammatical, syntactical and spelling errors is far more likely to be human."
  - Example 2: "Human text can have flaws in grammar and vocabulary that the AI-generated one cannot have (at least in my experience)."
  - Example 3: "In Human text flow is not perfect, it may have some inconsistencies."

  Overall, participants believe that grammatical and syntax errors are more likely to be indicators of human text, as AI would automatically correct such errors. This contrasts with the early days of text generative LLMs, which tended to make many such mistakes. These observations might have helped human reviewers, especially with short informal texts like tweets.

- **Stylistic Elements**: Participants looked at the style, structure, and formatting of texts to identify their origins. Below are three more examples of such descriptions:
  - Example 1: "AI-generated text is more formulaic and quite often far more polished than a human-written one."
  - Example 2: "AI loves to use some keywords like 'embark'. One can see usage of particular words being higher after introducing text generative LLMs."
  - Example 3: "Each paragraph structure AI-generated text has a more uniform paragraph structure with a beginning and a conclusion."

  In general, participants judged more coherent and polished texts as more likely to be AI-generated, and more free-flowing texts as more likely to be human. However, this might be a misconception, as AI models can easily write free-flowing text with the right prompts, especially in text completion mode.

- **Emotional and Personal Touch**: Some participants pointed out the emotional tone and personal touch as key differentiators for human texts. Below are three such examples:

- Example 1: "The human text is much more personal, it may even contain a point of view from an emotional side."
- Example 2: "Differing factors on each text, but mostly texts that appear more personal I classified as human."
- Example 3:"The ones that feel more 'bland' and with less emotion seem like they are AI-generated ones."

There seems to be a common feeling among some participants that more personal and emotional texts are human-generated, since AI cannot imitate human emotions precisely. However, this might be why most participants were deceived by the texts in the MAGE dataset, which contained personal-looking stories. Modern AI models are capable of writing in a personal style with the right prompt.

In conclusion, participants in our survey employed various clues—linguistic nuances, stylistic elements, and emotional tones—to distinguish between AI-generated and human-written texts. Despite these efforts, their overall accuracy in text classification remained only marginally better than random chance. This underscores the complexity of AI text detection, where modern models can simulate human-like language and styles with remarkable fidelity. The discrepancies in human performance across different text categories also highlight the evolving nature of AI capabilities and the need for deeper understanding of human cognition in assessing text authenticity. Future research should further explore these nuanced classification strategies to enhance the effectiveness of AI tools, ultimately improving collaboration between human reviewers and AI systems in text detection tasks.

5.2.5 Text paraphrasing task

Finally, we report on our results and findings on the final task of the survey, which involves users paraphrasing two short texts. The first short paragraph generated by ChatGPT was paraphrased by 15 different users in order to evade possible AI detection. These paraphrased paragraphs were then inputted into the RoBERTa-large-openai-detector and into our perplexity-based detector, which is optimized for this specific dataset with a threshold of 16.32. The perplexity of the base text was 8, placing it well within the AI-generated text zone. Additionally, ChatGPT-4o was instructed to create a similarity percentage metric based on the number of common words between texts, divided by the total words of each text. The results of this experiment are shown in Table 5.1 below.

Texts	**Text PPL**	**Fooled PPL**	**Fooled ROBERTA**	**Similarity percentage**
Text 1	23	Yes	Yes	78%
Text 2	10	No	Yes	98%
Text 3	15	No	Yes	96%
Text 4	17	Yes	Yes	95%
Text 5	26	Yes	Yes	93%
Text 6	14	No	Yes	92%
Text 7	44	Yes	Yes	85%
Text 8	12	No	No	99%
Text 9	11	No	Yes	95%
Text 10	11	No	No	99%
Text 11	29	Yes	Yes	64%
Text 12	9	No	No	99%
Text 13	9	No	No	99%
Text 14	18	Yes	No	78%
Text 15	9	No	No	99%

Table 5.1: Text paraphrasing experiment - Text 1

As observed in the table, the participants managed to fool the RoBERTa detector 9 out of 15 times and the perplexity-based detector 6 out of 15 times. Notably, texts with a similarity percentage below 85% should not be considered slight paraphrasing but rather a rewriting of the text, incorporating significant human

elements. If we disregard the 4 texts that are significantly rephrased, RoBERTa was deceived 6 out of 11 times, while the perplexity-based detector was fooled just 2 out of 11 times. This highlights the robustness issues that RoBERTa might have with slight paraphrasing, which were explored more thoroughly in Section 4. It suggests that a perplexity-based detector is harder to deceive, especially considering that one of the two cases where it was deceived was borderline (at perplexity 17). Additionally, the decent accuracy of our users in paraphrasing is noteworthy, as many of them managed to fool at least one detector, and everyone produced a text with higher perplexity than the base text, indicating a higher likelihood of being human-written.

The second short paragraph was paraphrased 11 times. We repeated the same process as for Text 1. The perplexity of this paragraph was 12, while the threshold remained at 16.32. Therefore, we hypothesized that this paragraph might be easier to paraphrase. The results are shown in Table 5.2 below.

Texts	**Text PPL**	**Fooled PPL**	**Fooled ROBERTA**	**Similarity percentage**
Text 1	32	Yes	Yes	61%
Text 2	22	Yes	Yes	95%
Text 3	13	No	No	100%
Text 4	23	Yes	Yes	93%
Text 5	47	Yes	Yes	85%
Text 6	13	No	No	91%
Text 7	27	Yes	Yes	81%
Text 8	14	No	No	98%
Text 9	20	Yes	Yes	73%
Text 10	13	No	No	96%
Text 11	48	Yes	Yes	59%

Table 5.2: Text paraphrasing experiment - Text 2

As shown in this table, the same texts managed to fool both the RoBERTa and perplexity-based detectors. Despite this text being closer to human text in theory and thereby easier to paraphrase, only 7 participants managed to paraphrase it correctly without changing too much of the text and maintaining a similarity percentage above 85%. Of these, 2 out of 6 participants managed to fool the detectors. Notably, some participants who successfully perturbed the first text felt that the second text already appeared human and were unsure how to change it. Overall, all participants were on the right track, increasing the perplexity of the base text.

This part of the study demonstrated the challenges and nuances involved in paraphrasing texts to evade AI detection. While participants were relatively successful in fooling the RoBERTa detector with slight paraphrasing, the perplexity-based detector proved in general more robust. The difficulty in paraphrasing varied between texts, with some participants finding the second text inherently more human-like and thus harder to modify without losing its original meaning. This finding suggests that while AI detectors can be deceived through strategic paraphrasing, the effectiveness of this approach varies, highlighting the importance of understanding the intricacies of human language and pattern recognition.

Chapter 6

Conclusion

In conclusion, our extensive analysis of the AI text detection task has provided significant insights into its current challenges and potential improvements. We scrutinized adversarial perturbations targeting state-of-the-art automated detection systems, exploring the delicate balance between achieving high accuracy, generalizability, and robustness while identifying particular weaknesses. By examining text perplexity as a reliable metric to gauge the unpredictability of a text to generative models, we developed a simple threshold-based detector. This detector not only serves as a solid baseline for future, more sophisticated detectors but also highlights that current datasets used in this task may not accurately reflect realistic scenarios. Additionally, our user survey, akin to a modern-day "Turing test" for advanced LLMs, revealed that human detection performance is approximately at random chance levels, indicating their unreliability in this task. Experiments with XAI methods aimed at enhancing human reviewer accuracy showed no significant effect, though a deeper analysis of human reviewer performance provided valuable insights that could guide the development of better explanatory tools in the future. Lastly, by observing humans paraphrasing AI-generated texts to appear human-like, we assessed their effectiveness in deceiving various detector schemes, contributing further to our understanding of the interplay between human and automated detection capabilities.

6.1 Discussion

Despite the significant advancements made in understanding the intricacies of the AI text detection task, our work has several limitations. Firstly, our focus was on the TextFooler adversarial attack framework due to its common usage, lightweight nature, and ease of use. However, the assumptions made about text fluency might be artifacts specific to this framework. These artifacts may not occur if other adversarial or counterfactual editors with better fluency, such as MiCE, were used. This suggests that our conclusions regarding fluency and detectability might be framework-dependent, and further research utilizing a broader range of adversarial tools is necessary for more generalizable insights.

Additionally, our results and assumptions are inherently tied to the datasets we used. Features and nuances can vary significantly depending on the context of each dataset, which is why we included a diverse range of datasets covering various text lengths and domains (tweets, news articles, scientific articles, etc.). This diversity aimed to provide a comprehensive overview, yet our findings underscore the limitations of many commonly used datasets. Notably, our study revealed that the MAGE dataset presents significant challenges for both AI and human detectors. The sophisticated prompt engineering involved in generating MAGE dataset entries effectively deceives both human and automated detection systems. While the dataset's introduction paper suggests that fine-tuning specialized detectors for long texts can achieve performance over 85%, this method's limitations must be noted. In realistic scenarios, the distribution from which a potentially AI-generated text originates is often unknown, limiting the practicality of such fine-tuning approaches.

Moreover, the challenges posed by the MAGE dataset indicate that text detection tasks will become increasingly difficult in the coming years. This, coupled with the unreliable performance of human detectors, suggests that practical detection may soon become unfeasible. In light of these challenges, it might be time to consider new directions, such as LLM authors implementing mechanisms to detect their own generated texts, like watermarking techniques. These proactive measures could offer a more robust solution to the evolving complexities of AI text detection, ensuring the integrity and reliability of generated content.

6.1.1 Ethics Statement

We acknowledge that using adversarial attack frameworks on AI text detectors might produce text that can often be misclassified by various detectors. We do not endorse the use of such text produced by those frameworks to evade detection of AI written content, or misrepresent human content as AI-written, for any purposes outside of research in the direction of better and more nuanced understanding of text detection.

6.2 Future Work

This work paves the way for numerous future studies in the context of AI text detection, which could have significant implications for society. Our comparisons between state-of-the-art AI-based text detectors and our perplexity-based detector suggest that while they achieve similarly high accuracy, they do so through different mechanisms. Perturbations designed to attack one type of detector typically do not deceive the other. Insights from our user study indicate that perplexity-based detectors might be more robust to human perturbations and function more similarly to how humans identify and flag potentially machine-generated texts. We hope this study encourages further research into perplexity-based systems, such as integrating a perplexity-based backdoor into traditional AI-based detectors. This integration could mitigate robustness issues, as any adversarial attack on the detector would also need to bypass the perplexity filter.

Another promising direction for future work arises from our user survey analysis. We found that local explainability techniques, like local feature importance and counterfactual explanations, are not very effective in assisting humans with AI text detection tasks. However, our analysis of human decisions revealed that humans rely more on global indicators, such as language, grammar, style, tone, and the overall feeling of the text. These factors are akin to traditional feature-based and stylometric methods. This insight suggests that global explanations, such as rule-based systems, might provide better support for human reviewers by aligning more closely with their natural evaluation processes. Therefore, exploring global explanation techniques could enhance human understanding and effectiveness in detecting AI-generated texts.

Finally, the last part of our work highlighted the significant potential of understanding how humans para-

phrase AI-generated content. With the increasing prevalence of using LLMs to refine human-written texts across various fields, examining the interplay between human and machine authorship presents a valuable area for future experimentation. Understanding this dynamic can inform the development of more sophisticated detection tools and contribute to maintaining the integrity of human and machine-generated content. By addressing these areas, future research can build on our findings to create more robust, reliable, and user-friendly AI text detection systems.

Chapter 7

Bibliography

[1] André, C. M. et al. "Detecting AI Authorship: Analyzing Descriptive Features for AI Detection". In: (2023).

[2] André, C. M. et al. *GPT vs. Human: A Corpus of Research Abstracts*. Retrieved December 2023. 2023. URL:

[3] Bao, G. et al. *Fast-DetectGPT: Efficient Zero-Shot Detection of Machine-Generated Text via Conditional Probability Curvature*. 2024. arXiv: 2310.05130 [cs.CL]. URL:

[4] Bhan, M. et al. "Enhancing textual counterfactual explanation intelligibility through Counterfactual Feature Importance". In: *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*. Ed. by A. Ovalle et al. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 221–231. DOI: 10.18653/v1/2023.trustnlp-1.19. URL:

[5] Cai, S. and Cui, W. *Evade ChatGPT Detectors via A Single Space*. 2023. arXiv: 2307.02599 [cs.CL].

[6] Calderon, N. et al. *DoCoGen: Domain Counterfactual Generation for Low Resource Domain Adaptation*. 2022. arXiv: 2202.12350 [cs.CL].

[7] Cer, D. et al. *Universal Sentence Encoder*. 2018. arXiv: 1803.11175 [cs.CL].

[8] Chakraborty, M. et al. *Counter Turing Test CT2: AI-Generated Text Detection is Not as Easy as You May Think – Introducing AI Detectability Index*. 2023. arXiv: 2310.05030 [cs.CL].

[9] Chakraborty, S. et al. *On the Possibilities of AI-Generated Text Detection*. 2023. arXiv: 2304.04736 [cs.CL].

[10] Clark, E. et al. *All That's 'Human' Is Not Gold: Evaluating Human Evaluation of Generated Text*. 2021. arXiv: 2107.00061 [cs.CL].

[11] Crothers, E., Japkowicz, N., and Viktor, H. *Machine Generated Text: A Comprehensive Survey of Threat Models and Detection Methods*. 2023. arXiv: 2210.07321 [cs.CL].

[12] Cui, J. et al. *Chatlaw: A Multi-Agent Collaborative Legal Assistant with Knowledge Graph Enhanced Mixture-of-Experts Large Language Model*. 2024. arXiv: 2306.16092 [id='cs.CL'].

[13] Dugan, L. et al. *RoFT: A Tool for Evaluating Human Detection of Machine-Generated Text*. 2020. arXiv: 2010.03070 [cs.CL].

[14] Dugan, L. et al. *Real or Fake Text?: Investigating Human Ability to Detect Boundaries Between Human-Written and Machine-Generated Text*. 2022. arXiv: 2212.12672 [cs.CL].

[15] Fan, A. et al. "ELI5: Long Form Question Answering". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Ed. by A. Korhonen, D. Traum, and L. Màrquez. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3558–3567. DOI: 10.18653/v1/P19-1346. URL:

[16] Filandrianos, G. et al. *Counterfactuals of Counterfactuals: a back-translation-inspired approach to analyse counterfactual editors*. 2023. arXiv: 2305.17055 [cs.CL]. URL:

[17] Foundation, W. *Wikimedia Downloads*. URL:

[18] Fröhling, L. and Zubiaga, A. "Feature-based detection of automated language models: tackling GPT-2, GPT-3 and Grover". en. In: *PeerJ Comput. Sci.* 7.e443 (Apr. 2021), e443.

[19] Gallé, M. et al. *Unsupervised and Distributional Detection of Machine-Generated Text*. 2021. arXiv: 2111.02878 [cs.CL]. URL:

[20] Gehrmann, S., Strobelt, H., and Rush, A. M. *GLTR: Statistical Detection and Visualization of Generated Text*. 2019. arXiv: 1906.04043 [cs.CL].
[21] Guidotti, R. et al. *A Survey Of Methods For Explaining Black Box Models*. 2018. arXiv: 1802.01933 [cs.CY].
[22] Guo, B. et al. *How Close is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection*. 2023. arXiv: 2301.07597 [cs.CL].
[23] Guo, C. et al. *On Calibration of Modern Neural Networks*. 2017. arXiv: 1706.04599 [cs.LG]. URL:
[24] Hanley, H. W. A. and Durumeric, Z. *Machine-Made Media: Monitoring the Mobilization of Machine-Generated Articles on Misinformation and Mainstream News Websites*. 2024. arXiv: 2305.09820 [id='cs.CY'].
[25] Hans, A. et al. *Spotting LLMs With Binoculars: Zero-Shot Detection of Machine-Generated Text*. 2024. arXiv: 2401.12070 [cs.CL].
[26] Holtzman, A. et al. *The Curious Case of Neural Text Degeneration*. 2020. arXiv: 1904.09751 [cs.CL].
[27] Hu, X., Chen, P.-Y., and Ho, T.-Y. *RADAR: Robust AI-Text Detection via Adversarial Learning*. 2023. arXiv: 2307.03838 [cs.CL].
[28] Huang, G. et al. *Are AI-Generated Text Detectors Robust to Adversarial Perturbations?* 2024. arXiv: 2406.01179 [id='cs.CL'].
[29] HuggingFace. *Perplexity of fixed-length models*.
[30] Ippolito, D. et al. "Automatic Detection of Generated Text is Easiest when Humans are Fooled". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by D. Jurafsky et al. Online: Association for Computational Linguistics, July 2020, pp. 1808–1822. DOI: 10.18653/v1/2020.acl-main.164. URL:
[31] Jawahar, G., Abdul-Mageed, M., and Lakshmanan, L. V. S. *Automatic Detection of Machine Generated Text: A Critical Survey*. 2020. arXiv: 2011.01314 [cs.CL].
[32] Ji, Z. et al. "Survey of Hallucination in Natural Language Generation". In: *ACM Computing Surveys* 55.12 (Mar. 2023), pp. 1–38. ISSN: 1557-7341. DOI: 10.1145/3571730. URL:
[33] Jin, D. et al. *Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment*. 2020. arXiv: 1907.11932 [cs.CL].
[34] Jones, C. R. and Bergen, B. K. *People cannot distinguish GPT-4 from a human in a Turing test*. 2024. arXiv: 2405.08007 [cs.HC]. URL:
[35] Kirchenbauer, J. et al. *A Watermark for Large Language Models*. 2023. arXiv: 2301.10226 [cs.LG].
[36] Kirchenbauer, J. et al. *On the Reliability of Watermarks for Large Language Models*. 2024. arXiv: 2306.04634 [cs.LG]. URL:
[37] Koike, R., Kaneko, M., and Okazaki, N. *OUTFOX: LLM-Generated Essay Detection Through In-Context Learning with Adversarially Generated Examples*. 2024. arXiv: 2307.11729 [cs.CL]. URL:
[38] Krishna, K. et al. *Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense*. 2023. arXiv: 2303.13408 [cs.CL].
[39] Kumarage, T. et al. *Stylometric Detection of AI-Generated Text in Twitter Timelines*. 2023. arXiv: 2303.03697 [cs.CL]. URL:
[40] Li, Y. et al. *MAGE: Machine-generated Text Detection in the Wild*. 2024. arXiv: 2305.13242 [cs.CL].
[41] Liang, W. et al. *GPT detectors are biased against non-native English writers*. 2023. arXiv: 2304.02819 [cs.CL].
[42] Liartis, J. et al. "Semantic Queries Explaining Opaque Machine Learning Classifiers." In: *DAO-XAI*. 2021.
[43] Liartis, J. et al. "Searching for explanations of black-box classifiers in the space of semantic queries". In: *Semantic Web* Preprint (), pp. 1–42.
[44] Lipton, P. "Contrastive Explanation". In: *Royal Institute of Philosophy Supplement* 27 (1990), pp. 247–266. DOI: 10.1017/s1358246100005130.
[45] Liu, A. et al. *An Unforgeable Publicly Verifiable Watermark for Large Language Models*. 2024. arXiv: 2307.16230 [cs.CL]. URL:
[46] Liu, Y. et al. *ArguGPT: evaluating, understanding and identifying argumentative essays generated by GPT models*. 2023. arXiv: 2304.07666 [cs.CL]. URL:
[47] Liu, Z. et al. *On the Detectability of ChatGPT Content: Benchmarking, Methodology, and Evaluation through the Lens of Academic Writing*. 2024. arXiv: 2306.05524 [cs.CL]. URL:

[48] Lundberg, S. and Lee, S.-I. *A Unified Approach to Interpreting Model Predictions*. 2017. arXiv: 1705.07874 [cs.AI]. URL:
[49] Massarelli, L. et al. *How Decoding Strategies Affect the Verifiability of Generated Text*. 2020. arXiv: 1911.03587 [cs.CL].
[50] Mastromichalakis, O. M., Liartis, J., and Stamou, G. *Beyond One-Size-Fits-All: Adapting Counterfactual Explanations to User Objectives*. 2024. arXiv: 2404.08721 [cs.LG]. URL:
[51] Mastromichalakis, O. M. et al. "Rule-Based Explanations of Machine Learning Classifiers Using Knowledge Graphs". In: *Proceedings of the AAAI Symposium Series*. Vol. 3. 1. 2024, pp. 193–202.
[52] Menis-Mastromichalakis, O. "Explainable Artificial Intelligence: An STS perspective". In: (2024).
[53] Menis-Mastromichalakis, O. et al. *Semantic Prototypes: Enhancing Transparency Without Black Boxes*. 2024. arXiv: 2407.15871 [cs.LG]. URL:
[54] Miao, Y. et al. *Efficient Detection of LLM-generated Texts with a Bayesian Surrogate Model*. 2024. arXiv: 2305.16617 [cs.LG]. URL:
[55] Mikros, G. K. et al. "AI-Writing Detection Using an Ensemble of Transformers and Stylometric Features". In: *IberLEF@SEPLN*. 2023. URL:
[56] Miller, T. "Explanation in Artificial Intelligence: Insights From the Social Sciences". In: *Artificial Intelligence* 267.C (2019), pp. 1–38. DOI: 10.1016/j.artint.2018.07.007.
[57] Mitchell, E. et al. *DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature*. 2023. arXiv: 2301.11305 [cs.CL].
[58] Molnar, C. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd ed. 2022. URL:
[59] Molnar, C. et al. *General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models*. 2021. arXiv: 2007.04131 [stat.ML].
[60] Morris, J. X. et al. *TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP*. 2020. arXiv: 2005.05909 [cs.CL].
[61] Oghaz, M. et al. *ChatGPT Classification Dataset*. Retrieved December 2023. 2023. URL:
[62] Oghaz, M. et al. *Detection and Classification of ChatGPT Generated Contents Using Deep Transformer Models*. Aug. 2023. DOI: 10.36227/techrxiv.23895951.v1.
[63] Opara, C. *StyloAI: Distinguishing AI-Generated Content with Stylometric Analysis*. 2024. arXiv: 2405.10129 [cs.CL]. URL:
[64] OpenAI. *OpenAI statement*. Retrieved November 2023. 2023. URL:
[65] Ouyang, L. et al. *Training language models to follow instructions with human feedback*. 2022. arXiv: 2203.02155 [cs.CL]. URL:
[66] Raffel, C. et al. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. 2023. arXiv: 1910.10683 [cs.LG]. URL:
[67] Ribeiro, M. T., Singh, S., and Guestrin, C. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. 2016. arXiv: 1602.04938 [cs.LG].
[68] Ribeiro, M. T., Singh, S., and Guestrin, C. *Model-Agnostic Interpretability of Machine Learning*. 2016. arXiv: 1606.05386 [stat.ML].
[69] Ross, A., Marasović, A., and Peters, M. E. *Explaining NLP Models via Minimal Contrastive Editing (MiCE)*. 2021. arXiv: 2012.13985 [cs.CL].
[70] Sadasivan, V. S. et al. *Can AI-Generated Text be Reliably Detected?* 2023. arXiv: 2303.11156 [cs.CL].
[71] Sanh, V. et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *ArXiv* abs/1910.01108 (2019).
[72] Sarvazyan, A. M. et al. "Overview of AuTexTification at IberLEF 2023: Detection and Attribution of Machine-Generated Text in Multiple Domains". In: *Procesamiento del Lenguaje Natural*. Jaén, Spain, Sept. 2023.
[73] Sarvazyan, A. M. et al. *Overview of AuTexTification at IberLEF 2023: Detection and Attribution of Machine-Generated Text in Multiple Domains*. 2023. arXiv: 2309.11285 [cs.CL].
[74] Shi, Z. et al. *Red Teaming Language Model Detectors with Language Models*. 2023. arXiv: 2305.19713 [cs.CL]. URL:
[75] Solaiman, I. et al. *Release Strategies and the Social Impacts of Language Models*. 2019. arXiv: 1908.09203 [cs.CL].
[76] Stokel-Walker, C. and Noorden, R. van. "What ChatGPT and generative AI mean for science". In: *Nature* 614 (2023), pp. 214–216. URL:

[77] Su, J. et al. *DetectLLM: Leveraging Log Rank Information for Zero-Shot Detection of Machine-Generated Text*. 2023. arXiv: 2306.05540 [cs.CL]. URL:
[78] submitters, arXiv.org. *arXiv Dataset*. 2024. DOI: 10.34740/KAGGLE/DSV/7548853. URL:
[79] Susnjak, T. *ChatGPT: The End of Online Exam Integrity?* 2022. arXiv: 2212.09292 [id='cs.AI'].
[80] Tian, E. and Cui, A. *GPTZero: Towards detection of AI-generated text using zero-shot and supervised methods"*. 2023. URL:
[81] Tulchinskii, E. et al. *Intrinsic Dimension Estimation for Robust Detection of AI-Generated Texts*. 2023. arXiv: 2306.04723 [cs.CL]. URL:
[82] Turing, A. M. "Computing Machinery and Intelligence". In: *Mind* 59.October (1950), pp. 433–60. DOI: 10.1093/mind/lix.236.433.
[83] Uchendu, A. et al. "Authorship Attribution for Neural Text Generation". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by B. Webber et al. Online: Association for Computational Linguistics, Nov. 2020, pp. 8384–8395. DOI: 10.18653/v1/2020.emnlp-main.673. URL:
[84] Uchendu, A. et al. *Does Human Collaboration Enhance the Accuracy of Identifying LLM-Generated Deepfake Texts?* 2023. arXiv: 2304.01002 [cs.CL]. URL:
[85] Vasilatos, C. et al. *HowkGPT: Investigating the Detection of ChatGPT-generated University Student Homework through Context-Aware Perplexity Analysis*. 2023. arXiv: 2305.18226 [cs.CL].
[86] Wang, Z. et al. *Implementing BERT and fine-tuned RobertA to detect AI generated news by ChatGPT*. 2023. arXiv: 2306.07401 [cs.CL].
[87] Weidinger, L. et al. *Ethical and social risks of harm from Language Models*. 2021. arXiv: 2112.04359 [id='cs.CL'].
[88] Wu, J. et al. *A Survey on LLM-Generated Text Detection: Necessity, Methods, and Future Directions*. 2024. arXiv: 2310.14724 [cs.CL].
[89] Wu, T. et al. *Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models*. 2021. arXiv: 2101.00288 [cs.CL].
[90] Yang, L., Jiang, F., and Li, H. *Is ChatGPT Involved in Texts? Measure the Polish Ratio to Detect ChatGPT-Generated Text*. 2023. arXiv: 2307.11380 [cs.CL]. URL:
[91] Yang, Y., Yih, W.-t., and Meek, C. "WikiQA: A Challenge Dataset for Open-Domain Question Answering". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015, pp. 2013–2018. DOI: 10.18653/v1/D15-1237. URL:
[92] Zhu, Y. et al. "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books". In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.