ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΠΛΗΡΟΦΟΡΙΑΚΑ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

Απάντηση μαθηματικών ερωτήσεων με χρήση Παραγωγικών Γλωσσικών Μοντέλων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Προκοπίου Πέτρου-Μάριου

Επιβλέπων : Σταματάτος Ευστάθιος

Μέλη εξεταστικής επιτροπής: Κωστούλας Θεόδωρος

Καβαλλιεράτου Εργίνα

Σάμος, Ιούνιος 2023

Πρόλογος και ευχαριστίες

Η παρούσα εργασία πραγματεύεται την αποτελεσματικότητα της δημιουργίας απαντήσεων σε μαθηματικές ερωτήσεις μέσω παραγωγικών γλωσσικών μοντέλων και την πληρότητα των τρόπων αξιολόγησής τους. Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Σταματάτο Ευστάθιο για τη σημαντική βοήθεια που μου προσέφερε κατά την εκπόνηση της εργασίας και τις γνώσεις που μου μετέδωσε την περίοδο αυτή. Επίσης, θα ήθελα να ευχαριστήσω την Ξαπλαντέρη Ευγενία για τη στήριξή της.

Πίνακας περιεχομένων

Λίστα Σχημάτων

Λίστα Πινάκων

Ακρωνύμια

QA	Question Answering
MathQA	Math Question Answering
GenAI	Generative Artificial Intelligence
IR	Information Retrieval
NLP	Natural Language Processing
ARQMath	Answer Retrieval for Questions about Math
CLEF	Conference and Labs of the Evaluation Forum
MIR	Mathematics Information Retrieval
MSE	Math Stack Exchange
API	Application Programming Interface
AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pre-trained Transformers
MOOC	Massive Online Open Courses
AR	Average Relevance
P@1	Precision at 1
LO	Lexical Overlap
CS	Contextual Similarity
BLOOM	BigScience Language Open-science Open-access Multilingual
OPT	Open Pre-trained Transformers
BPE	Byte Pair Encoding
LLM	Large Language Models
RNN	Recurrent Neural Networks
LSTM	(Long Short-Term Memory

Περίληψη

Αυτή η εργασία εξετάζει την αποτελεσματικότητα της χρήσης διάφορων παραγωγικών γλωσσικών μοντέλων για την δημιουργία απαντήσεων για ερωτήσεις μαθηματικού περιεχομένου. Τα τελευταία χρόνια υπάρχει αυξημένο ενδιαφέρον για ανάκτηση πληροφοριών μαθηματικού περιεχομένου στον κλάδο ανάκτησης πληροφοριών, και η χρήση αναδυόμενων τεχνολογιών, όπως τα παραγωγικά γλωσσικά μοντέλα, θα μπορούσε να βοηθήσει σημαντικά, δημιουργώντας εξατομικευμένες απαντήσεις σε περίπλοκα μαθηματικά ερωτήματα, έχοντας ως βάση μόνο την εκπαίδευση του μοντέλου. Για την εκπόνηση της εργασίας έγινε χρήση και εμπορικών αλλά και open source μοντέλων, και σε κάθε μοντέλο εφαρμόσθηκαν διάφορες κοινές προσεγγίσεις. Οι απαντήσεις που παρήχθησαν συγκρίθηκαν με τις απαντήσεις από τον διαγωνισμό ARQMath-3 που είχε ένα παρόμοιο θέμα, το οποίο αποτέλεσε και την έμπνευση για την εργασία αυτή. Επιπροσθέτως, έγιναν και προσπάθειες βελτίωσης των τρόπων αξιολόγησης των αποτελεσμάτων. Βρέθηκε ότι τα προεκπαιδευμένα μοντέλα που είναι διαθέσιμα στο κοινό επιτυγχάνουν παρόμοια αποτελέσματα, σύμφωνα με τις μετρικές που χρησιμοποιήθηκαν, με πιο εξειδικευμένες προσεγγίσεις, όπως αυτές που έγιναν για το ARQMath-3. Φάνηκε ότι η αποτελεσματικότητα επηρεάζεται τόσο από τον αριθμό των παραμέτρων και των άλλων τεχνικών χαρακτηριστικών των μοντέλων, όσο και από τον τρόπο που τίθενται τα ερωτήματα, τον τρόπο παραμετροποίησης και τον τρόπο προεκπαίδευσης του μοντέλου. Ταυτόχρονα υπήρξαν ενδείξεις ότι οι τωρινοί τρόποι αξιολόγησης δεν επαρκούν για την αξιολόγηση αποτελεσμάτων από παραγωγικά γλωσσικά μοντέλα, ότι υπάρχουν περιθώρια βελτίωσης τους, αλλά και ανάγκη εύρεσης επιπλέον τρόπων αξιολόγησης που θα ανταποκρίνονται στις ανάγκες της συγκεκριμένης προσέγγισης.

Λέξεις Κλειδιά: *Παραγωγικά Γλωσσικά Μοντέλα, Παραγωγικοί Προεκπαιδευμένοι Μετασχηματιστές, Εύρεση Μαθηματικών Πληροφοριών, Απάντηση Ερωτήσεων, Τεχνητή Νοημοσύνη, ARQMath*

Abstract

This thesis examines the effectiveness of using various language models for generating answers to mathematics-related questions. In recent years, there has been increased interest in information retrieval of mathematical content in the field of information retrieval, and the use of emerging technologies, such as language models, could greatly assist in creating personalized answers to complex mathematical questions based solely on the model's training. Both commercial and open-source models were used in this study, and different approaches were used for each model. The generated answers were compared with the answers from the ARQMath-3 competition, which had a similar theme and served as inspiration for this work. Additionally, efforts were made to improve the evaluation methods that were used for ARQMath-3. The results showed that publicly available pretrained models achieve similar results, according to the metrics used, with more specialized approaches like those used in ARQMath-3. The effectiveness was found to be influenced by both the number of parameters and other technical characteristics of the models, as well as the way questions are presented, and the parameterization and pretraining method of the model. At the same time, there were indications that the current evaluation methods are not sufficient for evaluating results from language models, that there is room for improvement, and a need for creating additional evaluation methods that meet the specific requirements of this approach.

Keywords: *Generative Language Models, Generative Pretrained Transformers, Math Information Retrieval, Question Answering, Artificial Intelligence, ARQMath*

1

Εισαγωγή

1.1 Machine Learning

Στην πορεία της ιστορίας, οι άνθρωποι έχουν εφεύρει διάφορα εργαλεία και μηχανές για να απλοποιήσουν τις εργασίες και να ικανοποιήσουν τις ανάγκες τους. Το machine learning είναι ένα υποσύνολο της τεχνητής νοημοσύνης με εστίαση στον σχεδιασμό υπολογιστικών συστημάτων που μπορούν να μάθουν από δεδομένα. Εμφανίστηκε από την ιδέα των συστημάτων που βελτιώνουν την απόδοσή τους μέσω της εμπειρίας. Έχει κερδίσει δημοτικότητα λόγω της αυξανόμενης ανάγκης εξαγωγής χρήσιμων πληροφοριών από μεγάλα σύνολα δεδομένων. Σύμφωνα με τον ορισμό του Tom M. Mitchell ένα πρόγραμμα υπολογιστή μαθαίνει όταν η απόδοσή του σε μια συγκεκριμένη εργασία βελτιώνεται με την εμπειρία που αποκτήθηκε από τα δεδομένα (Carta S., 2022). Ο Arthur Samuel, ορίζει το machine learning ως τον τομέα που επιτρέπει στους υπολογιστές να μαθαίνουν χωρίς ρητό προγραμματισμό (Mahesh et al., 2019).

Το machine learning έχει σημειώσει σημαντική πρόοδο τα τελευταία χρόνια, ειδικά σε πεδία όπως η μετάφραση, η αναγνώριση φωνής και η αναγνώριση εικόνας. Αυτή η πρόοδος μπορεί να αποδοθεί στην ανάπτυξη νευρωνικών μεθόδων, οι οποίες υπερέχουν των προηγούμενων προσεγγίσεων. Επιπλέον, η διαθεσιμότητα μεγάλου πλήθους δεδομένων εκπαίδευσης έχει παίξει έναν κρίσιμο ρόλο στην επιτυχία των συστημάτων αυτών (Kwiatkowski et al., 2019).

1.1.1 Τρόποι εκπαίδευσης

Ο κύριος στόχος του machine learning είναι η μάθηση μέσω των δεδομένων, και πολλοί ερευνητές έχουν εξερευνήσει διάφορες προσεγγίσεις για να επιτύχουν αυτόν τον στόχο. Το machine learning βασίζεται σε διάφορους αλγορίθμους για την επίλυση προβλημάτων που σχετίζονται με τα δεδομένα. Σημαντικό είναι να σημειωθεί ότι δεν υπάρχει ένας μοναδικός αλγόριθμος που να είναι κατάλληλος για όλα τα προβλήματα, και η επιλογή εξαρτάται από το συγκεκριμένο πρόβλημα, τις μεταβλητές που εμπλέκονται και το καταλληλότερο μοντέλο.

Υπάρχουν διάφοροι τρόποι εκπαίδευσης μοντέλων, το supervised learning, το unsupervised learning, το semi-supervised learning και το reinforcement learning. Το supervised learning είναι ένας τύπος machine learning που το σύστημα μαθαίνει βάσει ζευγών εισόδου-εξόδου. Εκπαιδεύεται με επισημειωμένα δεδομένα και μαθαίνει ποιά θα έπρεπε να είναι η «έξοδος» για μία συγκεκριμένη «είσοδο». Από την άλλη πλευρά, το unsupervised learning χρησιμοποιεί δεδομένα που δεν είναι επισημειωμένα και προσπαθεί να ανακαλύψει σχέσεις ανάμεσα στα δεδομένα χωρίς ανθρώπινη παρέμβαση. Το semi-supervised learning συνδυάζει επισημειωμένα και μη επισημειωμένα δεδομένα, κάτι που το καθιστά χρήσιμο όταν η απόκτηση επισημειωμένων δεδομένων είναι δύσκολη ή χρονοβόρα. Έτσι, μπορεί να χρησιμοποιήσει τα μη επισημειωμένα δεδομένα για να εκπαιδεύσει τον εαυτό του ακόμα καλύτερα απ' ότι θα μπορούσε χρησιμοποιώντας μόνο τα επισημειωμένα δεδομένα. Τέλος, το reinforcement learning μαθαίνει παίρνοντας feedback από το περιβάλλον του και χρησιμοποιώντας το για να μάθει πώς να εκτελεί μια ακολουθία ενεργειών. Το θετικό feedback ενισχύει μια συμπεριφορά/έξοδο, και το αρνητικό την αποθαρρύνει (Mahesh et al., 2019)(Bi et al., 2019)(Carta S., 2022) (Bell J., 2020).

1.1.2 Νευρωνικά δίκτυα

Μία από τις κατηγορίες αλγορίθμων μηχανικής μάθησης είναι τα νευρωνικά δίκτυα, τα οποία είναι εμπνευσμένα από τον ανθρώπινο εγκέφαλο και είναι μια σειρά αλγορίθμων που αναγνωρίζουν τις υποκείμενες σχέσεις στα δεδομένα. Αποτελούνται από επίπεδα εισόδου, κρυφά επίπεδα και επίπεδο εξόδου για την επεξεργασία και τον υπολογισμό των αποτελεσμάτων. Κατά την εκπαίδευσή τους τα επίπεδα αυτά χρησιμοποιούν μεγάλες ποσότητες δεδομένων και μαθαίνουν «βάρη», τα οποία στη συνέχεια εφαρμόζουν στα δεδομένα εισόδου που τους δίνονται για να παράξουν το τελικό αποτέλεσμα. Αυτή η εργασία εστιάζει στα νευρωνικά δίκτυα που εκπαιδεύονται για να δημιουργήσουν παραγωγικά μοντέλα, τα οποία μπορούν να προβλέψουν την συνέχεια των δεδομένων εισόδου (Mahesh et al., 2019).

1.2 Embeddings

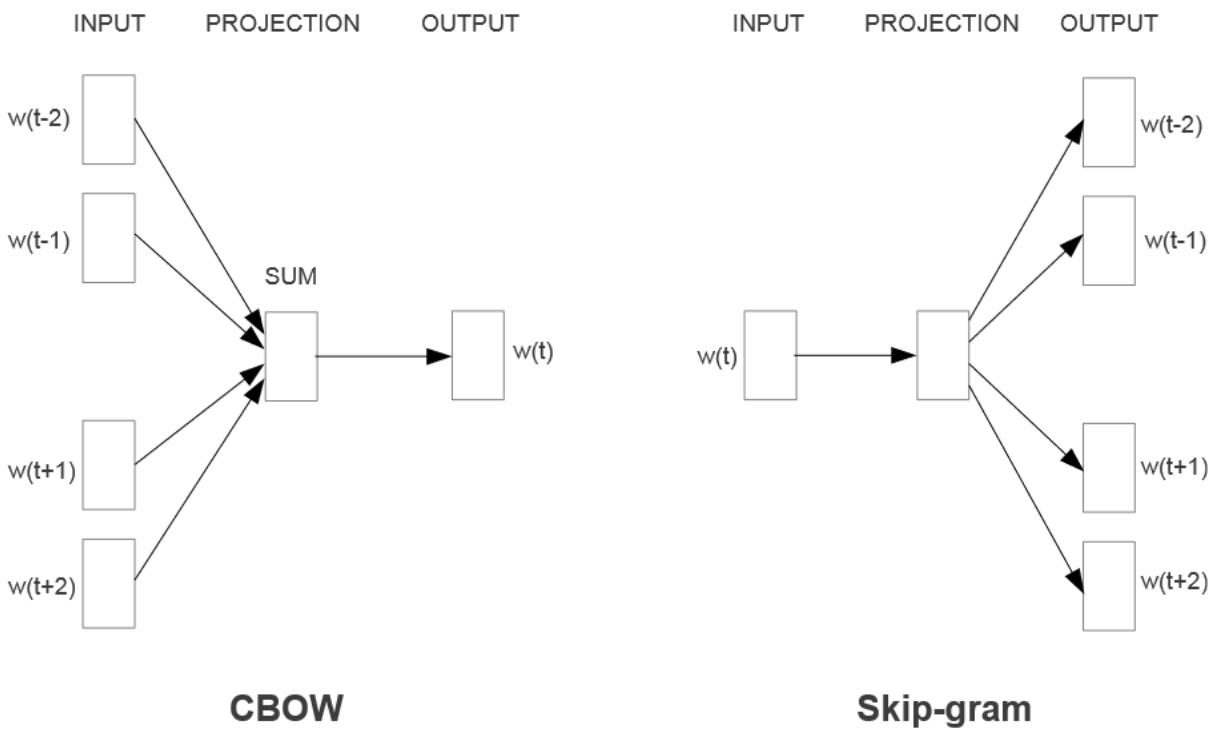
Το νόημα μιας λέξης μπορεί να εξαχθεί από τα περιβάλλοντα συμφραζόμενα στα οποία εμφανίζεται. Εξετάζοντας το πλαίσιο συνύπαρξης των λέξεων σε ένα μεγάλο σύνολο δεδομένων, οι ερευνητές, για παράδειγμα, έχουν παρατηρήσει ότι τα συμφραζόμενα στα οποία χρησιμοποιείται η λέξη "client" είναι πολύ παρόμοια με αυτά στα οποία εμφανίζεται η λέξη "customer", ενώ είναι λιγότερο παρόμοια με τα συμφραζόμενα των λέξεων "waitress" ή "retailer". Έτσι, γίνεται αντιληπτό ότι συνώνυμες λέξεις εμφανίζονται συχνά σε παρόμοια γλωσσικά περιβάλλοντα (Tripodi et al., 2017) (Jurafsky et al., 2023).

Για την αξιοποίηση αυτής της γλωσσικής ιδιότητας, έχουν αναπτυχθεί διάφοροι αλγόριθμοι. Από αυτούς τα word embeddings (πολυδιάστατες διανυσματικές αναπαραστάσεις λέξεων) έχουν κερδίσει ευρεία αποδοχή σε πολλές εφαρμογές Natural Language Processing(NLP). Τα embeddings βασίζονται στα νευρωνικά δίκτυα και έχουν αποδείξει την αποτελεσματικότητά τους στην αποτύπωση τόσο σημασιολογικών, όσο και συντακτικών χαρακτηριστικών των λέξεων (Tripodi et al., 2017). Πρέπει να σημειωθεί ότι αυτό επιτυγχάνεται χρησιμοποιώντας ως πηγή εκπαίδευσης ακατέργαστα κείμενα, χωρίς να βασίζονται σε επιπλέον πηγές πληροφοριών. Σε αυτήν την προσέγγιση, κάθε λέξη αναπαρίσταται ως ένα διάνυσμα, και λέξεις που συναντώνται με παρόμοια συμφραζόμενα αντιστοιχίζονται σε παρόμοια διανύσματα (Tripodi et al., 2017). Σημαντικό είναι να σημειωθεί ότι οι διαστάσεις αυτών των διανυσμάτων-λέξεων δεν είναι εύκολα ερμηνεύσιμες και δεν αντιστοιχούν σε συγκεκριμένες έννοιες με έναν σαφή τρόπο.

Όπως προαναφέρθηκε, τα embeddings χρησιμοποιούνται συνήθως για να παράξουν αναπαραστάσεις που αποτυπώνουν γλωσσική ομοιότητα. Η βασική αρχή είναι ότι λέξεις με κοντινές αναπαραστάσεις διανύσματος τείνουν να έχουν παρόμοια νοήματα. Ωστόσο, η έννοια του νοήματος είναι πολύπλευρη και υπάρχουν πολλές διαστάσεις με τις οποίες οι λέξεις μπορούν να είναι παρόμοιες. Για παράδειγμα, οι λέξεις "πάγος" και "κρύο" είναι παρόμοιες ως προς τη σημασία, οι λέξεις "πάγος" και "φωτιά" παρουσιάζουν συντακτικό και γραμματικό συσχετισμό ως ουσιαστικά αλλά εκφράζουν αντίθετα νοήματα, και οι λέξεις "πάγος" και "παγωμένος" σχετίζονται μορφολογικά καθώς έχουν ετυμολογική συγγένεια (Cotterell et al., 2019). Γι αυτό το λόγο, κάθε διάσταση των embeddings εξειδικεύεται στη σύγκριση ενός συγκεκριμένου χαρακτηριστικού των λέξεων.

Με την πάροδο του χρόνου τα embeddings έχουν εξελιχθεί ως αντικείμενο ξεχωριστής έρευνας χάρη στις εφαρμογές τους σε πολλές εργασίες NLP και την ικανότητά τους να κωδικοποιούν συντακτικές και σημασιολογικές σχέσεις με σχετική ακρίβεια (Almeida et al.,

2019). Πρόσφατες έρευνες έχουν συνδέσει τα φαινομενικά ανεξάρτητα πεδία των embeddings και των γλωσσικών μοντέλων (Almeida et al., 2019). Μία δημοφιλής οικογένεια μοντέλων δημιουργίας embeddings είναι αυτά που βασίζονται στην μεθοδολογία word2vec (Mikolov et al., 2013), η οποία χρησιμοποιεί νευρωνικά δίκτυα για την εκπαίδευση των μοντέλων. Στο πλαίσιο του word2vec προτείνονται δύο διαφορετικές αρχιτεκτονικές για τη δημιουργία των μοντέλων, οι οποίες βασίζονται στη μέθοδο της πρόβλεψης: το Continuous Bag-Of-Words (CBOW), που μεγιστοποιεί την πιθανότητα πρόβλεψης μιας λέξης δεδομένου του περιβάλλοντός της, και το Continuous Skip-Gram, που μεγιστοποιεί την πιθανότητα πρόβλεψης των γύρω λέξεων (πριν και μετά την τρέχουσα λέξη) δεδομένης της τρέχουσας λέξης (Mikolov et al., 2013) (Σχήμα 1). Αυτή η δυνατότητα πρόβλεψης είναι ιδιαίτερα χρήσιμη σε λογισμικά αναγνώρισης ομιλίας, όπου είναι απαραίτητο να αναγνωρίζονται σωστά οι προφερόμενες λέξεις ακόμη και υπό δύσκολες συνθήκες, όπως κακή ποιότητα του σήματος ή παρουσία έντονου θορύβου.



Σχήμα 1. Αναπαράσταση αρχιτεκτονικών CBOW και Skip-gram

Πηγή: Mikolov et al., 2013

Μια δεύτερη μεθοδολογία είναι το GloVe το οποίο δεν βασίζεται στη πρόβλεψη, αλλά στη μέτρηση του πόσο συχνά εμφανίζεται μαζί κάθε ζευγάρι λέξεων. Μέσω τη μέτρησης αυτής δημιουργείται ένας πίνακας συνύπαρξης μεταξύ των λέξεων και του context τους με βάση τον οποίο καθορίζονται οι συσχετισμοί μεταξύ των λέξεων (Σχήμα 2) (Pennington et al., 2014).

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

Σχήμα 2. Παράδειγμα πίνακα συνύπαρξης του GloVe

Πηγή: Pennington et al., 2014

Και οι δύο μεθοδολογίες βασίζονται στην υπόθεση ότι λέξεις με παρόμοια context, δηλαδή λέξεις που εμφανίζονται συχνά μαζί, έχουν παρόμοια νοήματα (Almeida et al., 2019) .

Λόγω του ότι όλες οι μεθοδολογίες χρειάζονται μεγάλη ποσότητα δεδομένων και πολύ χρόνο για την σωστή εκπαίδευσή τους, στις περισσότερες περιπτώσεις οι ερευνητές χρησιμοποιούν προεκπαιδευμένα μοντέλα αντί να εκπαιδεύσουν ένα καινούριο μοντέλο από την αρχή. Με αυτό τον τρόπο υπάρχει μεγάλη εξοικονόμηση χρόνου και πόρων, ενώ ταυτόχρονα η χρήση ήδη δοκιμασμένων μοντέλων εξασφαλίζει υψηλή αποτελεσματικότητα. Σε αυτή την εργασία έγινε χρήση του προεκπαιδευμένου μοντέλου MathBERTa για την μετατροπή λέξεων σε embeddings.

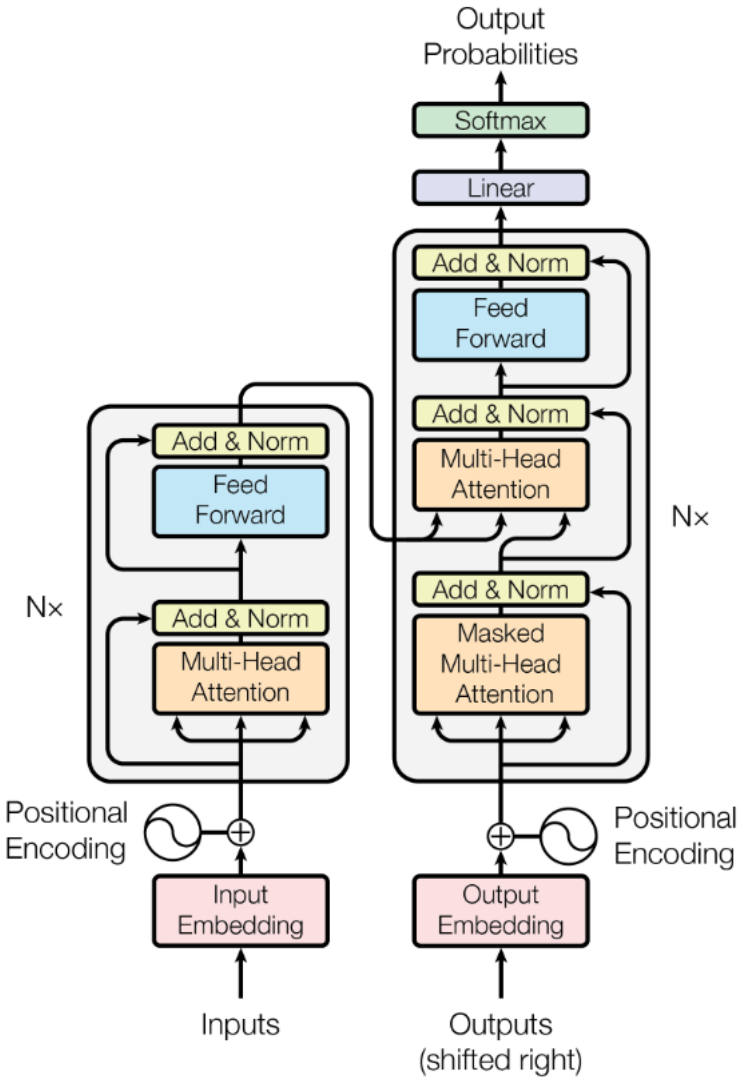
Ως προς την χρησιμότητά τους τα embeddings έχουν κερδίσει σημαντική προσοχή τα τελευταία χρόνια ως ένα ευέλικτο πλαίσιο αναπαράστασης λέξεων σε διάφορους τομείς. Έχει αναγνωριστεί ευρέως ότι η αναπαράσταση των λέξεων και των κειμένων ως διανυσματικών μεγεθών είναι ωφέλιμη, καθώς τα διανύσματα προσφέρουν μια εύληπτη ερμηνεία, επιτρέπουν χρήσιμες λειτουργίες, όπως πρόσθεση, αφαίρεση και μέτρηση αποστάσεων, και μπορούν να χρησιμοποιηθούν αποτελεσματικά σε διάφορους αλγορίθμους και στρατηγικές machine learning. Άλλωστε, η επιτυχία της χρήσης των embeddings στις εργασίες NLP, όπως η αναγνώριση των μερών του λόγου και ονοματικών συνόλων, η ανάλυση συναισθημάτων και η ανάκτηση εγγράφων, είχε ως αποτέλεσμα την αξιοποίησή τους στις κοινωνικές επιστήμες. Έτσι, τα embeddings έχουν αποκτήσει τεράστια δημοτικότητα τα τελευταία χρόνια, καθώς εφαρμόζονται σε διάφορους τομείς, από τη βελτίωση των εισροών για εργασίες επίβλεψης έως την παροχή ουσιαστικών εργαλείων, όπως η κατανόηση των αντίστοιχων αντιλήψεων που διαμορφώνονται με βάση τις διαφορές στις σημασιολογικές αποστάσεις (Rodriguez et al., 2021).

Όπως γίνεται αντιληπτό, τα embeddings έχουν δύο διακριτούς σκοπούς. Καταρχάς, λειτουργούν ως αναπαραστάσεις στοιχείων για άλλες διαδικασίες μάθησης. Δεύτερον, αποτελούν αντικείμενο ενδιαφέροντος για τη μελέτη της χρήσης και της σημασίας των λέξεων, περιλαμβανομένης της ανθρώπινης σημασιολογίας. Σημειώνεται ότι η επίτευξη καλής απόδοσης στον πρώτο σκοπό δεν αποτελεί προαπαιτούμενο καλής απόδοσης και στον δεύτερο (Rodriguez et al., 2021).

1.3 Transformers και Attention

Πριν την ανάπτυξη των Transformers, τα Recurrent Neural Networks (RNN), όπως το Long Short-Term Memory (LSTM), χρησιμοποιούνταν ευρέως για τη μοντελοποίηση ακολουθιών και τις διεργασίες μεταγωγής (Vaswani et al., 2017). Είχαν μεγάλη επιτυχία στη γλωσσική μοντελοποίηση και τη μηχανική μετάφραση. Ωστόσο, αυτά τα μοντέλα είχαν περιορισμούς λόγω της ακολουθιακής τους φύσης, η οποία περιόριζε την παραλληλοποίηση και συναντούσε σημαντικές δυσκολίες στις μεγαλύτερες ακολουθίες.

Για να αντιμετωπιστούν αυτοί οι περιορισμοί, δημιουργήθηκε η αρχιτεκτονική των



Σχήμα 3. Αρχιτεκτονική των transformers

transformers (Σχήμα 3).

Οι transformers βασίζονται στον μηχανισμό self-attention, ο οποίος δημιουργήθηκε με βάση τον μηχανισμό attention των προηγούμενων μοντέλων (Vaswani et al., 2017).

Το attention είναι ένας μηχανισμός που χρησιμοποιείται για να αξιολογηθεί η σχετικότητα των στοιχείων μεταξύ ακολουθιών. Επιτρέπει στο μοντέλο να επικεντρωθεί σε συγκεκριμένα μέρη της εισόδου κατά τη δημιουργία της εξόδου. Χρησιμοποιώντας το attention τα μοντέλα μπορούσαν να βελτιώσουν την απόδοσή τους σε διάφορες εργασίες επεξεργασίας φυσικής γλώσσας (Badhanau et al., 2014).

Το self-attention το οποίο υλοποιήθηκε στους transformers επιτρέπει στο μοντέλο να αξιολογήσει τη σχέση μεταξύ των διάφορων στοιχείων εντός της ίδιας ακολουθίας. Συγκρίνοντας κάθε στοιχείο στην ακολουθία με όλα τα υπόλοιπα, επιτρέπει στο μοντέλο να ανιχνεύει τις εξαρτήσεις μεταξύ όλων των στοιχείων. Το self-attention είναι ιδιαίτερα αποτελεσματικό στην κατανόηση των σχέσεων μεταξύ λέξεων σε εργασίες επεξεργασίας φυσικής γλώσσας, επειδή μπορεί να διατηρήσει περισσότερη πληροφορία σχετικά με την ακολουθία και να αναγνωρίσει εξαρτήσεις μεταξύ στοιχείων τα οποία βρίσκονται «μακριά» το ένα από το άλλο. Επίσης, από τη στιγμή που ο υπολογισμός για κάθε στοιχείο γίνεται ξεχωριστά, οι διεργασίες μπορούν να εκτελεστούν παράλληλα, μειώνοντας δραματικά το χρόνο που απαιτείται (Vaswani et al., 2017).

Στην υλοποίηση των transformers χρησιμοποιείται το multi-head attention το οποίο βελτιώνει την ισχύ και την ευελιξία του μηχανισμού self-attention. Αντί να βασίζεται σε ένα μόνο μηχανισμό self-attention, το multi-head attention χρησιμοποιεί ταυτόχρονα πολλές διαφορετικές υλοποιήσεις του μηχανισμού self-attention και κάθε μία από αυτές εστιάζει με διαφορετικό τρόπο στον υπολογισμό της εξάρτησης μεταξύ των στοιχείων. Αυτό επιτρέπει στο μοντέλο να αντιληφθεί πολλούς διαφορετικούς τύπους συσχετίσεων. Μετά τον υπολογισμό που γίνεται από κάθε μηχανισμό self-attention, τα αποτελέσματα συνενώνονται για να παραχθεί η τελική έξοδος του multi-head attention. Χρησιμοποιώντας multi-head attention, οι transformers μπορούν να «αντιληφθούν» διάφορες πτυχές των εξαρτήσεων και συσχετίσεων της ακολουθίας και να εκτελούν πιο πολύπλοκους υπολογισμούς, με αποτέλεσμα τη βελτίωση των δυνατοτήτων «κατανόησης» του μοντέλου και την ενίσχυση της απόδοσης σε πολύπλοκες εργασίες (Vaswani et al., 2017).

Για να συμπεριληφθεί η σειρά των στοιχείων της ακολουθίας, προστίθενται κωδικοποιήσεις θέσης στις εισόδους του κωδικοποιητή και του αποκωδικοποιητή. Αυτές οι κωδικοποιήσεις

παρέχουν πληροφορίες για τη σχετική ή απόλυτη θέση των στοιχείων στην ακολουθία. Μέσω της χρήσης των κωδικοποιήσεων θέσης, το μοντέλο μπορεί να εκμεταλλευτεί τη σειρά της ακολουθίας κατά την επεξεργασία (Vaswani et al., 2017).

Συνολικά, οι Transformers έχουν αναδειχθεί ως μια ισχυρή εναλλακτική στα RNN για την μοντελοποίηση ακολουθιών και τις διεργασίες μεταγωγής. Η αρχιτεκτονική τους με βάση το self-attention επιτρέπει περισσότερη παραλληλοποίηση, βελτιωμένη υπολογιστική απόδοση και αποτελέσματα υψηλής ποιότητας. Εκμεταλλευόμενοι το self-attention και τις κωδικοποιήσεις θέσης, οι transformers αποτυπώνουν εξαρτήσεις ανάμεσα σε μακρινά στοιχεία κι επεξεργάζονται αποτελεσματικά ακολουθίες διάφορων μηκών, οδηγώντας σε σημαντικές προόδους στη γλωσσική μοντελοποίηση και τη μηχανική μετάφραση.

1.4 Μοντέλα BERT και GPT

1.4.1 Μοντέλο Bidirectional Encoder Representations from Transformers (BERT)

Το BERT αναπτύχθηκε από την Google το 2018 ως ένας τρόπος να βοηθήσει τους υπολογιστές να κατανοήσουν τη σημασία της αμφιλεγόμενης γλώσσας σε κείμενο, χρησιμοποιώντας τα συμφραζόμενα της κάθε λέξης. Αντίθετα από τα παραδοσιακά μοντέλα που διαβάζουν το κείμενο αριστερά προς τα δεξιά ή δεξιά προς τα αριστερά, το BERT επεξεργάζεται ολόκληρη την πρόταση και προς τις δύο κατευθύνσεις, επιτρέποντάς του να λαμβάνει υπόψη το πλήρες πλαίσιο και τις εξαρτήσεις μεταξύ των λέξεων. Το BERT βασίζεται στη αρχιτεκτονική των transformers, η οποία χρησιμοποιεί τον μηχανισμο του self-attention για να αξιολογήσει τη σημασία κάθε λέξης σε μια πρόταση με βάση τη σχέση της με άλλες λέξεις. Το αρχικό μοντέλο BERT εκπαιδεύτηκε πάνω σε ένα μεγάλο αριθμό κειμένων από τη Wikipedia. Μετά την προ-εκπαίδευση, το BERT έχει τη δυνατότητα να εκπαιδευτεί περαιτέρω για συγκεκριμένες εργασίες, όπως ταξινόμηση κειμένου, απάντηση ερωτήσεων κ.α. . Κατά την φάση της εκπαίδευσης αυτής (fine-tuning), το BERT εκπαιδεύεται με τη χρήση ειδικών για την εργασία δεδομένων, όπου οι παράμετροι του μοντέλου προσαρμόζονται για να μπορούν να κάνουν ακριβείς προβλέψεις σχετικά με την επιθυμητή εργασία. Μία από τις βασικές χρήσεις του BERT είναι η μέτρηση της σημασιολογικής ομοιότητας μεταξύ προτάσεων. Αυτή θα είναι και ο βασικός τρόπος χρήσης του σε αυτή την εργασία (Devlin et al., 2019).

1.4.2 Μοντέλα Generative Pre-trained Transformers (GPT)

Τα μοντέλα GPT αποτελούν μια οικογένεια μοντέλων νευρωνικών δικτύων που χρησιμοποιούν την αρχιτεκτονική των transformers και αποτελούν σημαντική πρόοδο στον κλάδο του Artificial

Intelligence (AI), με πιο γνωστό παράδειγμα το ChatGPT. Το όνομα GPT δημιουργήθηκε από τον οργανισμό OpenAI, όμως έχουν εμφανιστεί και μοντέλα από άλλους οργανισμούς στα οποία ταιριάζει ο χαρακτηρισμός αυτός. Ορισμένα μοντέλα εντάσσουν τον χαρακτηρισμό και στο όνομά τους, όπως το Cerebras-GPT. Τα μοντέλα GPT δίνουν στις εφαρμογές τη δυνατότητα να δημιουργούν κείμενο που μοιάζει με ανθρώπινο και να απαντούν σε ερωτήσεις με διαλογικό τρόπο. Η δημιουργία και η κατακόρυφη άνοδος της φήμης των μοντέλων GPT αποτέλεσε σημαντικό παράγοντα στην ευρεία υιοθέτηση της τεχνολογίας AI, καθώς η τεχνολογία αυτή μπορεί τώρα να χρησιμοποιηθεί για την αυτοματοποίηση και βελτίωση μιας ευρείας γκάμας εργασιών, από μετάφραση κειμένων και περίληψη εγγράφων έως δημιουργία περιεχομένου, αλλαγή του στυλ ενός κειμένου, συγγραφή κώδικα και παροχή γενικών πληροφοριών. Τα μοντέλα transformers όπως το GPT έχουν αναζωπυρώσει την έρευνα γύρω από τα μοντέλα AI, δημιουργώντας εφαρμογές, οι οποίες σταδιακά χρησιμοποιούνται από διάφορους οργανισμούς για να φτάσουν σε νέα επίπεδα παραγωγικότητας, να εμπλουτίσουν τις εφαρμογές τους και να βελτιώσουν τις εμπειρίες των πελατών τους (Brown et al., 2020).

Στο πλαίσιο αυτής της εργασίας χρησιμοποιούνται μοντέλα GPT για την παραγωγή μαθηματικών απαντήσεων στις ερωτήσεις που τέθηκαν στον διαγωνισμό ARQMath.

1.5 Question Answering (QA)

Τα συστήματα QA σχεδιάζονται για να παρέχουν σχετικές απαντήσεις σε ερωτήσεις χρηστών σε φυσική γλώσσα (Rajpurkar et al., 2016). Ο κύριος στόχος των συστημάτων αυτών είναι να ανακτούν ακριβείς απαντήσεις σε ερωτήσεις, αντί να επιστρέφουν μια λίστα σχετικών εγγράφων ή παραθέσεων, όπως τα παραδοσιακά συστήματα ανάκτησης πληροφοριών. Τα συστήματα QA συνήθως αποτελούνται από τρία βασικά υποσυστήματα: Το σύστημα question processing, το οποίο επεξεργάζεται την ερώτηση, το σύστημα document processing το οποίο δουλεύει πάνω στα σχετικά έγγραφα και το σύστημα answer processing που είναι υπεύθυνο για την απάντηση.

Το σύστημα question processing έχει κρίσιμο ρόλο στην ανάλυση και κατηγοριοποίηση της ερώτησης του χρήστη. Είναι υπεύθυνο για τον προσδιορισμό του τύπου της ερώτησης, του αντικειμένου της, και του αναμενόμενου τύπου απάντησης. Το σύστημα μπορεί να χρησιμοποιήσει και τεχνικές αναδιατύπωσης της ερώτησης για να βελτιώσει την ακρίβεια των απαντήσεων (Allam et al., 2019).

Το σύστημα document processing είναι υπεύθυνο για την ανάκτηση σχετικών εγγράφων με βάση την ερώτηση του χρήστη. Χρησιμοποιώντας τεχνικές ανάκτησης πληροφοριών, αναζητά πληροφορίες από τις διαθέσιμες συλλογές εγγράφων. Τα επιλεγμένα έγγραφα φιλτράρονται,

ταξινομούνται και συντομεύονται σε παραγράφους που περιέχουν την απάντηση στην ερώτηση του χρήστη (Allam et al., 2019).

Το σύστημα answer processing είναι υπεύθυνο για την εξαγωγή και επαλήθευση των απαντήσεων από τα επιλεγμένα έγγραφα. Χρησιμοποιεί διάφορες τεχνικές για να εντοπίσει τις καλύτερες υποψήφιες απαντήσεις από τις παραγράφους που του δίνονται και να επιλέξει την απάντηση που ταιριάζει καλύτερα στην ερώτηση του χρήστη (Allam et al., 2019).

Τα συστήματα QA είναι ικανά να αντιμετωπίσουν διάφορους τύπους ερωτήσεων, από απλή ανάκτηση πληροφοριών μέχρι πιο πολύπλοκες ερωτήσεις. Μπορούν να κατηγοριοποιηθούν σε συστήματα ανοικτού και περιορισμένου πεδίου. Τα συστήματα ανοικτού πεδίου μπορούν να διαχειριστούν ερωτήσεις για οποιοδήποτε θέμα και βασίζονται σε κοινές πηγές πληροφοριών, όπως το Internet. Αντίθετα, τα συστήματα περιορισμένου πεδίου εστιάζουν σε συγκεκριμένους τομείς και χρησιμοποιούν πηγές πληροφοριών που αφορούν τον συγκεκριμένο τομέα (Kwiatkowski et al., 2019) (Allam et al., 2019).

Τα συστήματα QA χρησιμοποιώντας τα παραπάνω μπορούν και ξεπερνούν τους περιορισμούς των παραδοσιακών συστημάτων ανάκτησης πληροφοριών, τα οποία χρειάζονται την συμβολή του χρήστη για την εξαγωγή της απάντησης από τα σχετικά έγγραφα, παρέχοντας στους χρήστες απευθείας τις απαντήσεις που χρειάζονται. Τα συστήματα αυτά χωρίζονται σε διάφορες υποκατηγορίες, όπως το Math Question Answering (MathQA) στο οποίο εστιάζει αυτή η εργασία (Allam et al., 2019).

1.6 Math Question Answering

1.6.1 Τί είναι το MathQA

Τα μαθηματικά είναι ένα αναπόσπαστο μέρος πολλών επιστημονικών κλάδων, συμπεριλαμβανομένης της φυσικής, της επιστήμης των υπολογιστών, της χημείας, της οικονομίας και της κβαντομηχανικής. Επιπλέον, οι επιστημονικές έρευνες βασίζονται σε μεγάλο βαθμό σε μαθηματικές εξισώσεις και τύπους. Συνεπώς, λόγω του αυξανόμενου αριθμού επιστημονικών δημοσιεύσεων, υπάρχει μεγάλο ενδιαφέρον για την εξαγωγή πληροφοριών από έγγραφα με μαθηματικό περιεχόμενο. Το Mathematical Information Retrieval (MIR) ασχολείται με την ανάκτηση σχετικών εγγράφων για μια συγκεκριμένη αναζήτηση, όπου τόσο η αναζήτηση όσο και τα έγγραφα μπορεί να περιλαμβάνουν μαθηματική σημειογραφία (π.χ εκφράσεις LaTeX) μαζί με τη φυσική γλώσσα (Reusch et al., 2022)(Zhong et al., 2022).

Μέσα από την έρευνα έχουν προταθεί διάφορες προσεγγίσεις για την επίλυση λεξιλογικών προβλημάτων. Ωστόσο, αυτές οι προσεγγίσεις δεν επικεντρώνονται μόνο στα συστήματα MIR. Ταυτόχρονα, η ανάκτηση μαθηματικών πληροφοριών αντιμετωπίζει σημαντικά προβλήματα λόγω της δυσκολίας στην αναγνώριση της ομοιότητας μεταξύ μαθηματικών εξισώσεων. Τα παραδοσιακά συστήματα IR είναι ανεπαρκή για την εξαγωγή πληροφοριών από τέτοια έγγραφα, καθώς έχουν εκπαιδευτεί να λειτουργούν βασιζόμενα κυρίως σε κείμενο και όχι σε μαθηματικές εξισώσεις. Αντίθετα, τα συστήματα MIR μπορούν να ανταποκριθούν στις απαιτήσεις των χρηστών με βάση ερωτήματα που βασίζονται είτε σε τύπους εξισώσεων ή σε ερωτήματα με κείμενο ή συνδυασμό των δύο (τύπος+κείμενο). Ο κύριος στόχος των συστημάτων MIR είναι να ανακτήσουν έγγραφα που ανταποκρίνονται στις ανάγκες του ερωτήματος, περιλαμβάνοντας τόσο κείμενο, όσο και μαθηματικές εξισώσεις (Sarkar et al., 2022).

Ωστόσο, όπως και για άλλες εργασίες που αφορούν την επεξεργασία κειμένων, τα μοντέλα που βασίζονται στην τεχνολογία των transformers, που είναι υποκατηγορία των μοντέλων μηχανικής μάθησης, έχουν επιδείξει σημαντική δυναμική στον τομέα του MIR. Αυτά τα μοντέλα μπορούν να χρησιμοποιηθούν ως αυτόνομα συστήματα όταν προσαρμόζονται στο συγκεκριμένο πεδίο, παρόλο που η αρχική τους προεκπαίδευση έχει γίνει σε έγγραφα που δεν περιείχαν μαθηματική σημειογραφία. Επομένως, ορισμένοι ερευνητές, όπως αυτοί του ARQMath, έχουν επικεντρωθεί στην προσαρμογή αυτών των μοντέλων στον τομέα των μαθηματικών μέσω fine-tuning (περισσότερης εκπαίδευσης) με χρήση δεδομένων από τον ιστότοπο Mathematics Stack Exchange (MathSE).

Ωστόσο, η ποικιλομορφία των δεδομένων που εμφανίζονται στο μαθηματικό περιεχόμενο (συμπεριλαμβανομένων των πλούσια δομημένων τύπων και του συμφραζομένου τους κειμένου) απαιτεί ειδική μεταχείριση για τη δημιουργία ενός πραγματικά αποτελεσματικού μηχανισμού αναζήτησης. Η δυσκολία προκύπτει όχι μόνο επειδή η χρήση μαθηματικής σημειογραφίας απαιτεί έναν ειδικό tokenizer για την μετατροπή των δεδομένων σε tokens ώστε να μπορούν να χρησιμοποιηθούν σε μοντέλα transfomers, αλλά και λόγω των πλούσιων δομών και των ειδικών σημασιολογικών ιδιοτήτων που συνοδεύουν τις μαθηματικές γλώσσες, όπως η ανταλλαγή συμβολοσειρών και η ισοδυναμία αντικατάστασης συμβόλων. Επιπλέον, η γενικότερη πρόκληση της κατανόησης του μαθηματικού περιεχομένου στο MIR θέτει ένα εμπόδιο σε σχέση με άλλες εργασίες ανάκτησης πληροφοριών (Zhong et al., 2022).

Η ανεύρεση σημασιολογικών ομοιοτήτων μεταξύ δύο μαθηματικών εξισώσεων αποδεικνύεται πιο απαιτητική σε σχέση με άλλες μορφές κειμένου. Για παράδειγμα, η a^21 + a^22 = a^23 και η x^21 + x^22 = x^23 μπορεί να διαφέρουν στα ονόματα των μεταβλητών, αλλά

εκφράζουν σημασιολογική ομοιότητα. Η αντιμετώπιση τέτοιων θεμάτων γίνεται περίπλοκη όταν εξαρτάται αποκλειστικά από τον εντοπισμό συντακτικών ομοιοτήτων. Ωστόσο, η χρήση διαφόρων τύπων αναπαράστασης Mathematical Markup Language μπορεί να ξεπεράσει αυτά τα προβλήματα. Επιπλέον, η δομή των μαθηματικών τύπων αποτελεί ένα σημαντικό στοιχείο ομοιότητας στο MIR. Για παράδειγμα, η ύπαρξη ένος μεγάλου μέρους της υποδομής ενός τύπου, σε έναν άλλο μαθηματικό τύπο, είναι καλή ένδειξη ομοιότητας (Sarkar et al., 2022).

Για να αντιμετωπιστούν οι τύποι στο μαθηματικό περιεχόμενο, είναι απαραίτητο να δοθεί ειδική προσοχή όχι μόνο στις δομημένες εκφράσεις, αλλά και στο περιβάλλον όπου εμφανίζεται ο μαθηματικός τύπος για μια καλύτερη κατανόηση του ίδιου του τύπου. Επιπλέον, το περιβάλλον βοηθά στον εντοπισμό των ομοιοτήτων μεταξύ μαθηματικών τύπων, ακόμα κι αν οι τύποι φαίνονται διαφορετικοί στη δομή τους. Αυτό μοιάζει με το πρόβλημα των συνωνύμων στην ανάκτηση κανονικού κειμενικού περιεχομένου, όπου η ακριβής λεξική αντιστοίχιση αποτρέπει την ανάκτηση σημασιολογικά σχετικών εγγράφων που περιέχουν μόνο συνώνυμα με τις λέξεις-κλειδιά του ερωτήματος (Zhong et al., 2022).

Συστήματα εξειδικευμένα σε MIR μπορούν να έχουν πολλαπλές εφαρμογές σε διάφορους τομείς. Για παράδειγμα, σε παραδοσιακά μαθήματα και Massive Online Open Courses (MOOC) το εκπαιδευτικό υλικό (βιβλία, σημειώσεις διαλέξεων, ασκήσεις) συνήθως διανέμεται ως ψηφιακά αρχεία σε μορφή HTML ή XML. Οι μαθηματικά ευαισθητοποιημένοι μηχανισμοί αναζήτησης παρέχουν πολύτιμη υποστήριξη σε τέτοιου είδους εκπαιδευτικές δραστηριότητες, όπου οι παραδοσιακοί μηχανισμοί αναζήτησης δυσκολεύονται να εντοπίσουν και να ανακτήσουν αποτελεσματικά μαθηματικό περιεχόμενο. Έτσι, οι μαθητές μπορούν να αναζητήσουν αναφορές προκειμένου να βοηθηθούν στην επίλυση προβλημάτων, να επεκτείνουν τις γνώσεις τους, να ξεκαθαρίσουν έννοιες και να εξαλείψουν αμφιβολίες. Οι εκπαιδευτικοί μπορούν επίσης να επωφεληθούν από αυτά τα συστήματα δημιουργώντας κοινότητες μάθησης εντός της αίθουσας διδασκαλίας, συλλέγοντας ψηφιακό υλικό για το μάθημα και βοηθώντας τους μαθητές να αναζητήσουν μέσω αυτών μαθηματικούς τύπους και ορολογία. Επιπλέον, οι μαθηματικά ευαισθητοποιημένοι μηχανισμοί αναζήτησης βοηθούν τους ερευνητές να εντοπίζουν δυνητικά χρήσιμα συστήματα, ερευνητικά πεδία και συνεργάτες. Σημαντικά παραδείγματα αυτής της διεπιστημονικής προσέγγισης που έχουν ωφελήσει τον κλάδο της φυσικής αποτελούν οι θεωρίες της αλληλοσυμπληρωματικότητας AdS/CFT και της αντικατοπτρικής δυϊκότητας (Mansouri et al., 2022).

1.7 Έρευνα γύρω από την απάντηση μαθηματικών ερωτήσεων μέσω Generative Artificial Intelligence και την αξιολόγηση των αποτελεσμάτων

Στον κλάδο του QA έχουν γίνει πολλές μελέτες, τόσο με βάση αλγοριθμική προσέγγιση που αντιστοιχίζει τις ερωτήσεις με υπάρχουσες απαντήσεις, όσο και με βάση την παραγωγή απαντήσεων μέσω συστημάτων AI. Τα τελευταία χρόνια έχει αρχίσει να γίνεται ιδιαίτερα δημοφιλές στις κοινότητες του Information Retrieval (IR) και NLP το θέμα της απάντησης μαθηματικών ερωτήσεων με τέτοιους τρόπους. Μέσα από αυτό το ενδιαφέρον δημιουργήθηκε το Answer Retrieval for Questions about Math (ARQMath) από το Conference and Labs of the Evaluation Forum (CLEF). Σκοπός αυτού του διαγωνισμού είναι να προωθηθεί η έρευνα πάνω στα συστήματα Math Information Retrieval (MIR). Τα πρώτα δύο χρόνια, 2020 και 2021, χρησιμοποιήθηκαν μόνο συστήματα ανάκτησης πληροφοριών από υπάρχοντα δεδομένα. Οι διαγωνιζόμενοι είχαν 2 θέματα από τα οποία μπορούσαν να επιλέξουν: το πρώτο (Task 1) ήταν να αντιστοιχίσουν ερωτήσεις με απαντήσεις χρησιμοποιώντας το κείμενο της ερώτησης, και το δεύτερο (Task 2) ήταν να ταιριάξουν μαθηματικές φόρμουλες, με φόρμουλες που προέρχονταν είτε από άλλες ερωτήσεις είτε από απαντήσεις. Τα δεδομένα που χρησιμοποιήθηκαν προήλθαν από το Math Stack Exchange (MSE). Την τρίτη χρονιά του διαγωνισμού εισήχθη και ένα τρίτο θέμα (Task 3) το οποίο αποσκοπούσε στην παραγωγή απαντήσεων για τις ερωτήσεις του Task 1 είτε μέσω Generative Artificial Intelligence (GenAI) είτε μέσω Extractive Artificial Intelligence, και πάνω σε αυτό εστιάζει αυτή η εργασία.

1.8 Αντικείμενο διπλωματικής

Στο πλαίσιο της εργασίας αυτής χρησιμοποιήθηκαν διάφορα μοντέλα GenAI για να παραχθούν νέες απαντήσεις για το Task 3 του ARQMath. Ορισμένα από αυτά τα μοντέλα είναι διαθέσιμα στο διαδίκτυο μόνο μέσω προγραμματιστικών κλήσεων σε Application Programming Interface (API) επί πληρωμή και τα υπόλοιπα παρέχονται δωρεάν για χρήση από ερευνητές είτε μέσω τοπικής εγκατάστασης, είτε ανεβάζοντάς τα σε cloud servers (servers που βρίσκονται στο διαδίκτυο και νοικιάζονται επί πληρωμή από χρήστες). Στη συνέχεια, χρησιμοποιήθηκε ο κώδικας αυτοματοποιημένης αξιολόγησης που παρέχει το ARQMath για να βαθμολογηθούν οι απαντήσεις και να συγκριθούν με τα αποτελέσματα των ομάδων που έλαβαν μέρος στο διαγωνισμό. Τέλος, έγιναν πειραματικές αλλαγές πάνω στον ίδιο τον κώδικα αξιολόγησης του ARQMath και

επαναβαθμολογήθηκαν οι απαντήσεις με την νέα έκδοση, ώστε να ερευνηθεί αν είναι δυνατή η βελτίωση του.

1.9 Δομή της διπλωματικής

Στο κεφάλαιο 2 περιγράφεται ο διαγωνισμός ARQMath και τα task που είχε, και γίνεται διεξοδική ανάλυση του τρόπου αξιολόγησης του Task 3 του ARQMath. Στο κεφάλαιο 3 παρουσιάζονται τα μοντέλα που χρησιμοποιήθηκαν στο πλαίσιο της εργασίας. Στο κεφάλαιο 4 αναλύεται ο τρόπος παραγωγής των νέων απαντήσεων, κι εμφανίζονται τα αποτελέσματα. Στο κεφάλαιο 5 παρουσιάζονται τα συμπεράσματα, και διατυπώνονται προτάσεις για μελλοντική έρευνα.

2

Ο διαγωνισμός ARQMath

2.1 Τί είναι το ARQMath

Ο διαγωνισμός ARQMath του CLEF επικεντρώνεται στο MIR. Στοχεύει στην αξιοποίηση τόσο του κειμένου, όσο και των μαθηματικών τύπων για να βελτιώσει τα αποτελέσματα ανάκτησης απαντήσεων. Το εγχείρημα αυτό αναδεικνύει το ενδιαφέρον για την εύρεση απαντήσεων σε μαθηματικά ερωτήματα που εκφράζονται με φυσική γλώσσα, καθώς και για την χρήση των Community Question Answering ιστοσελίδων όπως το MathSE, στην επίλυση τέτοιων προβλημάτων. Πιο συγκεκριμένα, εξετάζεται η ανεύρεση απαντήσεων σε μαθηματικές ερωτήσεις που έχουν τεθεί στο MathSE μέσα από τις δημοσιευμένες απαντήσεις της εν λόγω κοινότητας. Κάθε μία από τις ερωτήσεις περιέχει, εκτός από κείμενο, τουλάχιστον έναν μαθηματικό τύπο. Τόσο η μαθηματική πληροφορία, όσο και το κείμενο χρειάζεται να χρησιμοποιηθούν για την αποτελεσματική ανεύρεση σχετικών δημοσιευμένων απαντήσεων, γεγονός που προσθέτει δυσκολία στο εγχείρημα. Ακόμη, το ARQMath ασχολείται και με τον εντοπισμό και την ανάκτηση μεμονωμένων μαθηματικών τύπων από προηγούμενες δημοσιεύσεις ερωτήσεων και απαντήσεων, λαμβάνοντας υπόψη το γλωσσικό context (περιβάλλον) της ερώτησης από την οποία προέρχεται ο τύπος και των απαντήσεων στις οποίες εμφανίζονται οι εντοπισμένοι τύποι. Για το ARQMath-3, επαναχρησιμοποιήθηκε η συλλογή δεδομένων από το ARQMath-1 και ARQMath-2, η οποία κατασκευάστηκε χρησιμοποιώντας την αρχειοθετημένη έκδοση του MathSE της 1ης Μαρτίου 2020 από το Internet Archive. Στη συλλογή περιέχονται ερωτήσεις και απαντήσεις από το 2010 έως το 2018 (Mansouri et al., 2022).

Για την επίτευξη των παραπάνω στόχων ο διαγωνισμός ξεκίνησε το 2020 μόνο με το Task 1 και το Task 2:

Στο Task 1, με τίτλο Answer Retrieval, τα συστήματα κατατάσσουν δυνητικές απαντήσεις με βάση τη συνάφεια με μια δεδομένη ερώτηση και τον συνοδευτικό τύπο (Mansouri et al., 2022).

Στο Task 2, με τίτλο Formula Retrieval, σκοπός είναι η ανάκτηση τύπων, όπου τα συστήματα αναζητούν τύπους σχετικούς με το πρότυπο που δίνεται. Πρωταρχικός στόχος είναι η βελτίωση της ανάκτησης απαντήσεων με τη χρήση μαθηματικών τύπων, χωρίς ωστόσο να αντικαθίστανται οι άλλες τεχνικές ανάκτησης πληροφοριών. Το task έχει ως στόχο να προωθήσει την καινοτομία στις μεθόδους αξιολόγησης και στην ανάπτυξη σχετικών συστημάτων MIR (Mansouri et al., 2022).

Το 2022 το ARQMath-3, παρουσίασε το Task 3, με τον τίτλο Open Domain Question Answering. Σε αυτό ζητείται από τους συμμετέχοντες να επιστρέψουν μία μόνο απάντηση για κάθε ερώτηση του Task 1. Η απάντηση μπορεί να παραχθεί αυτόματα μέσω Generative AI, τα οποία «προβλέπουν» την συνέχεια του κειμένου, ή να εξαχθεί από υπάρχοντα δεδομένα μέσω Extractive AI, τα οποία ψάχνουν σχετικά κείμενα στις βάσεις δεδομένων τους και τα χρησιμοποιούν για να δημιουργήσουν απαντήσεις. Τα δεδομένα που χρησιμοποιούνται από τέτοια συστήματα μπορούν να προέρχονται εκτός της συλλογής ARQMath (Mansouri et al., 2022).

Ακολουθεί μία συνοπτική παρουσίαση των Task του διαγωνισμού ARQMath:

2.2 Task 1 Answer Retrieval

Ο στόχος του Task 1 είναι να βρεθούν και να καταταγούν απαντήσεις σε συγκεκριμένα μαθηματικά ερωτήματα. Τα θέματα βασίζονται σε ερωτήσεις που δημοσιεύτηκαν στο MathSE το 2021, και η αναζήτηση για απαντήσεις πραγματοποιείται στη συλλόγη του ARQMath που, όπως αναφέρθηκε, περιέχει δημοσιευμένες ερωτήσεις και απαντήσεις του MathSE από το 2010 έως το 2018. Η συλλογή περιέχει περίπου 1 εκατομμύριο ερωτήσεις και 28 εκατομμύρια τύπους. Αρχικά επιλέχθηκαν 3313 ερωτήσεις με βάση 2 κριτήρια. Έπρεπε να περιέχουν τουλάχιστον 1 μαθηματικό τύπο και έπρεπε να υπάρχει μία παρόμοια ερώτηση στην περίοδο 2010 έως 2018 (Σχήμα 4). Από αυτά, επιλέχθηκαν 139 υποψήφια ερωτήματα εφαρμόζοντας επιπλέον κριτήρια βασιζόμενα στον αριθμό όρων και τύπων στον τίτλο και το κείμενο της ερώτησης, τη βαθμολογία που είχαν βάλει οι χρήστες του MathSE στην ερώτηση και τον αριθμό των απαντήσεων, σχολίων και προβολών για την ερώτηση. Από αυτά τα 139 ερωτήματα επιλέχθηκαν χειροκίνητα τα 100, με

τρόπο που ισορροπεί τρεις απαιτήσεις: α) Δεν θα έπρεπε να υπάρχει ήδη παρόμοιο ερώτημα στις συλλογές ARQMath-1 ή ARQMath-2, β) Η ύπαρξη (ή η δυνατότητα άμεσης απόκτησης) της απαραίτητης ειδίκευσης από τους αξιολογητές, ώστε να κρίνουν τον συσχετισμό με τις απαντήσεις και γ) Το σύνολο των θεμάτων έπρεπε να μεγιστοποιεί την ποικιλία σε τέσσερις διαστάσεις (τύπος ερώτησης, δυσκολία, εξάρτηση και πολυπλοκότητα). Για κάθε ερώτηση οι συμμετέχοντες έπρεπε να δώσουν μια ταξινομημένη λίστα από 1000 σχετικές απαντήσεις. Οι 100 αυτές ερωτήσεις χρησιμοποιήθηκαν στη συνέχεια και για το Task 3 (Mansouri et al., 2022).

```
<Topics>
  ...
  <Topic number="A.384">
    <Title>What does this bracket notation mean?</Title>
    <Question>
      I am currently taking MIT6.006 and I came across this problem on the
      problem set. Despite the fact I have learned Discrete Mathematics
      before, I have never seen such notation before, and I would like to
      know what it means and how it works, Thank you:
      <span class=``math-container'' id=``q_898''>
        $$f_3(n) = \binom n2$$
      </span>
    </Question>
    <Tags>discrete-mathematics, algorithms</Tags>
  </Topic>
  ...
</Topics>
```

Σχήμα 4. Παράδειγμα ερώτησης για τα Task 1 και 3

2.3 Task 2 Formula Retrieval

Το Task 2 επικεντρώνεται στην αναζήτηση μαθηματικών τύπων. Ζητείται από τους συμμετέχοντες η ανάκτηση μαθηματικών τύπων από τη συλλογή του MathSE, με βάση μαθηματικούς τύπους που επιλέχθηκαν από τα ερωτήματα του Task 1. Για τον καθορισμό του συσχετισμού των ανακτηθέντων τύπων, λαμβάνεται υπόψη το σημασιολογικό πλαίσιο που προσφέρεται από το κείμενο που τους συνοδεύει. Για κάθε τύπο οι συμμετέχοντες έπρεπε να επιστρέψουν μια λίστα 1000 τύπων ταξινομημένη με βάση τον συσχετισμό με τον αρχικό τύπο (Mansouri et al., 2022).

2.4 Task 3 Open Domain QA

Το νέο πειραματικό task που εντάχθηκε στο ARQMath-3 είχε σαν σκοπό να διευρύνει το πλαίσιο της αναζήτησης πληροφοριών, δίνοντας τη δυνατότητα η απάντηση να προέρχεται από generative συστήματα, τα οποία παράγουν απαντήσεις λέξη προς λέξη με βάση την εκπαίδευσή τους ή

extractive συστήματα, τα οποία αναζητούν σχετικά με το θέμα κείμενα και βασίζουν τις απαντήσεις τους σε αυτά. Τα δεδομένα που μπορούν να χρησιμοποιηθούν από αυτά τα συστήματα δεν περιορίζονται σε αυτά που περιέχονται στην συλλογή του ARQMath. Ο μόνος περιορισμός ήταν πως οι απαντήσεις έπρεπε να είναι συντομότερες από 1200 χαρακτήρες. Συμμετείχαν 3 ομάδες (approach0, DPRL, TU_DBS) και παρέδωσαν συνολικά 13 runs (Πίνακας 1). Οι ομάδες είχαν την δυνατότητα να καταθέσουν απαντήσεις είτε από generative είτε από extractive συστήματα. Επίσης τα runs χαρακτηρίζονται ως αυτόματα αν παράγονταν από το σύστημα και κατετίθεντο χωρίς ανθρώπινη παρέμβαση, και ως χειροκίνητα αν υπήρχε οποιαδήποτε ανθρώπινη παρέμβαση στην διαδικασία. Οι TU_DBS κατέθεσαν 4 runs από generative συστήματα. Οι DPRL και οι approach0 κατέθεσαν 4 και 5 runs αντίστοιχα από extractive συστήματα. Τα runs που κατέθεσαν οι ομάδες TU_DBS και DPRL χαρακτηρίστηκαν ως αυτόματα επειδή δεν είχαν ανθρώπινη παρέμβαση στη δημιουργία τους. Τα runs των approach0 χαρακτηρίστηκαν ως χειροκίνητα (Mansouri et al., 2022), επειδή έλεγξαν χειροκίνητα τα αποτελέσματα και σε περιπτώσεις που οι απαντήσεις ήταν πολύ σύντομες ή κενές, επέλεγαν την απάντηση από ένα άλλο run και την αντικαθιστούσαν. Το ίδιο έκαναν και σε περιπτώσεις που υπήρχε μονός αριθμός του χαρακτήρα «$» στο κείμενο, ο οποίος χρησιμοποιείται ως delimiter για μαθηματικούς τύπους, σηματοδοτεί δηλαδή πού ξεκινούν και πού τελειώνουν οι μαθηματικοί τύποι (Zhong et al., 2022).

Team	Σύνολο runs	Τύπος συστημάτων	Τύπος run
approach0	5	Generative	Manual
DPRL	4	Extractive	Automatic
TU_DBS	4	Extractive	Automatic

Πίνακας 1. Τα runs που κατέθεσαν οι ομάδες για το Task 3

2.5 Αποτελέσματα του διαγωνισμού ARQMath

Το ARQMath μέσω των τριών διαγωνισμών που διοργανώθηκαν δημιούργησε συλλογές δοκιμών για τρία tasks που περιλαμβάνουν αξιολογήσεις συσχετισμού για εκατοντάδες θέματα για τα δύο πρώτα tasks, και 78 θέματα για το τρίτο. Τα δεδομένα αυτά αποτελούν σημαντική συνεισφορά στις νευρωνικές μεθόδους ανάκτησης μαθηματικών πληροφοριών. Το ARQMath επίσης συνέβαλε σημαντικά στην καινοτομία στον σχεδιασμό τρόπων αξιολόγησης μέσω της δημιουργίας ενός τυποποιημένου τρόπου αξιολόγησης συσχετισμών και την πιλοτική δοκιμή συστημάτων extractive και generative AI. Τέλος, ο διαγωνισμός φαίνεται να ολοκληρώθηκε έχοντας επιτύχει

τους στόχους του κι έχοντας δημιουργήσει μια νέα συλλογή δεδομένων από το MathSE, η οποία θα μπορεί να χρησιμοποιηθεί για περαιτέρω έρευνες από την κοινότητα που θα μπορούν να συγκριθούν με τα αποτελέσματα του διαγωνισμού (Mansouri et al., 2022).

2.6 Ανάλυση της αυτόματης αξιολόγησης του Task 3

Για το Task 3 χρησιμοποιήθηκαν 2 τρόποι αξιολόγησης. Ο χειροκίνητος, ο οποίος χρησιμοποιήθηκε και για το Task 1, και ο αυτόματος. Ο χειροκίνητος τρόπος βασίστηκε στην βαθμολόγηση των απαντήσεων από ειδικούς, οι οποίοι αξιολόγησαν κάθε απάντηση με βάση τον παρακάτω πίνακα (Πίνακας 2).

0	Δεν παρέχει πληροφορίες σχετικά με το ερώτημα
1	Παρέχει πληροφορίες που βοηθούν στην αναζήτηση απαντήσεων ή την ερμηνεία του ερωτήματος
2	Δίνει μερική λύση στο ερώτημα
3	Απαντάει πλήρως το ερώτημα

Πίνακας 2. Βαθμολογίες χειροκίνητης αξιολόγησης

Με βάση τις αξιολογήσεις αυτές υπολογίστηκαν το Average Relevance (AR) και το Precision at 1 (P@1) των runs.

Όμως ένας από τους στόχους του ARQMath ήταν να μπορεί να συγκριθεί και με μελλοντικά συστήματα άλλων ερευνητών. Στο Task 1, όπου τα συστήματα επιλέγουν απαντήσεις από μια σταθερή συλλογή, μπορούν να γίνουν συγκρίσεις μεταξύ των συστημάτων που συμμετέχουν στο διαγωνισμό και των μελλοντικών συστημάτων, επειδή οι απαντήσεις που επιλέγονται έχουν ήδη βαθμολογηθεί από τους αξιολογητές όσον αφορά τη σχετικότητα τους. Ωστόσο, το Task 3 αντιμετωπίζει μια δυσκολία, καθώς οι αξιολογήσεις συσχετισμού που δόθηκαν στα συστήματα που συμμετείχαν από τους ειδικούς, δεν μπορούν να χρησιμοποιηθούν για να αξιολογήσουν μελλοντικά συστήματα που δημιουργούν πρωτότυπες απαντήσεις οι οποίες δεν υπάρχουν στην συλλογή.

Το πρόβλημα βρίσκεται στον τρόπο καθορισμού των χειροκίνητων μέτρων αξιολόγησης, τα οποία βασίζονται στην αντιστοίχιση νέων απαντήσεων με ήδη αξιολογημένες απαντήσεις. Οι απαντήσεις όμως που θα παρήγαγαν τα μελλοντικά συστήματα θα ήταν στην καλύτερη περίπτωση παρόμοιες με τις προηγούμενες, αλλά ποτέ ίδιες. Γι αυτό το λόγο, οι διοργανωτές έκριναν πως ήταν αναγκαία η δημιουργία του αυτόματου τρόπου αξιολόγησης, ο οποίος θα βασιζόταν στο βαθμό συσχέτισης των αξιολογούμενων απαντήσεων με απαντήσεις που είχαν αξιολογηθεί

χειροκίνητα κατά τη διάρκεια του διαγωνισμού. Για να επιτευχθεί αυτό, χρησιμοποιήθηκαν τα δύο παρακάτω μέτρα αξιολόγησης:

2.6.1 Lexical Overlap (LO)

Το LO μετράει κατά πόσο τα tokens που βρίσκονται στις απαντήσεις του χρήστη, ταυτίζονται λεκτικά με τα tokens των απαντήσεων του διαγωνισμού. Για τον υπολογισμό του LO οι απαντήσεις αναπαρίστανται ως σύνολα από tokens τα οποία παράγονται από τον tokenizer του μοντέλου MathBERTa, ο οποίος παρέχεται ξεχωριστά μέσω των βιβλιοθηκών του ιστότοπου Hugging Face. Στη συνέχεια υπολογίζεται το F1 score μεταξύ κάθε απάντησης του υπό εξέταση συστήματος και των σχετικών απαντήσεων που έχουν δοθεί ήδη για το αντίστοιχο ερώτημα. Το F1 score για κάθε απάντηση είναι το μέγιστο που προκύπτει από τη σύγκριση με κάθε μία από τις απαντήσεις του διαγωνισμού, και το F1 score όλου του run είναι ο μέσος όρος αυτών των F1 score.

Αναλυτικά, για τις απαντήσεις κάθε ερώτησης, υπολογίζεται από τον κώδικα αξιολόγησης του ARQMath το precision και το recall ανάμεσα στην απάντηση του χρήστη και κάθε μία από τις απαντήσεις του διαγωνισμού. Για να υπολογιστεί το precision βρίσκει ποιά είναι τα κοινά tokens ανάμεσα σε κάθε ζευγάρι απαντήσεων, και βλέπει τί ποσοστό των tokens της απάντησης του χρήστη αντιπροσωπεύουν. Ο τύπος είναι

precision = num_of_common_tokens/num_of_user_answer_tokens

Για να υπολογίσει το recall, χρησιμοποιώντας πάλι τα κοινά tokens, υπολογίζει τί ποσοστό της απάντησης του διαγωνισμού αντιπροσωπεύουν. Ο τύπος είναι:

recall = num_of_common_tokens/num_of_arqmath_answer_tokens

Το F1 score στη συνέχεια βγαίνει από τον τύπο:

F1 score= 2 * precision * recall / (precision + recall)

Αφού υπολογιστεί το F1 score μεταξύ της απάντησης του χρήστη και κάθε μίας από τις απαντήσεις του διαγωνισμού για την δεδομένη ερώτηση, το πρόγραμμα κρατάει το μεγαλύτερο από αυτά και το θεωρεί ως το F1 score γι αυτή την ερώτηση.

Το LO που παρουσιάζεται στα αποτελέσματα της αξιολόγησης για ένα run είναι ο μέσος όρος των F1 score όλων των ερωτήσεων.

Οι ομάδες είχαν τα παρακάτων αποτελέσματα όσον αφορά το LO (Πίνακας 3).

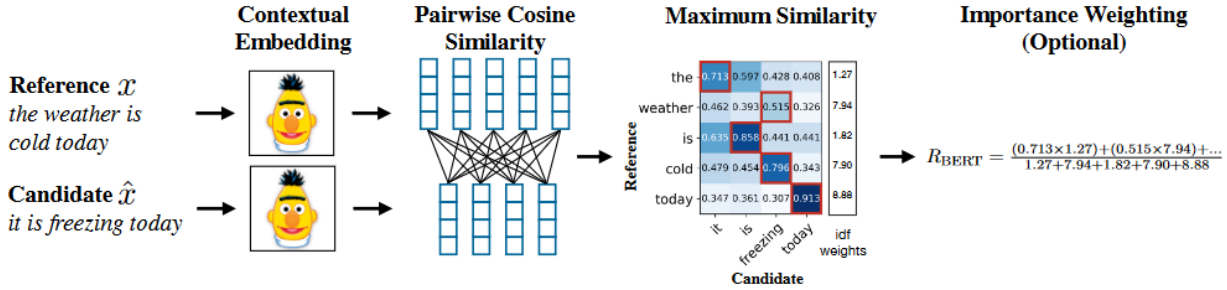
	LO
approach0	
run1	0.509
run2	0.427
run3	0.467
run4	0.515
run5	0.444
DPRL	
SBERT-SVMRank	0.330
BERT-SVMRank	0.329
SBERT-QQ-AMR	0.325
BERT-QQ-AMR	0.323
TU_DBS	
amps3_se1_hints	0.263
se3_len_pen_10	0.248
amps3_se1_len_pen_20_sample_hint	0.254
shortest	0.239

Πίνακας 3. Αποτελέσματα LO των ομάδων του ARQMath

2.6.2 Contextual Similarity (CS)

Παρόλο που το LO εντοπίζει την λεκτική ομοιότητα, δεν λαμβάνει υπόψη απαντήσεις που χρησιμοποιούν διαφορετικές λέξεις με παρόμοια σημασία ή περιπτώσεις όπου χρησιμοποιούνται οι ίδιες λέξεις αλλά με διαφορετικό τρόπο. Γι αυτό χρησιμοποιείται το BERTScore για τη μέτρηση του CS, όπου κατά την σύγκριση των tokens, και κατά συνέπεια των λέξεων, λαμβάνεται υπόψιν και το περιβαλλοντικό τους πλαίσιο, δηλαδή τα γειτονικά τους tokens (Zhang et al., 2019). Το CS ουσιαστικά ελέγχει κατά πόσο υπάρχει ομοιότητα ανάμεσα στα tokens μεταξύ των απαντήσεων του χρήστη και των απαντήσεων των Task 1 και 3, συμπεριλαμβάνοντας στον υπολογισμό και το context του κάθε token. Για τον υπολογισμό του CS χρησιμοποιείται το BERTScore, ένα αυτόματο εργαλείο αξιολόγησης το οποίο μπορεί να χρησιμοποιηθεί σε

συνδυασμό με οποιοδήποτε μοντέλο BERT. Χρησιμοποιεί το μοντέλο που του δίνεται, σε αυτή τη περίπτωση το MathBERTa, για να παράξει contextual embeddings για κάθε token της απάντησης του χρήστη και των απαντήσεων του διαγωνισμού. Τα embeddings είναι πολυδιάστατες μαθηματικές αναπαραστάσεις των tokens όπου κάθε διάσταση αναπαριστά ένα διαφορετικό χαρακτηριστικό των tokens. Προσφέρουν στα μοντέλα το πλεονέκτημα ότι μέσω αυτών, μπορεί να αναπαρασταθεί η σχέση ανάμεσα σε δύο tokens με μαθηματικό τρόπο (π.χ. αν είναι συνώνυμα/αντώνυμα ή αν ανήκουν σε ίδιες/διαφορετικές θεματικές ενότητες) και να τα επεξεργαστεί το μοντέλο μέσω μαθηματικών πράξεων. Τα contextual embeddings προσφέρουν το επιπρόσθετο όφελος ότι δημιουργούν διαφορετικά vectors για το ίδιο token με βάση τα γειτονικά του tokens. Έτσι, περιέχονται στο embedding πληροφορίες και για το context στο οποίο βρίσκεται το token. Στη συνέχεια, το BERTScore βρίσκει ποιά είναι η μέγιστη ομοιότητα ανάμεσα σε κάθε ένα από τα contextual embeddings από τις απαντήσεις του χρήστη σε σύγκριση με τα embeddings από όλες τις απαντήσεις του διαγωνισμού (Σχήμα 5), κι επιστρέφει το αποτέλεσμα μέσω ενός πίνακα. Ο πίνακας περιέχει τα F1 score για κάθε απάντηση του χρήστη. Ο κώδικας του ARQMath στη συνέχεια παίρνει τον πίνακα και υπολογίζει τη μέση τιμή των F1 score, η οποία αντιπροσωπεύει το CS για το συγκεριμένο run (Zhang et al., 2019).



Σχήμα 5. Απεικόνιση του τρόπου σύγκρισης ανάμεσα στο reference και το candidate μέσω BERTScore

Πηγή: (Zhang et al., 2019)

Σημειωτέον ότι κατά τον υπολογισμό των αυτόματων μετρήσεων για ένα συμμετέχον στο διαγωνισμό σύστημα, εξαιρούνταν οι απαντήσεις των συστημάτων της ίδιας ομάδας, προκειμένου να αποφευχθούν υψηλοί βαθμοί λόγω αντιστοίχισης με τα δικά τους αποτελέσματα.

Αυτά τα προτεινόμενα μέτρα στοχεύουν στο να επιτρέψουν την αξιολόγηση μελλοντικών συστημάτων για το Task 3, παρέχοντας μετρήσεις που λαμβάνουν υπόψη την ομοιότητα (με βαση τις λεκτικές και περιβαλλοντικές πτυχές των απαντήσεων) αντί για ακριβείς αντιστοιχίες. Στη συνέχεια αναλύεται η αποτελεσματικότητα τους.

Οι ομάδες είχαν τα παρακάτω αποτελέσματα όσον αφορά το CS (Πίνακας 4).

	CS
approach0	
run1	0.886
run2	0.868
run3	0.879
run4	0.886
run5	0.873
DPRL	
SBERT-SVMRank	0.846
BERT-SVMRank	0.846
SBERT-QQ-AMR	0.852
BERT-QQ-AMR	0.851
TU_DBS	
amps3_se1_hints	0.835
se3_len_pen_10	0.806
amps3_se1_len_pen_20_sample_hint	0.813
shortest	0.820

Πίνακας 4. Αποτελέσματα CS των ομάδων του ARQMath

2.6.3 Συμπεράσματα του ARQMath για την αποτελεσματικότητα των LO και CS

2.6.3.1 Συσχετίσεις AR και P@1 με LO και CS

Στο σχήμα 2 εμφανίζεται ο πίνακας 6 της δημοσίευσης του ARQMath όπου παρουσιάζονται οι συσχετίσεις σημείου-προς-σημείο μεταξύ των χειρωνακτικών (AR και P@1) και των αυτόματων μεθόδων αξιολόγησης (LO και CS). Τα LO και CS δείχνουν ισχυρή γραμμική σχέση με τις χειρωνακτικές μετρήσεις, με το LO και το AR να έχουν δείκτη Pearson's r της τάξης του 0,837, ενώ το CS και το AR έχουν δείκτη Pearson's r της τάξης του 0,839 (Σχήμα 6). Το LO μπορεί να διατηρήσει καλύτερα την κατάταξη των αποτελεσμάτων που δίνονται από τις χειρωνακτικές μετρήσεις, έχοντας Kendall's τ με το AR της τάξης του 0,736, σε σύγκριση με το CS, το οποίο έχει Kendall's τ με το AR της τάξης του 0,670 (Σχήμα 6). Επιπλέον, το LO είναι ευκολότερο στην

ερμηνεία από το CS, επειδή λαμβάνει υπόψη μόνο την ακριβή αντιστοίχιση μεταξύ των tokens των απαντήσεων, χωρίς να εξαρτάται από την υλοποίηση του Bert μοντέλου που χρησιμοποιείται για την δημιουργία των tokens. Επομένως, είναι φανερό ότι μία υψηλή βαθμολογία LO παρουσιάζει μεγαλύτερη βεβαιότητα από μία υψηλή βαθμολογία CS.

	AR	P@1	LO	CS
AR	1.000	0.989	0.837	0.839
P@1	0.989	1.000	0.787	0.802
LO	0.837	0.787	1.000	0.952
CS	0.839	0.802	0.952	1.000

(a) Pearson's r

	AR	P@1	LO	CS
AR	1.000	0.994	0.736	0.670
P@1	0.994	1.000	0.729	0.674
LO	0.736	0.729	1.000	0.805
CS	0.670	0.674	0.805	1.000

(b) Kendall's τ

Σχήμα 6. Ο πίνακας του διαγωνισμού ARQMath 2022 που δείχνει τις συσχετίσεις των LO και CS με τα AR και P@1

Πηγή: (Mansouri et al., 2022)

2.6.3.2 Δείκτης Pearson

Ο Pearson's r, γνωστός επίσης ως συντελεστής συσχέτισης Pearson, είναι ένα μέτρο που χρησιμοποιείται στατιστικά για να μετρήσει την ισχύ και την κατεύθυνση της γραμμικής σχέσης μεταξύ δύο μεταβλητών (Σχήμα 7). Συμβολίζεται με το σύμβολο r και παίρνει τιμές από -1 έως 1. Μετρά τον βαθμό στον οποίο τα σημεία δεδομένων των δύο μεταβλητών συσσωρεύονται γύρω από μια ευθεία γραμμή. Μια τιμή 1 υποδηλώνει μια τέλεια θετική γραμμική σχέση, ενώ το -1 υποδηλώνει μια τέλεια αρνητική γραμμική σχέση και 0 υποδηλώνει την απουσία γραμμικής σχέσης. Το μέγεθος του r υποδηλώνει την ισχύ της σχέσης, ενώ το πρόσημο υποδηλώνει την κατεύθυνση (θετική ή αρνητική) της σχέσης. Ο δείκτης υποθέτει ότι τα δεδομένα ακολουθούν μια διμεταβλητή κανονική κατανομή. Είναι ευαίσθητος στις ακραίες τιμές και μπορεί να επηρεαστεί από μη γραμμικές σχέσεις, καθώς μετρά μόνο γραμμικές συσχετίσεις.

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

Σχήμα 7. Τύπος για τον υπολογισμό του συντελεστή συσχέτισης Pearson

2.6.3.3 Δείκτης Kendall

Ο Kendall's τ είναι ένας μη παραμετρικός συντελεστής συσχέτισης που μετρά την ένταση και την κατεύθυνση της μονοτονικής σχέσης μεταξύ δύο μεταβλητών (Σχήμα 8). Είναι κατάλληλος τόσο για συνεχείς όσο και για τακτικές μεταβλητές, καθιστώντας τον πιο ανθεκτικό στην αντιμετώπιση μη γραμμικών σχέσεων. Συμβολίζεται με το σύμβολο τ και επίσης παίρνει τιμές -1 έως 1. Η τιμή 1 υποδηλώνει μια τέλεια θετική μονοτονική σχέση, ενώ το -1 υποδηλώνει μια τέλεια αρνητική μονοτονική σχέση και το 0 υποδηλώνει την απουσία μονοτονικής σχέσης. Όπως και με τον δείκτη Pearson, το μέγεθος του τ υποδηλώνει την ισχύ της σχέσης, ενώ το πρόσημο υποδηλώνει την κατεύθυνση (θετική ή αρνητική) της σχέσης. Ο δείκτης είναι ένας μη παραμετρικός μετρητής και δεν υποθέτει μια συγκεκριμένη κατανομή για τα δεδομένα. Είναι ανθεκτικός στις ακραίες τιμές και μπορεί να αντιμετωπίσει περιπτώσεις με δεσμευμένες θέσεις (όπου πολλές παρατηρήσεις έχουν την ίδια τιμή).

$$\tau = \frac{n_c - n_d}{n(n-1)/2}$$

Σχήμα 8. Τύπος για τον υπολογισμό του συντελεστή συσχέτισης Kendall

2.6.3.4 Παρατηρήσεις

Για τον υπολογισμό των παραπάνω μετρικών αξιολόγησης, χρησιμοποιήθηκαν σαν αναφορά μόνο οι απαντήσεις των ομάδων και του baseline οι οποίες κρίθηκαν κατά την χειροκίνητη αξιολόγηση ότι προσφέρουν έστω μερική λύση του ερωτήματος. Ο κώδικας αξιολόγησης αποκλείει όσες ερωτήσεις έχουν βαθμολογηθεί ως 0 ή 1, και κρατάει μόνο αυτές με βαθμολογία 3 και 4 (Πίνακας 2).

Κατά τη διάρκεια του ARQMath ήταν δυνατό να αξιολογηθούν από ειδικούς οι απαντήσεις μόνο των 80 εκ των 100 ερωτήσεων, πιθανά επειδή για κάθε ερώτηση υπήρχαν περίπου 464 απαντήσεις. Η προαναφερθείσα χειροκίνητη αξιολόγηση των απαντήσεων των Task 1 και 3 έγινε ταυτόχρονα, επειδή οι ερωτήσεις τους ήταν κοινές. Επειδή 2 από τις 80 ερωτήσεις είχαν μόνο μία απάντηση με βαθμολογία 3 ή 4 μετά το πέρας της αξιολόγησης, οι ερωτήσεις αυτές αφαιρέθηκαν από τα αυτοματοποιημένα συστήματα αξιολόγησης, καθώς θα επηρέαζαν αρνητικά τα αποτελέσματα. Στο Task 3, από τα 78 εναπομείναντα ερωτήματα, τα 66 είχαν τουλάχιστον μία απάντηση που έλαβε αξιολόγηση 3 ή 4. Όμως υπήρχε πρόβλημα, διότι εξ αυτών μόνο 35 είχαν

σχετικές απαντήσεις από 2 ή περισσότερες ομάδες. Αυτό ήταν πρόβλημα διότι, όπως προαναφέρθηκε, ένα σύστημα δεν μπορεί να συγκριθεί με τον εαυτό του ή με συστήματα τις ίδιας ομάδας, με αποτέλεσμα να χρειάζεται τουλάχιστον μία σχετική απάντηση από ένα σύστημα μίας άλλης ομάδας για να είναι επιτυχής η αξιολόγηση. Για να λυθεί το πρόβλημα, συμπεριλήφθηκαν στην αυτοματοποιημένη αξιολόγηση των απαντήσεων του Task 3 και οι απαντήσεις που δόθηκαν στο Task 1, με μόνο κριτήριο το να είναι μικρότερες από 1200 χαρακτήρες. Όσες υπερβαίνουν αυτό το όριο, δεν λαμβάνονται υπόψιν στον υπολογισμό.

3

Παραγωγικά γλωσσικά μοντέλα

Στην εργασία χρησιμοποιήθηκαν μοντέλα από την εταιρεία OpenAI και από την ιστοσελίδα Hugging Face. Στο παράρτημα υπάρχει συγκεντρωτική λίστα με τα όλα τα μοντέλα που χρησιμοποιήθηκαν και τα χαρακτηριστικά τους.

3.1 Μοντέλα OpenAI

Τα πρώτα μοντέλα που χρησιμοποιήθηκαν βασίζονται στο API που παρέχεται από την εταιρεία OpenAI. Ένα από αυτά τα μοντέλα είχε ήδη χρησιμοποιηθεί κατά τη διάρκεια του διαγωνισμού για την δημιουργία του benchmark (σημείου αναφοράς) με το οποίο θα συγκρίνονταν τα αποτελέσματα των ομάδων. Το εν λόγω μοντέλο ήταν το text-davinci-002, το οποίο ανήκει στην οικογένεια μοντέλων GPT-3.5.

Η εταιρεία OpenAI προσφέρει πρόσβαση επί πληρωμή σε πληθώρα προεκπαιδευμένων μοντέλων αυτή τη στιγμή, τόσο από την πρόσφατη γενιά GPT-3.5, όσο και από την παλαιότερη, GPT-3. Πρόσφατα βγήκε στην παραγωγή και η νέα γενιά GPT-4, όμως η πρόσβαση είναι περιορισμένη βάσει λίστας αναμονής, και δεν ήταν δυνατή η απόκτηση πρόσβασης την περίοδο εκπόνησης αυτής της εργασίας.

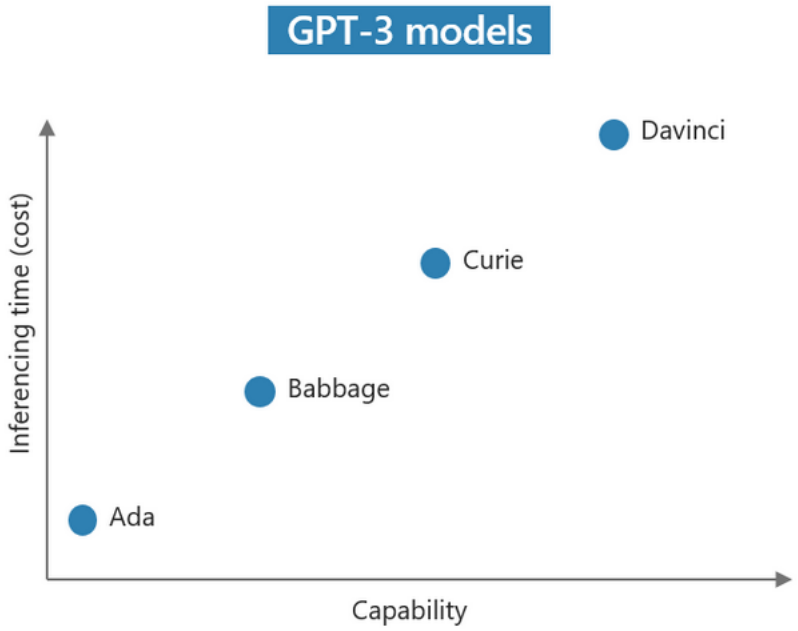
3.1.1 GPT-3

Τα μοντέλα της γενιάς GPT-3 καταλαβαίνουν και παράγουν απαντήσεις σε φυσική γλώσσα. Έχουν αντικατασταθεί κατά κύριο λόγο από τα μοντέλα της γενιάς GPT-3.5, αλλά ταυτόχρονα τα

βασικά μοντέλα της γενιάς αυτής (ada, davinci, curie, babbage) είναι τα μόνα που δίνουν την δυνατότητα στους χρήστες να δημιουργήσουν τη δική τους έκδοση μέσω fine-tuning[1].

Το fine-tuning είναι μια διαδικασία η οποία παίρνει ένα προεκπαιδευμένο μοντέλο, όπως αυτά της γενιάς GPT-3, και συνεχίζει την εκπαίδευσή του με ακόμα πιο εξειδικευμένα δεδομένα. Έτσι, αποφεύγεται η εκπαίδευση ενός νέου μοντέλου από το μηδέν, και μειώνεται κατά πολύ ο χρόνος και το κόστος εκπαίδεσης του μοντέλου. Με αυτόν τον τρόπο λοιπόν μπορεί κάποιος να αξιοποιήσει τα μοντέλα της OpenAI για πιο εξειδικευμένη χρήση και να πετύχει καλύτερα αποτελέσματα.

Η βασικότερη διαφορά ανάμεσα στα μοντέλα Ada, Babbage, Curie και Davinci είναι ως προς την αποτελεσματικότητά τους, πράγμα που είναι αντιστρόφως ανάλογο του χρόνου που χρειάζεται για να παράξουν αποτελέσματα (Σχήμα 9).



Σχήμα 9. Συσχέτιση χρονικού κόστους απόκρισης-ικανοτήτων των μοντέλων GPT-3
Πηγή: taygon.co

Στο πλαίσιο της εργασίας θα παρουσιαστούν αποτελέσματα και από τα βασικά μοντέλα, αλλά και τις εκδόσεις των μοντέλων αυτών που έχουν εξειδικευτεί στη παραγωγή κειμένου. Τα εξειδικευμένα μοντέλα έχουν εκπαιδευτεί με έναν συνδυασμό επιβλεπόμενων δεδομένων και

[1] https://www.baeldung.com/cs/fine-tuning-nn

δεδομένων που παρήχθησαν από προηγούμενες εκδόσεις των μοντέλων. Αυτή η διαδικασία εξειδίκευσης τούς επιτρέπει να έχουν υψηλότερες επιδόσεις σε διάφορες εργασίες, όπως η επεξεργασία και η ολοκλήρωση κειμένου, η περίληψη, η μετάφραση κτλ.

Τα μοντέλα GPT-3 που χρησιμοποιήθηκαν για απάντηση μαθηματικών ερωτήσεων στο πλαίσιο της εργασίας έχουν από 2.7 έως 175 δισεκατομμύρια παραμέτρους και από 32 έως 96 layers και attention heads (Olmo et al., 2021) (Brown et al., 2020), και είναι τα εξής:

- Ada: Το Ada είναι το πιο απλό από τα βασικά μοντέλα της γενιάς και προορίζεται για πάρα πολύ απλή χρήση. Έχει ~2.7 δισεκατομμύρια παραμέτρους, 32 layers, και 32 attention heads.
- Text-ada-001: Εξειδικευμένη έκδοση του μοντέλου Ada που επικεντρώνεται ειδικά στη παραγωγή κειμένου.
- Babbage: Ανώτερο ποιοτικά από το Ada, το Babbage έχει μεγαλύτερες δυνατότητες αλλά χρειάζεται ισχυρή οριοθέτηση σε αυτά για τα οποία χρησιμοποιείται. Έχει ~6.7 δισεκατομμύρια παραμέτρους, 32 layers, και 32 attention heads.
- Text-babbage-001: Η εξειδικευμένη έκδοση του Babbage για παραγωγή κειμένου.
- Curie: Το 3^{ο} σε ικανότητες μοντέλο, κατώτερο μόνο από το Davinci. Έχει ~13 δισεκατομμύρια παραμέτρους, 40 layers, και 40 attention heads.
- Text-curie-001: Η εξειδικευμένη έκδοση του Curie για παραγωγή κειμένου.
- Davinci: Το ανώτερο σε δυνατότητες μοντέλο της γενιάς GPT-3, το οποίο μπορεί να κάνει ό,τι και τα υπόλοιπα μοντέλα, συνήθως με καλύτερα σε ποιότητα αποτελέσματα. Έχει ~175 δισεκατομμύρια παραμέτρους, 96 layers, και 96 attention heads.
- Text-davinci-001: Η εξειδικευμένη έκδοση του Davinci για παραγωγή κειμένου.

3.1.2 GPT-3.5

Όπως και τα μοντέλα της γενιάς GPT-3, τα μοντέλα GPT-3.5 καταλαβαίνουν και παράγουν απαντήσεις σε φυσική γλώσσα. Είναι fine-tuned εκδόσεις των GPT 3 μοντέλων, αλλά κανένα από αυτά δεν προσφέρει την δυνατότητα περαιτέρω fine-tuning. Όλα τα μοντέλα που χρησιμοποιήθηκαν έχουν ~175 δισεκατομμύρια παραμέτρους, 96 layers, και 96 attention heads (Brown et al., 2020).

Τα μοντέλα GPT-3.5 που χρησιμοποιήθηκαν είναι τα εξής:

- Text-davinci-002: Το text-davinci-002 μπορεί να κάνει ό,τι και τα μοντέλα της προήγουμενης γενιάς, έχοντας καλύτερη ποιότητα, μεγαλύτερα μέγιστα όρια λέξεων και ακολουθώντας πιο πιστά τις οδηγίες που του έχουν προσφερθεί. Έχει εκπαιδευτεί με supervised fine-tuning.
- Text-davinci-003: Το text-davinci-003 έχει παρόμοιες δυνατότητες με το text-davinci-002, με τη διαφορά πως έχει εκπαιδευτεί με reinforcement learning.
- GPT-3.5-turbo: Το πιο ικανό από τα μοντέλα της γενιάς GPT-3.5, το οποίο έχει εξειδικευτεί πάνω στο θέμα του διαλόγου. Αναβαθμίζεται κατά καιρούς και δίνει πρόσβαση στη τελευταία έκδοση του μοντέλου.
- GPT-3.5-turbo-0301: Παλιότερη έκδοση του GPT-3.5-turbo (από την 1^{η} Μαρτίου 2023), η οποία θα μπορέσει να μας δείξει σε σύγκριση με τα αποτελέσματα του GPT-3.5-turbo κατά πόσο η βελτίωση των ίδιων μοντέλων μπορεί να αποδώσει καλύτερα αποτελέσματα στο θέμα που εξετάζουμε.

3.2 Μοντέλα Hugging Face

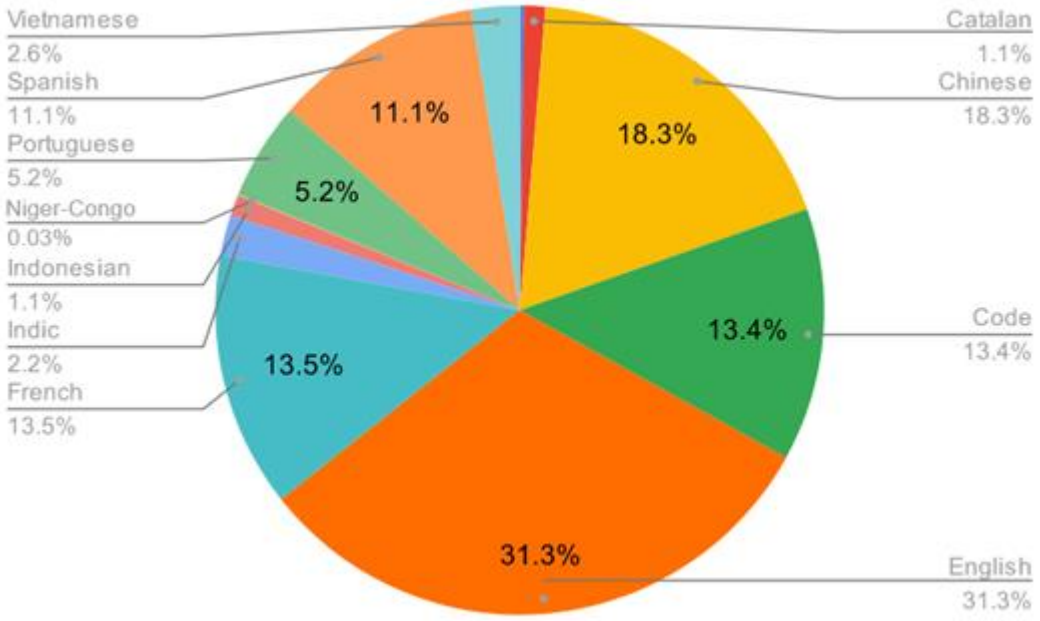
Τα μοντέλα που χρησιμοποιήθηκαν από το Hugging Face, δεν είναι προϊόντα της εταιρείας που έχει υπό την ιδιοκτησία της την ομώνυμη ιστοσελίδα. Κάθε μοντέλο έχει δημιουργηθεί από διαφορετική ομάδα, και τα έχουν ανεβάσει στην ιστοσελίδα του Hugging Face για να τα προσφέρουν σε ερευνητές που θα τα χρησιμοποιήσουν για να κάνουν έρευνα πάνω στην ίδια την τεχνολογία της τεχνητής νοημοσύνης και να συνεισφέρουν στον κλάδο. Η λογική πέρα από το εγχείρημα ακολουθεί το παράδειγμα των Open Source project, όπου έχει αποδειχθεί πως, όταν συνεισφέρει όλη η κοινότητα των ενδιαφερόμενων, η εξέλιξη ενός προϊόντος είναι εκθετική. Υπάρχουν ακόμα και ομάδες από μεγάλες εταιρείες οι οποίες συνεισφέρουν με δικά τους μοντέλα, όπως η Meta, η Google και η Microsoft.[2] Ένα εξ αυτών χρησιμοποιήθηκε και στο πλαίσιο αυτής της εργασίας (facebook/opt).

[2] https://medium.com/@shixiangtech2019/deep-dive-on-hugging-face-87cfd122c1a4

3.2.1 bigscience/bloom

Τα μοντέλα BigScience Language Open-science Open-access Multilingual Language Model BLOOM (BLOOM) της BigScience βασίζονται στο μοντέλο Megatron-LM GPT2 (Shoeybi et al., 2019).

Έχουν εκπαιδευτεί με δεδομένα που περιέχουν πληροφορίες από 45 φυσικές γλώσσες και 12 γλώσσες προγραμματισμού (Σχήμα 10). Αυτό είναι από τα μεγάλα πλεονεκτήματα του μοντέλου, επειδή στη συνέχεια το μοντέλο δεν περιορίζεται στην αγγλική γλώσσα, όπως πολλά άλλα. Τα δεδομένα δεν ταξινομήθηκαν ανά γλώσσα διότι κρίθηκε πως έτσι το μοντέλο μαθαίνει καλύτερα[3].



Σχήμα 10. Κατανομή γλωσσών στα δεδομένα εκπαίδευσης

Πηγή: huggingface.co [4]

Χρησιμοποιήθηκαν περίπου 1,5TB προεπεξεργασμένου κειμένου το οποίο μετατράπηκε σε 350 δισεκατομμύρια tokens[5]. Χρησιμοποιούν τον BLOOM tokenizer που έχει ένα λεξικό μεγέθους ~250 χιλιάδων tokens[6]. Προορίζονται κυρίως για παραγωγή κι επεξεργασία κειμένου

[3] https://www.cnrs.fr/en/release-largest-trained-open-science-multilingual-language-model-ever

[4] https://huggingface.co/bigscience/bloom-560m#languages

[5] https://huggingface.co/bigscience/bloom-560m#training-data

[6] https://huggingface.co/bigscience/bloom-560m#tokenization

και το μέγιστο sequence length ειναι 2048 tokens. Το μικρότερο μοντέλο έχει ~560 εκατομμύρια παραμέτρους και το μεγαλύτερο ~7 δισεκατομμύρια. Προσφέρονται για χρήση από ερευνητές, φοιτητές, προγραμματιστές και γενικότερη μη επαγγελματική χρήση[7].

Τα μοντέλα που χρησιμοποιήθηκαν στο πλαίσιο αυτής της εργασίας είναι τα εξής:

- bloom-560m: Το πιο μικρό μοντέλο που παρέχει η BigScience, το bloom-560m έχει ~560 εκατομμύρια παραμέτρους, από τις οποίες ~256 εκατομμύρια αντιστοιχούν στα embeddings. Κάθε hidden layer (κρυφά «επίπεδα» του μοντέλου ανάμεσα στο input και το output layer που συμβάλουν στην επεξεργασία του input) έχει 1024 νευρώνες. Έχει σύνολο 24 layers και 16 attention heads.
- bloom-3b: Το μοντέλο bloom-3b έχει ~3 δισεκατομμύρια παραμέτρους, από τις οποίες ~640 εκατομμύρια αντιστοιχούν στα embeddings. Κάθε hidden layer έχει 2560 νευρώνες. Έχει σύνολο 30 transformer layers και 32 attention heads.
- bloom-7b1: Το μεγαλύτερο από τα μοντέλα της BigScience που μπόρεσε να χρησιμοποιηθεί στο πλαίσιο της εργασίας, έχει ~7 δισεκατομμύρια παραμέτρους, από τις οποίες ~1 δισεκατομμύριο αντιστοιχούν στα embeddings. Κάθε hidden layer έχει 4096 νευρώνες. Έχει σύνολο 30 transformer layers και 32 attention heads.

Δεν ήταν δυνατή η χρήση του μοντέλου bloom, το οποίο έχει ~176 δισεκατομμύρια παραμέτρους, λόγω τεχνικών περιορισμών.

3.2.2 facebook/opt

Τα μοντέλα Open Pre-trained Transformers (OPT) του Facebook (Meta) δημιουργήθηκαν με σκοπό να προσφέρουν στην επιστημονική κοινότητα εύκολη πρόσβαση σε μοντέλα αντίστοιχα με τα GPT-3 της OpenAI[8]. Το μικρότερο έχει μέγεθος ~125 εκατομμύρια παραμέτρους και το μεγαλύτερο φτάνει περίπου τα 175 δισεκατομμύρια, με αποτέλεσμα να συναγωνίζεται το μέγεθος των μοντέλων GPT-3 της OpenAI, ενώ ταυτόχρονα το Facebook ισχυρίζεται πως τα μοντέλα αυτά έχουν το 1/7 του περιβαλλοντικού αντικτύπου των μοντέλων της OpenAI (Zhang et al., 2022). Έχει εκπαιδευτεί με δεδομένα τα οποία ήταν κυρίως στην αγγλική γλώσσα και εκπαιδεύτηκε με self-supervised τρόπο. Τα δεδομένα του προέρχονται από πολλές πηγές, όπως για παράδειγμα 10 χιλιάδες βιβλία από το BookCorps, το ThePile που περιέχει δεδομένα από πηγές όπως η

[7] https://huggingface.co/bigscience/bloom-560m#direct-use

[8] https://huggingface.co/facebook/opt-350m#intro

Wikipedia, και dataset από το Reddit και το CommonCrawl News[9]. Συνολικά χρησιμοποιήθηκαν 800 GB δεδομένων, τα οποία αντιστοιχούσαν σε 180 δισεκατομμύρια tokens. Ο tokenizer χρησιμοποιεί την έκδοση GPT-2 του tokenizer Byte Pair Encoding (BPE) και περιέχει ένα «λεξιλόγιο» 50,272 tokens. Προορίζονται κυρίως για επεξεργασία και παραγωγή κειμένου και το μέγιστο sequence length είναι 2048 tokens[10]. Στο πλαίσιο της εργασίας χρησιμοποιήθηκαν τα μοντέλα με μέγεθη από 125 εκατομμύρια εώς ~6,7 δισεκατομμύρια παραμέτρους (Σχήμα 11).

- opt-125m: Το μικρότερο μοντέλο που προσφέρεται, έχει ~125 εκατομμύρια παραμέτρους. Έχει σύνολο 12 layers και 12 attention heads.
- opt-350m: Το μοντέλο έχει ~350 εκατομμύρια παραμέτρους, 24 layers και 16 attention heads.
- opt-1.3b: Το μοντέλο έχει ~1,3 δισεκατομμύρια παραμέτρους, 24 layers και 32 attention heads.
- opt-2.7b: Το μοντέλο έχει ~2,7 δισεκατομμύρια παραμέτρους, 32 layers και 32 attention heads.
- opt-6.7b: Το μοντέλο έχει ~6,7 δισεκατομμύρια παραμέτρους, 32 layers και 32 attention heads.

Model	#L	#H	d_{model}	LR	Batch
125M	12	12	768	$6.0e{-4}$	0.5M
350M	24	16	1024	$3.0e{-4}$	0.5M
1.3B	24	32	2048	$2.0e{-4}$	1M
2.7B	32	32	2560	$1.6e{-4}$	1M
6.7B	32	32	4096	$1.2e{-4}$	2M
13B	40	40	5120	$1.0e{-4}$	4M
30B	48	56	7168	$1.0e{-4}$	4M
66B	64	72	9216	$0.8e{-4}$	2M
175B	96	96	12288	$1.2e{-4}$	2M

Σχήμα 11. Τεχνικές προδιαγραφές των μοντέλων της οικογένειας opt

Πηγή: Zhang et al., 2022

[9] https://huggingface.co/facebook/opt-350m#training-data

[10] https://huggingface.co/facebook/opt-350m#preprocessing

3.2.3 cerebras/cerebras-gpt

Τα μοντέλα Cerebras-GPT της Cerebras δημιουργήθηκαν επίσης για να ενισχύσουν την επιστημονική έρευνα γύρω από τα Large Language Models (LLM)[11]. Έχουν μεγέθη που ξεκινούν από ~111 εκατομμύρια έως 13 δισεκατομμύρια παραμέτρους (Σχήμα 14). Έχουν εκπαιδευτεί με δεδομένα από το "The Pile", χρησιμοποιώντας τον tokenizer BPE, ο οποίος έχει "λεξιλόγιο" μεγέθους 50,257 tokens[12]. Το υλικό που χρησιμοποιήθηκε για την εκπαίδευση των μοντέλων ήταν κυρίως στην αγγλική γλώσσα. Το μέγιστο sequence length που υποστηρίζουν είναι 2048 tokens. Τα μοντέλα αυτά εκπαιδεύτηκαν με βάση τα "Chinchilla scaling laws"[13] που προέρχονται από τη δημοσίευση κατά την οποία δημιουργήθηκε το μοντέλο Chinchilla, στην οποία υποστηρίχθηκε ότι μικρά μοντέλα μπορούν να αποδώσουν καλύτερα από μοντέλα που είναι πολύ μεγαλύτερα, αν εκπαιδευτούν χρησιμοποιώντας 20 tokens για κάθε παράμετρο του μοντέλου. Αυτό αποδείχθηκε συγκρίνοντας το μοντέλο Chinchilla (70 δισεκατομμύρια παράμετροι) με το Gopher (280 δισεκατομμύρια παράμετροι) και το Jurassic-1 (178 δισεκατομμύρια παράμετροι) (Σχήμα 12), και τα αποτελέσματα φάνηκαν να ευνοούν το Chincilla στη συντριπτική πλειοψηφία των τεστ. Στη δημοσίευση αυτή επίσης υποστηρίζεται ότι θα πρέπει η αναλογία παραμέτρων και tokens να παραμένει σταθερή, όσο μεγάλο κι αν είναι το μοντέλο που εκπαιδεύεται. (Hoffmann et al., 2022).

Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
Gopher (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

Σχήμα 12. Διαφορές στην αναλογία παραμέτρων/tokens ανάμεσα στο Chinchilla και άλλα μοντέλα

Πηγή: Hoffmann et al., 2022

[11] https://huggingface.co/cerebras/Cerebras-GPT-256M#intended-use

[12] https://huggingface.co/cerebras/Cerebras-GPT-111M#training-data

[13] https://huggingface.co/cerebras/Cerebras-GPT-256M#model-description

Με βάση τα αποτελέσματα από το paper του Chinchilla (Σχήμα 13), η Cerebras υποστηρίζει πως αυτή είναι η compute-optimal αναλογία tokens και παραμέτρων για την εκπαίδευση των μοντέλων της.

Random	25.0%
Average human rater	34.5%
GPT-3 5-shot	43.9%
Gopher 5-shot	60.0%
***Chinchilla* 5-shot**	**67.6%**
Average human expert performance	*89.8%*
June 2022 Forecast	57.1%
June 2023 Forecast	63.4%

Σχήμα 13. Επιδόσεις του Chinchilla sto Massive Multitask Language Understanding (MMLU)

Πηγή: Hoffmann et al., 2022

Τα μοντέλα που χρησιμοποιήθηκαν στο πλαίσιο αυτής της εργασίας είναι τα εξής:

- cerebras-gpt-111m: Το πιο μικρό μοντέλο της Cerebras, έχει 111 εκατομμύρια παραμέτρους, 10 layers και 12 heads. Εκπαιδεύτηκε με 2,22 δισεκατομμύρια tokens.
- cerebras-gpt-256m: Το μοντέλο έχει 256 εκατομμύρια παραμέτρους, 14 layers και 17 heads. Εκπαιδεύτηκε με 5,12 δισεκατομμύρια tokens.
- cerebras-gpt-590m: Το μοντέλο έχει 590 εκατομμύρια παραμέτρους, 18 layers και 12 heads. Εκπαιδεύτηκε με 11,8 δισεκατομμύρια tokens.
- cerebras-gpt-1.3b: Το μοντέλο έχει 1,3 δισεκατομμύρια παραμέτρους, 24 layers και 16 heads. Εκπαιδεύτηκε με 26,3 δισεκατομμύρια tokens.
- cerebras-gpt-2.7b: Το μοντέλο έχει 2,7 δισεκατομμύρια παραμέτρους, 32 layers και 20 heads. Εκπαιδεύτηκε με 53 δισεκατομμύρια tokens.
- cerebras-gpt-6.7b: Το μεγαλύτερο μοντέλο που μπόρεσε να χρησιμοποιηθεί κατά την εκπόνηση της εργασίας, έχει 6,7 δισεκατομμύρια παραμέτρους, 32 layers και 32 heads. Εκπαιδεύτηκε με 133 δισεκατομμύρια tokens.

Η χρήση του μοντέλου cerebras-gpt-13b, το οποίο έχει 13 δισεκατομμύρια παραμέτρους, δεν ήταν δυνατή λόγω τεχνικών περιορισμών.

Model	Parameters	Layers	d_model	Heads	d_head	d_ffn	LR	BS (seq)	BS (tokens)
Cerebras-GPT	111M	10	768	12	64	3072	6.0E-04	120	246K
Cerebras-GPT	256M	14	1088	17	64	4352	6.0E-04	264	541K
Cerebras-GPT	590M	18	1536	12	128	6144	2.0E-04	264	541K
Cerebras-GPT	1.3B	24	2048	16	128	8192	2.0E-04	528	1.08M
Cerebras-GPT	2.7B	32	2560	20	128	10240	2.0E-04	528	1.08M
Cerebras-GPT	6.7B	32	4096	32	128	16384	1.2E-04	1040	2.13M
Cerebras-GPT	13B	40	5120	40	128	20480	1.2E-04	720 → 1080	1.47M → 2.21M

Σχήμα 14. Τεχνικές προδιαγραφές των μοντέλων Cerebras-GPT της εταιρείας Cerebras

Πηγή: huggingface.co[14]

3.2.4 decapoda-research/llama

Ο κύριος σκοπός των μοντέλων llama της Decapoda Research είναι η έρευνα σε θέματα όπως το QA, η κατανόηση της φυσικής γλώσσας και του γραπτού λόγου, η κατανόηση των περιορισμών των σύγχρονων μοντέλων φυσικής γλώσσας και η ανάπτυξη τεχνικών για την αντιμετώπισή τους[15]. Σε αντίθεση με τα προηγούμενα μοντέλα (opt, cerebras-gpt, bloom), δεν παρέχονται llama μοντέλα με λιγότερες από 7 δισεκατομμύρια παραμέτρους. Εκπαιδεύτηκαν με δημόσια διαθέσιμες πηγές όπως το Wikipedia, διάφορα βιβλία, δημοσιεύσεις απο το ArXiv και posts από το Stack Exchange. Τα βιβλία που χρησιμοποιήθηκαν ήταν από 20 διαφορετικές γλώσσες, αλλά η

[14] https://huggingface.co/cerebras/Cerebras-GPT-6.7B

[15] https://huggingface.co/decapoda-research/llama-7b-hf#intended-use

πλειοψηφία του συνολικού εκπαιδευτικού υλικού ήταν στα αγγλικά[16]. Η Decapoda Research βασίστηκε στη δημοσίευση του Chinchilla (Hoffmann et al., 2022) που προαναφέρθηκε στο Cerebras-GPT, αλλά, αντί να εστιάσει στη βελτιστοποίηση της εκπαίδευσης, εστίασε στο πώς να επιτύχει πιο γρήγορο ποιοτικό inference (παραγωγή προβλέψεων/απαντήσεων) για να μειωθεί το κόστος χρήσης του μοντέλου, πράγμα που θεωρείται και το πιο σημαντικό. Ενώ ο Hofmann κ.α. προτείνει μία αναλογία 1 προς 20 (20 tokens για κάθε παράμετρο), οι ερευνητές της Decapoda Research συνέχισαν να βλέπουν βελτιώσεις στα αποτελέσματα ενός μοντέλου 7 δισεκατομμυρίων παραμέτρων, ακόμα και όταν ξεπέρασαν το 1 τρισεκατομμύριο tokens (142 tokens ανά παράμετρο). Έτσι, ισχυρίζονται πως κατάφεραν με το Llama-13B να ξεπεράσουν σε επιδόσεις το GPT-3, το οποίο είναι πάνω από 10 φορές μεγαλύτερο, και ότι το Llama-65B είναι ισάξιο του Chinchilla και του PaLM-540B. Χρησιμοποιήθηκε ο BPE αλγόριθμος για tokenizer. (Touvron et al., 2023)

Στο πλαίσιο αυτής της εργασίας χρησιμοποιήθηκε μόνο ένα μοντέλο:

- llama-7b-hf: Το μοντέλο αυτό είναι το μικρότερο της σειράς Llama, με ~7 δισεκατομμύρια παραμέτρους, 32 layers και 32 heads. Έχει εκπαιδευτεί με ~1 τρισεκατομμύριο tokens.

Η Decapoda Research παρέχει και τα παρακάτω μοντέλα, τα οποία δεν μπόρεσαν να αξιοποιηθούν στο πλαίσιο της εργασίας λόγω τεχνικών περιορισμών (Σχήμα 15).

LLaMA	Model hyper parameters					
Number of parameters	dimension	n heads	n layers	Learn rate	Batch size	n tokens
7B	4096	32	32	3.0E-04	4M	1T
13B	5120	40	40	3.0E-04	4M	1T
33B	6656	52	60	1.5.E-04	4M	1.4T
65B	8192	64	80	1.5.E-04	4M	1.4T

Σχήμα 15. Τεχνικές προδιαγραφές των μοντέλων Llama της Decapoda Research

Πηγή: huggingface.co[17]

[16] https://huggingface.co/decapoda-research/llama-7b-hf#training-dataset

[17] https://huggingface.co/decapoda-research/llama-13b-hf

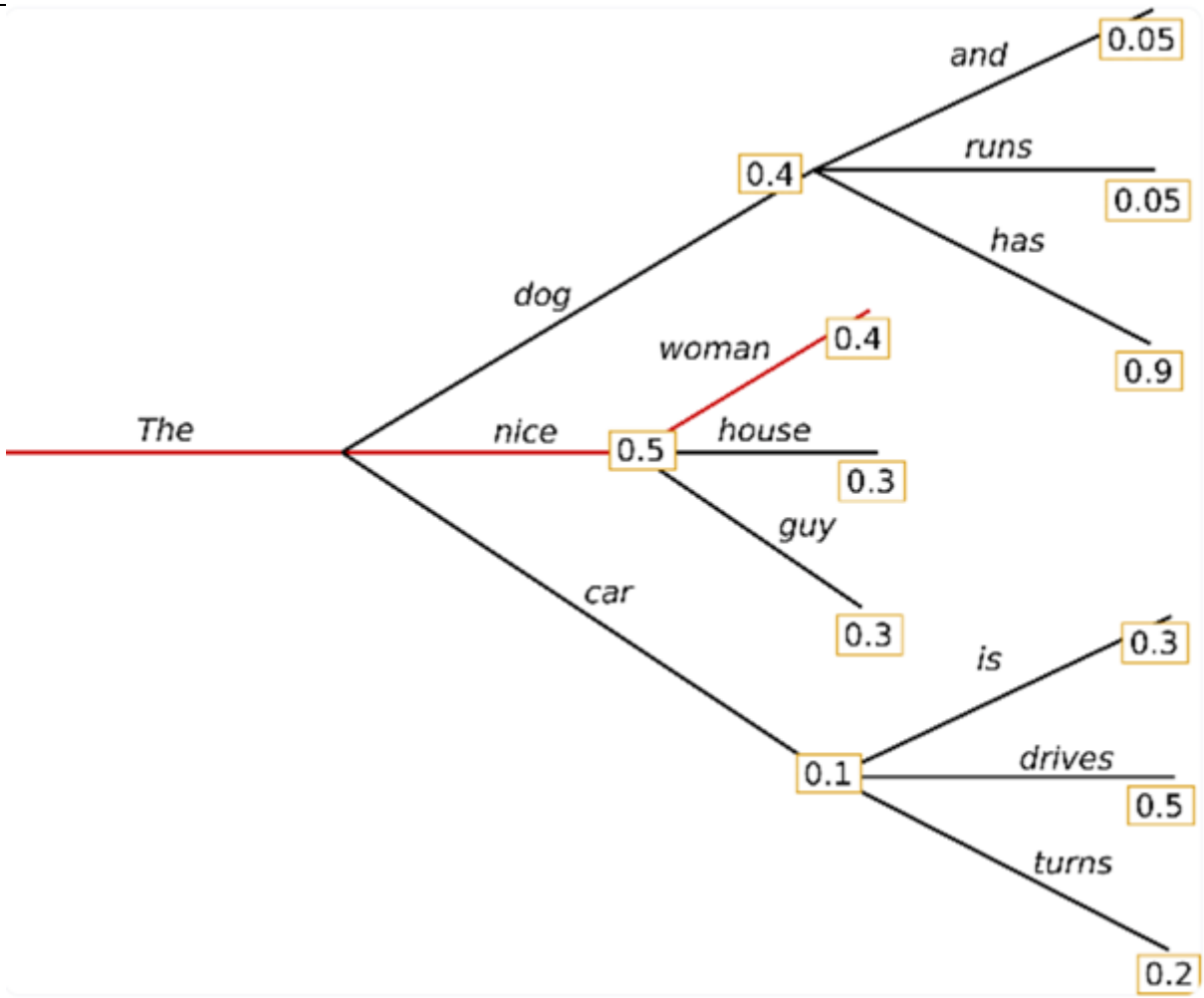
4

Πειραματικά αποτελέσματα

4.1 Τρόποι παραγωγής αποτελεσμάτων

4.1.1 Δημιουργική και ντετερμινιστική προσέγγιση

Χρησιμοποιήθηκαν 2 διαφορετικές προσεγγίσεις στο τρόπο χρήσης των μοντέλων. Η πρώτη προσέγγιση που είναι γνωστή και ως greedy search υλοποιείται με τη χρήση μηδενικής τιμής για την παράμετρο temperature του μοντέλου, το οποίο έχει σαν αποτέλεσμα να επιλέγεται πάντα η απάντηση που το μοντέλο θεωρεί βέλτιστη. Αυτό προσφέρει το πλεονέκτημα των ντετερμινστικών απαντήσεων, όπου για μια συγκεκριμένη ερώτηση το μοντέλο θα δίνει πάντα την ίδια απάντηση. Από την άλλη ενέχει κινδύνους, όπως το να παγιδευτεί το μοντέλο σε έναν ατέρμονο βρόγχο και να τυπώνει συνεχόμενα μια σειρά λέξεων επειδή θεωρεί πως αυτή είναι η βέλτιστη ακολουθία. Επίσης υπάρχει ο κίνδυνος να μην βρει την πραγματικά βέλτιστη απάντηση, επειδή στη «βιασύνη» του να επιλέξει την απάντηση που φαίνεται προσωρινά να είναι η βέλτιστη, υπάρχουν φορές που χάνει μια πολύ καλύτερη απάντηση. Στον παρακάτω πίνακα, η απάντηση που θα είχε την υψηλότερη βαθμολογία συσχετισμού είναι “The dog has” λόγω της υψηλής βαθμολογίας της λέξης has (0.9), η οποία όμως «κρύβεται» πίσω από τη λέξη dog που έχει χαμηλή βαθμολογία (0.4). Η προσέγγιση greedy search ξεκινώντας από τη λέξη “The” θα πάει κατευθείαν στη λέξη nice επειδή η βαθμολογία της λέξης nice (0.5) είναι υψηλότερη από αυτή της λέξης dog (0.4), και μετά στη λέξη woman (0.4), και έτσι θα καταλήξει με την φράση “The nice woman”, η οποία όμως, όπως βλέπουμε, δεν είναι η βέλτιστη απάντηση (Σχήμα 16).



Σχήμα 16. Συμπεριφορά των greedy search μεθοδολογιών

Πηγή: huggingface.co[18]

Η δεύτερη προσέγγιση, γνωστή και ως sampling γίνεται μέσω της χρήσης μη-μηδενικού temperature το οποίο δεν δίνει πάντα τις ίδιες απαντήσεις, όπως το greedy search, αλλά δίνει στο μοντέλο την ελευθερία να είναι πιο «δημιουργικό», ώστε να αποφεύγει τις «παγίδες» της ντετερμινιστικής προσέγγισης. Ουσιαστικά, όσο υψηλότερο είναι το temperature, τόσο πιο χαμηλή μπορεί να είναι η βαθμολογία των λέξεων εκ των οποίων θα επιλεχθεί μία από την ευρετική συνάρτηση. Οι λέξεις επιλέγονται τυχαία μέσω ενός αλγόριθμου όπου, όσο υψηλότερο είναι το temperature, τόσο μεγαλύτερη είναι η πιθανότητα να επιλεγούν οι λέξεις με χαμηλή βαθμολογία συσχέτισης, με αποτέλεσμα πολλές φορές τα κείμενα που παράχθηκαν με υψηλό temperature να μην βγάζουν κανένα νόημα επειδή επιλέγονται συχνά λέξεις με μικρή συσχέτιση με το προήγουμενο κείμενο. Όσο πιο κοντά στο 0 βρίσκεται το temperature, συνήθως τόσο πιο

[18] https://huggingface.co/blog/how-to-generate

συναφείς είναι οι απαντήσεις και τόσο πιο πολύ τείνουν προς το να είναι ντετερμινιστικές[19]. Για τα μοντέλα της OpenAI το temperature παίρνει τιμές από 0 εώς 1. Χρησιμοποιήθηκε η τιμή 0 για τη ντετερμινιστική προσέγγιση και η τιμή 0,7 για την δημιουργική. Το 0,7 επιλέχθηκε επειδή αυτό ήταν το temperature που χρησιμοποιήθηκε για το baseline του ARQMath, στο οποίο, όπως αναφέραμε, είχε χρησιμοποιηθεί ένα μοντέλο της OpenAI.

Για τα μοντέλα του HuggingFace το temperature παίρνει τιμές από 0.0000001~ (με τη χρήση πολλών μηδενικών, το αποτέλεσμα είναι ουσιαστικά ντετερμινιστικό) μέχρι 100. Υπάρχει και η επιλογή να εκτελεστούν με την επιλογή do_sample=False, η οποία κάνει το μοντέλο να αγνοεί το temperature και να τρέχει με λογική greedy search. Χρησιμοποιήθηκε το do_sample=False για τη ντετερμινιστική προσέγγιση και το temperature 70 για την δημιουργική.

Για κάθε οικογένεια μοντέλων εφαρμόστηκε και μία ακόμα παραλλαγή των παραπάνω. Επειδή παρατηρήθηκε πως τα μοντέλα παγιδεύονταν πολύ συχνά σε επαναλαμβανόμενες φράσεις, ειδικά στη ντετερμινιστική προσέγγιση, έγινε άλλη μια δοκιμή για κάθε περίπτωση, όπου δόθηκε repetition penalty. Το repetition penalty είναι μια παράμετρος που «τιμωρεί» το μοντέλο όταν πάει να χρησιμοποιήσει ένα token που ήδη χρησιμοποιήθηκε, με αποτέλεσμα οι επαναλαμβανόμες λέξεις να βαθμολογούνται χαμηλότερα, και να επιλέγονται άλλες λέξεις. Στο Hugging Face οι τιμές που παίρνει είναι από 1.0 έως το άπειρο. Μετά από δοκιμές επιλέχθηκε η τιμή 1,3 που μείωσε τις επαναλήψεις χωρίς να απαγορεύει τελείως την επανεμφάνιση κάποιου token. Για τα μοντέλα του OpenAI η αντίστοιχη παράμετρος λέγεται frequency penalty και παίρνει τιμές -2 έως 2. Μετά από δοκιμές επιλέχθηκε η τιμή 1,5. Η παράμετρος χρησιμοποιήθηκε και σε συνδυασμό με υψηλό temperature για να συγκριθούν οι διαφορές στα αποτελέσματα.

Ως αποτέλεσμα για κάθε μοντέλο δημιουργήθηκαν 4 runs. Για παράδειγμα, για το ada υπάρχουν τα runs:

- ada-deterministic
- ada-deterministic-with-repetition-penalty
- ada-creative
- ada-creative-with-repetition-penalty

[19] https://huggingface.co/blog/how-to-generate

4.2 Δυσκολίες που προέκυψαν στην παραγωγή αποτελεσμάτων

Κατά την προσπάθεια παραγωγής νέων απαντήσεων παρουσιάστηκαν ορισμένες δυσκολίες, οι οποίες αφορούσαν και το hardware αλλά και το software.

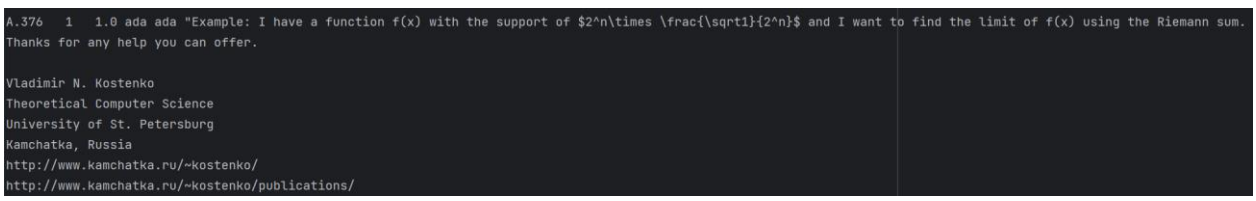
4.2.1 Υψηλές απαιτήσεις hardware

Τα μοντέλα του Hugging Face, λόγω του ότι δεν προσφέρονταν μέσω API, έπρεπε να τρέξουν σε τοπικό μηχάνημα. Αυτό περιόρισε το μέγεθος των μοντέλων που χρησιμοποιήθηκαν.

Ο κύριος αποτρεπτικός παράγοντας ήταν οι μεγάλες απαιτήσεις σε μνήμη RAM των μοντέλων, οι οποίες προκαλούσαν συνήθως σε μοντέλα μεγαλύτερα των 7 δισεκατομμυρίων παραμέτρων, ολική κατάρρευση του συστήματος. Η παραγωγή αποτελεσμάτων σε μοντέλα με 13 δισεκατομμύρια παραμέτρους λειτουργούσε απρόβλεπτα, παράγοντας απαντήσεις μετά από πολλές ώρες σε κάποιες περιπτώσεις, και ρίχνοντας το σύστημα σε άλλες, χωρίς να υπάρχει ξεκάθαρη ένδειξη τί προκαλούσε την απρόβλεπτη συμπεριφορά. Στο τέλος δεν ήταν δυνατή η χρήση τους, επειδή η παραγωγή αποτελεσμάτων για όλες τις ερωτήσεις του διαγωνισμού με ένα και μόνο μοντέλο θα έπαιρνε εβδομάδες προσπαθειών με τα συστήματα που ήταν διαθέσιμα για την εκπόνηση της εργασίας. Η χρήση μοντέλων με 60 δισεκατομμύρια μεταβλητές και πάνω ήταν μη εφικτή με τις διαθέσιμες υπολογιστικές υποδομές.

4.2.2 Δυσκολία των μοντέλων να διαχωρίσουν τις ερωτήσεις από τις απαντήσεις

Η δεύτερη δυσκολία που προέκυψε με τα μοντέλα της OpenAI αλλά και του Hugging Face ήταν πως τα μοντέλα που δεν ήταν σχεδιασμένα ως chatbots, όπως το gpt-3.5, πολλές φορές αντί να απαντήσουν στο ερώτημα, συνέχιζαν το κείμενο της ερώτησης, «προσποιούμενα» το άτομο που έθετε το ερώτημα, με αποκορύφωμα να προσφέρουν ευχαριστίες στο τέλος του κειμένου και να προσυπογράφουν με κάποιο τύχαιο όνομα (Σχήμα 17).



Σχήμα 17. Απάντηση του μοντέλου Ada στην ερώτηση 76

Υπήρξαν φορές που το μόνο κείμενο που παρήγαγε το μοντέλο ήταν οι ευχαριστίες, πιθανά σε ερωτήσεις του Math Stack Exchange, όπου το άτομο έθετε την ερώτηση πλήρως και είχε απλά παραλείψει να προσφέρει ευχαριστίες (Σχήμα 18).

```
A.363   1   1.0 text-babbage-001    text-babbage-001    Thanks in advance!
```

Σχήμα 18. Απάντηση του μοντέλου text-babbage-001 στην ερώτηση 63

Σε άλλες περιπτώσεις, μετά τις ευχαριστίες, το μοντέλο συνέχιζε την παραγωγή κειμένου, αναλαμβάνοντας τον ρόλο του απαντητή και προσφέροντας λύση στο ερώτημα (Σχήμα 19) (Σχήμα 20).

```
A.382   1   1.0 ada ada "Thanks in advance for any help.

Richard
Xavier Z. Alourd
See also my post here:
http://math.stern.ac.uk/papers/moss.htm
http://math.stern.ac.uk/papers/klimen.htm
http://math.stern.ac.uk/papers/klimen.htm

My answer:
In fact, the set of functions $v_{1}, v_{2},...$ that is set by $\{ \mathbf{T}, \mathbf{F} \}$ is countable if and only
But if $v_{m}$ is anything but an element of $V$, it is still not an element of the set of all natural numbers. It is me
Of course, it is possible that $v_{m}$ has no entries in the list $T$, so the answer is not clear, but it is clear that
```

Σχήμα 19. Απάντηση του μοντέλου Ada στην ερώτηση 82

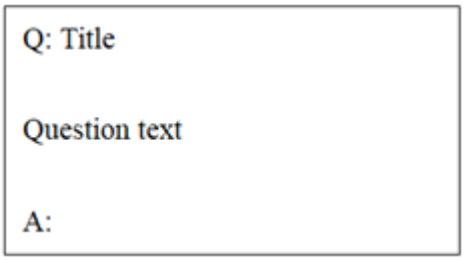
```
A.375   1   1.0 babbage babbage "Thanks!

Answer:

The product of two diophantine equations is diophantine. That means that you can't solve one equation and expect to be a

For example, if you multiply $x^2 + 7$ and $y^2 + 10$ you get $x^2 + (y^2 + 10) + 7 = y^2 + (x^2 + 10) + 7$ which is an
```

Σχήμα 20. Απάντηση του μοντέλου Babbage στην ερώτηση 75

Το πρόβλημα λύθηκε αλλάζοντας λίγο το κείμενο της ερώτησης. Ο λόγος πίσω από το πρόβλημα φαίνεται να ήταν ο τρόπος που χρησιμοποιήθηκε από το ARQMath στο baseline σύστημα για να τεθούν οι ερωτήσεις, και τον οποίο μιμήθηκε αρχικά αυτή η εργασία.

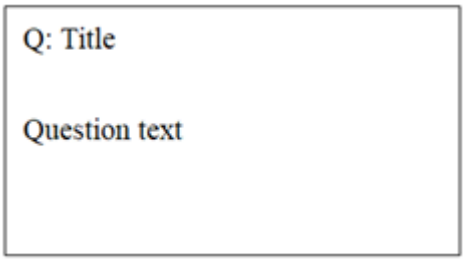
Έστω ο τίτλος ερώτησης «Title» και το κείμενο «Question text». Ο κώδικας έπαιρνε αυτά τα δύο δεδομένα και δημιουργούσε το εξής input, “Q: Title\n\nQuestion text\n\n”.

Οι χαρακτήρες \n μεταφράζονται από τα συστήματα κειμένου ως αλλαγή γραμμής, οπότε η μορφή της ερώτησης ήταν η εξής (Σχήμα 21):



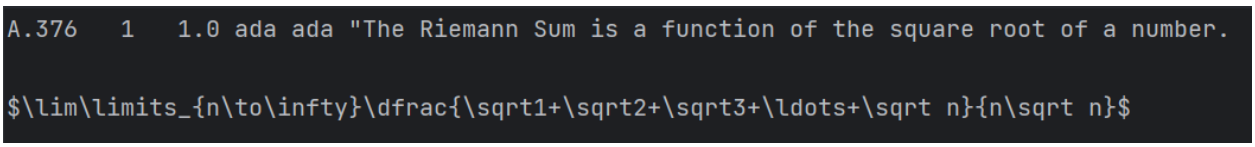
Σχήμα 21. Αρχική μορφή ερωτήσεων

Αυτό φαίνεται να είχε ως αποτέλεσμα το κάθε μοντέλο, που δεν ήταν υλοποιημένο ως chatbot ώστε να γνωρίζει πως ο ρόλος του είναι καθαρά η παραγωγή απαντήσεων, να κάνει generate την συνέχεια της ερώτησης. Ο τρόπος που αντιμετωπίστηκε αυτό το πρόβλημα ήταν απλά η ένταξη, στο τέλος του ερωτήματος, των χαρακτήρων «A:». Στο προηγούμενο παράδειγμα αυτό θα φαινόταν ως εξής (Σχήμα 22):

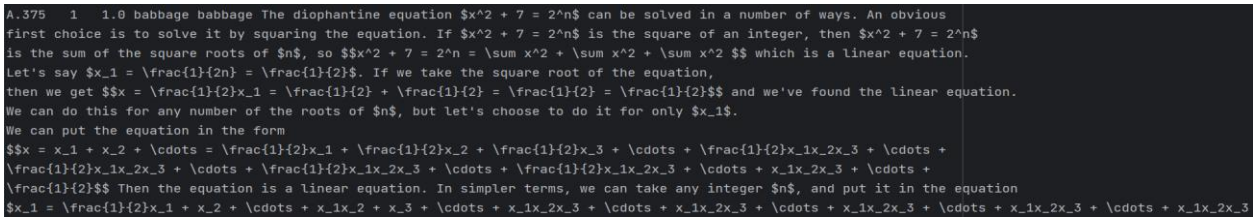


Σχήμα 22. Νέα μορφή ερωτήσεων

Μετά την ένταξη αυτών των χαρακτήρων, τα μοντέλα φαίνεται να «αντιλαμβάνονται» πως σε εκείνο το σημείο ξεκινούσε η απάντηση, και απαντούσαν το ερώτημα (Σχήμα 23) (Σχήμα 24).



Σχήμα 23. Νέα απάντηση του μοντέλου Ada στην ερώτηση 76



Σχήμα 24. Νέα απάντηση του μοντέλου Babbage στην ερώτηση 75

4.2.3 Υψηλή συχνότητα επανάληψης φράσεων

Η τρίτη δυσκολία που παρουσίαστηκε ήταν ότι στην προσπάθεια παραγωγής ντετερμινιστικών απαντήσεων, τα μοντέλα συνήθως επαναλάμβαναν συνεχώς τις ίδιες φράσεις. Για την αντιμετώπιση αυτού του προβλήματος, έγινε ακόμα μία εκτέλεση των μοντέλων στην οποία χρησιμοποιήθηκε μία επιπλέον παράμετρος, το repetition penalty για τα μοντέλα του Hugging Face, και το frequency penalty για τα μοντέλα του OpenAI, οι οποίες περιγράφονται με περισσότερη λεπτομέρεια στην ενότητα 6.1.1. Τα chatbot μοντέλα GPT-3.5 ήταν τα μόνα που φάνηκαν να μην χρειάζονται αυτή τη παράμετρο, αφού οι απαντήσεις τους δεν περιείχαν επαναλαμβανόμενες φράσεις.

4.3 Αποτελέσματα

Για την αξιολόγηση των απαντήσεων του Task 3 χρησιμοποιήθηκε κώδικας γραμμένος σε Python, ο οποίος παρέχεται από το ARQMath[20].

Ο κώδικας αυτός υπολογίζει το LO και το CS των απαντήσεων του χρήστη σε σύγκριση με τις απαντήσεις από το Task 1 και το Task 3 του διαγωνισμού.

Για την αξιολόγηση των αποτελεσμάτων χρησιμοποιήθηκαν 4 τρόποι. Ο πρώτος ήταν με χρήση του αρχικού κώδικα αξιολόγησης. Ο δεύτερος ήταν με αλλαγές, ώστε να χρησιμοποιείται ο μέσος όρος των F1 score και όχι το υψηλότερο για κάθε απάντηση. Ο τρίτος ήταν με αλλαγές, ώστε να φιλτράρονται οι «κοινές» λέξεις (λέξεις όπως «no», «it», «both» κτλ.). Και τέλος, ο τέταρτος ήταν με ένα άλλο φίλτρο, το οποίο αφαιρούσε τις επαναλαμβανόμενες φράσεις.

Έγινε και μία δοκιμή με φίλτρο για τα επιπλέον whitespaces (δηλαδή παραπανίσια κενά ανάμεσα στις λέξεις, πολλές αλλαγές γραμμών κτλ.) αλλά δεν υπήρξαν διαφορές στα αποτελέσματα, γεγονός που σημαίνει ότι δεν λαμβάνονται υπόψιν στους υπολογισμούς.

Για πιο αποτελεσματική ερμηνεία των αποτελεσμάτων δημιουργήθηκε χειροκίνητα και ένα run με ψεύτικες απαντήσεις, το οποίο χρησιμοποιήθηκε για να αξιολογηθεί η απόδοση των μετρικών LO και CS σε αδιαμφισβήτητα λανθασμένες απαντήσεις. Οι απαντήσεις σε όλα τα ερωτήματα ήταν το κείμενο «Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop».

[20] https://drive.google.com/drive/u/0/folders/1iE8gEU_7tT5wb40TSTJb_tUZsWChumVi

Επιπροσθέτως έγιναν συμπληρωματικά και μερικές ενδεικτικές δοκιμές αξιολόγησης του τρόπου βαθμολόγησης CS, χρησιμοποιώντας διάφορους συνδυασμούς του BERTScore με άλλα μοντέλα της οικογένειας BERT.

Τέλος, εμφανίζονται ενδεικτικά οι διαφορές ανάμεσα στις διαφορετικές εκδόσεις ενός μοντέλου, χρησιμοποιώντας το Cerebras-GPT σαν παράδειγμα.

4.3.1 Αξιολόγηση μέσω του αρχικού κώδικα

Ο πρώτος τρόπος ήταν η χρήση του κώδικα αξιολόγησης του ARQMath χωρίς καμία αλλαγή. Αρχικά επιβεβαιώθηκαν όλα τα αποτελέσματα που εμφανίζονται στη δημοσίευση, εκτός από δύο τα οποία ανήκαν στην ομάδα DPRL (SBERT-SVMRank και BERT-SVMRank) και παρουσίασαν μικρές διαφορές, χωρίς να υπάρχει ξεκάθαρη ένδειξη όσον αφορά τον λόγο απόκλισης. Στον πίνακα αποτελεσμάτων της παρούσας εργασίας (βλ. Παράρτημα) αναγράφονται οι νέες τιμές λόγω αδυναμίας επιβεβαίωσης των προηγούμενων. Ο κώδικας έτρεξε 3 φορές, χρησιμοποιώντας κάθε φορά διαφορετικούς συνδυασμούς των δεδομένων του διαγωνισμού ως reference σε κάθε εκτέλεση.

Οι 3 παραλλαγές ήταν οι εξής:

- Σύγκριση με τις απαντήσεις του Task 1 και του Task 3
- Σύγκριση μόνο με τις απαντήσεις του Task 1
- Σύγκριση μόνο με τις απαντήσεις του Task 3

Η δεύτερη και η τρίτη παραλλαγή είχαν σαν στόχο την αξιολόγηση του αν οι απαντήσεις των μοντέλων είχαν μεγαλύτερη συσχέτιση με τις ανθρώπινες απαντήσεις του Task 1 ή με τις απαντήσεις που έδωσαν τα άλλα συστήματα αυτόματης παραγωγής κειμένου. Επίσης, θα εξυπηρετήσουν και σαν ένδειξη για το αν τα αποτελέσματα που πήραν υψηλή βαθμολογία ήταν επειδή είχαν υψηλή συσχέτιση με ανθρώπινες ή με generated απαντήσεις.

Σε κάθε πίνακα παρουσιάζονται τα υψηλότερα αποτελέσματα LO και CS για κάθε οικογένεια μοντέλων που χρησιμοποιήθηκαν στην παρούσα εργασία και για κάθε ομάδα του ARQMath. Στις περιπτώσεις που διαφορετικά μοντέλα της ίδιας οικογένειας είχαν καλύτερο LO και CS, εμφανίζονται και τα δύο.

Στον Πίνακας 5 βλέπουμε ότι η ομάδα των approach0 παραμένει στην πρώτη θέση με διαφορά. Ανάμεσα όμως στα αποτελέσματα των υπόλοιπων ομάδων του ARQMath και τα αποτελέσματα των pretrained μοντέλων που χρησιμοποιήθηκαν, δεν εμφανίζεται σημαντική

διαφορά. Αξιοσημείωτο είναι το γεγονός ότι, παρότι τα extractive συστήματα, όπως αυτά που χρησιμοποίησε η ομάδα DPRL, θεωρείται πως έχουν πλεονέκτημα έναντι των generative, δεν παρατηρείται κάτι τέτοιο στα αποτελέσματα.

Όπως ήταν αναμενόμενο, για κάθε μοντέλο του Hugging Face, οι εκδόσεις με τις περισσότερες παραμέτρους είχαν τις περισσότερες φορές τα καλύτερα αποτελέσματα. Επίσης παρατηρείται πως για κάθε οικογένεια μοντέλων, τα runs με τις υψηλότερες βαθμολογίες ήταν συνήθως τα ντετερμινιστικά, τα οποία είχαν την τάση να επαναλαμβάνουν μικρές φράσεις που περιείχαν μαθηματικές λέξεις-κλειδιά, χωρίς να αναπτύσσουν περισσότερο το θέμα, μέχρι να φτάσουν το όριο των 1200 χαρακτήρων.

Παρατηρείται επίσης ότι ακόμα και απαντήσεις με τυχαίες λέξεις χωρίς νόημα (Fake run) βαθμολογήθηκαν με CS 0.742, ενώ ταυτόχρονα η υψηλότερη βαθμολογία που εμφανίζεται με τον ίδιο τρόπο αξιολόγησης ήταν 0.886 . Αυτό δείχνει πως το εύρος στο οποίο βρίσκονται οι τιμές για το CS είναι στη πραγματικότητα περίπου από 0.74 έως 1 αντί για 0 έως 1. Οπότε τα αποτελέσματα θα πρέπει να ερμηνευτούν με βάση αυτό το εύρος, το οποίο σημαίνει πως τα συστήματα που είχαν CS κοντά στο 0.74, όπως για παράδειγμα το Cerebras-GPT-6.7B-creative-with-repetition-penalty που είχε CS 0.76 (Παράρτημα Ι), στην πραγματικότητα είχαν πολύ χαμηλή βαθμολογία. Ταυτόχρονα όμως αυτό σημαίνει πως, έστω και μικρές διαφορές της τάξεως του 0.01, είναι πιο σημαντικές απ’ όσο φαίνονται εκ πρώτης όψεως.

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.317	0.851
Approach0		
run4	0.515	0.886
DPRL		
SBERT-SVMRank	0.330	0.846
SBERT-QQ-AMR	0.325	0.852
TU_DBS		
amps3_se1_hints	0.263	0.835
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.025	0.742
OpenAI text generators		
text-davinci-003-deterministic	0.357	0.853
text-davinci-002-deterministic	0.348	0.858
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.357	0.849
gpt-3.5-turbo-creative	0.351	0.850
Hugging Face text generators		
bloom-7b1-deterministic	0.325	0.834
Cerebras-GPT-6.7B-deterministic	0.337	0.845
llama-7b-hf-deterministic	0.298	0.811
opt-2.7b-deterministic	0.319	0.819
opt-6.7b-deterministic	0.315	0.821

Πίνακας 5. Σύγκριση με όλες τις απαντήσεις του διαγωνισμού μέσω του αρχικού αλγόριθμου

Στον Πίνακας 6 παρατηρείται ότι οι απαντήσεις των συστημάτων έχουν μεγαλύτερη συσχέτιση με τις απαντήσεις που δόθηκαν στο Task 1 απ' ότι με τις απαντήσεις που δόθηκαν για το Task 3. Όμως αυτό πιθανά προέρχεται από τον τρόπο λειτουργίας του κώδικα αξιολόγησης. Όπως αναφέρθηκε στο κεφάλαιο 4, για κάθε ερώτηση, ο κώδικας συγκρίνει την απάντηση του αξιολογούμενου συστήματος με κάθε σχετική απάντηση που δόθηκε κατά τη διάρκεια του διαγωνισμού, και στη συνέχεια κρατάει τον υψηλότερο βαθμό συσχέτισης που βρίσκει. Αυτό σημαίνει πως όσο περισσότερες απαντήσεις υπάρχουν για μία ερώτηση, τόσο αυξάνονται οι πιθανότητες ένα σύστημα να λάβει υψηλές βαθμολογίες CS και LO. Στη συγκεκριμένη περίπτωση γνωρίζουμε πως οι σχετικές απαντήσεις από το Task 1 ήταν συνολικά πολύ περισσότερες από αυτές του Task 3. Ο κώδικας φαίνεται να χρησιμοποιεί 2246 απαντήσεις από το Task 1 και μόνο 170 από το Task 3. Οπότε για κάθε απάντηση του αξιολογούμενου συστήματος, υπάρχει μεγαλύτερη πιθανότητα να υπάρξει υψηλότερη συσχέτιση όταν συγκρίνεται με τις απαντήσεις του Task 1 απ' ότι όταν συγκρίνεται με του Task 3. Αυτό το θέμα διερευνάται περισσότερο στο επόμενο κεφάλαιο.

Model	LO Task 1	LO Task 3	CS Task 1	CS Task 3
ARQMath				
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.308	0.270	0.850	0.835
Approach0				
run4	0.510	0.250	0.885	0.828
DPRL				
SBERT-SVMRank	0.326	0.238	0.847	0.821
TU_DBS				
amps3_se1_hints	0.234	0.201	0.831	0.813
New runs				
Fake run				
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.024	0.008	0.742	0.722
OpenAI text generators				
text-davinci-003-deterministic	0.333	0.319	0.847	0.836
OpenAI chatbots				
gpt-3.5-turbo-deterministic	0.343	0.326	0.845	0.839
Hugging Face text generators				
bloom-7b1-deterministic	0.301	0.282	0.830	0.824
Cerebras-GPT-6.7B-deterministic	0.297	0.306	0.836	0.835
llama-7b-hf-deterministic	0.273	0.265	0.807	0.799
opt-6.7b-deterministic	0.283	0.276	0.817	0.805

Πίνακας 6. Ξεχωριστές συγκρίσεις με τις απαντήσεις των Task 1 και 3 μέσω του αρχικού αλγόριθμου

4.3.2 Αξιολόγηση με χρήση του μέσου F1 score

Ο δεύτερος τρόπος αξιολόγησης ήταν με αλλαγές στον κώδικα αξιολόγησης του ARQMath, όπου αντί να κρατάει το υψηλότερο F1 score για κάθε σετ απαντήσεων, κάθε απάντηση του χρήστη συγκρίνεται ξεχωριστά με κάθε μία από τις απαντήσεις των tasks, και στο τέλος υπολογίζεται ο μέσος όρος από όλα τα F1 score. Για αυτή την έκδοση του κώδικα αξιολόγησης εκτελέστηκαν επίσης οι παραλλαγές που αναφέρονται στο 7.1, όπου οι απαντήσεις αξιολογούνται και σε σύγκριση με τον συνδυασμό απαντήσεων των Task 1 και 3, αλλά και με κάθε συλλογή απαντήσεων ξεχωριστά.

Συγκρίνοντας τον Πίνακας 7 με τον Πίνακας 5, παρατηρείται ότι τα runs των approach0 χάνουν το σημαντικό «προβάδισμα» που είχαν έναντι συστημάτων όπως το GPT-3.5, τόσο στο CS που αρχικά υπήρχαν διαφορές της τάξεως του 0.04, το οποίο, όπως αναφέρθηκε στο κεφάλαιο 7.1, σηματοδοτούσε μη-αμελητέα διαφορά, αλλά και στο LO στο οποίο τα συστήματα των approach0 είχαν ένα εντυπωσιακό προβάδισμα της τάξεως του 0.15. Βλέπουμε πως χρησιμοποιώντας το μέσο F1 score, το GPT-3.5 ξεπερνάει το σύστημα των approach0 στο LO και πρακτικά το ισοφαρίζει όσον αφορά το CS. Αυτό σηματοδοτεί πως πιθανά το σύστημα των approach0 να λειτουργούσε με τέτοιο τρόπο, ώστε στη σύγκριση με συγκεκριμένες από τις απαντήσεις να λάμβανε πολύ υψηλές βαθμολογίες LO και CS, οι οποίες επιλέγονταν από το αρχικό σύστημα αξιολόγησης, αγνοώντας τη χαμηλή συσχέτιση που υπήρχε με τις υπόλοιπες σχετικές απαντήσεις.

Κατά τ’ άλλα παρατηρείται πως μειώθηκαν οι βαθμολογίες όλων των συστημάτων (σε μικρότερο βαθμό απ’ όσο μειώθηκε η βαθμολογία των approach0), το οποίο ήταν αναμενόμενο αποτέλεσμα της επιλογής του μέσου όρου αντί για το μέγιστο F1 score.

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.195	0.821
Approach0		
run1	0.222	0.822
DPRL		
SBERT-SVMRank	0.185	0.811
TU_DBS		
amps3_se1_len_pen_20_sample_hint	0.114	0.773
amps3_se1_hints	0.110	0.791
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.004	0.716
OpenAI text generators		
text-davinci-003-creative	0.223	0.820
OpenAI chatbots		
gpt-3.5-turbo-creative	0.235	0.820
Hugging Face text generators		
bloom-7b1-deterministic	0.184	0.799
bloom-7b1-deterministic-with-repetition-penalty	0.163	0.805
Cerebras-GPT-6.7B-deterministic	0.181	0.807
llama-7b-hf-deterministic	0.178	0.783
llama-7b-hf-deterministic-with-repetition-penalty	0.146	0.785
opt-6.7b-deterministic	0.182	0.789

Πίνακας 7. Σύγκριση με όλες τις απαντήσεις του διαγωνισμού με χρήση του μέσου F1 score

Στη συνέχεια, βλέπουμε ότι στον Πίνακας 8 όπου οι απαντήσεις συγκρίνονται ξεχωριστά με αυτές του Task 1 και του Task 3, οι αλλαγές στον κώδικα αξιολόγησης είχαν σαν αποτέλεσμα τα συστήματα να παρουσιάσουν μεγαλύτερη συσχέτιση με τις απαντήσεις του Task 3 απ' ότι με τις απαντήσεις του Task 1. Αυτό έρχεται σε αντίθεση με αυτό που παρατηρείται στο κεφάλαιο 7.1 κι ενισχύει την πιθανότητα τα αποτελέσματα εκείνου του κεφαλαίου να είναι διαφορετικά επειδή ο αρχικός αλγόριθμος «πριμοδοτεί» τις συσχετίσεις με το Task 1, λόγω της ύπαρξης περισσότερων απαντήσεων συγκριτικά με το Task 3, άρα και περισσότερων πιθανοτήτων να βρεθεί υψηλή συσχέτιση (υπάρχουν 2246 Task 1 απαντήσεις και 170 Task 3 απαντήσεις). Όταν όμως εξαλείφεται αυτό το πλεονέκτημα με την χρήση του μέσου όρου αντί της υψηλότερης τιμής για το F1 score, εμφανίζεται η ύπαρξη μεγαλύτερης συσχέτισης των generative συστημάτων με άλλα αυτόματα συστήματα παραγωγής απαντήσεων, παρά με απαντήσεις που προέρχονται από ανθρώπινες πηγές.

Model	LO Task 1	LO Task 3	CS Task 1	CS Task 3
ARQMath				
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.192	0.243	0.820	0.827
Approach0				
run1	0.221	0.238	0.822	0.827
DPRL				
SBERT-SVMRank	0.184	0.210	0.811	0.813
TU_DBS				
amps3_se1_hints	0.107	0.159	0.790	0.804
New runs				
OpenAI text generators				
text-davinci-003-creative	0.219	0.281	0.819	0.831
OpenAI chatbots				
gpt-3.5-turbo-creative	0.231	0.288	0.819	0.829
Hugging Face text generators				
bloom-7b1-deterministic-with-repetition-penalty	0.161	0.184	0.805	0.808
Cerebras-GPT-6.7B-deterministic	0.176	0.251	0.806	0.822
llama-7b-hf-deterministic-with-repetition-penalty	0.146	0.152	0.785	0.784
opt-6.7b-deterministic	0.179	0.220	0.789	0.795

Πίνακας 8. Ξεχωριστές συγκρίσεις με τις απαντήσεις των Task 1 και 3 με χρήση του μέσου F1 score

Επίσης για πρώτη φορά αναδεικνύονται σε μεγάλο βαθμό τα συστήματα που έτρεξαν με repetition penalty, τα οποία φαίνεται να επηρεάστηκαν λιγότερο από την αλλαγή στον κώδικα αξιολόγησης, παρουσιάζοντας πολύ μικρότερες πτώσεις τόσο στο LO όσο και στο CS σε σχέση με τα απλά ντετερμινιστικά συστήματα. Ενδεικτικά αναφέρονται τα παρακάτω συστήματα

(Πίνακας 9). Σημειώνονται με bold οι μεγαλύτερες τιμές κατά τη σύγκριση μεταξύ runs του ιδίου συστήματος, με και χωρίς repetition penalty.

Model	Lexical overlap before	Lexical overlap after	Contextual similarity before	Contextual similarity after
babbage				
babbage-deterministic	**0.319**	0.187	0.816	0.784
babbage-deterministic-with-repetition-penalty	0.308	**0.201**	**0.825**	**0.801**
text-davinci-002				
text-davinci-002-deterministic	**0.348**	0.186	**0.858**	0.814
text-davinci-002-deterministic-with-repetition-penalty	0.342	**0.220**	0.847	**0.818**
opt-1.3b				
opt-1.3b-deterministic	**0.289**	0.163	**0.807**	0.775
opt-1.3b-deterministic-with-repetition-penalty	0.268	**0.166**	0.803	**0.780**

Πίνακας 9. Σύγκριση μοντέλων με και χωρίς repetition penalty

4.3.3 Αξιολόγηση με φιλτράρισμα των «κοινών» λέξεων

Ο τρίτος τρόπος αξιολόγησης ήταν με φιλτράρισμα των «κοινών» λέξεων από τις απαντήσεις των αξιολογούμενων συστημάτων και από τις απαντήσεις που χρησιμοποιούνται ως μέτρο σύγκρισης. Αυτό έγινε χρησιμοποιώντας το σετ των stopwords από την βιβλιοθήκη NLTK για να αναγνωριστούν οι λέξεις και να αφαιρεθούν από το κείμενο.

Συγκρίνοντας τα αποτελέσματα του Πίνακας 10 με αυτά του Πίνακας 5 παρατηρείται πως βελτιώνονται όλες οι βαθμολογίες.

Στην περίπτωση του LO αυτό πιθανά συμβαίνει επειδή κάθε σύστημα χρησιμοποίησε διαφορετικές συνδετικές λέξεις ανάμεσα στις λέξεις-κλειδιά, οπότε η αφαίρεσή τους μείωσε τη λεξιλογική απόκλιση. Για παράδειγμα οι φράσεις “There is no linear relationship between the results” και “There is linear relationship between the results”, μετά την αφαίρεση των «κοινών» λέξεων θα περιέχουν και οι δύο το κείμενο “linear relationship results” το οποίο αυξάνει το LO.

Όμως η αφαίρεση της λέξης "no" από την πρώτη φράση αλλάζει όλο το νόημα, το οποίο σημαίνει πως ενώ το LO φτάνει το 100%, δεν υπάρχει 100% ταύτιση του νοήματος.

Στην περίπτωση του CS παρατηρείται κάτι παρόμοιο. Η αφαίρεση των «κοινών» λέξεων, ενώ αυξάνει την βαθμολογία των μοντέλων, επηρεάζει αρνητικά την αξιολόγηση, επειδή αυτές οι «κοινές» λέξεις, παρ' όλο που δεν έχουν εξειδικευμένο νόημα, χρησιμοποιούνται ως context από το BERTSscore. Αφαιρώντας τη λέξη "no" αλλάζει σημαντικά το context των υπόλοιπων λέξεων της φράσης, με αποτέλεσμα να χάνεται πληροφορία η οποία χρειάζεται για τη σωστή αξιολόγηση μέσω του BERTScore.

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.480	0.879
Approach0		
run1	0.604	0.907
DPRL		
SBERT-QQ-AMR	0.457	0.871
TU_DBS		
amps3_se1_hints	0.464	0.862
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.029	0.737
OpenAI text generators		
text-davinci-002-deterministic	0.519	0.879
OpenAI chatbots		
gpt-3.5-turbo-0301-creative	0.471	0.873
Hugging Face text generators		
bloom-7b1-deterministic	0.482	0.860
Cerebras-GPT-6.7B-deterministic	0.505	0.872
llama-7b-hf-deterministic	0.427	0.836

opt-2.7b-deterministic	0.459	0.850

Πίνακας 10. Σύγκριση με τις απαντήσεις των Task 1 και Task 3 με φιλτράρισμα των «κοινών» tokens

4.3.4 Αξιολόγηση με φιλτράρισμα των επαναλαμβανόμενων φράσεων

Ο τέταρτος τρόπος αξιολόγησης ήταν με φιλτράρισμα των επαναλαμβανόμενων φράσεων από τις απαντήσεις των αξιολογούμενων συστημάτων και από τις απαντήσεις που χρησιμοποιούνται ως μέσο σύγκρισης. Αυτό ήταν δυνατό μέσω της βιβλιοθήκης NLTK, η οποία δίνει τη δυνατότητα διαχωρισμού των απαντήσεων σε μεμονωμένες φράσεις. Στη συνέχεια αφαιρέθηκαν οι διπλότυπες φράσεις και οι απαντήσεις ξαναδημιουργήθηκαν χρησιμοποιώντας τις εναπομείνασες φράσεις, με την σειρά που υπήρχαν στο κείμενο.

Συγκρίνοντας τα αποτελέσματα του Πίνακας 11 με αυτά του Πίνακας 5 παρατηρείται πως η μόνη επιρροή που είχε η συγκεκριμένη μέθοδος ήταν να αυξηθεί το CS των ντετερμινιστικών συστημάτων. Το LO παρέμεινε σταθερό για όλα τα μοντέλα. Από αυτό συμπεραίνεται ότι η υπάρξη επαναλαμβανόμενων φράσεων φαίνεται να μειώνει τη βαθμολογία CS των συστημάτων όταν χρησιμοποιείται ο κώδικας αξιολόγησης του ARQMath. Ωστόσο, η βαθμολογία δεν μειώνεται αρκετά ώστε να μπορέσουν να αναδειχθούν τα αποτελέσματα από runs που έχουν μεγαλύτερη ποικιλία λέξεων και απαντήσεις με καθαρότερο νόημα, τα οποία όμως δεν εστιάζουν στην χρήση φράσεων με λίγες λέξεις-κλειδιά, όπως κάνουν τα ντετερμινιστικά.

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.317	0.851
Approach0		
run4	0.516	0.886
DPRL		
SBERT-SVMRank	0.338	0.848
SBERT-QQ-AMR	0.325	0.852
TU_DBS		
amps3_se1_hints	0.263	0.837
New runs		
Fake run		
“Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop”	0.025	0.742
OpenAI text generators		
text-davinci-003-deterministic	0.357	0.853
text-davinci-002-deterministic	0.348	0.858
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.357	0.849
Hugging Face text generators		
bloom-7b1-deterministic	0.325	0.841
Cerebras-GPT-6.7B-deterministic	0.337	0.854
llama-7b-hf-deterministic	0.298	0.827

opt-2.7b-deterministic	0.319	0.841

Πίνακας 11. Σύγκριση με τις απαντήσεις των Task 1 και Task 3 με φιλτράρισμα επαναλαμβανόμενων φράσεων

4.3.5 Αποτελέσματα CS με χρήση διαφορετικών μοντέλων BERT

Για την αξιολόγηση της επιρροής που έχει στην κατάταξη των runs η χρήση του MathBERTa, σε συνδυασμό με το BERTScore για την παραγωγή του δείκτη CS, έγιναν εκτελέσεις του κώδικα αξιολόγησης με τη χρήση άλλων μοντέλων της οικογένειας BERT, ώστε να μελετηθεί το κατά πόσο τα runs θα κατατάσσονταν με διαφορετικό τρόπο σε περίπτωση της χρήσης κάποιου άλλου μοντέλου.

Για αυτές τις δοκιμές επιλέχθηκαν ενδεικτικά τα εξής runs

- Fake run
- ARQMath baseline
- gpt-3.5-turbo-creative
- bloom-560-creative

Στον Πίνακας 12 παρατηρείται ότι όσο διαφορετικές κι αν είναι οι βαθμολογίες ανάμεσα στις τιμές που επιστρέφει το BERTScore σε συνδυασμό με κάθε υλοποίηση του BERT, τα συστήματα διατηρούν την μεταξύ τους κατάταξη, και τις «αποστάσεις» ανάμεσα στις αξιολογήσεις τους.

	Fake run	bloom-560	ARQMath baseline	gpt-3.5
bart-base	0.358	0.628	0.678	0.683
distilbert-base-uncased	0.617	0.827	0.873	0.879
deberta-base	0.396	0.757	0.813	0.830
bigbird-base-trivia-itc	0.317	0.735	0.787	0.798
deberta-v3-large	0.115	0.743	0.823	0.841
deberta-v3-base	0.125	0.740	0.824	0.844
deberta-v3-small	0.133	0.744	0.829	0.849
bart-large	0.159	0.669	0.739	0.752
longformer-large-4096-finetuned-triviaqa	0.262	0.704	0.762	0.775
deberta-xlarge-mnli	0.159	0.721	0.805	0.824

Πίνακας 12. Αποτελέσματα του BERTScore με χρήση διαφορετικών μοντέλων BERT

4.3.6 Διαφορές ανάμεσα στις εκδόσεις του ίδιου μοντέλου

Συγκρίνοντας τα αποτελέσματα των εκδόσεων με διαφορετικό αριθμό παραμέτρων, είναι εμφανές ότι στις περισσότερες περιπτώσεις η χρήση περισσότερων παραμέτρων παράγει καλύτερα αποτελέσματα όσον αφορά τα LO και CS. Οι παρακάτω πίνακες (Πίνακας 13 και Πίνακας 14) περιέχουν τα αποτελέσματα όπου χρησιμοποιείται ο μέσος όρος για τον υπολογισμό του τελικού F1 score, από τις αξιολογήσεις που έγιναν για το κεφάλαιο 7.2, γι αποφυγή περιπτώσεων όπου οι απαντήσεις κάποιας έκδοσης των μοντέλων έχουν μη φυσιολογικά υψηλή συσχέτιση με ορισμένες απαντήσεις του διαγωνισμού.

Model	Lexical overlap	Contextual similarity
Cerebras-GPT-111M-deterministic	0.117	0.756
Cerebras-GPT-256M-deterministic	0.147	0.771
Cerebras-GPT-590M-deterministic	0.158	0.787
Cerebras-GPT-1.3B-deterministic	0.171	0.801
Cerebras-GPT-2.7B-deterministic	0.161	0.796
Cerebras-GPT-6.7B-deterministic	0.181	0.807

Πίνακας 13. Αποτελέσματα των διάφορων εκδόσεων του μοντέλου Cerebras-GPT με ντετερμινιστική παραμετροποίηση

Model	Lexical overlap	Contextual similarity
opt-125m-deterministic	0.172	0.782
opt-350m-deterministic	0.155	0.777
opt-1.3b-deterministic	0.163	0.775
opt-2.7b-deterministic	0.182	0.790
opt-6.7b-deterministic	0.182	0.789

Πίνακας 14. Αποτελέσματα των διάφορων εκδόσεων του μοντέλου opt με ντετερμινιστική παραμετροποίηση

5

Συμπεράσματα

Παρατηρήθηκε στην ανάλυση των αποτελεσμάτων ότι, με εξαίρεση τα αποτελέσματα των approach0 τα οποία παρήχθησαν μέσω extractive συστημάτων σε συνδυασμό με manual παρέμβαση, τα υπόλοιπα συστήματα που χρησιμοποιήθηκαν από τις ομάδες του διαγωνισμού, δεν παρήγαγαν καλύτερα αποτελέσματα από τα pretrained μοντέλα της OpenAI και του Hugging Face όσον αφορά τις μετρήσεις LO και CS. Είναι ιδιαίτερα ενδιαφέρον ότι πολλά μοντέλα του Hugging Face με λίγα δισεκατομμύρια παραμέτρους και χωρίς fine-tuning, κατάφεραν να έχουν παρόμοια ή καλύτερα αποτελέσματα από τις ομάδες DPRL και TU_DBS. Αν λάβουμε υπόψιν και τις δοκιμές του κεφαλαίου 7.2 όπου χρησιμοποιήθηκε το μέσο F1 score των απαντήσεων, κανένα από τα εξειδικευμένα συστήματα του διαγωνισμού δεν απέδωσε καλύτερα από αυτά που χρησιμοποιήθηκαν σε αυτή την εργασία. Με βάση αυτά τα δεδομένα θα μπορούσαμε να συμπεράνουμε ότι τα επαρκώς εκπαιδευμένα pretrained μοντέλα δεν χρειάζονται περαιτέρω εξειδίκευση για να παράξουν ικανοποιητικά αποτελέσματα σε μαθηματικά ερωτήματα.

Επίσης επισημαίνεται το γεγονός ότι, ενώ τα extractive συστήματα θεωρείται πως έχουν πλεονέκτημα έναντι των generative, τα extractive συστήματα των DPRL δεν μπόρεσαν να ξεπεράσουν τις επιδόσεις των generative συστημάτων με καμία μέθοδο αξιολόγησης. Όσον αφορά τα generative συστήματα των approach0 δεν είναι ξεκάθαρο χωρίς εις βάθος ανάλυση αν οι υψηλές αποδόσεις τους οφείλονται στη χρήση καλύτερων μοντέλων, στην καλύτερη εκπαίδευση τους, στην manual παρέμβαση που έκανε η ομάδα ή σε overfitting των δεδομένων που χρησιμοποιούνται ως πηγή πληροφοριών για τις απαντήσεις. Όσον αφορά τη σύγκριση των υπολοίπων συστημάτων μεταξύ τους, υπάρχουν ενδείξεις πως παίζει μεγαλύτερο ρόλο η σωστή

παραμετροποίηση και ο τρόπος προεκπαίδευσης των μοντέλων, απ' ότι ο αριθμός των παραμέτρων τους, χωρίς αυτό να σημαίνει πως η συμβολή των παραμέτρων είναι αμελητέα. Για παράδειγμα, το Cerebras-GPT-1.3B-deterministic(0.320, 0.832) είχε καλύτερες επιδόσεις από το llama-7b-hf-deterministic (0.298, 0.811), και το opt-6.7b-deterministic(0.315, 0.821).

Η συμβολή του αριθμού των παραμέτρων είναι προφανής κατά τη σύγκριση μοντέλων της ίδιας οικογένειας, όπου η μόνη διαφορά είναι ο αριθμός των παραμέτρων, όπως τα παραδείγματα του κεφαλαίου 7.6. Είναι προφανές ότι ο αριθμός των παραμέτρων επηρεάζει τα αποτελέσματα, με την πλειοψηφία των παραδειγμάτων να δείχνουν ότι, όσο περισσότερες παραμέτρους έχει ένα μοντέλο, τόσο καλύτερα είναι τα αποτελέσματα. Βέβαια στις περιπτώσεις των μοντέλων όπως το Llama και το Cerebras-GPT όπου τα μοντέλα με περισσότερες παραμέτρους εκπαιδεύτηκαν με περισσότερα tokens, δεν είναι εύκολο να διαχωριστεί η επιρροή του αριθμού των παραμέτρων από αυτήν της ποσότητας του εκπαιδευτικού υλικού.

Όσον αφορά την παραμετροποίηση, ως επί το πλείστον όλα τα μονέλα παρουσίασαν σε γενικές γραμμές καλύτερα αποτελέσματα όταν εκτελέστηκαν με greedy search και χωρίς repetition penalty. Αυτό άλλαξε όταν χρησιμοποιήθηκε το μέσο F1 score για την αξιολόγηση των απαντήσεων, όπου σε πολλές περιπτώσεις αναδείχτηκαν τα runs που χρησιμοποίησαν greedy search μαζί με repetition penalty. Όσον αφορά τα runs με υψηλό temperature, σπάνια είχαν ανταγωνιστικές επιδόσεις, ανεξαρτήτως της εκδοχής του κώδικα αξιολόγησης που χρησιμοποιήθηκε, στο οποίο πιθανά να συνέβαλε το ύψος του temperature που επιλέχθηκε. Επίσης, τα μόνα μοντέλα που έδωσαν ανταγωνιστικά αποτελέσματα με υψηλό temperature, συγκρινόμενα με τα greedy search αποτελέσματα των ίδιων μοντέλων ήταν της OpenAI, οπότε υπάρχει η πιθανότητα να επηρεάζονται περαιτέρω τα αποτελέσματα από την επιρροή που έχουν οι διάφορες τιμές temperature στο κάθε μοντέλο. Πιθανά το temperature 0,7 στην κλίμακα 0 έως 1 της OpenAI να έχει διαφορετικό αποτέλεσμα από την τιμή 70 στην κλίμακα 0 έως 100 του Hugging Face.

Επιπροσθέτως, επιβεβαιώνονται τα συμπεράσματα της δημοσίευσης του ARQMath σχετικά με το LO (Mansouri et al., 2022). Με βάση τις μετρήσεις, φαίνεται να είναι καλύτερη ένδειξη για τη ποιότητα των απαντήσεων, δίνοντας μόλις 0.025 στην αξιολόγηση των ψεύτικων απαντήσεων και πιο ρεαλιστικές αξιολογήσεις στα υπόλοιπα runs, με τις τιμές να βρίσκονται στο εύρος 0.025-51.5.

Κοιτώντας όμως τις απαντήσεις των μοντέλων μπορούμε να δούμε ότι και οι δύο τρόποι μέτρησης αποτυγχάνουν να αξιολογήσουν κατά πόσο οι απαντήσεις είναι σωστές, αν βγάζουν νόημα ή αν είναι συντακτικά σωστές. Απαντήσεις που απλά περιέχουν τις κατάλληλες λέξεις-

κλειδιά, βαθμολογούνται υψηλά λόγω της ταύτισης με τις λέξεις-κλειδιά των απαντήσεων του διαγωνισμού. Αυτό φαίνεται παρατηρώντας το γεγονός ότι μοντέλα με σύντομες επαναλαμβανόμενες φράσεις, όπως για παράδειγμα το Cerebras-GPT-6.7B με greedy search χωρίς repetition penalty, είχε LO 0.337 και CS 0.845, ενώ πρακτικά στις περισσότερες απαντήσεις του παγιδευόταν σε λούπες και έλεγε τις ίδιες λέξεις μέχρι να εξαντλήσει το όριο των 1200 χαρακτήρων. Ακόμα και στις περιπτώσεις αξιολόγησης με χρήση του μέσου F1 score αντί για το μέγιστο, τα greedy search runs χωρίς repetition penalty παρέμειναν σε υψηλές θέσεις, ανταγωνιζόμενα τα greedy search runs με repetition penalty, τα οποία δεν παγιδεύονταν σε ατέρμονους βρόγχους επανάληψης λέξεων-κλειδιών. Θα υπήρχε μεγαλύτερη εμπιστοσύνη στους δείκτες, αν επαρκούσε ο καθορισμός του βαθμού συσχέτισης μεταξύ των κειμένων, όπως σε ένα τυπικό σύστημα IR το οποίο επιστρέφει σχετικά έγγραφα, όμως τα αποτελέσματα δείχνουν πως όσον αφορά τη δημιουργία απαντήσεων μέσω generative συστημάτων, είναι εξίσου σημαντική και η αξιολόγηση της ορθότητας των απαντήσεων, τόσο νοηματικά όσο και συντακτικά.

Ταυτόχρονα, η επιλογή του μέγιστου F1 score που γίνεται από τον πρωταρχικό κώδικα αξιολόγησης φαίνεται να ωφελεί κάποια συστήματα περισσότερο από άλλα, λόγω του ότι έτσι αγνοεί όλες τις σχετικές απαντήσεις του διαγωνισμού με τις οποίες υπάρχει χαμηλή συσχέτιση. Αυτό έχει επίσης σαν αποτέλεσμα να αυξάνεται η πιθανότητα υψηλής βαθμολογίας συσχέτισης όταν υπάρχει μεγάλο πλήθος απαντήσεων στα δεδομένα που χρησιμοποιούνται ως σημείο αναφοράς, χωρίς να λαμβάνεται υπόψιν αν η απάντηση έχει χαμηλή συσχέτιση με την πλειοψηφία των απαντήσεων. Θεωρητικά μία απάντηση θα μπορούσε να έχει μηδενική συσχέτιση με όλες τις απαντήσεις εκτός από μία, και αυτό δεν θα ήταν εμφανές στα αποτελέσματα. Από τη στιγμή που οι απαντήσεις αυτές είναι συνδεδεμένες με το ίδιο ερώτημα, δεν θα έπρεπε να αγνοείται ο χαμηλός βαθμός συσχέτισης όταν αντιστοιχεί σε σημαντικό ποσοστό των δεδομένων. Η αλλαγή της υλοποίησης ώστε να χρησιμοποιείται το μέσο F1 score φαίνεται να εξαλείφει αυτό το πρόβλημα.

Η υλοποίηση φίλτρου για «κοινές» λέξεις, ενώ εμφάνισε βελτίωση στις βαθμολογίες των LO και CS, κρίνεται ως αντιπαραγωγική, λόγω του ότι αφαιρεί από τα κείμενα κρίσιμες πληροφορίες οι οποίες χρησιμοποιούνται από τον κώδικα αξιολόγησης ως context. Παρομοίως, το φίλτρο για τις διπλότυπες φράσεις φάνηκε να ωφελεί μόνο τα greedy search συστήματα που δεν είχαν repetition penalty, το οποίο δεν αυξάνει την αξιοπιστία της αξιολόγησης λόγω του ότι οι απαντήσεις που ωφελήθηκαν ήταν σύντομες επαναλήψεις λέξεων-κλειδιών.

Τέλος, η αποτελεσματικότητα του BERTScore στο να κατατάσσει τα συστήματα φαίνεται να είναι ανεξάρτητη του μοντέλου BERT με το οποίο χρησιμοποιείται. Παρ’ ότι ο συνδυασμός με το

μοντέλο MathBERTa βαθμολόγησε με CS 0.742 το ψεύτικο run που δημιουργήθηκε, η «θέση» στην οποία κατατάσσει τα υπόλοιπα runs επιβεβαιώνεται από τα υπόλοιπα μοντέλα BERT, με τα «καλύτερα» runs να βρίσκονται πάντα σε υψηλότερη θέση από τα «χειρότερα», και με το ψεύτικο run σταθερά στην τελευταία θέση.

Με βάση τα ανωτέρω συμπεράσματα και τις δυσκολίες που συναντήθηκαν κατά την εκπόνηση αυτής της εργασίας, προτείνονται οι παρακάτω παρατηρήσεις που θα μπορούσαν να αποτελέσουν αφορμή για μελλοντική έρευνα.

Η πιο σημαντική ανάγκη που παρατηρείται, είναι η ανάγκη για ακόμα καλύτερους τρόπους αυτόματης αξιολόγησης των συστημάτων παραγωγής μαθηματικών απαντήσεων. Οι αυτοματοποιημένοι τρόποι που υπάρχουν αυτή τη στιγμή δεν φαίνεται να δίνουν ξεκάθαρες ενδείξεις, και δεν έχουν τρόπο να αξιολογήσουν το κατά πόσο οι απαντήσεις που παράγονται καλύπτουν πλήρως και σωστά τα ερωτήματα που τίθενται στο μοντέλο. Επίσης, δεν φαίνεται να υπάρχει κάποιος τυποποιημένος τρόπος εφαρμογής των τρόπων αξιολόγησης. Ενώ πολλοί συμφωνούν με την προσέγγιση του LO και του CS, η υλοποίηση αφήνεται στην κρίση του καθενός. Όπως παρατηρήθηκε, η επιλογή χρήσης του μέγιστου ή του μέσου F1 score από μόνη της εμφάνισε τεράστιες διαφορές στα αποτελέσματα. Επίσης, ενώ το BERTScore βαθμολογεί πολύ διαφορετικά ανάλογα με το μοντέλο της οικογένειας BERT με το οποίο θα συνδυαστεί, δεν υπάρχει συμφωνία ανάμεσα στα μέλη της κοινότητας για το ποιό θα έπρεπε να είναι το μοντέλο που θα χρησιμοποιείται ως σημείο αναφοράς. Με βάση αυτά κρίνεται ότι είναι απαραίτητη η βελτίωση και τυποποίηση κατά το δυνατόν των αυτοματοποιημένων τρόπων αξιολόγησης απαντήσεων σε μαθηματικές ερωτήσεις.

Επίσης, για να γίνει πιο προσβάσιμη η έρευνα από περισσότερα μέλη της κοινότητας, θα βοηθούσε να εφευρεθούν τρόποι εκτέλεσης μεγαλύτερων μοντέλων σε υπολογιστικά συστήματα με χαμηλότερες προδιαγραφές. Προς το παρόν, για να γίνει εις βάθος έρευνα, ειδικά αν συμπεριλαμβάνεται και η εκπαίδευση μοντέλων, είναι απαραίτητη η αγορά ή μίσθωση συστημάτων με υψηλές προδιαγραφές.

Τέλος, χρειάζεται να δημιουργηθούν περισσότερες υποδομές σχετικά με την έρευνα πάνω στο MIR. Αυτό περιλαμβάνει τη συγκέντρωση μεγάλων ποσοτήτων δεδομένων που θα μπορούν να χρησιμοποιηθούν για τη εκπαίδευση των εξειδικευμένων μοντέλων, τη δημιουργία pretrained μοντέλων συγκεκριμένα για MIR και τη δημιουργία μίας κεντρικής πηγής πληροφόρησης για ενδιαφερόμενους ερευνητές.

Βιβλιογραφία

Allam, A.M., & Haggag, M.H. (2016). The Question Answering Systems : A Survey .

Almeida, F., & Xexéo, G. (2019). Word Embeddings: A Survey. *ArXiv, abs/1901.09069.*

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR, abs/1409.0473.*

Bell, J. (2020). Machine Learning (2nd ed.). Wiley. Retrieved from *https://www.perlego.com/book/2770789/machine-learning-handson-for-developers-and-technical-professionals-pdf* (Original work published 2020)

Bi, Q., Goodman, K.E., Kaminsky, J., & Lessler, J. (2019). What Is Machine Learning: a Primer for the Epidemiologist. *American journal of epidemiology.*

Bordes, A., Weston, J., Collobert, R., & Bengio, Y. (2011). Learning Structured Embeddings of Knowledge Bases. *Proceedings of the AAAI Conference on Artificial Intelligence.*

Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T.J., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language Models are Few-Shot Learners. *ArXiv, abs/2005.14165*.

Carta, S. (2022). Machine Learning and the City. Wiley. Retrieved from *https://www.perlego.com/book/3512252/machine-learning-and-the-city-applications-in-architecture-and-urban-design-pdf* (Original work published 2022)

Cotterell, R., & Schütze, H. (2019). Morphological Word-Embeddings. *North American Chapter of the Association for Computational Linguistics.*

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv, abs/1810.04805*.

Drozd, A., Rogers, A., & Matsuoka, S. (2016). Word Embeddings, Analogies, and Machine Learning: Beyond king - man + woman = queen. *International Conference on Computational Linguistics.*

Geletka, M., Kalivoda, V., Štefánik, M., Toma, M., & Sojka, P. (2022). Diverse Semantics Representation is King. *Conference and Labs of the Evaluation Forum.*

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D.D., Hendricks, L.A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., Driessche, G.V., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Rae, J.W., Vinyals, O., & Sifre, L. (2022). Training Compute-Optimal Large Language Models. *ArXiv, abs/2203.15556.*

Jurafsky, D., and Martin, J.H. (2023) Speech and Language Processing (3rd ed. draft). *Stanford University Press.*

Kane, A., Ng, Y.K., and Tompa, F.W. (2022) Dowsing for Answers to Math Questions: Doing Better with Less. *Conference and Labs of the Evaluation Forum.*

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A.P., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M., Dai, A.M.,

Uszkoreit, J., Le, Q.V., & Petrov, S. (2019). Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, 7, 453-466.

Langsenkamp, M., Mansouri, B., & Zanibbi, R. (2022) Expanding Spatial Regions and Incorporating IDF for PHOC-Based Math Formula Retrieval at ARQMath-3. *Conference and Labs of the Evaluation Forum.*

Mahesh, B. (2019). Machine Learning Algorithms -A Review. *10.21275/ART20203995.*

Mansouri, B., Novotný, V., Agarwal, A., Oard, D.W., and Zanibbi, R. (2022). Overview of ARQMath-3 (2022): Third CLEF Lab on Answer Retrieval for Questions on Math (Working Notes Version). *Conference and Labs of the Evaluation Forum.*

Mikolov, T., Chen, K., Corrado, G.S., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. International Conference on Learning Representations.

Novotný, V., and Štefánik, M. (2022) Combining Sparse and Dense Information Retrieval Soft Vector Space Model and MathBERTa at ARQMath-3 Task 1 (Answer Retrieval). *Conference and Labs of the Evaluation Forum.*

Olmo, A., Sreedharan, S., & Kambhampati, S. (2021). GPT3-to-plan: Extracting plans from text using GPT-3. ArXiv, abs/2106.07131.

Pennington, J., Socher, R., & Manning, C.D. (2014). GloVe: Global Vectors for Word Representation. Conference on Empirical Methods in Natural Language Processing.

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. *ArXiv, abs/1606.05250.*

Reddy, S., Chen, D., & Manning, C.D. (2018). CoQA: A Conversational Question Answering Challenge. *Transactions of the Association for Computational Linguistics*, 7, 249-266.

Reusch, A., Thiele, M., & Lehner, W. (2022). Transformer-Encoder and Decoder Models for Questions on Math. *Conference and Labs of the Evaluation Forum.*

Rodriguez, P.L., & Spirling, A. (2021). Word Embeddings: What Works, What Doesn't, and How to Tell the Difference for Applied Research. *The Journal of Politics*, 84, 101 - 115.

Sarkar, S., Das, D., Pakray, P., and Pinto, D. (2022). Formula Retrieval Using Structural Similarity. *Conference and Labs of the Evaluation Forum.*

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., & Catanzaro, B. (2019). Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. *ArXiv, abs/1909.08053.*

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., and Lample, G. (2023), LLaMA: Open and Efficient Foundation Language Models. *ArXiv, abs/2302.13971.*

Tripodi, R., & Li Pira, S. 2017. Analysis of Italian Word Embeddings. In Basili, R., Nissim, M., & Satta, G. (Eds.), *Proceedings of the Fourth Italian Conference on Computational Linguistics CLiC-it 2017:* 11-12 December 2017, Rome. Torino: Accademia University Press. doi:10.4000/books.aaccademia.2475

Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is All you Need. *NIPS.*

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M.T., Li, X., Lin, X.V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, P.S., Sridhar, A., Wang, T., & Zettlemoyer, L. (2022). OPT: Open Pre-trained Transformer Language Models. *ArXiv, abs/2205.01068.*

Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., & Artzi, Y. (2019). BERTScore: Evaluating Text Generation with BERT. *ArXiv, abs/1904.09675.*

Zhong, W., Xie, Y., and Lin, J. (2022). Applying Structural and Dense Semantic Matching for the ARQMath Lab 2022, CLEF. *Conference and Labs of the Evaluation Forum*.

Πηγές από το Διαδίκτυο

https://www.baeldung.com/cs/fine-tuning-nn

https://www.taygan.co/blog/2023/02/01/azure-openai

https://medium.com/@shixiangtech2019/deep-dive-on-hugging-face-87cfd122c1a4

https://www.cnrs.fr/en/release-largest-trained-open-science-multilingual-language-model-ever

https://huggingface.co/bigscience/bloom-560m#languages

https://huggingface.co/bigscience/bloom-560m#training-data

https://huggingface.co/bigscience/bloom-560m#tokenization

https://huggingface.co/bigscience/bloom-560m#direct-use

https://huggingface.co/facebook/opt-350m#intro

https://huggingface.co/facebook/opt-350m#training-data

https://huggingface.co/facebook/opt-350m#preprocessing

https://huggingface.co/cerebras/Cerebras-GPT-256M#intended-use

https://huggingface.co/cerebras/Cerebras-GPT-111M#training-data

https://huggingface.co/cerebras/Cerebras-GPT-256M#model-description

https://huggingface.co/cerebras/Cerebras-GPT-6.7B

https://huggingface.co/decapoda-research/llama-7b-hf#intended-use

https://huggingface.co/decapoda-research/llama-7b-hf#training-dataset

https://huggingface.co/decapoda-research/llama-13b-hf

https://huggingface.co/blog/how-to-generate

https://huggingface.co/blog/how-to-generate

https://drive.google.com/drive/u/0/folders/1iE8gEU_7tT5wb40TSTJb_tUZsWChumVi

Παράρτημα

	Παράμετροι μοντέλου	Layers	Attention heads
GPT-3			
Ada	2,7 δισεκατομμύρια	32	32
Text-ada-001	2,7 δισεκατομμύρια	32	32
Babbage	6,7 δισεκατομμύρια	32	32
Text-babbage-001	6,7 δισεκατομμύρια	32	32
Curie	13 δισεκατομμύρια	40	40
Text-curie-001	13 δισεκατομμύρια	40	40
Davinci	175 δισεκατομμύρια	96	96
Text-davinci-001	175 δισεκατομμύρια	96	96
GPT-3.5			
Text-davinci-002	175 δισεκατομμύρια	96	96
Text-davinci-003	175 δισεκατομμύρια	96	96
GPT-3.5-turbo	175 δισεκατομμύρια	96	96
GPT-3.5-turbo-0301	175 δισεκατομμύρια	96	96
bigscience/bloom			
bloom-560m	560 εκατομμύρια	24	16
bloom-3b	3 δισεκατομμύρια	30	32
bloom-7b1	7 δισεκατομμύρια	30	32
facebook/opt			
opt-125m	125 εκατομμύρια	12	12

opt-350m	350 εκατομμύρια	24	16
opt-1.3b	1,3 δισεκατομμύρια	24	32
opt-2.7b	2,7 δισεκατομμύρια	32	32
opt-6.7b	6,7 δισεκατομμύρια	32	32
cerebras/cerebras-gpt			
cerebras-gpt-111m	111 εκατομμύρια	10	12
cerebras-gpt-256m	256 εκατομμύρια	14	17
cerebras-gpt-590m	590 εκατομμύρια	18	12
cerebras-gpt-1.3b	1,3 δισεκατομμύρια	24	16
cerebras-gpt-2.7b	2,7 δισεκατομμύρια	32	20
cerebras-gpt-6.7b	6,7 δισεκατομμύρια	32	32
decapoda-research/llama			
llama-7b-hf	7 δισεκατομμύρια	32	32

Πίνακας 15. Μοντέλα που χρησιμοποιήθηκαν για την εργασία

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	**0.317**	**0.851**
Approach0		
run1	0.509	0.886
run4	**0.515**	**0.886**
run3	0.467	0.879
run2	0.427	0.868
run5	0.444	0.873
DPRL		
SBERT-SVMRank	**0.338**	0.848
BERT-SVMRank	0.337	0.848
SBERT-QQ-AMR	0.325	**0.852**

BERT-QQ-AMR	0.323	0.851
TU_DBS		
amps3_se1_hints	**0.263**	**0.835**
se3_len_pen_10	0.248	0.806
amps3_se1_len_pen_20_sample_hint	0.254	0.813
shortest	0.239	0.820
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	**0.025**	**0.742**
OpenAI text generators		
ada-deterministic	0.276	0.798
ada-deterministic-with-repetition-penalty	**0.308**	**0.831**
ada-creative	0.289	0.808
ada-creative-with-repetition-penalty	0.287	0.815
babbage-deterministic	**0.319**	0.816
babbage-deterministic-with-repetition-penalty	0.308	**0.825**
babbage-creative	0.312	0.824
babbage-creative-with-repetition-penalty	0.282	0.814
curie-deterministic	**0.326**	0.818
curie-deterministic-with-repetition-penalty	0.318	**0.831**
curie-creative	0.308	0.826
curie-creative-with-repetition-penalty	0.289	0.823
davinci-deterministic	**0.337**	0.820
davinci-deterministic-with-repetition-penalty	0.326	**0.833**
davinci-creative	0.321	0.829
davinci-creative-with-repetition-penalty	0.297	0.821
text-ada-001-deterministic	0.270	0.829

text-ada-001-deterministic-with-repetition-penalty	0.263	**0.832**
text-ada-001-creative	0.256	0.826
text-ada-001-creative-with-repetition-penalty	**0.274**	0.828
text-babbage-001-deterministic	0.282	0.836
text-babbage-001-deterministic-with-repetition-penalty	0.283	0.832
text-babbage-001-creative	0.270	0.832
text-babbage-001-creative-with-repetition-penalty	**0.285**	**0.836**
text-curie-001-deterministic	**0.302**	**0.841**
text-curie-001-deterministic-with-repetition-penalty	0.296	0.839
text-curie-001-creative	0.285	0.837
text-curie-001-creative-with-repetition-penalty	0.291	0.838
text-davinci-002-deterministic	**0.348**	**0.858**
text-davinci-002-deterministic-with-repetition-penalty	0.342	0.847
text-davinci-002-creative	0.330	0.849
text-davinci-002-creative-with-repetition-penalty	0.327	0.841
text-davinci-003-deterministic	**0.357**	**0.853**
text-davinci-003-deterministic-with-repetition-penalty	0.330	0.839
text-davinci-003-creative	0.342	0.852
text-davinci-003-creative-with-repetition-penalty	0.331	0.837
OpenAI chatbots		
gpt-3.5-turbo-deterministic	**0.357**	0.849
gpt-3.5-turbo-creative	0.351	**0.850**
gpt-3.5-turbo-0301-deterministic	**0.356**	0.849
gpt-3.5-turbo-0301-creative	0.355	**0.850**
Hugging Face text generators		
bloom-560m-deterministic	**0.315**	**0.825**
bloom-560m-deterministic-with-repetition-penalty	0.206	0.818
bloom-560m-creative	0.203	0.789

bloom-560m-creative-with-repetition-penalty	0.153	0.780
bloom-3b-deterministic	**0.322**	**0.829**
bloom-3b-deterministic-with-repetition-penalty	0.228	0.824
bloom-3b-creative	0.194	0.784
bloom-3b-creative-with-repetition-penalty	0.166	0.779
bloom-7b1-deterministic	**0.325**	**0.834**
bloom-7b1-deterministic-with-repetition-penalty	0.235	0.825
bloom-7b1-creative	0.176	0.775
bloom-7b1-creative-with-repetition-penalty	0.150	0.774
Cerebras-GPT-111M-deterministic	0.211	0.780
Cerebras-GPT-111M -deterministic-with-repetition-penalty	**0.220**	**0.812**
Cerebras-GPT-111M -creative	0.183	0.769
Cerebras-GPT-111M -creative-with-repetition-penalty	0.187	0.770
Cerebras-GPT-256M-deterministic	**0.268**	0.805
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.241	**0.823**
Cerebras-GPT-256M -creative	0.187	0.770
Cerebras-GPT-256M -creative-with-repetition-penalty	0.168	0.768
Cerebras-GPT-590M-deterministic	**0.274**	0.813
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.255	**0.825**
Cerebras-GPT-590M-creative	0.198	0.775
Cerebras-GPT-590M-creative-with-repetition-penalty	0.177	0.773
Cerebras-GPT-1.3B-deterministic	**0.320**	**0.832**
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.243	0.826
Cerebras-GPT-1.3B-creative	0.198	0.773
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.171	0.767
Cerebras-GPT-2.7B-deterministic	**0.319**	**0.834**
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.254	0.831
Cerebras-GPT-2.7B-creative	0.213	0.770

Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.186	0.767
Cerebras-GPT-6.7B-deterministic	**0.337**	**0.845**
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.258	0.829
Cerebras-GPT-6.7B-creative	0.209	0.766
Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.175	0.760
llama-7b-hf-deterministic	**0.298**	**0.811**
llama-7b-hf-deterministic-with-repetition-penalty	0.227	0.804
llama-7b-hf-creative	0.201	0.764
llama-7b-hf-creative-with-repetition-penalty	0.176	0.764
opt-125m-deterministic	**0.303**	**0.807**
opt-125m-deterministic-with-repetition-penalty	0.265	0.791
opt-125m-creative	0.178	0.774
opt-125m-creative-with-repetition-penalty	0.181	0.768
opt-350m-deterministic	**0.278**	**0.805**
opt-350m-deterministic-with-repetition-penalty	0.247	0.799
opt-350m-creative	0.200	0.773
opt-350m-creative-with-repetition-penalty	0.177	0.763
opt-1.3b-deterministic	**0.289**	**0.807**
opt-1.3b-deterministic-with-repetition-penalty	0.268	0.803
opt-1.3b-creative	0.199	0.771
opt-1.3b-creative-with-repetition-penalty	0.179	0.767
opt-2.7b-deterministic	**0.319**	**0.819**
opt-2.7b-deterministic-with-repetition-penalty	0.244	0.791
opt-2.7b-creative	0.204	0.773
opt-2.7b-creative-with-repetition-penalty	0.166	0.758
opt-6.7b-deterministic	**0.315**	**0.821**
opt-6.7b-deterministic-with-repetition-penalty	0.266	0.806
opt-6.7b-creative	0.203	0.768

opt-6.7b-creative-with-repetition-penalty	0.178	0.763

Πίνακας 16. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του διαγωνισμού μέσω του αρχικού αλγόριθμου

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.308	0.850
Approach0		
run1	0.503	0.884
run4	0.510	0.885
run3	0.462	0.878
run2	0.423	0.867
run5	0.440	0.872
DPRL		
SBERT-SVMRank	0.326	0.847
BERT-SVMRank	0.325	0.847
SBERT-QQ-AMR	0.318	0.851
BERT-QQ-AMR	0.315	0.850
TU_DBS		
amps3_se1_hints	0.234	0.831
se3_len_pen_10	0.229	0.802
amps3_se1_len_pen_20_sample_hint	0.227	0.809
shortest	0.219	0.817
New runs		

Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.024	0.742
OpenAI text generators		
ada-deterministic	0.255	0.794
ada-deterministic-with-repetition-penalty	0.294	0.829
ada-creative	0.271	0.805
ada-creative-with-repetition-penalty	0.282	0.814
babbage-deterministic	0.287	0.810
babbage-deterministic-with-repetition-penalty	0.297	0.823
babbage-creative	0.299	0.821
babbage-creative-with-repetition-penalty	0.276	0.813
curie-deterministic	0.301	0.813
curie-deterministic-with-repetition-penalty	0.306	0.829
curie-creative	0.294	0.823
curie-creative-with-repetition-penalty	0.282	0.821
davinci-deterministic	0.307	0.815
davinci-deterministic-with-repetition-penalty	0.313	0.830
davinci-creative	0.305	0.826
davinci-creative-with-repetition-penalty	0.287	0.820
text-ada-001-deterministic	0.249	0.826
text-ada-001-deterministic-with-repetition-penalty	0.248	0.829
text-ada-001-creative	0.239	0.824
text-ada-001-creative-with-repetition-penalty	0.250	0.825
text-babbage-001-deterministic	0.248	0.832
text-babbage-001-deterministic-with-repetition-penalty	0.249	0.828
text-babbage-001-creative	0.244	0.829
text-babbage-001-creative-with-repetition-penalty	0.253	0.832

text-curie-001-deterministic	0.275	0.838
text-curie-001-deterministic-with-repetition-penalty	0.278	0.836
text-curie-001-creative	0.269	0.835
text-curie-001-creative-with-repetition-penalty	0.270	0.835
text-davinci-002-deterministic	0.297	0.848
text-davinci-002-deterministic-with-repetition-penalty	0.319	0.843
text-davinci-002-creative	0.298	0.843
text-davinci-002-creative-with-repetition-penalty	0.311	0.838
text-davinci-003-deterministic	0.333	0.847
text-davinci-003-deterministic-with-repetition-penalty	0.314	0.835
text-davinci-003-creative	0.322	0.847
text-davinci-003-creative-with-repetition-penalty	0.316	0.834
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.343	0.845
gpt-3.5-turbo-creative	0.335	0.845
gpt-3.5-turbo-0301-deterministic	0.343	0.845
gpt-3.5-turbo-0301-creative	0.338	0.845
Hugging Face text generators		
bloom-560m-deterministic	0.284	0.820
bloom-560m-deterministic-with-repetition-penalty	0.203	0.818
bloom-560m-creative	0.199	0.788
bloom-560m-creative-with-repetition-penalty	0.149	0.780
bloom-3b-deterministic	0.295	0.826
bloom-3b-deterministic-with-repetition-penalty	0.219	0.822
bloom-3b-creative	0.192	0.784
bloom-3b-creative-with-repetition-penalty	0.164	0.778
bloom-7b1-deterministic	0.301	0.830
bloom-7b1-deterministic-with-repetition-penalty	0.228	0.823

bloom-7b1-creative	0.174	0.774
bloom-7b1-creative-with-repetition-penalty	0.148	0.774
Cerebras-GPT-111M-deterministic	0.193	0.777
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.215	0.812
Cerebras-GPT-111M -creative	0.181	0.769
Cerebras-GPT-111M -creative-with-repetition-penalty	0.186	0.769
Cerebras-GPT-256M-deterministic	0.249	0.802
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.235	0.821
Cerebras-GPT-256M -creative	0.186	0.770
Cerebras-GPT-256M -creative-with-repetition-penalty	0.167	0.768
Cerebras-GPT-590M-deterministic	0.254	0.811
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.250	0.824
Cerebras-GPT-590M-creative	0.197	0.775
Cerebras-GPT-590M-creative-with-repetition-penalty	0.176	0.773
Cerebras-GPT-1.3B-deterministic	0.273	0.825
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.236	0.825
Cerebras-GPT-1.3B-creative	0.196	0.773
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.169	0.767
Cerebras-GPT-2.7B-deterministic	0.282	0.828
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.250	0.831
Cerebras-GPT-2.7B-creative	0.211	0.769
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.185	0.767
Cerebras-GPT-6.7B-deterministic	0.297	0.836
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.254	0.828
Cerebras-GPT-6.7B-creative	0.209	0.766
Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.174	0.760
llama-7b-hf-deterministic	0.273	0.807
llama-7b-hf-deterministic-with-repetition-penalty	0.225	0.804

llama-7b-hf-creative	0.199	0.764
llama-7b-hf-creative-with-repetition-penalty	0.175	0.764
opt-125m-deterministic	0.263	0.803
opt-125m-deterministic-with-repetition-penalty	0.263	0.790
opt-125m-creative	0.176	0.774
opt-125m-creative-with-repetition-penalty	0.181	0.768
opt-350m-deterministic	0.256	0.800
opt-350m-deterministic-with-repetition-penalty	0.244	0.797
opt-350m-creative	0.199	0.773
opt-350m-creative-with-repetition-penalty	0.176	0.763
opt-1.3b-deterministic	0.263	0.803
opt-1.3b-deterministic-with-repetition-penalty	0.266	0.802
opt-1.3b-creative	0.197	0.771
opt-1.3b-creative-with-repetition-penalty	0.179	0.766
opt-2.7b-deterministic	0.284	0.814
opt-2.7b-deterministic-with-repetition-penalty	0.242	0.791
opt-2.7b-creative	0.202	0.772
opt-2.7b-creative-with-repetition-penalty	0.166	0.758
opt-6.7b-deterministic	0.283	0.817
opt-6.7b-deterministic-with-repetition-penalty	0.264	0.806
opt-6.7b-creative	0.200	0.768
opt-6.7b-creative-with-repetition-penalty	0.177	0.763

Πίνακας 17. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του Task 1 του διαγωνισμού μέσω του αρχικού αλγόριθμου

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.270	0.835
Approach0		
run1	0.254	0.833
run4	0.250	0.828
run3	0.261	0.832
run2	0.250	0.829
run5	0.246	0.827
DPRL		
SBERT-SVMRank	0.237	0.820
BERT-SVMRank	0.238	0.821
SBERT-QQ-AMR	0.213	0.817
BERT-QQ-AMR	0.219	0.818
TU_DBS		
amps3_se1_hints	0.201	0.813
se3_len_pen_10	0.195	0.786
amps3_se1_len_pen_20_sample_hint	0.200	0.793
shortest	0.182	0.798
New runs		

Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.008	0.722
OpenAI text generators		
ada-deterministic	0.243	0.785
ada-deterministic-with-repetition-penalty	0.271	0.816
ada-creative	0.262	0.796
ada-creative-with-repetition-penalty	0.244	0.802
babbage-deterministic	0.282	0.806
babbage-deterministic-with-repetition-penalty	0.268	0.815
babbage-creative	0.276	0.815
babbage-creative-with-repetition-penalty	0.237	0.804
curie-deterministic	0.289	0.809
curie-deterministic-with-repetition-penalty	0.286	0.822
curie-creative	0.275	0.811
curie-creative-with-repetition-penalty	0.246	0.811
davinci-deterministic	0.305	0.809
davinci-deterministic-with-repetition-penalty	0.292	0.822
davinci-creative	0.285	0.816
davinci-creative-with-repetition-penalty	0.265	0.814
text-ada-001-deterministic	0.235	0.808
text-ada-001-deterministic-with-repetition-penalty	0.223	0.811
text-ada-001-creative	0.214	0.802
text-ada-001-creative-with-repetition-penalty	0.233	0.808
text-babbage-001-deterministic	0.242	0.815
text-babbage-001-deterministic-with-repetition-penalty	0.243	0.811
text-babbage-001-creative	0.237	0.813
text-babbage-001-creative-with-repetition-penalty	0.242	0.812

text-curie-001-deterministic	0.245	0.814
text-curie-001-deterministic-with-repetition-penalty	0.237	0.815
text-curie-001-creative	0.236	0.816
text-curie-001-creative-with-repetition-penalty	0.236	0.813
text-davinci-002-deterministic	0.313	0.842
text-davinci-002-deterministic-with-repetition-penalty	0.315	0.838
text-davinci-002-creative	0.299	0.836
text-davinci-002-creative-with-repetition-penalty	0.290	0.827
text-davinci-003-deterministic	0.319	0.836
text-davinci-003-deterministic-with-repetition-penalty	0.296	0.828
text-davinci-003-creative	0.323	0.842
text-davinci-003-creative-with-repetition-penalty	0.294	0.825
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.326	0.839
gpt-3.5-turbo-creative	0.321	0.839
gpt-3.5-turbo-0301-deterministic	0.324	0.839
gpt-3.5-turbo-0301-creative	0.324	0.839
Hugging Face text generators		
bloom-560m-deterministic	0.267	0.811
bloom-560m-deterministic-with-repetition-penalty	0.167	0.803
bloom-560m-creative	0.168	0.775
bloom-560m-creative-with-repetition-penalty	0.123	0.767
bloom-3b-deterministic	0.269	0.810
bloom-3b-deterministic-with-repetition-penalty	0.194	0.811
bloom-3b-creative	0.160	0.773
bloom-3b-creative-with-repetition-penalty	0.133	0.769
bloom-7b1-deterministic	0.282	0.824
bloom-7b1-deterministic-with-repetition-penalty	0.211	0.816

bloom-7b1-creative	0.143	0.762
bloom-7b1-creative-with-repetition-penalty	0.116	0.761
Cerebras-GPT-111M-deterministic	0.177	0.770
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.171	0.791
Cerebras-GPT-111M -creative	0.138	0.756
Cerebras-GPT-111M -creative-with-repetition-penalty	0.142	0.756
Cerebras-GPT-256M-deterministic	0.207	0.784
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.197	0.803
Cerebras-GPT-256M -creative	0.145	0.759
Cerebras-GPT-256M -creative-with-repetition-penalty	0.119	0.755
Cerebras-GPT-590M-deterministic	0.231	0.802
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.206	0.804
Cerebras-GPT-590M-creative	0.155	0.762
Cerebras-GPT-590M-creative-with-repetition-penalty	0.135	0.759
Cerebras-GPT-1.3B-deterministic	0.294	0.822
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.191	0.808
Cerebras-GPT-1.3B-creative	0.158	0.761
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.127	0.752
Cerebras-GPT-2.7B-deterministic	0.276	0.821
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.193	0.808
Cerebras-GPT-2.7B-creative	0.172	0.756
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.136	0.752
Cerebras-GPT-6.7B-deterministic	0.306	0.835
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.201	0.810
Cerebras-GPT-6.7B-creative	0.167	0.753
Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.126	0.747
llama-7b-hf-deterministic	0.265	0.799
llama-7b-hf-deterministic-with-repetition-penalty	0.180	0.790

llama-7b-hf-creative	0.160	0.751
llama-7b-hf-creative-with-repetition-penalty	0.140	0.752
opt-125m-deterministic	0.265	0.796
opt-125m-deterministic-with-repetition-penalty	0.213	0.779
opt-125m-creative	0.142	0.761
opt-125m-creative-with-repetition-penalty	0.133	0.755
opt-350m-deterministic	0.231	0.790
opt-350m-deterministic-with-repetition-penalty	0.196	0.785
opt-350m-creative	0.149	0.757
opt-350m-creative-with-repetition-penalty	0.134	0.749
opt-1.3b-deterministic	0.248	0.793
opt-1.3b-deterministic-with-repetition-penalty	0.205	0.786
opt-1.3b-creative	0.151	0.756
opt-1.3b-creative-with-repetition-penalty	0.129	0.752
opt-2.7b-deterministic	0.282	0.808
opt-2.7b-deterministic-with-repetition-penalty	0.177	0.775
opt-2.7b-creative	0.154	0.758
opt-2.7b-creative-with-repetition-penalty	0.117	0.743
opt-6.7b-deterministic	0.276	0.805
opt-6.7b-deterministic-with-repetition-penalty	0.202	0.790
opt-6.7b-creative	0.157	0.756
opt-6.7b-creative-with-repetition-penalty	0.131	0.749

Πίνακας 18. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του Task 3 του διαγωνισμού μέσω του αρχικού αλγόριθμου

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.195	0.821
Approach0		
run1	0.222	0.822
run4	0.219	0.819
run3	0.213	0.822
run2	0.216	0.820
run5	0.209	0.819
DPRL		
SBERT-SVMRank	0.185	0.811
BERT-SVMRank	0.184	0.811
SBERT-QQ-AMR	0.157	0.809
BERT-QQ-AMR	0.158	0.809
TU_DBS		
amps3_se1_hints	0.110	0.791
se3_len_pen_10	0.114	0.765

amps3_se1_len_pen_20_sample_hint	0.114	0.773
shortest	0.112	0.780
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.004	0.716
OpenAI text generators		
ada-deterministic	0.155	0.768
ada-deterministic-with-repetition-penalty	0.201	0.805
ada-creative	0.189	0.781
ada-creative-with-repetition-penalty	0.192	0.794
babbage-deterministic	0.187	0.784
babbage-deterministic-with-repetition-penalty	0.201	0.801
babbage-creative	0.206	0.798
babbage-creative-with-repetition-penalty	0.183	0.792
curie-deterministic	0.204	0.788
curie-deterministic-with-repetition-penalty	0.214	0.808
curie-creative	0.204	0.798
curie-creative-with-repetition-penalty	0.189	0.801
davinci-deterministic	0.203	0.789
davinci-deterministic-with-repetition-penalty	0.217	0.808
davinci-creative	0.214	0.802
davinci-creative-with-repetition-penalty	0.194	0.801
text-ada-001-deterministic	0.159	0.794
text-ada-001-deterministic-with-repetition-penalty	0.159	0.798
text-ada-001-creative	0.146	0.789
text-ada-001-creative-with-repetition-penalty	0.161	0.794
text-babbage-001-deterministic	0.149	0.797

text-babbage-001-deterministic-with-repetition-penalty	0.153	0.793
text-babbage-001-creative	0.147	0.794
text-babbage-001-creative-with-repetition-penalty	0.159	0.796
text-curie-001-deterministic	0.160	0.802
text-curie-001-deterministic-with-repetition-penalty	0.162	0.802
text-curie-001-creative	0.161	0.798
text-curie-001-creative-with-repetition-penalty	0.166	0.801
text-davinci-002-deterministic	0.186	0.814
text-davinci-002-deterministic-with-repetition-penalty	0.220	0.818
text-davinci-002-creative	0.193	0.812
text-davinci-002-creative-with-repetition-penalty	0.208	0.811
text-davinci-003-deterministic	0.223	0.817
text-davinci-003-deterministic-with-repetition-penalty	0.215	0.810
text-davinci-003-creative	0.223	0.820
text-davinci-003-creative-with-repetition-penalty	0.217	0.810
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.239	0.819
gpt-3.5-turbo-creative	0.235	0.820
gpt-3.5-turbo-0301-deterministic	0.236	0.819
gpt-3.5-turbo-0301-creative	0.234	0.820
Hugging Face text generators		
bloom-560m-deterministic	0.177	0.793
bloom-560m-deterministic-with-repetition-penalty	0.136	0.797
bloom-560m-creative	0.136	0.770
bloom-560m-creative-with-repetition-penalty	0.097	0.763
bloom-3b-deterministic	0.184	0.797
bloom-3b-deterministic-with-repetition-penalty	0.151	0.803
bloom-3b-creative	0.126	0.767

bloom-3b-creative-with-repetition-penalty	0.102	0.761
bloom-7b1-deterministic	0.184	0.799
bloom-7b1-deterministic-with-repetition-penalty	0.163	0.805
bloom-7b1-creative	0.114	0.754
bloom-7b1-creative-with-repetition-penalty	0.095	0.756
Cerebras-GPT-111M-deterministic	0.117	0.756
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.141	0.784
Cerebras-GPT-111M -creative	0.118	0.753
Cerebras-GPT-111M -creative-with-repetition-penalty	0.117	0.751
Cerebras-GPT-256M-deterministic	0.147	0.771
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.158	0.795
Cerebras-GPT-256M -creative	0.121	0.757
Cerebras-GPT-256M -creative-with-repetition-penalty	0.104	0.752
Cerebras-GPT-590M-deterministic	0.158	0.787
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.163	0.799
Cerebras-GPT-590M-creative	0.126	0.757
Cerebras-GPT-590M-creative-with-repetition-penalty	0.115	0.757
Cerebras-GPT-1.3B-deterministic	0.171	0.801
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.152	0.797
Cerebras-GPT-1.3B-creative	0.129	0.755
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.106	0.750
Cerebras-GPT-2.7B-deterministic	0.161	0.796
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.152	0.797
Cerebras-GPT-2.7B-creative	0.140	0.751
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.119	0.751
Cerebras-GPT-6.7B-deterministic	0.181	0.807
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.162	0.798
Cerebras-GPT-6.7B-creative	0.138	0.751

Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.106	0.742
llama-7b-hf-deterministic	0.178	0.783
llama-7b-hf-deterministic-with-repetition-penalty	0.146	0.785
llama-7b-hf-creative	0.130	0.748
llama-7b-hf-creative-with-repetition-penalty	0.113	0.750
opt-125m-deterministic	0.172	0.782
opt-125m-deterministic-with-repetition-penalty	0.171	0.769
opt-125m-creative	0.119	0.755
opt-125m-creative-with-repetition-penalty	0.116	0.753
opt-350m-deterministic	0.155	0.777
opt-350m-deterministic-with-repetition-penalty	0.160	0.774
opt-350m-creative	0.121	0.751
opt-350m-creative-with-repetition-penalty	0.109	0.744
opt-1.3b-deterministic	0.163	0.775
opt-1.3b-deterministic-with-repetition-penalty	0.166	0.780
opt-1.3b-creative	0.126	0.753
opt-1.3b-creative-with-repetition-penalty	0.107	0.748
opt-2.7b-deterministic	0.182	0.790
opt-2.7b-deterministic-with-repetition-penalty	0.151	0.771
opt-2.7b-creative	0.127	0.752
opt-2.7b-creative-with-repetition-penalty	0.098	0.741
opt-6.7b-deterministic	0.182	0.789
opt-6.7b-deterministic-with-repetition-penalty	0.167	0.783
opt-6.7b-creative	0.129	0.752
opt-6.7b-creative-with-repetition-penalty	0.107	0.747

Πίνακας 19. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του διαγωνισμού μέσω του μέσου F1 score

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.192	0.820
Approach0		
run1	0.221	0.822
run4	0.218	0.818
run3	0.212	0.822
run2	0.216	0.819
run5	0.209	0.819
DPRL		
SBERT-SVMRank	0.184	0.811
BERT-SVMRank	0.182	0.811
SBERT-QQ-AMR	0.155	0.809
BERT-QQ-AMR	0.155	0.809
TU_DBS		
amps3_se1_hints	0.107	0.790
se3_len_pen_10	0.111	0.764

amps3_se1_len_pen_20_sample_hint	0.111	0.772
shortest	0.109	0.779
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.004	0.716
OpenAI text generators		
ada-deterministic	0.152	0.767
ada-deterministic-with-repetition-penalty	0.199	0.805
ada-creative	0.186	0.780
ada-creative-with-repetition-penalty	0.191	0.794
babbage-deterministic	0.183	0.783
babbage-deterministic-with-repetition-penalty	0.199	0.800
babbage-creative	0.203	0.797
babbage-creative-with-repetition-penalty	0.182	0.792
curie-deterministic	0.200	0.787
curie-deterministic-with-repetition-penalty	0.211	0.807
curie-creative	0.201	0.797
curie-creative-with-repetition-penalty	0.187	0.801
davinci-deterministic	0.199	0.788
davinci-deterministic-with-repetition-penalty	0.214	0.807
davinci-creative	0.211	0.802
davinci-creative-with-repetition-penalty	0.191	0.801
text-ada-001-deterministic	0.156	0.793
text-ada-001-deterministic-with-repetition-penalty	0.156	0.798
text-ada-001-creative	0.143	0.789
text-ada-001-creative-with-repetition-penalty	0.158	0.794
text-babbage-001-deterministic	0.146	0.796

text-babbage-001-deterministic-with-repetition-penalty	0.149	0.792
text-babbage-001-creative	0.143	0.794
text-babbage-001-creative-with-repetition-penalty	0.155	0.795
text-curie-001-deterministic	0.156	0.802
text-curie-001-deterministic-with-repetition-penalty	0.159	0.801
text-curie-001-creative	0.158	0.798
text-curie-001-creative-with-repetition-penalty	0.164	0.801
text-davinci-002-deterministic	0.180	0.814
text-davinci-002-deterministic-with-repetition-penalty	0.216	0.817
text-davinci-002-creative	0.188	0.811
text-davinci-002-creative-with-repetition-penalty	0.205	0.811
text-davinci-003-deterministic	0.218	0.816
text-davinci-003-deterministic-with-repetition-penalty	0.211	0.810
text-davinci-003-creative	0.219	0.819
text-davinci-003-creative-with-repetition-penalty	0.214	0.809
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.234	0.818
gpt-3.5-turbo-creative	0.231	0.819
gpt-3.5-turbo-0301-deterministic	0.231	0.818
gpt-3.5-turbo-0301-creative	0.230	0.819
Hugging Face text generators		
bloom-560m-deterministic	0.174	0.793
bloom-560m-deterministic-with-repetition-penalty	0.136	0.797
bloom-560m-creative	0.135	0.770
bloom-560m-creative-with-repetition-penalty	0.096	0.763
bloom-3b-deterministic	0.181	0.796
bloom-3b-deterministic-with-repetition-penalty	0.150	0.803
bloom-3b-creative	0.125	0.767

bloom-3b-creative-with-repetition-penalty	0.101	0.760
bloom-7b1-deterministic	0.180	0.797
bloom-7b1-deterministic-with-repetition-penalty	0.161	0.805
bloom-7b1-creative	0.113	0.754
bloom-7b1-creative-with-repetition-penalty	0.094	0.756
Cerebras-GPT-111M-deterministic	0.114	0.755
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.141	0.784
Cerebras-GPT-111M -creative	0.118	0.753
Cerebras-GPT-111M -creative-with-repetition-penalty	0.117	0.751
Cerebras-GPT-256M-deterministic	0.145	0.771
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.156	0.795
Cerebras-GPT-256M -creative	0.121	0.757
Cerebras-GPT-256M -creative-with-repetition-penalty	0.104	0.752
Cerebras-GPT-590M-deterministic	0.156	0.787
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.162	0.799
Cerebras-GPT-590M-creative	0.126	0.757
Cerebras-GPT-590M-creative-with-repetition-penalty	0.116	0.758
Cerebras-GPT-1.3B-deterministic	0.165	0.800
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.151	0.797
Cerebras-GPT-1.3B-creative	0.129	0.755
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.106	0.750
Cerebras-GPT-2.7B-deterministic	0.156	0.795
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.150	0.797
Cerebras-GPT-2.7B-creative	0.140	0.751
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.118	0.751
Cerebras-GPT-6.7B-deterministic	0.176	0.806
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.161	0.798
Cerebras-GPT-6.7B-creative	0.137	0.751

Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.106	0.742
llama-7b-hf-deterministic	0.174	0.782
llama-7b-hf-deterministic-with-repetition-penalty	0.146	0.785
llama-7b-hf-creative	0.129	0.748
llama-7b-hf-creative-with-repetition-penalty	0.113	0.750
opt-125m-deterministic	0.169	0.781
opt-125m-deterministic-with-repetition-penalty	0.171	0.769
opt-125m-creative	0.119	0.755
opt-125m-creative-with-repetition-penalty	0.116	0.753
opt-350m-deterministic	0.152	0.776
opt-350m-deterministic-with-repetition-penalty	0.160	0.774
opt-350m-creative	0.121	0.751
opt-350m-creative-with-repetition-penalty	0.108	0.744
opt-1.3b-deterministic	0.160	0.774
opt-1.3b-deterministic-with-repetition-penalty	0.166	0.779
opt-1.3b-creative	0.125	0.753
opt-1.3b-creative-with-repetition-penalty	0.107	0.749
opt-2.7b-deterministic	0.178	0.789
opt-2.7b-deterministic-with-repetition-penalty	0.152	0.771
opt-2.7b-creative	0.127	0.752
opt-2.7b-creative-with-repetition-penalty	0.099	0.741
opt-6.7b-deterministic	0.179	0.789
opt-6.7b-deterministic-with-repetition-penalty	0.167	0.783
opt-6.7b-creative	0.129	0.752
opt-6.7b-creative-with-repetition-penalty	0.107	0.747

Πίνακας 20. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του Task 1 του διαγωνισμού μέσω του μέσου F1 score

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.243	0.827
Approach0		
run1	0.238	0.827
run4	0.232	0.823
run3	0.243	0.827
run2	0.229	0.824
run5	0.227	0.821
DPRL		
SBERT-SVMRank	0.210	0.813
BERT-SVMRank	0.209	0.813
SBERT-QQ-AMR	0.196	0.810
BERT-QQ-AMR	0.200	0.812
TU_DBS		
amps3_se1_hints	0.159	0.804
se3_len_pen_10	0.160	0.779

amps3_se1_len_pen_20_sample_hint	0.163	0.787
shortest	0.156	0.793
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.004	0.713
OpenAI text generators		
ada-deterministic	0.199	0.775
ada-deterministic-with-repetition-penalty	0.237	0.807
ada-creative	0.226	0.791
ada-creative-with-repetition-penalty	0.212	0.795
babbage-deterministic	0.239	0.796
babbage-deterministic-with-repetition-penalty	0.238	0.807
babbage-creative	0.245	0.807
babbage-creative-with-repetition-penalty	0.200	0.795
curie-deterministic	0.255	0.800
curie-deterministic-with-repetition-penalty	0.251	0.816
curie-creative	0.245	0.804
curie-creative-with-repetition-penalty	0.215	0.805
davinci-deterministic	0.258	0.801
davinci-deterministic-with-repetition-penalty	0.252	0.813
davinci-creative	0.256	0.811
davinci-creative-with-repetition-penalty	0.228	0.805
text-ada-001-deterministic	0.200	0.801
text-ada-001-deterministic-with-repetition-penalty	0.191	0.802
text-ada-001-creative	0.181	0.796
text-ada-001-creative-with-repetition-penalty	0.199	0.797
text-babbage-001-deterministic	0.194	0.803

text-babbage-001-deterministic-with-repetition-penalty	0.195	0.798
text-babbage-001-creative	0.202	0.801
text-babbage-001-creative-with-repetition-penalty	0.211	0.802
text-curie-001-deterministic	0.206	0.806
text-curie-001-deterministic-with-repetition-penalty	0.201	0.806
text-curie-001-creative	0.197	0.805
text-curie-001-creative-with-repetition-penalty	0.193	0.801
text-davinci-002-deterministic	0.261	0.827
text-davinci-002-deterministic-with-repetition-penalty	0.277	0.828
text-davinci-002-creative	0.260	0.823
text-davinci-002-creative-with-repetition-penalty	0.251	0.818
text-davinci-003-deterministic	0.281	0.828
text-davinci-003-deterministic-with-repetition-penalty	0.263	0.820
text-davinci-003-creative	0.281	0.831
text-davinci-003-creative-with-repetition-penalty	0.255	0.816
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.293	0.829
gpt-3.5-turbo-creative	0.288	0.829
gpt-3.5-turbo-0301-deterministic	0.292	0.829
gpt-3.5-turbo-0301-creative	0.290	0.828
Hugging Face text generators		
bloom-560m-deterministic	0.215	0.796
bloom-560m-deterministic-with-repetition-penalty	0.143	0.796
bloom-560m-creative	0.146	0.770
bloom-560m-creative-with-repetition-penalty	0.103	0.761
bloom-3b-deterministic	0.233	0.806
bloom-3b-deterministic-with-repetition-penalty	0.170	0.805
bloom-3b-creative	0.139	0.768

bloom-3b-creative-with-repetition-penalty	0.114	0.763
bloom-7b1-deterministic	0.237	0.814
bloom-7b1-deterministic-with-repetition-penalty	0.184	0.808
bloom-7b1-creative	0.125	0.757
bloom-7b1-creative-with-repetition-penalty	0.100	0.755
Cerebras-GPT-111M-deterministic	0.146	0.762
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.151	0.783
Cerebras-GPT-111M -creative	0.120	0.751
Cerebras-GPT-111M -creative-with-repetition-penalty	0.116	0.749
Cerebras-GPT-256M-deterministic	0.164	0.770
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.171	0.795
Cerebras-GPT-256M -creative	0.124	0.755
Cerebras-GPT-256M -creative-with-repetition-penalty	0.100	0.751
Cerebras-GPT-590M-deterministic	0.186	0.793
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.178	0.797
Cerebras-GPT-590M-creative	0.130	0.756
Cerebras-GPT-590M-creative-with-repetition-penalty	0.111	0.754
Cerebras-GPT-1.3B-deterministic	0.242	0.813
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.164	0.797
Cerebras-GPT-1.3B-creative	0.136	0.755
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.107	0.748
Cerebras-GPT-2.7B-deterministic	0.223	0.808
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.172	0.802
Cerebras-GPT-2.7B-creative	0.146	0.751
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.119	0.747
Cerebras-GPT-6.7B-deterministic	0.251	0.822
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.173	0.798
Cerebras-GPT-6.7B-creative	0.151	0.750

Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.102	0.739
llama-7b-hf-deterministic	0.225	0.792
llama-7b-hf-deterministic-with-repetition-penalty	0.152	0.784
llama-7b-hf-creative	0.144	0.746
llama-7b-hf-creative-with-repetition-penalty	0.115	0.747
opt-125m-deterministic	0.219	0.787
opt-125m-deterministic-with-repetition-penalty	0.176	0.770
opt-125m-creative	0.123	0.754
opt-125m-creative-with-repetition-penalty	0.113	0.751
opt-350m-deterministic	0.196	0.787
opt-350m-deterministic-with-repetition-penalty	0.167	0.777
opt-350m-creative	0.129	0.751
opt-350m-creative-with-repetition-penalty	0.110	0.743
opt-1.3b-deterministic	0.206	0.786
opt-1.3b-deterministic-with-repetition-penalty	0.172	0.780
opt-1.3b-creative	0.131	0.749
opt-1.3b-creative-with-repetition-penalty	0.108	0.745
opt-2.7b-deterministic	0.238	0.806
opt-2.7b-deterministic-with-repetition-penalty	0.148	0.770
opt-2.7b-creative	0.132	0.750
opt-2.7b-creative-with-repetition-penalty	0.095	0.736
opt-6.7b-deterministic	0.220	0.795
opt-6.7b-deterministic-with-repetition-penalty	0.169	0.783
opt-6.7b-creative	0.133	0.751
opt-6.7b-creative-with-repetition-penalty	0.104	0.742

Πίνακας 21. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του Task 3 του διαγωνισμού μέσω του μέσου F1 score

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.480	0.879
Approach0		
run1	0.604	0.907
run4	0.603	0.906
run3	0.586	0.899
run2	0.527	0.890
run5	0.551	0.894
DPRL		
SBERT-SVMRank	0.452	0.869
BERT-SVMRank	0.451	0.868
SBERT-QQ-AMR	0.457	0.871
BERT-QQ-AMR	0.456	0.870
TU_DBS		
amps3_se1_hints	0.464	0.862
se3_len_pen_10	0.449	0.842

amps3_se1_len_pen_20_sample_hint	0.448	0.845
shortest	0.439	0.847
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.029	0.737
OpenAI text generators		
ada-deterministic	0.384	0.825
ada-deterministic-with-repetition-penalty	0.416	0.852
ada-creative	0.380	0.830
ada-creative-with-repetition-penalty	0.337	0.835
babbage-deterministic	0.449	0.845
babbage-deterministic-with-repetition-penalty	0.397	0.848
babbage-creative	0.407	0.851
babbage-creative-with-repetition-penalty	0.334	0.837
curie-deterministic	0.445	0.848
curie-deterministic-with-repetition-penalty	0.410	0.852
curie-creative	0.389	0.848
curie-creative-with-repetition-penalty	0.364	0.845
davinci-deterministic	0.457	0.850
davinci-deterministic-with-repetition-penalty	0.426	0.855
davinci-creative	0.402	0.855
davinci-creative-with-repetition-penalty	0.367	0.841
text-ada-001-deterministic	0.382	0.842
text-ada-001-deterministic-with-repetition-penalty	0.379	0.847
text-ada-001-creative	0.343	0.835
text-ada-001-creative-with-repetition-penalty	0.365	0.839
text-babbage-001-deterministic	0.404	0.850

text-babbage-001-deterministic-with-repetition-penalty	0.388	0.844
text-babbage-001-creative	0.379	0.846
text-babbage-001-creative-with-repetition-penalty	0.405	0.850
text-curie-001-deterministic	0.422	0.852
text-curie-001-deterministic-with-repetition-penalty	0.412	0.850
text-curie-001-creative	0.407	0.852
text-curie-001-creative-with-repetition-penalty	0.393	0.850
text-davinci-002-deterministic	0.519	0.879
text-davinci-002-deterministic-with-repetition-penalty	0.482	0.871
text-davinci-002-creative	0.491	0.876
text-davinci-002-creative-with-repetition-penalty	0.442	0.864
text-davinci-003-deterministic	0.480	0.872
text-davinci-003-deterministic-with-repetition-penalty	0.444	0.862
text-davinci-003-creative	0.469	0.872
text-davinci-003-creative-with-repetition-penalty	0.445	0.859
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.468	0.872
gpt-3.5-turbo-creative	0.455	0.872
gpt-3.5-turbo-0301-deterministic	0.462	0.872
gpt-3.5-turbo-0301-creative	0.471	0.873
Hugging Face text generators		
bloom-560m-deterministic	0.451	0.846
bloom-560m-deterministic-with-repetition-penalty	0.357	0.845
bloom-560m-creative	0.288	0.820
bloom-560m-creative-with-repetition-penalty	0.251	0.808
bloom-3b-deterministic	0.452	0.850
bloom-3b-deterministic-with-repetition-penalty	0.378	0.852
bloom-3b-creative	0.279	0.817

bloom-3b-creative-with-repetition-penalty	0.242	0.805
bloom-7b1-deterministic	0.482	0.860
bloom-7b1-deterministic-with-repetition-penalty	0.383	0.852
bloom-7b1-creative	0.270	0.805
bloom-7b1-creative-with-repetition-penalty	0.249	0.802
Cerebras-GPT-111M-deterministic	0.326	0.814
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.279	0.822
Cerebras-GPT-111M -creative	0.262	0.798
Cerebras-GPT-111M -creative-with-repetition-penalty	0.230	0.790
Cerebras-GPT-256M-deterministic	0.412	0.835
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.337	0.840
Cerebras-GPT-256M -creative	0.263	0.801
Cerebras-GPT-256M -creative-with-repetition-penalty	0.222	0.792
Cerebras-GPT-590M-deterministic	0.418	0.840
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.349	0.840
Cerebras-GPT-590M-creative	0.263	0.805
Cerebras-GPT-590M-creative-with-repetition-penalty	0.227	0.798
Cerebras-GPT-1.3B-deterministic	0.492	0.862
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.374	0.846
Cerebras-GPT-1.3B-creative	0.268	0.804
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.225	0.790
Cerebras-GPT-2.7B-deterministic	0.496	0.863
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.389	0.849
Cerebras-GPT-2.7B-creative	0.271	0.798
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.233	0.788
Cerebras-GPT-6.7B-deterministic	0.505	0.872
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.387	0.850
Cerebras-GPT-6.7B-creative	0.279	0.799

Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.206	0.779
llama-7b-hf-deterministic	0.427	0.836
llama-7b-hf-deterministic-with-repetition-penalty	0.290	0.821
llama-7b-hf-creative	0.256	0.792
llama-7b-hf-creative-with-repetition-penalty	0.210	0.783
opt-125m-deterministic	0.426	0.838
opt-125m-deterministic-with-repetition-penalty	0.283	0.807
opt-125m-creative	0.265	0.803
opt-125m-creative-with-repetition-penalty	0.228	0.790
opt-350m-deterministic	0.397	0.833
opt-350m-deterministic-with-repetition-penalty	0.278	0.815
opt-350m-creative	0.225	0.792
opt-350m-creative-with-repetition-penalty	0.191	0.777
opt-1.3b-deterministic	0.416	0.834
opt-1.3b-deterministic-with-repetition-penalty	0.285	0.819
opt-1.3b-creative	0.253	0.796
opt-1.3b-creative-with-repetition-penalty	0.202	0.782
opt-2.7b-deterministic	0.459	0.850
opt-2.7b-deterministic-with-repetition-penalty	0.253	0.803
opt-2.7b-creative	0.239	0.792
opt-2.7b-creative-with-repetition-penalty	0.182	0.770
opt-6.7b-deterministic	0.444	0.850
opt-6.7b-deterministic-with-repetition-penalty	0.286	0.821
opt-6.7b-creative	0.246	0.789
opt-6.7b-creative-with-repetition-penalty	0.190	0.776

Πίνακας 22. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του διαγωνισμού με φιλτράρισμα των "κοινών" λέξεων

Model	Lexical overlap	Contextual similarity
ARQMath		
ARQMath baseline (text-davinci-002 with 0.7 temperature)	0.317	0.851
Approach0		
run1	0.509	0.886
run4	0.516	0.886
run3	0.467	0.879
run2	0.427	0.868
run5	0.444	0.873
DPRL		
SBERT-SVMRank	0.338	0.848
BERT-SVMRank	0.337	0.848
SBERT-QQ-AMR	0.325	0.852
BERT-QQ-AMR	0.323	0.851
TU_DBS		
amps3_se1_hints	0.263	0.837
se3_len_pen_10	0.248	0.809

amps3_se1_len_pen_20_sample_hint	0.255	0.817
shortest	0.239	0.821
New runs		
Fake run		
"Flibberdoodle Quibblywop Blibberblab Snickerdoodle Wobbleglop"	0.025	0.742
OpenAI text generators		
ada-deterministic	0.276	0.820
ada-deterministic-with-repetition-penalty	0.308	0.831
ada-creative	0.289	0.813
ada-creative-with-repetition-penalty	0.287	0.815
babbage-deterministic	0.319	0.836
babbage-deterministic-with-repetition-penalty	0.308	0.826
babbage-creative	0.312	0.827
babbage-creative-with-repetition-penalty	0.282	0.814
curie-deterministic	0.326	0.838
curie-deterministic-with-repetition-penalty	0.318	0.831
curie-creative	0.308	0.828
curie-creative-with-repetition-penalty	0.289	0.823
davinci-deterministic	0.337	0.837
davinci-deterministic-with-repetition-penalty	0.326	0.833
davinci-creative	0.321	0.830
davinci-creative-with-repetition-penalty	0.297	0.821
text-ada-001-deterministic	0.270	0.830
text-ada-001-deterministic-with-repetition-penalty	0.263	0.832
text-ada-001-creative	0.256	0.826
text-ada-001-creative-with-repetition-penalty	0.274	0.828
text-babbage-001-deterministic	0.282	0.836

text-babbage-001-deterministic-with-repetition-penalty	0.283	0.833
text-babbage-001-creative	0.270	0.832
text-babbage-001-creative-with-repetition-penalty	0.285	0.836
text-curie-001-deterministic	0.302	0.841
text-curie-001-deterministic-with-repetition-penalty	0.296	0.839
text-curie-001-creative	0.285	0.837
text-curie-001-creative-with-repetition-penalty	0.291	0.838
text-davinci-002-deterministic	0.348	0.858
text-davinci-002-deterministic-with-repetition-penalty	0.342	0.847
text-davinci-002-creative	0.330	0.849
text-davinci-002-creative-with-repetition-penalty	0.327	0.841
text-davinci-003-deterministic	0.357	0.853
text-davinci-003-deterministic-with-repetition-penalty	0.330	0.839
text-davinci-003-creative	0.343	0.852
text-davinci-003-creative-with-repetition-penalty	0.332	0.837
OpenAI chatbots		
gpt-3.5-turbo-deterministic	0.357	0.849
gpt-3.5-turbo-creative	0.351	0.850
gpt-3.5-turbo-0301-deterministic	0.356	0.849
gpt-3.5-turbo-0301-creative	0.355	0.850
Hugging Face text generators		
bloom-560m-deterministic	0.315	0.837
bloom-560m-deterministic-with-repetition-penalty	0.206	0.818
bloom-560m-creative	0.203	0.789
bloom-560m-creative-with-repetition-penalty	0.153	0.780
bloom-3b-deterministic	0.322	0.837
bloom-3b-deterministic-with-repetition-penalty	0.228	0.824
bloom-3b-creative	0.194	0.784

bloom-3b-creative-with-repetition-penalty	0.166	0.779
bloom-7b1-deterministic	0.325	0.841
bloom-7b1-deterministic-with-repetition-penalty	0.235	0.825
bloom-7b1-creative	0.176	0.775
bloom-7b1-creative-with-repetition-penalty	0.150	0.774
Cerebras-GPT-111M-deterministic	0.211	0.820
Cerebras-GPT-111M -deterministic-with-repetition-penalty	0.220	0.812
Cerebras-GPT-111M -creative	0.183	0.769
Cerebras-GPT-111M -creative-with-repetition-penalty	0.187	0.770
Cerebras-GPT-256M-deterministic	0.268	0.830
Cerebras-GPT-256M -deterministic-with-repetition-penalty	0.241	0.823
Cerebras-GPT-256M -creative	0.187	0.770
Cerebras-GPT-256M -creative-with-repetition-penalty	0.168	0.768
Cerebras-GPT-590M-deterministic	0.274	0.833
Cerebras-GPT-590M-deterministic-with-repetition-penalty	0.255	0.825
Cerebras-GPT-590M-creative	0.199	0.775
Cerebras-GPT-590M-creative-with-repetition-penalty	0.177	0.773
Cerebras-GPT-1.3B-deterministic	0.320	0.846
Cerebras-GPT-1.3B-deterministic-with-repetition-penalty	0.243	0.826
Cerebras-GPT-1.3B-creative	0.199	0.773
Cerebras-GPT-1.3B-creative-with-repetition-penalty	0.171	0.767
Cerebras-GPT-2.7B-deterministic	0.319	0.845
Cerebras-GPT-2.7B-deterministic-with-repetition-penalty	0.254	0.832
Cerebras-GPT-2.7B-creative	0.213	0.770
Cerebras-GPT-2.7B-creative-with-repetition-penalty	0.186	0.767
Cerebras-GPT-6.7B-deterministic	0.337	0.854
Cerebras-GPT-6.7B-deterministic-with-repetition-penalty	0.258	0.829
Cerebras-GPT-6.7B-creative	0.209	0.766

Cerebras-GPT-6.7B-creative-with-repetition-penalty	0.175	0.760
llama-7b-hf-deterministic	0.298	0.827
llama-7b-hf-deterministic-with-repetition-penalty	0.228	0.804
llama-7b-hf-creative	0.201	0.764
llama-7b-hf-creative-with-repetition-penalty	0.177	0.764
opt-125m-deterministic	0.303	0.818
opt-125m-deterministic-with-repetition-penalty	0.265	0.792
opt-125m-creative	0.178	0.774
opt-125m-creative-with-repetition-penalty	0.181	0.768
opt-350m-deterministic	0.278	0.824
opt-350m-deterministic-with-repetition-penalty	0.247	0.800
opt-350m-creative	0.200	0.773
opt-350m-creative-with-repetition-penalty	0.177	0.763
opt-1.3b-deterministic	0.289	0.826
opt-1.3b-deterministic-with-repetition-penalty	0.268	0.802
opt-1.3b-creative	0.199	0.771
opt-1.3b-creative-with-repetition-penalty	0.179	0.767
opt-2.7b-deterministic	0.319	0.841
opt-2.7b-deterministic-with-repetition-penalty	0.244	0.791
opt-2.7b-creative	0.205	0.773
opt-2.7b-creative-with-repetition-penalty	0.166	0.758
opt-6.7b-deterministic	0.315	0.842
opt-6.7b-deterministic-with-repetition-penalty	0.266	0.806
opt-6.7b-creative	0.203	0.768
opt-6.7b-creative-with-repetition-penalty	0.178	0.763

Πίνακας 23. Σύγκριση όλων των αποτελεσμάτων με τις απαντήσεις του διαγωνισμού με φιλτράρισμα των επαναλαμβανόμενων φράσεων