



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**  
**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ**  
**ΕΠΙΚΟΙΝΩΝΙΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Πτυχιακή Εργασία**

|                           |  |
|---------------------------|--|
| Τίτλος Πτυχιακής Εργασίας | Αυτόματη Ταξινόμηση Μουσικολογικού Είδους Με Βάση Τους Στίχους |
| Thesis Title              | Lyrics-Based Music Genre Classification                        |
| Όνοματεπώνυμο Φοιτητή     | ΕΥΑΓΓΕΛΟΣ ΒΡΥΣΕΛΛΑΣ  |
| Πατρώνυμο                 | ΗΛΙΑΣ  |
| Αριθμός Μητρώου           | Π/11021  |
| Επιβλέπων                 | ΔΙΟΝΥΣΙΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ, ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΗΣ                    |

## Copyright ©

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς. Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

## **ΔΙΜΕΛΗ ΕΠΙΤΡΟΠΗ ΕΞΕΤΑΣΗΣ**

2-Member Examination Committee

**ΔΙΟΝΥΣΙΟΣ ΣΩΤΗΡΟΠΟΥΛΟΣ**

DIONISIS SOTIROPOYLOS

**ΓΕΩΡΓΙΟΣ ΤΣΙΧΡΙΝΤΖΗΣ**

GEORGE TSIHRINTZIS

## Ευχαριστίες

---

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Διονύσιο Σωτηρόπουλο για την εμπιστοσύνη που μου έδειξε δίνοντας μου την ευκαιρία να εκπονήσω αυτήν την πτυχιακή εργασία και την υποστήριξή του κατά τη διάρκεια της εκπόνησης της εργασίας. Επίσης, θέλω να ευχαριστήσω τους γονείς μου Ηλία και Σταματία για την στήριξη σε όλο αυτό τον αγώνα για την ολοκλήρωση της πτυχιακής μου και γενικά των σπουδών μου. Ιδιαίτερο ευχαριστώ στην κοπέλα μου την Αθηνά που χωρίς αυτή δεν θα κατάφερα τίποτα από όλα αυτά.

## Περίληψη

---

Η παρούσα πτυχιακή εργασία διερευνά την ανάπτυξη και βελτιστοποίηση ενός μοντέλου βασισμένου στο BERT για την αυτόματη ταξινόμηση μουσικών ειδών. Μέσω εκτεταμένης προεπεξεργασίας δεδομένων και προσαρμογής του μοντέλου, αντιμετωπίσαμε τις προκλήσεις που θέτουν τα μη ισορροπημένα σύνολα δεδομένων και οι κοινές λέξεις μεταξύ των ειδών. Οι αρχικές δοκιμές με πέντε είδη απέδωσαν μη ικανοποιητικά αποτελέσματα, οδηγώντας σε προσαρμογές στην επιλογή των ειδών για βελτίωση της απόδοσης του μοντέλου. Οι επακόλουθες πειραματικές διαδικασίες με λιγότερα είδη πέτυχαν σημαντικές βελτιώσεις, με το τελικό μοντέλο να ταξινομεί τρία είδη με εντυπωσιακή ακρίβεια 93%. Η έρευνα αυτή όχι μόνο αναδεικνύει την αποτελεσματικότητα των μοντέλων μετασχηματιστών σε εργασίες ταξινόμησης κειμένου, αλλά παρέχει επίσης χρήσιμες πληροφορίες για τη βελτιστοποίηση των ροών εργασίας μηχανικής μάθησης για πρακτικές εφαρμογές στην ταξινόμηση μουσικών ειδών.

### Λέξεις Κλειδιά:

- Ταξινόμηση Μουσικών Ειδών
- Επεξεργασία Φυσικής Γλώσσας (NLP)
- BERT
- Ταξινόμηση Κειμένων
- Μηχανική Μάθηση
- Μοντέλα Μετασχηματιστών
- Προεπεξεργασία Δεδομένων
- Βελτιστοποίηση Μοντέλου
- Νευρωνικά Δίκτυα

## Abstract

---

This thesis investigates the development and optimization of a BERT-based model for the automatic classification of music genres. Through extensive data preprocessing and model fine-tuning, we addressed the challenges posed by imbalanced datasets and common word overlaps across genres. Initial trials with five genres yielded suboptimal results, leading to adjustments in genre selection to improve model performance. Subsequent experiments with fewer genres achieved notable improvements, with the final model accurately classifying three genres with an impressive 93% accuracy. This research not only highlights the efficacy of transformer models in text classification tasks but also provides insights into optimizing machine learning workflows for practical applications in music genre classification.

### *Key Words:*

- *Music Genre Classification*
- *Natural Language Processing (NLP)*
- *BERT*
- *Text Classification*
- *Machine Learning*
- *Transformer Models*
- *Data Preprocessing*
- *Model Optimization*
- *Neural Networks*

## Αφιέρωσεις

---

Αφιερωμένο στην ανιψιά μου την Ηλιάνα!

## Πίνακας Περιεχομένων

---

|  |    |
|--|----|
| Ευχαριστίες.....   | 3  |
| Περίληψη (Abstract) .....  | 4  |
| Αφιερώσεις.....  | 7  |
| Πίνακας Περιεχομένων .....                                       | 8  |
| Κατάλογος Εικόνων.....   | 9  |
| Κατάλογος Διαγραμμάτων.....                                      | 10 |
| Εισαγωγή .....   | 11 |
| 1. Σχετικά Έργα.....   | 13 |
| 2. Αρχιτεκτονική Bert.....                                       | 14 |
| 2.1 Μετάβαση στο Bert.....                                       | 14 |
| 2.2 Αρχιτεκτονική.....   | 15 |
| 2.2.1 Encoding.....  | 16 |
| 2.3 Προετοιμασία του περιβάλλοντος εισαγωγής προεκπαίδευσης..... | 18 |
| 2.3.1 Πρόβλεψη επόμενης πρότασης.....                            | 20 |
| 2.3.2 Προεκπαίδευση και τελειοποίηση ενός μοντέλου BERT.....     | 22 |
| 3. Τεχνολογία.....   | 23 |
| 3.1 Python.....  | 23 |
| 3.2 Deep learning Frameworks.....                                | 23 |
| 3.2.1 Tensorflow.....  | 23 |
| 3.2.2 Keras.....   | 23 |
| 3.2.3 PyTorch.....   | 24 |
| 3.2.4 Βασικές Διαφορές .....                                     | 24 |
| 3.3 Άλλες βιβλιοθήκες Python .....                               | 25 |
| 3.3.1 Numpy .....  | 25 |
| 3.3.2 Pandas .....   | 25 |
| 3.3.3 Scikit-learn .....   | 25 |
| 3.3.4 Matplotlib.....  | 25 |
| 4. Ανάλυση και Προεργασία Δεδομένων.....                         | 26 |
| 4.1 Βάση Δεδομένων.....  | 26 |
| 4.1.1 Έυρεση.....  | 26 |
| 4.1.2 Περιγραφή Βάσης.....                                       | 26 |
| 4.2 Ανάλυση Βάσης.....   | 27 |
| 4.3 Προεπεξεργασία Δεδομένων.....                                | 29 |
| 5. Μοντέλο.....  | 31 |
| 5.1 Δημιουργία Μοντέλου.....                                     | 31 |
| 5.2 Αποτελέσματα.....  | 31 |
| 5.2.1 Μοντέλο 5 Ειδών.....                                       | 31 |
| 5.2.2 Μοντέλο 4 Ειδών.....                                       | 32 |
| 5.2.3 Μοντέλο 3 Ειδών.....                                       | 34 |
| 6. Συμπεράσματα.....   | 35 |



## Κατάλογος Εικόνων

---

|  |   |
|--|---|
| Εικόνα 1. encoding .....               | 3 |
| Εικόνα 2. Models size .....            | 4 |
| Εικόνα 3. Bert two step Framework..... | 5 |
| Εικόνα 4. Λεζάντα εικόνας.....         | 6 |

## Κατάλογος Διαγραμμάτων

---

|   |    |
|---|----|
| Διάγραμμα 1. Distribution of Genres .....   | 7  |
| Διάγραμμα 2. Distribution of Language ..... | 8  |
| Διάγραμμα 3. Distribution of Lyrics.....    | 10 |



## Εισαγωγή

---

Στον σύγχρονο κόσμο, η μουσική είναι απαραίτητη για την ύπαρξή μας, καθώς επηρεάζει τα συναισθήματά μας, μας διασκεδάζει και προωθεί τις διαπολιτισμικές σχέσεις. Επιπλέον, η κατηγοριοποίηση τραγουδιών ανά είδος έχει επίσης πρακτικές συνέπειες για τα συστήματα συστάσεων και την εξατομικευμένη παράδοση περιεχομένου. Αλλά υπάρχει ένα μεγάλο πρόβλημα όταν οι άνθρωποι προσπαθούν να κατανοήσουν τους συναισθηματικούς τόνους ενός τραγουδιού διαβάζοντας απλώς τις λέξεις. Όταν η μελωδία απουσιάζει και διαβάζει κανείς τους στίχους μόνος του, το πνεύμα και η διάθεση της μουσικής συχνά χάνονται και είναι δύσκολο να ερμηνεύσει τη συναισθηματική χροιά των στίχων. Οι άνθρωποι αδυνατούν να εκτιμήσουν και να κατανοήσουν πλήρως το βάθος της μουσικής έκφρασης ως αποτέλεσμα αυτού του περιορισμού. Αυτός είναι ο λόγος για τον οποίο θέλουμε να καταλάβουμε εάν είναι δυνατόν να προβλεφθούν βασικές πληροφορίες για ένα τραγούδι, συμπεριλαμβανομένου του είδους του, της κατά προσέγγιση ηλικίας του, ακόμη και της πιθανής επιτυχίας του όσον αφορά τις προβολές σελίδας.

Αντιμετωπίζοντας αυτό το πρόβλημα, θέλουμε να χρησιμοποιήσουμε τεχνικές επεξεργασίας φυσικής γλώσσας (NLP) για να αναλύσουμε και να ερμηνεύσουμε ολοκληρωμένα τους στίχους των τραγουδιών. Θέλουμε να αποκτήσουμε σημαντικές γνώσεις για τον κόσμο της μουσικής μέσω της προτεινόμενης λύσης, αυξάνοντας την απόλαυση και την προσβασιμότητα όλων.

Επιπλέον, υπάρχουν εκατομμύρια τραγούδια στις μουσικές πλατφόρμες που απαιτούν οργάνωση συνήθως με βάση τα είδη τους. Επομένως, γίνεται δύσκολο για τους διαχειριστές να οργανώσουν τραγούδια σε τέτοιες μουσικές πλατφόρμες. Οι τεχνικές μηχανικής μάθησης έχουν χρησιμοποιηθεί για την ταξινόμηση ειδών μουσικής χρησιμοποιώντας διάφορους τύπους δεδομένων που εξάγονται από τραγούδια. Ενώ ορισμένες μελέτες χρησιμοποίησαν ηχητικά σήματα [7, 30], άλλες χρησιμοποίησαν στίχους τραγουδιών για ταξινόμηση ειδών μουσικής θεωρώντας την εργασία ως εργασία ταξινόμησης κειμένου. Δεδομένου ότι οι στίχοι περιέχουν εξαιρετικές πληροφορίες για το είδος της μουσικής, προσπαθούμε να αντιμετωπίσουμε το πρόβλημα ταξινόμησης του είδους μουσικής με τους στίχους σε αυτή τη μελέτη. Η ταξινόμηση κειμένων είναι ένα πολύ γνωστό πρόβλημα στην επεξεργασία φυσικής γλώσσας (NLP). Ο στόχος της εργασίας είναι να αντιστοιχίσετε προκαθορισμένες κατηγορίες σε ένα δεδομένο κείμενο. Προηγούμενες μελέτες χρησιμοποιούσαν νευρωνικά μοντέλα για την κατηγοριοποίηση κειμένων όπως το Νευρωνικό Δίκτυο συνέλιξης (CNN) [10, 27], τα επαναλαμβανόμενα νευρωνικά δίκτυα (RNN) [15, 34] και ο μηχανισμός προσοχής [13, 32]. Πρόσφατα, προεκπαιδευμένα γλωσσικά μοντέλα έχουν χρησιμοποιηθεί για την εκμάθηση γλωσσικών αναπαραστάσεων χρησιμοποιώντας μεγάλο αριθμό δεδομένων χωρίς ετικέτα. Υπάρχουν δύο πρόσφατα μοντέλα αναπαράστασης γλώσσας: BERT [5] και DistilBERT [24]. Το BERT βασίζεται σε έναν πολυστρωματικό αμφίδρομο μετασχηματιστή [31] και το DistilBERT είναι μια «αποσταγμένη» έκδοση του BERT, η οποία είναι συγκριτικά μικρότερη και ταχύτερη από το BERT.

Σε αυτή τη μελέτη, διερευνούμε τη χρήση των BERT στην ταξινόμηση μουσικών ειδών . Επιπλέον, συγκρίνουμε τρεις τύπους μοντέλων ως προς την πολυπλοκότητά τους αναλύοντας τους χρόνους υπολογισμού τους αναδιπλώσεις: 1- ακρίβεια και 2- υπολογιστικοί χρόνοι Εάν σκοπεύουμε να χρησιμοποιήσουμε τα μοντέλα σε μια εφαρμογή σε πραγματικό χρόνο, οι χρόνοι απόκρισης και εκπαίδευσης θα ήταν κρίσιμοι Λαμβάνοντας υπόψη και τις δύο πτυχές (ακρίβεια και υπολογιστικό χρόνο) σε τέτοια μοντέλα, τα αποτελέσματα δείχνουν ότι το BERT θα ήταν μια λογική λύση σε μια εφαρμογή σε πραγματικό χρόνο και σε πραγματικό κόσμο. Το υπόλοιπο αυτής της εργασίας είναι οργανωμένη ως εξής: 1 Κεφάλαιο Εισαγωγή , 2 Κεφάλαιο Περιγραφή του Bert , 3 Κεφάλαιο Ανάλυση της Βάσης Δεδομένων και προεπεξεργασία και 4 Κεφάλαιο Ανάλυση των αποτελεσμάτων.

## 1. Σχετικά Έργα

---

Σε αυτή τη μελέτη, προτείνουμε ένα μοντέλο ταξινόμησης μουσικών ειδών χρησιμοποιώντας στίχους. Στη βιβλιογραφία, διάφορες μελέτες ταξινόμησης κειμένων χρησιμοποιούν βαθιά επαναλαμβανόμενα νευρωνικά δίκτυα. Σε αυτήν την ενότητα, θα καλύψουμε προηγούμενες μελέτες σχετικά με την ταξινόμηση των ειδών μουσικής χρησιμοποιώντας αλγόριθμους μηχανικής μάθησης και μοντέλα αναπαράστασης γλώσσας που χρησιμοποιούνται για την ταξινόμηση κειμένων. Ως αλγόριθμοι μηχανικής μάθησης, οι Howard et al. [6] χρησιμοποίησαν τον αλγόριθμο Naive Bayes για το είδος μουσικής χρησιμοποιώντας ένα πολύγλωσσο σύνολο δεδομένων. Οι Ying et al. [33] χρησιμοποιούν διάφορους αλγόριθμους ταξινόμησης όπως Naive Bayes, k-NN (k κοντινότερος γείτονας), SVM (υποστηρικτικές διανυσματικές μηχανές) για την κατηγοριοποίηση των μουσικών ειδών και διαθέσεων σε ένα τραγούδι. Οι Oramas et al. [21, 22] προτείνουν ένα μοντέλο συνελκτικού νευρωνικού δικτύου (CNN) που χρησιμοποιεί χαρακτηριστικά που εξάγονται από ήχο, εικόνες και κείμενο κάθε τραγουδιού. Οι Lima et al. [2] προτείνουν ένα μοντέλο BiLSTM για να ταξινομήσει ένα σύνολο στίχων βραζιλιάνικων τραγουδιών. Αντίστοιχα, ο Τσαπτσινός χρησιμοποιεί ένα ιεραρχικό δίκτυο προσοχής (HAN) [29] για την ταξινόμηση των ειδών μουσικής που βασίζεται επίσης σε στίχους. Άλλες μελέτες διερευνούν τη χρήση ήχου και στίχων [16, 18, 20], ή ομοιοκαταληξίες και στυλ σε στίχους [17] για ταξινόμηση ειδών μουσικής. Η ακραία ταξινόμηση κειμένου πολλών ετικετών είναι μια πρόκληση για την επισήμανση ενός κειμένου με βάση ένα εξαιρετικά μεγάλο σύνολο ετικετών, το οποίο είναι γενικά πάνω από χιλιάδες. Οι Chang et al. [3] χρησιμοποίησαν το BERT για αυτόν τον σκοπό. Οι Munikar et al. [19] και Li et al. [12] χρησιμοποίησαν το BERT για την ταξινόμηση συναισθημάτων. Sun et al. [28] διεξάγουν πειράματα βελτιστοποιώντας το BERT χρησιμοποιώντας διαφορετικές μεθόδους και εκτελούν πειράματα στο μοντέλο βάσης BERT χωρίς περίβλημα για ταξινόμηση αγγλικού κειμένου και στο κινεζικό μοντέλο βάσης BERT για ταξινόμηση κινεζικών κειμένων. Η απόσταση εφαρμόζεται επίσης στην ταξινόμηση κειμένου. Οι Chia et al. [4] εφαρμόσαν απόσταση χρησιμοποιώντας Open AI GPT [23] ως δάσκαλο και ένα δίκτυο BiLSTM, ένα ρηχό δίκτυο CNN, μια νέα δομή CNN χρησιμοποιούνται ως μαθητές. Ο Adhikari [1] προτείνει μια απόσταση BERT- μεγάλου έως μικρού LSTM, χρησιμοποιώντας έτσι 30 φορές λιγότερο αριθμό παραμέτρων.

## 2. Αρχιτεκτονική BERT

---

### 2.1 Μετάβαση στο BERT

Σκεφτείτε το αρχικό Transformer ως μοντέλο κατασκευασμένο με τούβλα LEGO®. Το σετ κατασκευής περιέχει τούβλα όπως κωδικοποιητές, αποκωδικοποιητές, στρώματα ενσωμάτωσης, μεθόδους κωδικοποίησης θέσης, στρώματα προσοχής πολλαπλών κεφαλών, στρώματα προσοχής με μάσκα πολλαπλών κεφαλών, κανονικοποίηση μετά το επίπεδο, υποστρώματα τροφοδοσίας προς τα εμπρός και στρώματα γραμμικής εξόδου. Τα τούβλα έρχονται σε διάφορα μεγέθη και μορφές. Μπορείτε να ξοδέψετε ώρες κατασκευάζοντας κάθε είδους μοντέλα χρησιμοποιώντας το ίδιο κιτ κατασκευής! Ορισμένες κατασκευές θα απαιτήσουν μόνο μερικά από τα τούβλα. Άλλες κατασκευές θα προσθέσουν ένα νέο κομμάτι, όπως ακριβώς όταν λαμβάνουμε πρόσθετα τούβλα για ένα μοντέλο που έχει κατασκευαστεί με εξαρτήματα LEGO®. Η BERT πρόσθεσε ένα νέο κομμάτι στο κιτ κατασκευής Transformer: ένα αμφίδρομο υποστρώμα προσοχής πολλαπλών κεφαλών. Όταν εμείς οι άνθρωποι αντιμετωπίζουμε προβλήματα στην κατανόηση μιας πρότασης, δεν κοιτάμε μόνο τις προηγούμενες λέξεις. Ο BERT, όπως και εμείς, κοιτάζει όλες τις λέξεις στην ίδια πρόταση ταυτόχρονα. Σε αυτό το κεφάλαιο, θα διερευνήσουμε πρώτα την αρχιτεκτονική των αναπαραστάσεων αμφίδρομου κωδικοποιητή από μετασχηματιστές (BERT). Το BERT χρησιμοποιεί μόνο τα μπλοκ των κωδικοποιητών του Transformer με νέο τρόπο και δεν χρησιμοποιεί τη στοίβα αποκωδικοποιητών. Στη συνέχεια, θα βελτιστοποιήσουμε ένα προεκπαιδευμένο μοντέλο BERT. Το μοντέλο BERT που θα βελτιστοποιήσουμε εκπαιδεύτηκε από τρίτο μέρος και ανέβηκε στο Hugging Face. Οι μετασχηματιστές μπορούν να είναι προεκπαιδευμένοι. Στη συνέχεια, ένα προεκπαιδευμένο BERT, για παράδειγμα, μπορεί να ρυθμιστεί με ακρίβεια σε πολλές εργασίες NLP. Θα ζήσουμε αυτή τη συναρπαστική εμπειρία χρήσης του μετασχηματιστή κατάντη χρησιμοποιώντας μονάδες Hugging Face.

## 2.2 Αρχιτεκτονική

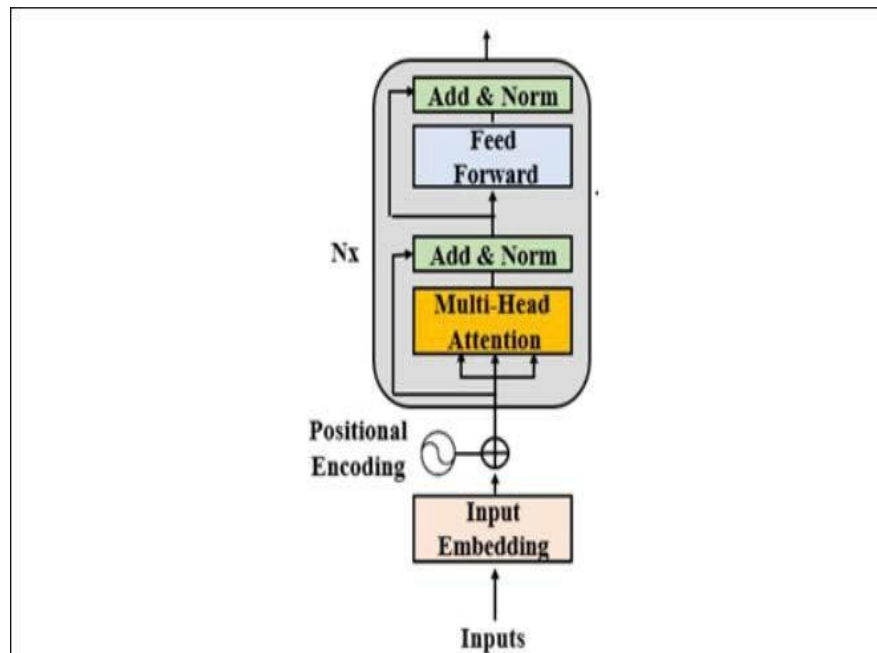
Ο BERT ή Bidirectional Encoder Representations from Transformers είναι ένα μοντέλο φυσικής γλώσσας που χρησιμοποιεί τους αμφίδρομους μετασχηματιστές για να κατανοήσει τη σημασία των λέξεων στο πλαίσιο τους εκπαιδεύεται με την προσέγγιση του masked language modeling όπου κάποιες λέξεις καλύπτονται τυχαία και το μοντέλο προσπαθεί να τις προβλέψει λαμβάνοντας υπόψη και τα δύο γειτονικά περιβάλλοντα χρησιμοποιείται για πολλές εφαρμογές όπως η ανάλυση συναισθήματος η απάντηση σε ερωτήσεις και η ταξινόμηση κειμένων

Το BERT διαθέτει δύο εκδόσεις BERT Base και BERT Large με διαφορετικό αριθμό παραμέτρων και επιπέδων μετασχηματιστών χρησιμοποιεί την αυτοπροσοχή για να αναλύσει τις σχέσεις μεταξύ των λέξεων το εκπαιδευτικό του σύνολο περιλαμβάνει βιβλία και άρθρα της Wikipedia βελτιώνει σημαντικά την απόδοση σε διάφορες γλωσσικές εργασίες ενσωματώνει προκατασκευασμένα μοντέλα που μπορούν να προσαρμοστούν σε συγκεκριμένα καθήκοντα μέσω fine-tuning μειώνει την ανάγκη για μεγάλες ποσότητες επισημασμένων δεδομένων κατά την εκπαίδευση και προσφέρει υψηλή ευελιξία και ακρίβεια στη γλωσσική κατανόηση



### 2.2.1 Encoding

Το πρώτο δομικό στοιχείο που θα πάρουμε από το αρχικό μοντέλο Transformer είναι ένα επίπεδο κωδικοποιητή.



Το μοντέλο BERT δεν χρησιμοποιεί στρώματα αποκωδικοποιητή. Ένα μοντέλο BERT έχει στοίβα κωδικοποιητή αλλά όχι στοίβες αποκωδικοποιητών. Τα καλυμμένα διακριτικά (απόκρυψη των διακριτικών για πρόβλεψη) βρίσκονται στα επίπεδα προσοχής του κωδικοποιητή, όπως θα δούμε όταν κάνουμε ζουμ σε ένα επίπεδο κωδικοποιητή BERT στις ακόλουθες ενότητες. Ο αρχικός μετασχηματιστής περιέχει μια στοίβα  $N=6$  στρώσεων. Ο αριθμός των διαστάσεων του αρχικού Transformer είναι :  $d_{model} = 512$

Ο αριθμός των κεφαλών προσοχής του αρχικού Transformer είναι  $A=8$ . Οι διαστάσεις μιας κεφαλής του αρχικού Transformer είναι:

$$d_k = \frac{d_{model}}{A} = \frac{512}{8} = 64$$

Τα επίπεδα κωδικοποιητή BERT είναι μεγαλύτερα από το αρχικό μοντέλο Transformer. Μπορούν να κατασκευαστούν δύο μοντέλα BERT με τα επίπεδα κωδικοποιητή:

$BERT_{model}$ , που περιέχει μια στοίβα  $N=12$  στρωμάτων κωδικοποιητή.  $d_{model} = 768$  και μπορεί επίσης να εκφράζεται ως  $H=768$ , όπως στο έγγραφο BERT. Ένα υπο-στρώμα προσοχής πολλαπλών κεφαλών περιέχει  $A=12$  κεφαλές. Οι διαστάσεις κάθε κεφαλής  $Z_a$  παραμένουν 64 όπως στο αρχικό μοντέλο Transformer:

$$d_k = \frac{d_{model}}{A} = \frac{768}{12} = 64$$

Η έξοδος κάθε υποστρώματος προσοχής πολλαπλών κεφαλών πριν από τη συνένωση θα είναι η έξοδος των 12 κεφαλών:

$$\text{output multi-head attention} = \{z_0, z_1, z_2, \dots, z_{11}\}$$

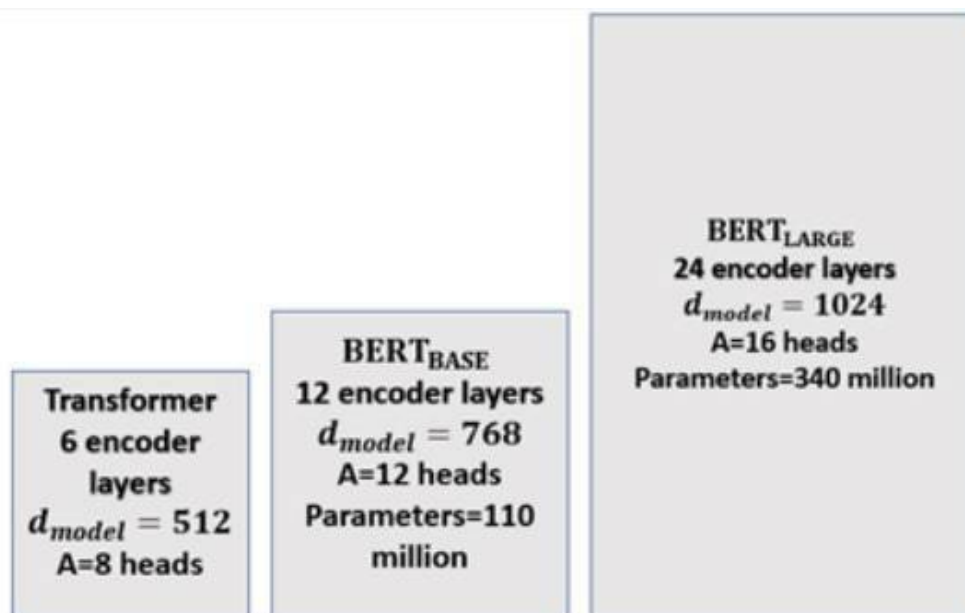
$$d_k = \frac{d_{model}}{A} = \frac{1064}{16} = 64$$

$Bert_{large}$  που περιέχει μια στοίβα  $N=24$  στρωμάτων κωδικοποιητή.  $d_{model} = 1064$  Οι διαστάσεις κάθε κεφαλής  $Z_a$  παραμένουν επίσης 64 όπως στο αρχικό μοντέλο Transformer:

Η έξοδος κάθε υποστρώματος προσοχής πολλαπλών κεφαλών πριν από τη συνένωση θα είναι η έξοδος των 16 κεφαλών:

$$\text{output multi-head attention} = \{z_0, z_1, z_2, \dots, z_{14}, z_{15}, z_{16}\}$$

Τα μεγέθη των μοντέλων μπορούν να συνοψιστούν ως εξής:



Το μέγεθος και οι διαστάσεις παίζουν ουσιαστικό ρόλο στην προπόνηση τύπου BERT. Τα μοντέλα BERT είναι σαν τους ανθρώπους. Τα μοντέλα BERT παράγουν καλύτερα αποτελέσματα με περισσότερη μνήμη εργασίας (διαστάσεις) και περισσότερη γνώση (δεδομένα). Τα μεγάλα μοντέλα μετασηματιστών που μαθαίνουν μεγάλους όγκους δεδομένων θα προεκπαιδευτούν καλύτερα για εργασίες μεταγενέστερου NLP. Ας πάμε τώρα στο πρώτο υποεπίπεδο και ας δούμε τις θεμελιώδεις πτυχές της ενσωμάτωσης εισόδου και της κωδικοποίησης θέσης σε ένα μοντέλο BERT.

### 2.3 Προετοιμασία του περιβάλλοντος εισαγωγής προεκπαίδευσης

Το μοντέλο BERT δεν έχει στοίβα αποκωδικοποιητών επιπέδων. Ως εκ τούτου, δεν έχει καλυμμένο υποστρώμα προσοχής πολλαπλών κεφαλών. Ο BERT προχωρά περαιτέρω και δηλώνει ότι ένα καλυμμένο στρώμα προσοχής πολλαπλών κεφαλών που καλύπτει την υπόλοιπη ακολουθία εμποδίζει τη διαδικασία προσοχής. Ένα καλυμμένο στρώμα προσοχής πολλαπλών κεφαλών καλύπτει όλα τα διακριτικά που βρίσκονται πέρα από την παρούσα θέση. Για παράδειγμα, πάρτε την ακόλουθη πρόταση:

Η γάτα κάθισε πάνω του γιατί ήταν ωραίο χαλί.

Εάν έχουμε μόλις φτάσει στη λέξη "it", η είσοδος του κωδικοποιητή θα μπορούσε να είναι: Η γάτα κάθισε πάνω του <μασκοφόρος ακολουθία>

Το κίνητρο αυτής της προσέγγισης είναι να εμποδίσει το μοντέλο να δει το αποτέλεσμα που υποτίθεται ότι θα προβλέψει. Αυτή η προσέγγιση από αριστερά προς τα δεξιά παράγει σχετικά καλά αποτελέσματα.

Ωστόσο, το μοντέλο δεν μπορεί να μάθει πολλά με αυτόν τον τρόπο. Για να ξέρουμε σε τι αναφέρεται το «αυτό», πρέπει να δούμε ολόκληρη την πρόταση για να φτάσουμε στη λέξη «χαλί» και να καταλάβουμε ότι «αυτό» ήταν το χαλί.

Οι συντάκτες του BERT βρήκαν μια ιδέα. Γιατί να μην εκπαιδεύσετε το μοντέλο για να κάνει προβλέψεις χρησιμοποιώντας μια διαφορετική προσέγγιση;

Οι συγγραφείς του Bert κατέληξαν στην αμφίδρομη προσοχή, αφήνοντας το κεφάλι της προσοχής να παρακολουθεί όλες τις λέξεις τόσο από αριστερά προς τα δεξιά όσο και από δεξιά προς τα αριστερά. Με άλλα λόγια, η μάσκα αυτοπροσοχής ενός κωδικοποιητή θα μπορούσε να κάνει τη δουλειά χωρίς να παρεμποδίζεται από τα καλυμμένα υποστρώματα προσοχής πολλαπλών κεφαλών του αποκωδικοποιητή.

Το μοντέλο εκπαιδεύτηκε με δύο εργασίες. Η πρώτη μέθοδος είναι το Masked Language Modeling (MLM). Η δεύτερη μέθοδος είναι η πρόβλεψη επόμενης πρότασης (NSP). Ας ξεκινήσουμε με τη μοντελοποίηση γλώσσας με μάσκα.

### **Μοντελοποίηση γλώσσας με μάσκα**

Η μοντελοποίηση καλυμμένης γλώσσας δεν απαιτεί εκπαίδευση ενός μοντέλου με μια ακολουθία ορατών λέξεων που ακολουθείται από μια καλυμμένη ακολουθία για πρόβλεψη. Ο BERT εισάγει την αμφίδρομη ανάλυση μιας πρότασης με μια τυχαία μάσκα σε μια λέξη της πρότασης. Είναι σημαντικό να σημειωθεί ότι ο Bert εφαρμόζει το WordPiece, μια μέθοδο τμηματοποίησης υπο-λέξεων, tokenization στις εισόδους. Χρησιμοποιεί επίσης κωδικοποίηση θέσης, όχι την προσέγγιση ημιτόνου-συνημιτονοειδούς.

Μια πιθανή ακολουθία εισόδου θα μπορούσε να είναι:

«Η γάτα κάθισε πάνω της γιατί ήταν ωραίο χαλί».

Ο αποκωδικοποιητής θα κάλυπτε την ακολουθία προσοχής αφού το μοντέλο φτάσει στη λέξη "it":

«Η γάτα κάθισε πάνω της <μασκοφόρος ακολουθία>».

Αλλά ο κωδικοποιητής BERT κρύβει ένα τυχαίο διακριτικό για να κάνει μια πρόβλεψη: "Η γάτα κάθισε πάνω της [ΜΑΣΚΑ] ήταν ένα ωραίο χαλί."

Το υπο-στρώμα πολλαπλής προσοχής μπορεί τώρα να δει ολόκληρη τη σειρά, να εκτελέσει τη διαδικασία αυτοπροσοχής και να προβλέψει το μασκοφόρο διακριτικό.

Τα διακριτικά εισόδου καλύφθηκαν με έναν δύσκολο τρόπο για να εξαναγκάσουν το μοντέλο να εκπαιδευτεί περισσότερο, αλλά να παράγουν καλύτερα αποτελέσματα με τρεις μεθόδους:

Εκπλήξτε το μοντέλο μην καλύπτοντας ούτε ένα διακριτικό στο 10% του συνόλου δεδομένων, για παράδειγμα:

«Η γάτα κάθισε πάνω του [γιατί] ήταν ωραίο χαλί».

Εκπλήξτε το μοντέλο αντικαθιστώντας το διακριτικό με ένα τυχαίο διακριτικό στο 10% του συνόλου δεδομένων, για παράδειγμα:

«Η γάτα καθόταν πάνω του [συχνά] ήταν ωραίο χαλί».

Αντικατάσταση ενός διακριτικού με ένα διακριτικό [MASK] στο 80% του συνόλου δεδομένων. για παράδειγμα:

«Η γάτα κάθισε πάνω της [ΜΑΣΚΑ] ήταν ένα ωραίο χαλί».

Η τολμηρή προσέγγιση των συγγραφέων αποφεύγει την υπερβολική προσαρμογή και αναγκάζει το μοντέλο να εκπαιδεύεται αποτελεσματικά. Ο BERT εκπαιδεύτηκε επίσης να κάνει πρόβλεψη επόμενης πρότασης.

### 2.3.1 Πρόβλεψη επόμενης πρότασης

Η δεύτερη μέθοδος που βρέθηκε για την εκπαίδευση του BERT είναι η Πρόβλεψη Επόμενης Προτάσεως (NSP). Η εισαγωγή περιέχει δύο προτάσεις. Είναι σημαντικό να σημειωθεί ότι το BERT εφαρμόζει το WordPiece, μια μέθοδο τμηματοποίησης υπο- λέξεων, tokenization στις εισόδους. Χρησιμοποιεί επίσης μαθημένη κωδικοποίηση θέσης, όχι την ημιτονοειδή-συνημιτονική προσέγγιση. Κεφάλαιο 2 [49]

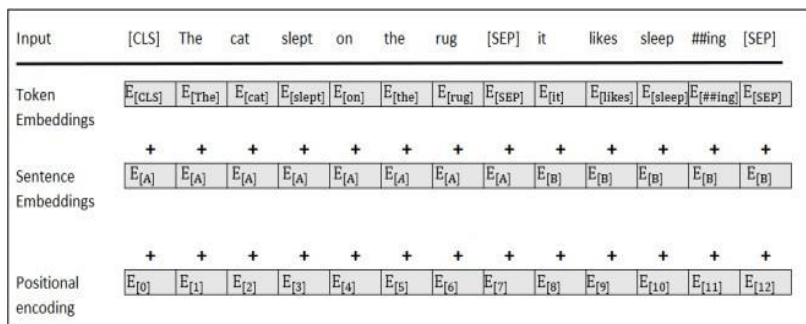
Προστέθηκαν δύο νέα διακριτικά:

- 1) Το [CLS] είναι ένα διακριτικό δυαδικής ταξινόμησης που προστίθεται στην αρχή της πρώτης ακολουθίας για να προβλέψει εάν η δεύτερη ακολουθία ακολουθεί την πρώτη ακολουθία. Ένα θετικό δείγμα είναι συνήθως ένα ζεύγος διαδοχικών προτάσεων που λαμβάνονται από ένα σύνολο δεδομένων. Ένα αρνητικό δείγμα δημιουργείται χρησιμοποιώντας ακολουθίες από διαφορετικά έγγραφα.
- 2) Το [SEP] είναι ένα διακριτικό διαχωρισμού που σηματοδοτεί το τέλος μιας ακολουθίας.

Για παράδειγμα, οι προτάσεις εισαγωγής που λαμβάνονται από ένα βιβλίο θα μπορούσαν να είναι: "Η γάτα κοιμήθηκε στο χαλί. Της αρέσει να κοιμάται όλη μέρα." Αυτές οι δύο προτάσεις θα γίνουν μια πλήρης ακολουθία εισαγωγής:

[CLS] η γάτα κοιμήθηκε στο χαλί [SEP] της αρέσει να κοιμάται ## όλη μέρα[SEP]

Αυτή η προσέγγιση απαιτεί πρόσθετες πληροφορίες κωδικοποίησης για τη διάκριση της ακολουθίας A από την ακολουθία B. Εάν συνδυάσουμε ολόκληρη τη διαδικασία ενσωμάτωσης, λαμβάνουμε:



Οι ενσωματώσεις εισόδου λαμβάνονται αθροίζοντας τις ενσωματώσεις διακριτικών, τις ενσωματώσεις του τμήματος (πρόταση, φράση, λέξη) και τις ενσωματώσεις κωδικοποίησης θέσης. Το υποεπίπεδο ενσωμάτωσης εισόδου και κωδικοποίησης θέσης ενός μοντέλου BERT μπορεί να συνοψιστεί ως εξής:

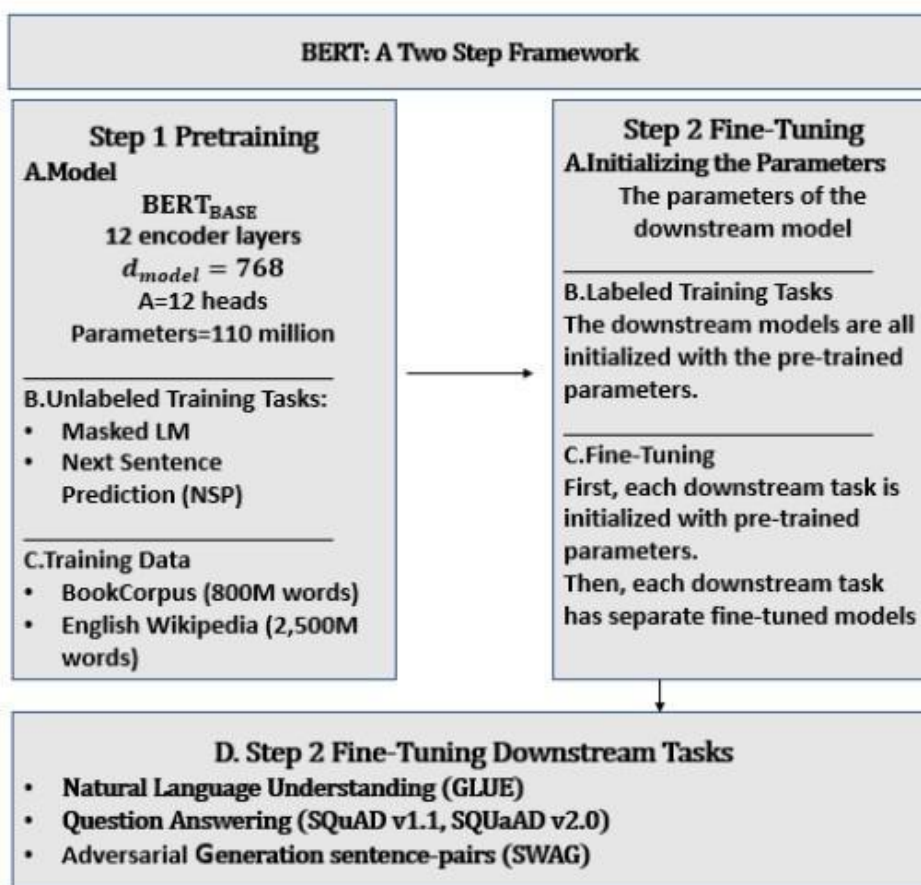
- A sequence of words is broken down into WordPiece tokens.
- Ένα διακριτικό [MASK] θα αντικαταστήσει τυχαία τα αρχικά διακριτικά λέξεων για εκπαίδευση μοντελοποίησης γλώσσας με μάσκα

- Ένα διακριτικό ταξινόμησης [CLS] εισάγεται στην αρχή μιας ακολουθίας για σκοπούς ταξινόμησης.
- Ένα διακριτικό [SEP] διαχωρίζει δύο προτάσεις (τμήματα, φράσεις) για εκπαίδευση NSP.
- Η ενσωμάτωση πρότασης προστίθεται στην ενσωμάτωση διακριτικού, έτσι ώστε η πρόταση A να έχει διαφορετική τιμή ενσωμάτωσης πρότασης από την πρόταση B.
- Η κωδικοποίηση θέσης μαθαίνεται. Η μέθοδος κωδικοποίησης θέσης ημιτονοειδούς-συνημιτονοειδούς του αρχικού μετασχηματιστή δεν εφαρμόζεται.
- Ορισμένα επιπλέον βασικά χαρακτηριστικά είναι:
- Το BERT χρησιμοποιεί αμφίδρομη προσοχή σε όλα τα υποεπίπεδα προσοχής πολλαπλών κεφαλών, ανοίγοντας τεράστιους ορίζοντες μάθησης και κατανόησης των σχέσεων μεταξύ των διακριτικών.
- Η BERT εισάγει σενάρια ενσωμάτωσης χωρίς επίβλεψη, μοντέλων προεκπαίδευσης με κείμενο χωρίς ετικέτα. Αυτό αναγκάζει το μοντέλο να σκέφτεται περισσότερο κατά τη διάρκεια της διαδικασίας εκμάθησης της προσοχής πολλαπλών κεφαλών. Αυτό καθιστά τον BERT σε θέση να μάθει πώς δημιουργούνται οι γλώσσες και να εφαρμόζει αυτή τη γνώση σε εργασίες κατάντη χωρίς να χρειάζεται να προπονείται κάθε φορά.
- Το BERT χρησιμοποιεί επίσης εποπτευόμενη μάθηση, καλύπτοντας όλες τις βάσεις στη διαδικασία προεκπαίδευσης.

Η BERT έχει βελτιώσει το περιβάλλον εκπαίδευσης των μετασχηματιστών. Ας δούμε τώρα το κίνητρο της προ-προπόνησης και πώς βοηθάει τη διαδικασία τελειοποίησης.

### 2.3.2 Προεκπαίδευση και τελειοποίηση ενός μοντέλου BERT

Το BERT είναι ένα πλαίσιο δύο βημάτων. Το πρώτο βήμα είναι η προ-προπόνηση και το δεύτερο είναι η τελειοποίηση



Η εκπαίδευση ενός μοντέλου μετασχηματιστή μπορεί να διαρκέσει ώρες, αν όχι ημέρες. Χρειάζεται αρκετός χρόνος για τη μηχανική της αρχιτεκτονικής και των παραμέτρων και την επιλογή των κατάλληλων συνόλων δεδομένων για την εκπαίδευση ενός μοντέλου μετασχηματιστή.

Η προ εκπαίδευση είναι το πρώτο βήμα του πλαισίου BERT που μπορεί να αναλυθεί σε δύο υποστάδια:

- Καθορισμός της αρχιτεκτονικής του μοντέλου: αριθμός στρωμάτων, αριθμός κεφαλών, διαστάσεις και τα άλλα δομικά στοιχεία του μοντέλου
- Εκπαίδευση του μοντέλου σε εργασίες Μοντελοποίησης Μασικής Γλώσσας (MLM) και NSP

Το δεύτερο βήμα του πλαισίου BERT είναι η τελειοποίηση, η οποία μπορεί επίσης να αναλυθεί σε δύο υποστάδια:

- Εκκίνηση του κατάντη μοντέλου που επιλέχθηκε με τις εκπαιδευμένες παραμέτρους του προεκπαιδευμένου μοντέλου BERT
- Βελτιστοποίηση των παραμέτρων για συγκεκριμένες εργασίες μεταγενέστερου σταδίου, όπως Αναγνώριση Κειμενικής Ενέργειας (RTE), Απάντηση Ερωτήσεων (SQuAD v1.1, SQuAD v2.0) και Καταστάσεις με αντίθετες γενιές (SWAG)

## 3. Τεχνολογία

---

### 3.1 Python

Η Python θεωρείται ευρέως ως επιλογή, για την ανάπτυξη μοντέλων τεχνητής νοημοσύνης, επειδή είναι εύκολο στην εκμάθηση και τη χρήση. Η απλή σύνταξή της απλοποιεί την κωδικοποίηση αλγορίθμων AI. Η Python προσφέρει επίσης υποστήριξη για εξειδικευμένα εργαλεία και βιβλιοθήκες που έχουν σχεδιαστεί ειδικά για AI, όπως το TensorFlow, το Keras και το PyTorch. Αυτοί οι πόροι διευκολύνουν σημαντικά την διαδικασία κατασκευής μοντέλων AI. Επιπλέον, η Python μπορεί να υπερηφανεύεται για μια κοινότητα χρηστών που συνεργάζονται ενεργά, ανταλλάσσουν ιδέες και παρέχουν βοήθεια ο ένας στον άλλον. Αυτό το περιβάλλον συνεργασίας διευκολύνει την εύρεση λύσεων και την ανακάλυψη τεχνικών στον τομέα της AI. Δεδομένων αυτών των παραγόντων, δεν αποτελεί έκπληξη το γεγονός ότι η Python έχει κερδίσει δημοτικότητα και αποδεικνύεται πολύτιμο για έργα τεχνητής νοημοσύνης.

### 3.2 Deep learning Frameworks

#### 3.2.1 Tensorflow

Το TensorFlow, που αναπτύχθηκε από την Google είναι ένα πλαίσιο ανοιχτού κώδικα, για μηχανική μάθηση. Είναι γνωστό για το παράδειγμα υπολογιστικού γραφήματος, όπου βρίσκονται οι υπολογισμοί αναπαρίσταται ως κατευθυνόμενο γράφημα. Αρχικά, το TensorFlow χρησιμοποιούσε έναν στατικό υπολογισμό γράφημα, που σημαίνει ότι ολόκληρη η δομή του γραφήματος έπρεπε να καθοριστεί πριν από οποιοδήποτε θα μπορούσε να πραγματοποιηθεί υπολογισμός. Ωστόσο, το TensorFlow2.x εισήγαγε την ανυπόμονη εκτέλεση, επιτρέποντας ευελιξία και διαισθητική ανάπτυξη. Υποστηρίζει επιλογές υλικού όπως CPU, GPU και TPU (Tensor Processing Units). Με μια ενεργή κοινότητα, Το TensorFlow προσφέρει πληθώρα πόρων και βιβλιοθηκών, για εργασίες μηχανικής μάθησης.

#### 3.2.2 Keras

Το Keras ξεκίνησε ως ένα ανεξάρτητο API νευρωνικών δικτύων υψηλού επιπέδου, σχεδιασμένο να είναι φιλική προς το χρήστη και να παρέχει ως απλή διεπαφή για μοντέλα κατασκευής και εκπαίδευσης. Το Keras στοχεύει να παρέχει ένα αθέμιτο πλεονέκτημα σε κάθε προγραμματιστή που θέλει να στείλει εφαρμογές που υποστηρίζονται από τη μηχανική μάθηση. Το Keras εστιάζει στην ταχύτητα εντοπισμού σφαλμάτων, την κομψότητα του κώδικα και συνοπτικότητα, δυνατότητα συντήρησης και δυνατότητα ανάπτυξης. Με τη χρήση του κώδικα Keras η βάση είναι μικρότερη, πιο ευανάγνωστα, ευκολότερα στην επανάληψη. Τα μοντέλα Keras λειτουργούν πιο γρήγορα χάρη στη συλλογή XLA με JAX και TensorFlow και είναι πιο εύκολο να αναπτυχθούν σε κάθε επιφάνεια (διακομιστής, κινητό, πρόγραμμα περιήγησης, ενσωματωμένο) χάρη στα εξαρτήματα εξυπηρέτησης από τα οικοσυστήματα TensorFlow και PyTorch



### **3.2.3 PyTorch**

Το PyTorch είναι ένα πλαίσιο μηχανικής μάθησης ανοιχτού κώδικα που αναπτύχθηκε από το Facebook

Ερευνητικό εργαστήριο AI (FAIR). Είναι γνωστό για το δυναμικό υπολογιστικό γράφημά του, το οποίο επιτρέπει πιο ευέλικτη και διαισθητική ανάπτυξη. Σε αντίθεση με τις προηγούμενες εκδόσεις του TensorFlow, το PyTorch υιοθετεί ένα πιο Pythonic και επιτακτικό στυλ, καθιστώντας το δημοφιλές

μεταξύ ερευνητών και επαγγελματιών που προτιμούν μια δυναμική προσέγγιση στην κατασκευή μοντέλων. Το PyTorch κέρδισε σημαντική δημοτικότητα στην κοινότητα αναζήτησης εκεί λόγω ευκολία χρήσης, ισχυρή υποστήριξη για την επιτάχυνση GPU και μια εξαιρετικά ενεργή κοινότητα.

### **3.2.4 Βασικές διαφορές**

#### **Υπολογιστικά Γραφήματα**

Το TensorFlow αρχικά χρησιμοποιούσε στατικά γραφήματα υπολογισμού, ενώ το PyTorch χρησιμοποιεί γραφήματα δυναμικού υπολογισμού. Το Keras, όταν ενσωματώνεται με το TensorFlow, ακολουθεί το παράδειγμα υπολογιστικού γραφήματος του TensorFlow.

#### **Ευκολία στη χρήση**

Το TensorFlow είχε μια πιο απότομη καμπύλη μάθησης αρχικά, αλλά με το TensorFlow2.x και πρόθυμο εκτέλεση, έγινε πιο διαισθητικό. Το Keras ήταν πάντα σχεδιασμένο για ευκολία στη χρήση και φιλικότητα προς τον χρήστη. Ο PyTorch είναι γνωστός για τον διαισθητικό και Pythonic προγραμματισμό του στυλ.

#### **Ευελιξία και Έλεγχος**

Το TensorFlow παρέχει API υψηλού και χαμηλού επιπέδου, προσφέροντας μια ισορροπία μεταξύ αφαίρεση και έλεγχος. Το Keras αφαιρεί πολλές λεπτομέρειες χαμηλού επιπέδου, παρέχοντας λιγότερη ευελιξία σε σύγκριση με τα API χαμηλότερου επιπέδου του TensorFlow. Το PyTorch προσφέρει ένα εξαιρετικά ευέλικτο και δυναμική προσέγγιση, επιτρέποντας υψηλό βαθμό ελέγχου στα μοντέλα.

#### **Κοινότητα και Οικοσύστημα**

Το TensorFlow έχει μια μεγάλη αποσταθεροποιημένη κοινότητα με εκτεταμένους πόρους και βιβλιοθήκες τρίτων. Το Keras επωφελείται από το οικοσύστημα του TensorFlow και έχει το δικό του κοινότητα επίσης. Το PyTorch απέκτησε ταχεία δημοτικότητα στις ερευνητικές κοινότητες, ιδιαίτερα για το δυναμικό υπολογιστικό γράφημα.

#### **Υποστήριξη υλικού**

Το TensorFlow παρέχει εκτεταμένη υποστήριξη για διάφορα υλικά, συμπεριλαμβανομένων των CPU, των GPU, και TPUs. Το PyTorch είναι κατάλληλο για επιτάχυνση GPU και προσαρμόζεται σε διαφορετικά ρυθμίσεις υλικού.

Τελικά, η επιλογή μεταξύ TensorFlow, Keras και PyTorch εξαρτάται από τις ατομικές προτιμήσεις, τις απαιτήσεις του έργου και την εξοικείωση με το πλαίσιο. Και τα τρία είναι ισχυρά εργαλεία με ενεργές κοινότητες και χρησιμοποιούνται ευρέως και στα δύο έρευνα και βιομηχανία.

### 3.3 Άλλες βιβλιοθήκες Python

#### 3.3.1 NumPy

Το NumPy, συντομογραφία του Numerical Python, είναι ένα θεμελιώδες πακέτο για επιστημονικά υπολογισμούς σε Python. Είναι διάσημο για το ισχυρό αντικείμενο της σειράς N-διάστατων, που είναι ένα ευέλικτο δοχείο για μεγάλες και πολυδιάστατες συστοιχίες. Πίνακες NumPy διευκολύνουν προηγμένες μαθηματικές και στατιστικές πράξεις, καθώς είναι βελτιστοποιημένες για απόδοση και επιτρέπουν την αποτελεσματική επεξεργασία συστοιχιών. Η βιβλιοθήκη παρέχει επίσης εργαλεία για την ενσωμάτωση κώδικα C/C++ και FORTRAN, επιτρέποντας περαιτέρω βελτιστοποίηση και ταχύτητα. Η ευρεία διάδοση της δημοτικότητάς του μεταξύ των επιστημόνων δεδομένων και των ερευνητών πηγάζει από τις υψηλού επιπέδου μαθηματικές του λειτουργίες, την ευκολία ενσωμάτωσης με άλλες βιβλιοθήκες, και τον κεντρικό του ρόλο στο ευρύτερο οικοσύστημα ανάλυσης δεδομένων, μηχανικής μάθησης και επιστημονικός υπολογισμός στην Python.

#### 3.3.2 Pandas

Το Pandas είναι μια βιβλιοθήκη Python που χρησιμοποιείται ευρέως για χειρισμό και ανάλυση δεδομένα. Προσφέρει δομές δεδομένων υψηλού επιπέδου, όπως DataFrame και Series που απλοποιούν χειρισμό και επεξεργασία πληροφοριών. Με τα Pandas τα άτομα μπορούν αποτελεσματικά εκτελεί εργασίες όπως καθαρισμός και μετατροπή δεδομένων συγχώνευσης συνόλων δεδομένων και δημιουργίας οπτικοποιήσεων. Η ικανότητά του να εισάγει και να εξάγει πληροφορίες δεδομένων όπως CSV, Excel και οι βάσεις δεδομένων SQL το έχουν κάνει ένα προτιμώμενο εργαλείο, μεταξύ των επιστημόνων δεδομένων και των αναλυτών. Η φιλική προς το χρήστη διεπαφή του Panda, σε συνδυασμό με τη γκάμα των δυνατοτήτων του, έχει σταθερά το καθιέρωσε ως στοιχείο της εργαλειοθήκης ανάλυσης δεδομένων Python.

#### 3.3.3 Scikit-learn

Το Scikit Learn, γνωστό και ως sklearn είναι μια χρησιμοποιημένη βιβλιοθήκη μηχανικής εκμάθησης, για την Python που εκτιμάται ιδιαίτερα για την απλότητα και την αποτελεσματικότητά του. Προσφέρει μια σειρά από εργαλεία για εργασίες μηχανικής μάθησης όπως ταξινόμηση, παλινδρόμηση, ομαδοποίηση και μείωση διαστάσεων. Το Scikitlearn βασίζεται στα θεμέλια των NumPy και SciPy παρέχοντας παράλληλα μια διεπαφή για τη δημιουργία και την τελειοποίηση μοντέλων μηχανικής εκμάθησης ολοκληρωμένη τεκμηρίωση, χαρακτήρας χρήστη και ικανότητα χειρισμού τύπων δεδομένων είναι μια ευνοϊκή επιλογή τόσο για αρχάριους όσο και για έμπειρους ασκούμενους, στον τομέα της επιστήμης δεδομένων και της μηχανικής μάθησης.

#### 3.3.4 Matplotlib

Το Matplotlib είναι μια γνωστή βιβλιοθήκη Python που χρησιμοποιείται ευρέως για τη δημιουργία κινούμενων εικόνων και διαδραστικές απεικονίσεις. Έχει κερδίσει δημοτικότητα, για τη φύση του χρήστη και ευελιξία που επιτρέπει στους χρήστες να δημιουργούν διάφορους τύπους γραφημάτων και γραφημάτων χρησιμοποιώντας μόνο λίγες γραμμές κώδικα. Ένα από τα πλεονεκτήματα του Matplotlib είναι το επίπεδο προσαρμογής του, το οποίο παρέχει στους χρήστες τη δυνατότητα να ρυθμίζουν με ακρίβεια σχεδόν κάθε πτυχή της πλοκής τους. Από άξονες όρια και ετικέτες στο συνολικό στυλ και χρωματικό συνδυασμό. Αυτή η προσαρμοστικότητα κάνει είναι ένα απαραίτητο εργαλείο στην Python για την οπτικοποίηση δεδομένων σε επαγγελματικά περιβάλλοντα που εξυπηρετούν σκοπούς όπως η διερεύνηση προτύπων δεδομένων που παρουσιάζουν τα ερευνητικά ευρήματα και απεικονίζουν σύνθετα σύνολα δεδομένων.

## 4. Ανάλυση και Προπεξεργασία Δεδομένων

---

### 4.1 Βάση Δεδομένων

#### 4.1.1 Εύρεση

Για αυτό το έργο χρειαζόμαστε μια βάση δεδομένων που όχι μόνο θα έπρεπε να είναι τεράστια ώστε το μοντέλο να παίρνει τυχαία τραγούδια από οποιοδήποτε είδος, αλλά και να έχει μεγάλη ποικιλία σε κάθε είδος και ακριβείς στίχους. Η δημιουργία μιας βάσης δεδομένων όπως αυτή από την αρχή ήταν εκτός θέματος και σχεδόν κάθε βάση δεδομένων στον Ιστό ήταν μικρή ή ελάχιστη από ποικιλία σε είδη και ακόμη και τραγούδια. Το καλύτερο που βρήκα ήταν στο Kaggle με περίπου 290000 τραγούδια. Αποδείχθηκε ότι είναι μια απίστευτα καλή βάση δεδομένων και προτείνω σε όλους όσους εργάζονται στην ανάλυση δεδομένων ή στη μηχανική μάθηση για ταξινόμηση είδους να τη χρησιμοποιήσουν. Με κάποιες αλλαγές και γυάλισμα θα πρέπει να είναι αριστούργημα.

#### 4.1.2 Περιγραφή Βάσης

Αυτή η βάση δεδομένων έχει περίπου 290.000 διαφορετικά τραγούδια και σε κάθε τραγούδι, έχουμε τις στήλες Καλλιτέχνης, Όνομα τραγουδιού, Γλώσσα, Είδος και στίχοι. Έχουμε 10 διαφορετικά είδη και 33 διαφορετικές γλώσσες.

## 4.2 Ανάλυση Βάσης

Η απόκτηση τόσο τεράστιων και απρόβλεπτων δεδομένων έχει επίσης τα μειονεκτήματά της. Πρέπει να το φέρουμε στο σχήμα που είναι καλύτερο για την ακρίβεια του προβλήματός μας. Πριν ξεκινήσουμε, πρέπει να ρίξουμε διπλότυπα, ώστε να μην επηρεάσουν τα αποτελέσματα και να απορρίψουμε επίσης αυτά όπου οι στίχοι είναι NULL.

Ξεκινώντας από τα βασικά, εξετάζουμε την κεφαλή των δεδομένων για να γνωρίζουμε την ακριβή δομή:

|   | Artist    | Song                    | Genre | Language | Lyrics  |
|---|-----------|-------------------------|-------|----------|---|
| 0 | 12 stones | world so cold           | Rock  | en       | It starts with pain, followed by hate\nFueled ... |
| 1 | 12 stones | broken                  | Rock  | en       | Freedom!\nAlone again again alone\nPatiently w... |
| 2 | 12 stones | 3 leaf loser            | Rock  | en       | Biting the hand that feeds you, lying to the v... |
| 3 | 12 stones | anthem for the underdog | Rock  | en       | You say you know just who I am\nBut you can't ... |
| 4 | 12 stones | adrenaline              | Rock  | en       | My heart is beating faster can't control these... |

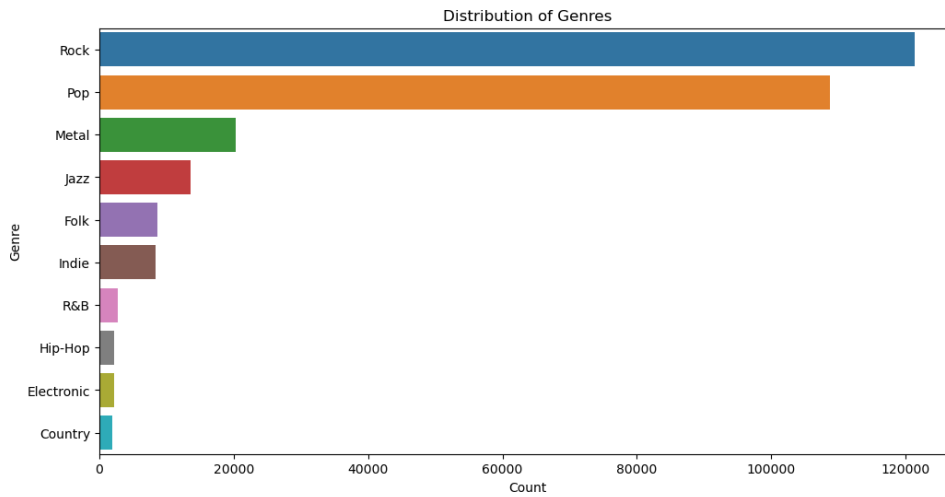
Και χρειαζόμαστε επίσης κάποιες κύριες πληροφορίες των δεδομένων:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 290183 entries, 0 to 290182
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Artist      290183 non-null object
1   Song        290182 non-null object
2   Genre       290183 non-null object
3   Language    290179 non-null object
4   Lyrics      290148 non-null object
dtypes: object(5)
memory usage: 11.1+ MB
```

Και για πιο αναλυτική περιγραφή:

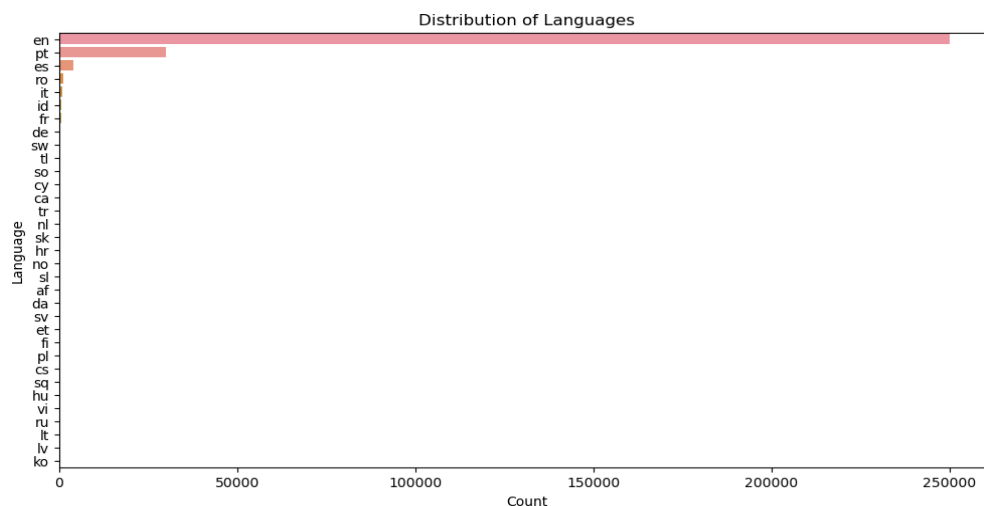
|               | Artist        | Song   | Genre  | Language | Lyrics       |
|---------------|---------------|--------|--------|----------|--------------|
| <b>count</b>  | 290183        | 290182 | 290183 | 290179   | 290148       |
| <b>unique</b> | 11152         | 164357 | 10     | 33       | 249297       |
| <b>top</b>    | elvis presley | intro  | Rock   | en       | Instrumental |
| <b>freq</b>   | 1611          | 163    | 121404 | 250197   | 540          |

Στη συνέχεια πηγαίνουμε σε πιο εις βάθος ανάλυση, ελέγχουμε πρώτα την Κατανομή των Ειδών στη βάση δεδομένων:



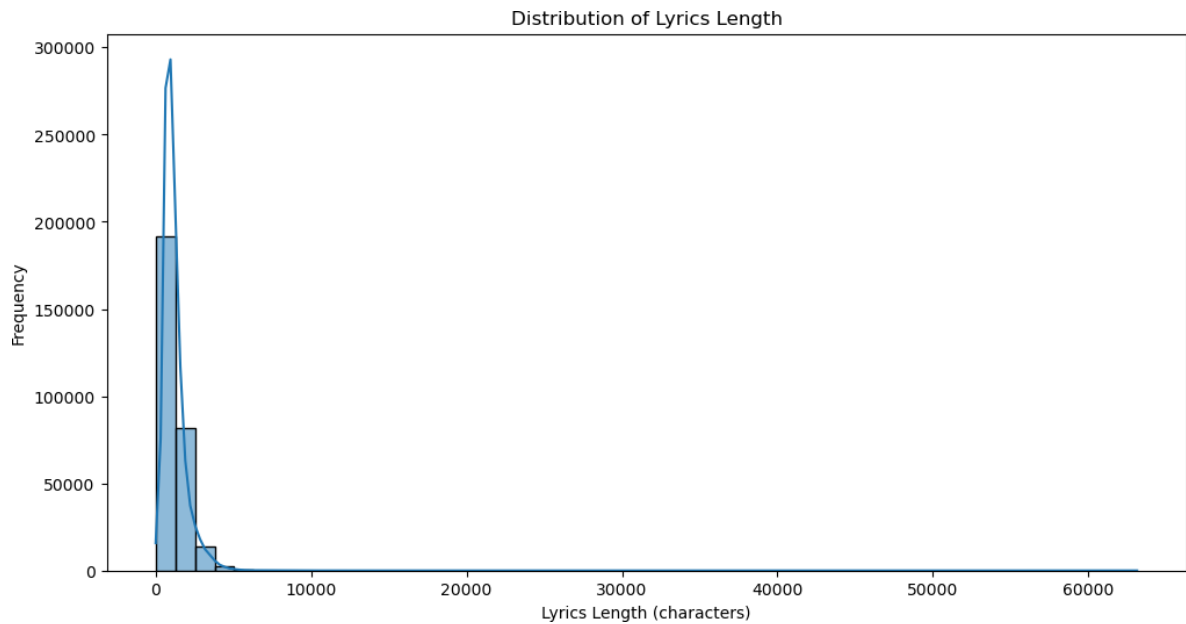
Είναι σαφές ότι δεν είναι ισορροπημένο, επομένως χρειαζόμαστε περισσότερες πληροφορίες, επομένως περιορίζουμε τα δεδομένα και λαμβάνουμε μόνο τα καλύτερα χρησιμοποιώντας δεδομένα.

Το επόμενο βήμα είναι να ελέγξετε την κατανομή της γλώσσας::



Αγγλικά, Πορτογαλικά, Ισπανικά είναι οι πιο χρησιμοποιούμενες γλώσσες με τα Αγγλικά να είναι μακράν οι πιο χρησιμοποιούμενες.

Έλεγχος της κατανομής του μήκους των στίχων:



Τα περισσότερα τραγούδια έχουν κατά μέσο όρο το ίδιο μήκος στίχων που είναι καλό για εξισορρόπηση.

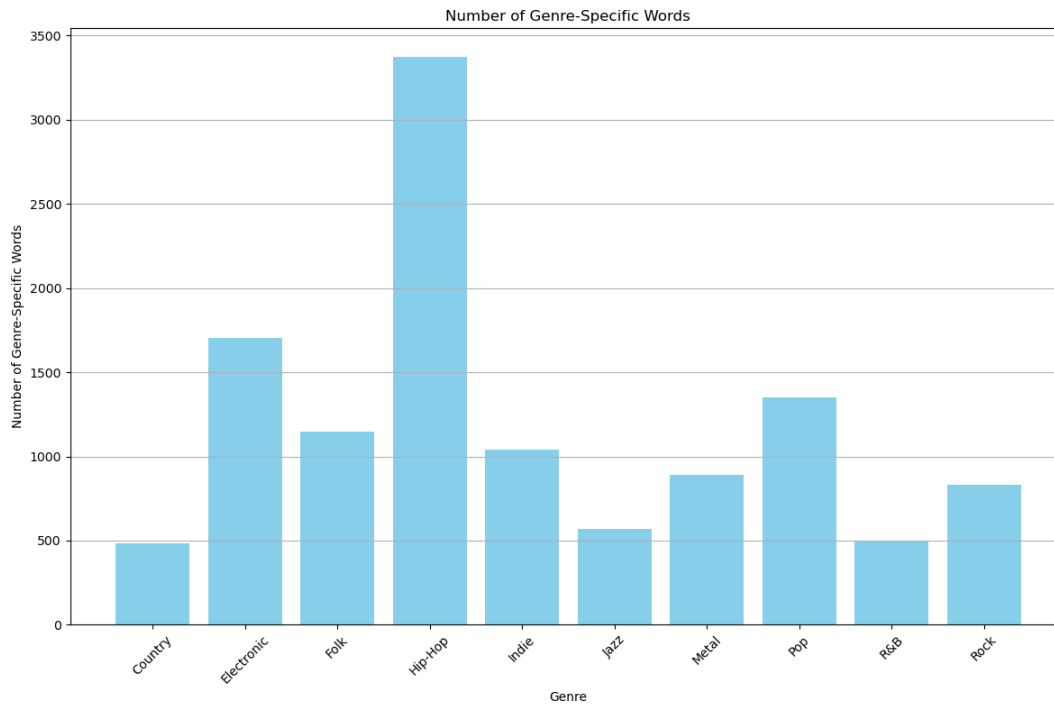
### 4.3 Προεπεξεργασία Δεδομένων

Αφού λάβουμε και αναλύσουμε πληροφορίες και δεδομένα, πρέπει να κάνουμε κάποιες αλλαγές στα δεδομένα για να επιτύχουμε τα καλύτερα αποτελέσματα. Διατηρώ μόνο τις 3 γλώσσες που χρησιμοποιούνται περισσότερο και ρίχνω όλες τις άλλες γλώσσες.

Μετά από αυτό, θα πρέπει να ελαχιστοποιήσω τυχαία τη βάση δεδομένων, ώστε να υπάρχει ο ίδιος αριθμός τραγουδιών σε κάθε είδος. Διαλέγω 200 τραγούδια από κάθε είδος:

```
Genre
Country      200
Electronic   200
Folk         200
Hip-Hop      200
Indie        200
Jazz         200
Metal        200
Pop          200
R&B          200
Rock         200
Name: count, dtype: int64
```

Το επόμενο βήμα είναι να μάθετε σε ποιο είδος υπάρχουν περισσότερες μοναδικές λέξεις, ώστε να είναι πιο εύκολο για το μοντέλο να έχει καλύτερα αποτελέσματα.



Με διαφορά το Hip-Hop έχει τις πιο μοναδικές λέξεις σε κάθε στίχο τραγουδιού. Θα κρατήσουμε τα πέντε είδη που έχουν πιο μοναδικά. Αυτά θα είναι Hip-Hop, Pop, Folk, Electronic και Indie για τουλάχιστον τις αρχικές προσπάθειες με το μοντέλο μας. Υπάρχει πάντα ο κίνδυνος με μόνο στίχους και χωρίς ήχο για πολλά είδη να έχουν πολλές ίδιες λέξεις και είναι πραγματικά πολύ δύσκολο έργο. Θα δοκιμάσω πολλούς συνδυασμούς ειδών για να φτάσω τη μέγιστη ακρίβεια και θα επιλέξω διαφορετικό αριθμό ειδών από 3 έως 5.

## 5. Μοντέλο

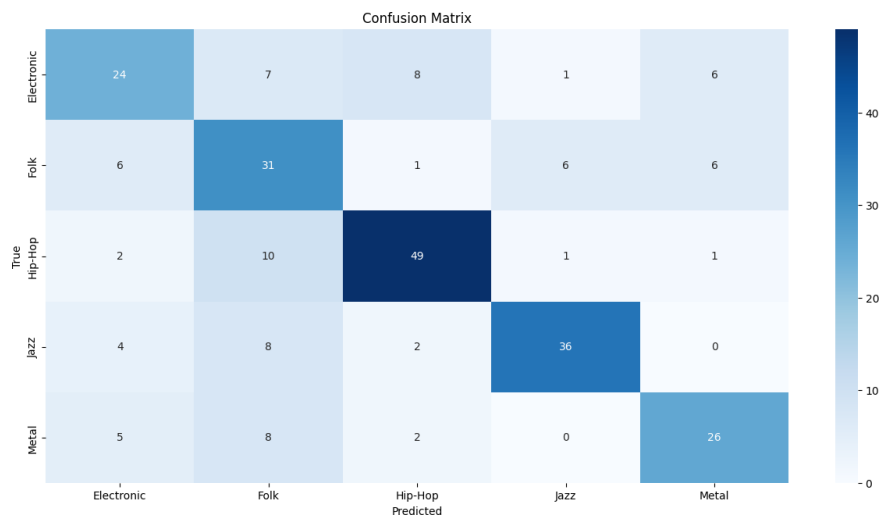
### 5.1 Δημιουργία Μοντέλου

Το μοντέλο μας θα είναι ένα τυπικό προεκπαιδευμένο μοντέλο με βάση το μπερτ. Κωδικοποίησα τα είδη δεδομένων χρησιμοποιώντας έναν καυτό κωδικοποιητή επειδή είναι διαφορετικά αντικείμενα. Επιπλέον, χώρισα τα δεδομένα σε προπονήσεις και σετ δοκιμών με το σύνολο δοκιμών να είναι το 10% του συνόλου. Οι επόμενες παρτίδες ξεκίνησαν στις 16, αλλά γρήγορα ορίστηκαν σε 64. Προσθέτω την εγκατάλειψη για να ελέγξω την υπερπροσαρμογή και ορίστηκε στο 0,1 αλλά άλλαξε σε 0,5 μετά από πολλές προσπάθειες. Τελικά ο ρυθμός εκμάθησης ξεκίνησε στο  $3e-5$ , αλλά ρυθμίζεται στο  $2e-5$  ενώ εγώ επιλέγω 15 εποχές.

### 5.2 Αποτελέσματα

#### 5.2.1 Μοντέλο 5 ειδών

Το εκπαιδευτικό μοντέλο με 5 πρώτα είδη είχε πολύ άσχημα αποτελέσματα. Αυτό ήταν αποτέλεσμα των πολύ κοινών λέξεων που χρησιμοποιούνται στην ποπ και τη λαϊκή και στο indie. Έτσι άλλαξα τα είδη σε Metal, Jazz, Hip-Hop, Folk, Electronic. Η πρώτη προσπάθεια ήταν πραγματικά πολλά υποσχόμενη με αυτά τα αποτελέσματα:





```

Validation Loss: 1.0125
Validation Accuracy: 0.6640
4/4 [=====] - 21s 3s/step

```

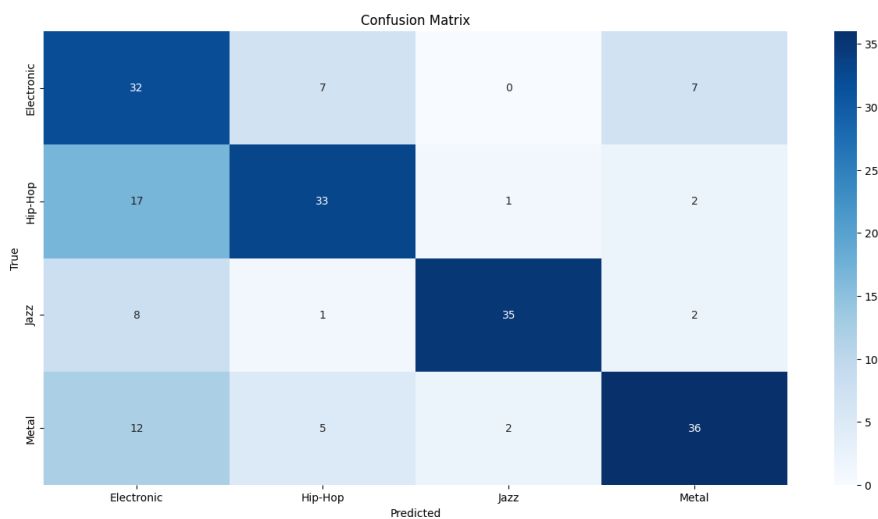
|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| Electronic   | 0.59      | 0.52   | 0.55     | 46      |
| Folk         | 0.48      | 0.62   | 0.54     | 50      |
| Hip-Hop      | 0.79      | 0.78   | 0.78     | 63      |
| Jazz         | 0.82      | 0.72   | 0.77     | 50      |
| Metal        | 0.67      | 0.63   | 0.65     | 41      |
| accuracy     |           |        | 0.66     | 250     |
| macro avg    | 0.67      | 0.65   | 0.66     | 250     |
| weighted avg | 0.68      | 0.66   | 0.67     | 250     |

Η βαθμολογία του είναι πολύ καλή αν λάβουμε υπόψη ότι πολλές άλλες σχετικές εργασίες έχουν ακρίβεια 44-65%.

### 5.2.2 Μοντέλο 4 Ειδών

Επόμενο βήμα είναι για ακόμα καλύτερα αποτελέσματα, έριξα τα είδη στο 4 και μετά το 3 με την καλύτερη πρόβλεψη και ακρίβεια.

Πρώτα, έριξα το Folk και το έτρεξα ξανά. Τα αποτελέσματα ήταν ελαφρώς καλύτερα και πλησίασαν το 70%:



```

7/7 [=====] - 9s 1s/step - loss: 0.8984 - accuracy: 0.6800
Validation Loss: 0.8984
Validation Accuracy: 0.6800
7/7 [=====] - 14s 1s/step
      precision    recall  f1-score   support

 Electronic      0.46      0.70      0.56         46
   Hip-Hop      0.72      0.62      0.67         53
     Jazz      0.92      0.76      0.83         46
     Metal      0.77      0.65      0.71         55

 accuracy                   0.68         200
 macro avg      0.72      0.68      0.69         200
weighted avg      0.72      0.68      0.69         200
    
```

### 5.2.3 Μοντέλο 3 Ειδών

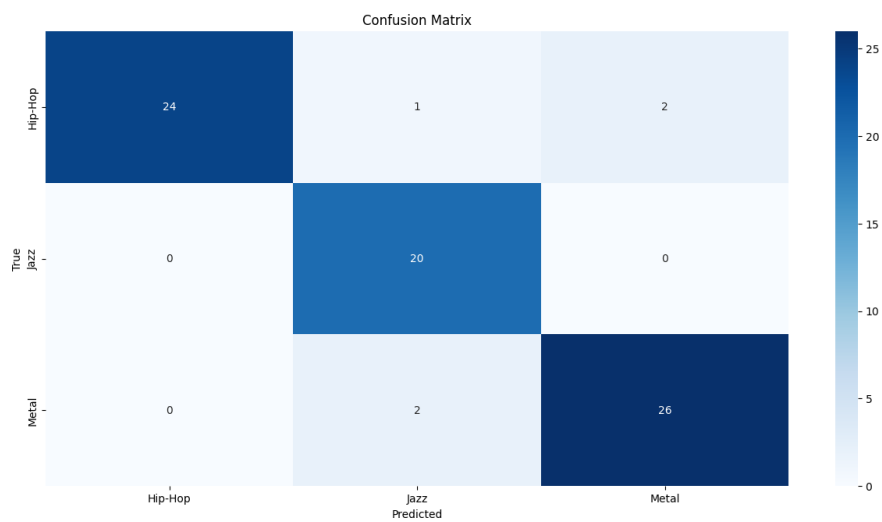
Τέλος, έριξα και το Electronic αφήνοντας το μοντέλο για μόλις 3 είδη και τα αποτελέσματα ήταν εκπληκτικά:

```

5/5 [=====] - 3s 685ms/step - loss: 0.3487 - accuracy: 0.9333
Validation Loss: 0.3487
Validation Accuracy: 0.9333
5/5 [=====] - 7s 526ms/step
      precision    recall  f1-score   support

 Hip-Hop         1.00      0.89      0.94         27
      Jazz         0.87      1.00      0.93         20
      Metal        0.93      0.93      0.93         28

 accuracy                   0.93         75
 macro avg                  0.93      0.94      0.93         75
 weighted avg               0.94      0.93      0.93         75
    
```



Είναι μια σχεδόν τέλεια βαθμολογία με ακρίβεια 93% και απώλεια επικύρωσης 0,3

## 6. Συμπεράσματα

---

Η παρούσα πτυχιακή εργασία επικεντρώθηκε στην ανάπτυξη ενός αποτελεσματικού μοντέλου για την αυτόματη ταξινόμηση μουσικών ειδών χρησιμοποιώντας το μοντέλο BERT (Bidirectional Encoder Representations from Transformers). Η πρόκληση της ακριβούς ταξινόμησης μουσικών ειδών αποκλειστικά βάσει των στίχων αντιμετωπίστηκε μέσω εκτεταμένης προεπεξεργασίας δεδομένων, βελτιστοποίησης του μοντέλου και επαναλαμβανόμενων πειραμάτων. Ο κύριος στόχος ήταν η διερεύνηση της αποτελεσματικότητας των μοντέλων μετασχηματιστών σε εργασίες ταξινόμησης κειμένου και η επίτευξη υψηλής ακρίβειας. Αρχικά, οι δοκιμές με πέντε μουσικά είδη δεν απέδωσαν τα επιθυμητά αποτελέσματα, γεγονός που οδήγησε σε προσαρμογές και αναπροσαρμογή των ειδών που χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου. Οι επόμενες πειραματικές διαδικασίες με λιγότερα είδη οδήγησαν σε σημαντικές βελτιώσεις, με το τελικό μοντέλο να ταξινομεί τρία είδη με εντυπωσιακή ακρίβεια 93%. Η έρευνα αυτή επιβεβαίωσε την αποτελεσματικότητα των μοντέλων μετασχηματιστών, όπως το BERT, στην ταξινόμηση κειμένων και παρείχε χρήσιμες πληροφορίες για τη βελτιστοποίηση των ροών εργασίας μηχανικής μάθησης για πρακτικές εφαρμογές στην ταξινόμηση μουσικών ειδών. Επιπλέον, αυτή η μελέτη υπογραμμίζει τη σημασία της προσεκτικής επιλογής και προεπεξεργασίας δεδομένων, καθώς και της συνεχούς αναπροσαρμογής και βελτίωσης του μοντέλου για την επίτευξη βέλτιστων αποτελεσμάτων. Η χρήση του BERT αποδείχθηκε ιδιαίτερα αποτελεσματική στην αντιμετώπιση των προκλήσεων που θέτουν τα μη ισορροπημένα σύνολα δεδομένων και οι κοινές λέξεις μεταξύ των μουσικών ειδών, επιτυγχάνοντας υψηλή απόδοση στην ταξινόμηση.

### **Μελλοντική Εργασία και Πειράματα**

Η έρευνα αυτή ανοίγει το δρόμο για περαιτέρω μελέτες και βελτιώσεις. Παρακάτω παρατίθενται κάποιες προτάσεις για μελλοντική εργασία και πειράματα:

#### **Επέκταση των Ειδών:**

Η εισαγωγή περισσότερων μουσικών ειδών στο μοντέλο μπορεί να προσφέρει μια πιο ευρεία ταξινόμηση. Εξετάζοντας τη δυνατότητα ταξινόμησης σε δέκα ή περισσότερα είδη, το μοντέλο θα γίνει πιο ευέλικτο και χρήσιμο σε πραγματικές εφαρμογές.

#### **Συνδυασμός Πολυτροπικών Δεδομένων:**

Η ενσωμάτωση ηχητικών χαρακτηριστικών, εκτός από τους στίχους, μπορεί να βελτιώσει την ακρίβεια του μοντέλου. Ο συνδυασμός κειμένου και ήχου μπορεί να παράσχει μια πληρέστερη εικόνα του είδους της μουσικής.

#### **Χρήση Προηγμένων Μοντέλων Μετασχηματιστών:**

Η διερεύνηση και χρήση πιο σύγχρονων μοντέλων μετασχηματιστών, όπως το GPT-3 ή άλλων εκδόσεων του BERT, μπορεί να αποδώσει ακόμα καλύτερα αποτελέσματα.

**Αντιμετώπιση Μη Ισορροπημένων Δεδομένων:**

Η εφαρμογή τεχνικών υπερδειγματοληψίας ή υποδειγματοληψίας για την αντιμετώπιση των μη ισορροπημένων δεδομένων μπορεί να βελτιώσει την απόδοση του μοντέλου σε κατηγορίες με λιγότερα δεδομένα.

**Βελτιστοποίηση Υπερπαραμέτρων:**

Η συστηματική βελτιστοποίηση των υπερπαραμέτρων του μοντέλου μπορεί να οδηγήσει σε περαιτέρω βελτίωση της ακρίβειας. Η χρήση αλγορίθμων όπως το Grid Search ή το Bayesian Optimization μπορεί να είναι χρήσιμη.

**Εφαρμογή σε Πραγματικό Χρόνο:**

Η ανάπτυξη και δοκιμή του μοντέλου σε εφαρμογές πραγματικού χρόνου, όπως συστήματα συστάσεων μουσικής, μπορεί να αναδείξει την πρακτική του χρησιμότητα και να προσφέρει ανατροφοδότηση για περαιτέρω βελτιώσεις.

**Πολυγλωσσική Υποστήριξη:**

Η προσαρμογή του μοντέλου για να υποστηρίζει στίχους σε πολλές γλώσσες μπορεί να το καταστήσει πιο χρήσιμο σε παγκόσμιο επίπεδο. Η επέκταση αυτή μπορεί να περιλαμβάνει την προεκπαίδευση του μοντέλου σε πολυγλωσσικά σύνολα δεδομένων.

**Μελέτη Συναισθημάτων:**

Η ενσωμάτωση της ανάλυσης συναισθημάτων των στίχων για την ταξινόμηση των ειδών μπορεί να προσφέρει πιο ακριβείς προβλέψεις, ιδιαίτερα σε είδη που χαρακτηρίζονται από συγκεκριμένα συναισθηματικά μοτίβα.

Η παρούσα εργασία αποτελεί σημαντική συμβολή στον τομέα της αυτόματης ταξινόμησης μουσικών ειδών και θέτει τις βάσεις για μελλοντικές ερευνητικές προσπάθειες που μπορούν να οδηγήσουν σε περαιτέρω καινοτομίες και βελτιώσεις.

## Βιβλιογραφία

---

- [1] Adhikari, Ashutosh, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. “Docbert: Bert for Document Classification.” arXiv Preprint arXiv:1904.08398.
- [2] Araújo Lima, Raul de, Rômulo César Costa de Sousa, Simone Diniz Junqueira Barbosa, and Hélio Cortês Vieira Lopes. 2020. “Brazilian Lyrics-Based Music Genre Classification Using a Blstm Network.” <http://arxiv.org/abs/2003.05377>.
- [3] Chang, Wei-Cheng, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit Dhillon. 2019. “X-Bert: EXtreme Multi-Label Text Classification with Using Bidirectional Encoder Representations from Transformers.” arXiv Preprint arXiv:1905.02331.
- [4] Chia, Yew Ken, Sam Witteveen, and Martin Andrews. 2019. “Transformer to Cnn: Label-Scarce Distillation for Efficient Text Classification.” <http://arxiv.org/abs/1909.03508>.
- [5] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” <http://arxiv.org/abs/1810.04805>.
- [6] Howard, Sam, Carlos N. Silla Jr, and Colin G. Johnson. 2011. “Automatic LyricsBased Music Genre Classification in a Multilingual Setting.” In Thirteenth Brazilian Symposium on Computer Music. <https://kar.kent.ac.uk/33266/>. [7] Huang, Derek A, Arianna A Serafini, and Eli J Pugh. n.d. “Music Genre Classification.”
- [8] “Hugging Face – on a Mission to Solve Nlp, One Commit at a Time.” n.d. Hugging Face – on a Mission to Solve NLP, One Commit at a Time. <https://huggingface.co/>.
- [9] Johnson, Rie, and Tong Zhang. 2016. “Supervised and Semi-Supervised Text Categorization Using Lstm for Region Embeddings.” <http://arxiv.org/abs/1602.02373>.
- [10] Johnson, Rie, and Tong Zhang. 2017. “Deep Pyramid Convolutional Neural Networks for Text Categorization.” In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 562–70.
- [11] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
- [12] Li, Wenting, Shangbing Gao, Hong Zhou, Zihong Huang, Kewen Zhang, and Wei Li. 2019. “The Automatic Text Classification Method Based on Bert and Feature Union.” In 2019 IEEE 25th International Conference on Parallel and Distributed Systems (Icnp), 774–77. IEEE.
- [13] Lin, Zhouhan, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. “A Structured Self-Attentive Sentence Embedding.” arXiv Preprint arXiv:1703.03130.
- [14] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2019. On the variance of the adaptive learning rate and beyond. arXiv preprint arXiv:1908.03265 (2019).
- [15] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. 2016. “Recurrent Neural Network for Text Classification with Multi-Task Learning.” arXiv Preprint arXiv:1605.05101.
- [16] Mayer, Rudolf, Robert Neumayer, and Andreas Rauber. 2008a. “Combination of Audio and Lyrics Features for Genre Classification in Digital Audio Collections.” In Proceedings of the 16th Acm International Conference on Multimedia, 159–68.
- [17] Mayer, Rudolf, Robert Neumayer, and Andreas Rauber. 2008b. “Rhyme and Style Features for Musical Genre Classification by Song Lyrics.” In Ismir, 337–42.
- [18] Mayer, Rudolf, and Andreas Rauber. 2011. “Musical Genre Classification by Ensembles of Audio and Lyrics Features.” In Proceedings of International Conference on Music Information Retrieval, 675–80.

- [19] Radford, Alec, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. "Improving Language Understanding with Unsupervised Learning." Technical Report, OpenAI.
- [20] Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. "DistilBERT, a Distilled Version of Bert: Smaller, Faster, Cheaper and Lighter." <http://arxiv.org/abs/1910.01108>.
- [21] Schuster, Mike, and Kuldip Paliwal. 1997a. "Bidirectional Recurrent Neural Networks." *Signal Processing, IEEE Transactions on* 45 (December): 2673–81. <https://doi.org/10.1109/78.650093>.
- [22] Schuster, Mike, and Kuldip K Paliwal. 1997b. "Bidirectional Recurrent Neural Networks." *IEEE Transactions on Signal Processing* 45 (11): 2673–81.
- [23] Shen, Dinghan, Yizhe Zhang, Ricardo Henao, Qinliang Su, and Lawrence Carin. 2018. "Deconvolutional Latent-Variable Model for Text Sequence Matching." In *Thirty-Second Aaai Conference on Artificial Intelligence*.
- [24] Sun, Chi, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. "How to Fine-Tune Bert for Text Classification?" In *China National Conference on Chinese Computational Linguistics*, 194–206. Springer.
- [25] Tsaptsinos, Alexandros. 2017. "Music Genre Classification by Lyrics Using a Hierarchical Attention Network." In *ICME*.
- [26] Tzanetakis, George, and Perry Cook. 2002. "Musical Genre Classification of Audio Signals." *IEEE Transactions on Speech and Audio Processing* 10 (5): 293–302.
- [27] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." In *Advances in Neural Information Processing Systems*, 5998–6008.
- [28] Yang, Zichao, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. "Hierarchical Attention Networks for Document Classification." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–9.
- [29] Ying, Teh Chao, Shyamala Doraisamy, and Lili Nurliyana Abdullah. 2012. "Genre and Mood Classification Using Lyric Features." In *2012 International Conference on Information Retrieval & Knowledge Management*, 260–63. IEEE.
- [30] Yogatama, Dani, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. "Generative and Discriminative Text Classification with Recurrent Neural Networks." *arXiv Preprint arXiv:1703.0189*