ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αξιολόγηση Μηχανισμών Αυτόματης Κατηγοριοποίησης Ελληνικών Νομικών Κειμένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΠΑΝΑΓΙΩΤΗ ΧΑΤΖΗΓΙΑΝΝΑΚΗ

Επιβλέπων : Παναγιώτης Τσανάκας
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αξιολόγηση Μηχανισμών Αυτόματης Κατηγοριοποίησης Ελληνικών Νομικών Κειμένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΠΑΝΑΓΙΩΤΗ ΧΑΤΖΗΓΙΑΝΝΑΚΗ

Επιβλέπων : Παναγιώτης Τσανάκας
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 31η Οκτωβρίου 2023.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Παναγιώτης Τσανάκας	Ανδρέας-Γεώργιος Σταφυλοπάτης	Ευγενία Τζαννίνη
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Επίκουρος Καθηγήτρια Ε.Μ.Π.

Αθήνα, Οκτώβριος 2023

.....

ΠΑΝΑΓΙΩΤΗΣ ΧΑΤΖΗΓΙΑΝΝΑΚΗΣ

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

Copyright © Παναγιώτης Χατζηγιαννάκης, 2023

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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. Ερωτήματα που αφορούν τη χρήση της εργασίας γιακερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Η Επεξεργασία Φυσικής Γλώσσας είναι μία περιοχή της Τεχνητής Νοημοσύνης που προσελκύει όλο και περισσότερο επιστημονικό ενδιαφέρον και σταδιακά γίνεται μέρος της καθημερινής ζωής του ανθρώπου. Από τα πιο θεμελιώδη προβλήματα που αντιμετωπίζει είναι το πρόβλημα της κατηγοριοποίησης κειμένου, δηλαδή της κατάταξης εγγράφων κειμένου σε προκαθορισμένες ομάδες. Η υποπεριοχή του προβλήματος που μας απασχολεί στην εργασία αυτή είναι η κατηγοριοποίηση νομικών κειμένων. Από την πλευρά των δικηγόρων, η κατηγοριοποίηση νομικών κειμένων μπορεί να βοηθήσει στην γρήγορη αναγνώριση σχετικών νομικών προηγουμένων και καταστατικών, γλιτώνοντας έτσι πολύτιμο χρόνο στην νομική έρευνα. Από την πλευρά των πολιτών, καθιστά ευκολότερη την πρόσβαση σε νομικές πληροφορίες, δίνοντάς τους έτσι την δυνατότητα και το κίνητρο να κατανοήσουν τα δικαιώματά τους και τις υποχρεώσεις τους.

Το ακριβές αντικείμενο της διπλωματικής είναι η σύγκριση αλγορίθμων μηχανικής μάθησης στο πλαίσιο της Κατηγοριοποίησης Πολλαπλών Ετικετών Ελληνικών νομικών κειμένων. Στο πρόβλημα της Κατηγοριοποίησης κειμένου Πολλαπλών Ετικετών είσοδος είναι ένα κείμενο και έξοδος πολλαπλές, μη αμοιβαία αποκλειόμενες ετικέτες - κατηγορίες. Εκτελούμε ένα πείραμα σε κείμενα Ελληνικής νομοθεσίας όπου οι ετικέτες προέρχονται από τον θησαυρό Eurovoc και ένα πείραμα σε κείμενα νομολογίας όπου οι ετικέτες προέρχονται από τον Άρειο Πάγο. Οι αλγόριθμοι ταξινόμησης τους οποίους υλοποιούμε εκτείνονται από παραδοσιακή μηχανική μάθηση (Naïve Bayes, K-Nearest Neighbor, Decision Tree, Random Forest, Bagging) μέχρι και μοντέλα που βασίζονται στους μετασχηματιστές (BERT – Bidirectional Encoder Representations from Transformers). Ύστερα από την εκτέλεση των πειραμάτων, αξιολογούμε τις επιδόσεις των αλγορίθμων με βάση μετρικών. Το συμπέρασμα στο οποίο καταλήγουμε είναι ότι στην κατηγοριοποίηση νομοθεσίας την καλύτερη επίδοση έχουν οι αλγόριθμοι Naïve Bayes και BERT, ενώ στην κατηγοριοποίηση νομολογίας την καλύτερη επίδοση έχουν οι αλγόριθμοι BERT και K-Nearest Neighbor.

Λέξεις Κλειδιά: Νομικά Κείμενα, Κατηγοριοποίηση πολλαπλών ετικετών, μηχανική μάθηση, μετασχηματιστές

Abstract

Natural Language Processing is a field of Artificial Intelligence that keeps attracting more and more scientific interest and is gradually becoming a part of our everyday lives. One of the most fundamental problems it attempts to solve is text classification, which involves categorizing text documents into predefined categories. Our specific area of interest is legal document classification. For a lawyer, legal text classification can help quickly identify relevant precedents and statutes, saving valuable time in legal research. For a citizen, it can enable easier access to legal information, empowering them to understand their rights and obligations.

More precisely, the aim of this thesis is the comparison of machine learning algorithms in the task of Multi-Label Classification of Greek legal documents. In Multi-Label Text Classification, the input is a text document and the output consists of multiple, mutually non-exclusive labels. We perform an experiment on Greek legislation documents where the labels are from the Eurovoc theasaurus and another experiment on case laws where the labels are from Areios Pagos. We implement various classification algorithms ranging from traditional machine learning (Naïve Bayes, K-Nearest Neighbor, Decision Tree, Random Forest, Bagging) to state-of-the-art models based on transformers (BERT – Bidirectional Encoder Representations from Transformers). After conducting the experiments, we evaluate the performance of each algorithm with the use of metrics. We come to the conclusion that in the case of legislation the most efficient algorithms are Naïve Bayes and BERT whereas in the case of case laws the most efficient algorithms are BERT and K-Nearest Neighbor.

Keywords: legal documents, Multi-Label Classification, machine learning, transformers

Πίνακας περιεχομένων

1 Εισαγωγή 1

1.1 Εισαγωγή 1
1.2 Κίνητρο και συνεισφορά διπλωματικής 2
1.3 Οργάνωση κειμένου 3

2 Θεωρητικό υπόβαθρο 4

2.1 Προεπεξεργασία (Preprocessing) 4
2.1.1 TF-IDF Vectorization 4
2.2 Μετασχηματισμός Προβλήματος 6
2.2.1 Μετασχηματισμός Label Powerset 6
2.3 Ταξινόμηση (Classification) 6
2.3.1 Naïve Bayes 7
2.3.2 K-Nearest Neighbor (kNN) 9
2.3.3 Decision Tree 9
2.3.4 Random Forest 12
2.3.5 Bagging 13
2.3.6 BERT 14

3 Σχετικές εργασίες 17

3.1 Κατηγοριοποίηση κειμένων 17
3.2 Κατηγοριοποίηση νομικών κειμένων 18
3.3 Κατηγοριοποίηση Πολλαπλών Ετικετών νομικών κειμένων 19
3.4 Εργαλεία κατηγοριοποίησης μέσω του θησαυρού Eurovoc 20
3.4.1 Jex JRC Eurovoc Indexer 20
3.4.2 PyEuroVoc 21

4 Αυτόματη Κατηγοριοποίηση Νομοθεσίας 22

4.1 Σώμα Κειμένων – Επισημασμένο σύνολο δεδομένων αληθείας 22
4.2 Προεπεξεργασία 27
4.3 Αλγόριθμοι 27
4.3.1 Naïve Bayes 28
4.3.2 K-Nearest Neighbor (kNN) 28
4.3.3 Decision Tree 29
4.3.4 Random Forest 29
4.3.5 Bagging 29
4.3.6 BERT 30
4.4 Παράμετροι Αξιολόγησης 31
4.5 Αποτελέσματα 32
4.6 Σύνοψη συμπερασμάτων αξιολόγησης 37

5 Αυτόματη Κατηγοριοποίηση Νομολογίας 38

5.1 Σώμα Κειμένων – Επισημασμένο σύνολο δεδομένων αληθείας 38
5.2 Προεπεξεργασία 40
5.3 Αλγόριθμοι 40
5.3.1 Naïve Bayes *41*
5.3.2 K-Nearest Neighbor (kNN) *42*
5.3.3 Decision Tree *42*
5.3.4 Random Forest *42*
5.3.5 Bagging *43*
5.3.6 BERT *43*
5.4 Παράμετροι Αξιολόγησης 44
5.5 Αποτελέσματα 44
5.6 Σύνοψη συμπερασμάτων αξιολόγησης 46

6 Επίλογος 47

6.1 Σύνοψη και συμπεράσματα 47
6.2 Μελλοντικές επεκτάσεις 48

7 Βιβλιογραφία 50

1

Εισαγωγή

1.1 Εισαγωγή

Η τεχνητή νοημοσύνη (Artificial Intelligence - AI) αλλάζει τον τρόπο με τον οποίο ο νομικός τομέας λειτουργεί. Αν και η προσαρμογή της στον τομέα αυτό είναι ακόμη κάτι το καινούργιο, οι δικηγόροι σήμερα έχουν μία μεγάλη ποικιλία ευφυών εργαλείων στη διάθεσή τους. Μία απο τις πιο σημαντικές εκφάνσεις της AI για αυτόν τον σκοπό είναι η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP). Η NLP είναι ένα υποσύνολο της AI που επεξεργάζεται φυσική ανθρώπινη γλώσσα, είτε σε μορφή κειμένου είτε σε μορφή ήχου – φωνής.

Αρχικά, συμβάλλει στην επιτάχυνση της νομικής έρευνας, της οποίας η διεξαγωγή σε βάθος είναι απαραίτητη για όλες τις νομικές διαδικασίες αλλά είναι και η αιτία που διαρκούν μεγάλο χρονικό διάστημα. Οι νομικές μηχανές αναζήτησης που χρησιμοποιούν NLP μπορούν να μεταφράσουν την απλή γλώσσα σε νομική ορολογία, καθιστώντας ευκολότερη την αναζήτηση σχετικών εγγράφων και υποθέσεων. Περισσότερο προηγμένα NLP προγράμματα μπορούν να κάνουν αναζήτηση για έννοιες, όχι απλώς συγκεκριμένες λέξεις – κλειδιά, βοηθώντας έτσι τους δικηγόρους να βρουν αυτό που θέλουν γρηγορότερα. Επίσης μπορούν να αναλύσουν μία μελέτη περίπτωσης ή ένα έγγραφο και να προτείνουν άλλες παρόμοιες υποθέσεις για εξέταση. Παράλληλα, η NLP μπορεί να βοηθήσει τους δικηγόρους στην σύνταξη εγγράφων αποφεύγοντας σφάλματα που σχετίζονται με ασάφειες σε νομικά έγγραφα που μπορούν να οδηγήσουν σε λάθος ερμηνείες. Υπάρχει επίσης η δυνατότητα προγράμματα να επεξεργάζονται έγγραφα σε πολλές γλώσσες, να δημιουργούν αυτόματα πρότυπα βασισμένα σε έναν δοθέντα νόμο, συμφωνία ή εταιρική πολιτική.

Επιπροσθέτως, υπάρχει μεγάλη συνεισφορά στον αυτοματισμό εργασιών, αν και η NLP δεν είναι από μόνη της τεχνολογία αυτοματισμών. Το πιο δημοφιλές παράδειγμα αυτής της εφαρμογής είναι τα chatbots, που παρέχουν στήριξη ακόμη και τις ώρες που οι δικηγόροι δεν εργάζονται. Από τις πιο πρόσφατες αλλά πολύ σημαντικές εφαρμογές NLP στον νομικό τομέα είναι και η δημιουργία μοντέλων πρόβλεψης για μία δίκη μετά από ανάλυση περιπτώσεων του παρελθόντος. Καθ'αυτό τον τρόπο, έχουμε μοντέλα NLP που μπορούν να προβλέψουν με μεγάλη πιθανότητα την τελική απόφαση ενός δικαστηρίου που εφόσον βασίζονται στην μηχανική μάθηση (machine learning), όσο περισσότερο χρησιμοποιούνται στον νομικό χώρο τόσο πιο πολύ αυξάνεται η ακρίβειά τους όσον αφορά τις επιδόσεις.
Από την άλλη πλευρά, το πεδίο της κατηγοριοποίησης νομικών κειμένων με χρήση NLP εξακολουθεί να στερείται το βάθος της έρευνας και ανάπτυξης που απαιτείται για ολοκληρωμένες και αποτελεσματικές λύσεις. Αν και έχουν σημειωθεί αξιοσημείωτα βήματα στην εφαρμογή τεχνικών NLP στην ανάλυση νομικών κειμένων, η ύπαρξη μεγάλων δημοσίως διαθέσιμων datasets είναι σπάνια, γεγονός που εμποδίζει την εκπαίδευση ισχυρών μοντέλων. Η περίπλοκη φύση της νομικής γλώσσας προσθέτει επίπεδα πολυπλοκότητας που απαιτούν πιο διαφοροποιημένες προσεγγίσεις. Ωστόσο, καθώς οι δυνατότητες για την αυτοματοποίηση χρονοβόρων εργασιών στο νομικό επάγγελμα είναι τεράστιες, υπάρχει επιτακτική ανάγκη για προώθηση αυτού του τομέα.

1.2 Κίνητρο και συνεισφορά διπλωματικής

Υπάρχουν χιλιάδες νομικά κείμενα τα οποία αν μπορούσαν να κατηγοριοποιηθούν με έναν δομημένο τρόπο, τα οφέλη θα ήταν πολλά τόσο για φοιτητές Νομικής όσο και για δικηγόρους. Για τους φοιτητές, η ταξινόμηση νομικών κειμένων με καλά ορισμένες κατηγορίες παίζει σημαντικό ρόλο στην εκπαίδευσή τους και στην κατανόηση νομικών θεμάτων. Παρέχει έναν οργανωμένο τρόπο μελέτης νομικού υλικού. Για τους δικηγόρους, διευκολύνει την αναζήτηση συγκεκριμένων νομικών εγγράφων και με αυτόν τον τρόπο εξοικονομείται σημαντικός χρόνος στην νομική έρευνα. Επίσης προσδίδει ένα επίπεδο συνέπειας και κανονικοποίησης αφού δημιουργείται ένας κοινός κώδικας επικοινωνίας στην νομική κοινότητα.
Το αντικείμενο της διπλωματικής είναι η σύγκριση διαφόρων αλγορίθμων μηχανικής μάθησης στο πρόβλημα της αυτόματης κατηγοριοποίησης κειμένων Ελληνικής νομοθεσίας και νομολογίας. Στα κείμενα νομοθεσίας οι κατηγορίες προέρχονται από τον θησαυρό Eurovoc ενώ στα κείμενα νομολογίας οι κατηγορίες προέρχονται από τον Άρειο Πάγο. Έχοντας ως χαρακτηριστικό ότι σε κάθε νομικό έγγραφο ο εκάστοτε αλγόριθμος πρέπει να αναθέτει μία η περισσότερες κατηγορίες, το πρόβλημα κατατάσσεται στην κατηγορία των προβλημάτων Κατηγοριοποίησης Πολλαπλών Ετικετών (Multi Label Classification - MLC). Ένα πρόβλημα αυτού του είδους στην Ελληνική γλώσσα είναι δύσκολο να αντιμετωπιστεί διότι εμφανίζονται προκλήσεις για μοντέλα τα οποία έχουν σχεδιαστεί αποκλειστικά για Latin-based γλώσσες. Οι προκλήσεις αυτές αφορούν αναγνώριση λεξικών μονάδων (tokenization), κωδικοποίηση χαρακτήρων (character encoding) και font support. Επίσης, σε σύγκριση με τα Αγγλικά ή τα Ισπανικά, η Ελληνική γλώσσα έχει περιορισμένα δεδομένα εκπαίδευσης, το οποίο καθιστά δύσκολο να δημιουργηθούν μοντέλα μηχανικής μάθησης υψηλής ακρίβειας.

Η συνεισφορά της διπλωματικής συνοψίζεται ως εξής:

1. Μελετήσαμε συστήματα και αλγορίθμους κατηγοριοποίησης κειμένων νομικής φύσεως και μη από τη σχετική βιβλιογραφία
2. Υλοποιήσαμε έξι αλγορίθμους ταξινόμησης (Naïve Bayes, K-Nearest Neighbor, Decision Tree, Random Forest, Bagging, και BERT - Bidirectional Encoder Representations from Transformers)
3. Αξιολογήσαμε την επίδοση των αλγορίθμων βάσει μετρικών σε κείμενα Ελληνικής νομοθεσίας και νομολογίας
4. Τα ευρήματά μας είναι τα εξής: Όσον αφορά την κατηγοριοποίηση νομοθεσίας την καλύτερη επίδοση έχουν οι αλγόριθμοι Naïve Bayes και BERT, ενώ όσον αφορά την κατηγοριοποίηση νομολογίας την καλύτερη επίδοση έχουν οι αλγόριθμοι BERT και K-Nearest Neighbor (kNN)

1.3 Οργάνωση κειμένου

Η δομή της διπλωματικής είναι η εξής:

Στο κεφάλαιο 2 αναλύουμε την απαραίτητη θεωρία όσον αφορά το υπόβαθρο μηχανικής μάθησης. Στη συνέχεια, στο κεφάλαιο 3 παρουσιάζουμε εργασίες σχετικές με το θέμα. Ακολούθως, στα κεφάλαια 4 και 5 παρουσιάζουμε την μεθοδολογία την οποία ακολουθήσαμε για καθένα από τα πειράματά μας μαζί με την αξιολόγηση των αλγορίθμων μηχανικής μάθησης που χρησιμοποιήσαμε. Τέλος, στο κεφάλαιο 6 συνοψίζουμε τα συμπεράσματά μας και αναφέρουμε πιθανές μελλοντικές επεκτάσεις της διπλωματικής εργασίας.

2

Θεωρητικό υπόβαθρο

Στο κεφάλαιο αυτό αναπτύσσεται το θεωρητικό υπόβαθρο που θα βοηθήσει στην κατανόηση των βασικών εννοιών που λαμβάνουν μέρος στο παρόν έργο. Αρχικά αναλύουμε μεθόδους προεπεξεργασίας κειμένου, στη συνέχεια περιγράφουμε τεχνικές μετασχηματισμού προβλήματος και τέλος εξετάζουμε αλγορίθμους μηχανικής μάθησης.

2.1 Προεπεξεργασία (Preprocessing)

Στο επίπεδο της προεπεξεργασίας κειμένου, έχουν προταθεί διάφορες τεχνικές. Όσον αφορά την συντακτική αναπαράσταση λέξεων, μία από αυτές είναι η n-gram [1], ένα σύνολο n-λέξεων που εμφανίζονται με συγκεκριμένη σειρά σε ένα κείμενο. Όσον αφορά την περίπτωση λέξεων με βάρος κυριαρχούν τα μοντέλα Bag-of-Words (BoW) [2], μία απλουστευμένη αναπαράσταση ενός εγγράφου κειμένου από επιλεγμένα κομμάτια του με βάση συγκεκριμένα κριτήρια, όπως πχ η συχνότητα λέξεων, και η Term Frequency - Inverse Document Frequency (TF-IDF) [3]. Στο πλαίσιο των διανυσματικών παραστάσεων λέξεων (word embeddings) σημαντική είναι η τεχνική Word2Vec [4] που χρησιμοποιεί νευρωνικά δίκτυα με 2 κρυμμένα layers, η τεχνική GloVe (Global Vectors for Word Representation) [5] που δεν χρησιμοποιεί νευρωνικά δίκτυα αλλά εκμεταλλεύεται το γεγονός ότι λέξεις με παρόμοια σημασία τείνουν να συνυπάρχουν σε παρόμοια συμφραζόμενα, και η FastText [6], μία μέθοδος που δημιούργησε το Facebook AI Research lab και αποτελεί επέκταση της Word2Vec. Τέλος, μία ακόμη μέθοδος διανυσματικής αναπαράστασης λέξεων είναι η context2vec [7] που χρησιμοποιεί αμφίδρομο δίκτυο long short-term memory (LSTM).
Από αυτές, εκείνη που θα μας απασχολήσει περισσότερο και αναλύουμε παρακάτω είναι η TF-IDF.

2.1.1 TF-IDF Vectorization

Η **TF-IDF (Term Frequency – Inverse Document Frequency)** είναι μία στατιστική μετρική που χρησιμοποιείται στην ανάκτηση πληροφοριών (Information Retrieval) και εκτιμά πόσο σχετική είναι μία λέξη για ένα έγγραφο σε μία συλλογή από έγγραφα. Αυτή η εκτίμηση

προκύπτει από το γινόμενο δύο ποσοτήτων: τον αριθμό των φορών που μια λέξη εμφανίζεται σε ένα κείμενο (term frequency) και το πόσο συχνή ή σπάνια μία λέξη είναι σε ολόκληρο το σύνολο των εγγράφων (inverse document frequency). Η δεύτερη ποσότητα υπολογίζεται παίρνοντας τον συνολικό αριθμό των εγγράφων, διαιρώντας τον με τον αριθμό των εγγράφων που περιέχουν την λέξη, και υπολογίζοντας τον λογάριθμο του πηλίκου αυτού. Σε μαθηματικούς όρους, η TF-IDF για μια λέξη t στο έγγραφο d από το σύνολο εγγράφων D (πλήθος εγγράφων N) δίνεται από την σχέση:

$$tf\ idf(t,d,D) = tf(t,d)\ .idf(t,D)$$

Εξίσωση 2.1: TF-IDF

Όπου:

$$tf(t,d) = \log\left(1 + freq(t,d)\right)$$

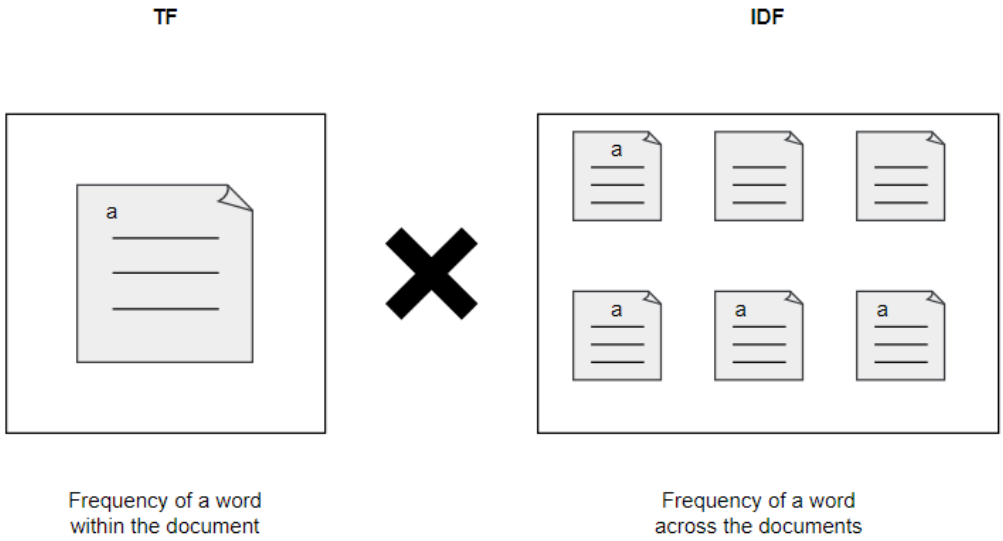
Εξίσωση 2.2: τύπος Term Frequency

και

$$idf(t,D) = \log\left(\frac{N}{count(d \in D : t \in d)}\right)$$

Εξίσωση 2.3: τύπος Inverse Document Frequency

Στο πλαίσιο της μηχανικής μάθησης με φυσική γλώσσα, το βασικό εμπόδιο είναι ότι οι αλγόριθμοι διαχειρίζονται αριθμητικά δεδομένα, ενώ η φυσική γλώσσα είναι κείμενο. Η TF-IDF λοιπόν, μετατρέπει το κείμενο σε αριθμούς - διανύσματα (Text Vectorization) ώστε να δοθεί έγκυρα σαν είσοδος σε διάφορους αλγορίθμους ταξινόμησης.



Εικόνα 2.1: TF-IDF

2.2 Μετασχηματισμός Προβλήματος

Με τον όρο του μετασχηματισμού προβλήματος αναφερόμαστε στην μετατροπή ενός προβλήματος πολλαπλών ετικετών (Multi-Label) σε ένα πρόβλημα μίας ετικέτας (Single-Label). Για τον σκοπό αυτό, έχουν προταθεί τρεις μέθοδοι - μετασχηματισμοί:

i) Η μέθοδος Binary Relevance [8], που χειρίζεται κάθε ετικέτα ανεξάρτητα, και έπειτα οι πολλαπλές ετικέτες διαχωρίζονται ως πρόβλημα ταξινόμησης μίας κλάσης. Με αυτή την μέθοδο, υπάρχει ο κίνδυνος να χαθούν οι συσχετίσεις μεταξύ των ετικετών.

ii) Η μέθοδος Classifier Chains [9], όπου χρησιμοποιούνται πολλαπλοί ταξινομητές σε μία αλυσίδα. Ο πρώτος ταξινομητής χτίζεται χρησιμοποιώντας τα δεδομένα εισόδου. Οι επόμενοι ταξινομητές εκπαιδεύονται με βάση τις εξόδους των αμέσως προηγουμένων δημιουργώντας, έτσι, μία ακολουθιακή διαδικασία. Με αυτή την μέθοδο λύνεται το πρόβλημα των συσχετίσεων ετικετών της προηγούμενης μεθόδου.

iii) Η μέθοδος Label Powerset [10], η οποία θα μας απασχολήσει περισσότερο και γι'αυτόν τον λόγο την αναλύουμε παρακάτω.

2.2.1 Μετασχηματισμός Label Powerset

Ο μετασχηματισμός **Label Powerset**, που μπορεί να χρησιμοποιηθεί σαν τεχνική σε συνδυασμό με διάφορους αλγορίθμους ταξινόμησης, μπορεί να μετασχηματίσει ένα πρόβλημα πολλαπλών ετικετών (Multi-Label Classification) σε ενα πρόβλημα πολλαπλών κλάσεων (Multi-Class Classification). Ο τρόπος με τον οποίο επιτυγχάνεται αυτό είναι δημιουργώντας έναν δυαδικό ταξινομητή (binary classifier) για κάθε συνδυασμό ετικετών στο training set. Για παράδειγμα, ας θεωρήσουμε ένα πρόβλημα με 3 ετικέτες: Α, Β και Γ. Η αναπαράσταση με Label Powerset αυτού του προβλήματος είναι ένα πρόβλημα πολλαπλών κλάσεων με $2^3 = 8$ κλάσεις που καθορίζονται από την παρουσία ή όχι καθεμίας ετικέτας. Έτσι, συμβολίζοντας με 1 την παρουσία και με 0 την απουσία μίας ετικέτας, οι κλάσεις θα είναι οι [0 0 0], [0 0 1], [0 1 0], [0 1 1], [1 0 0], [1 0 1], [1 1 0], [1 1 1], όπου για παράδειγμα η κλάση [1 0 1] δηλώνει την παρουσία των κλάσεων Α και Γ, αλλά όχι της Β.

2.3 Ταξινόμηση (Classification)

Στο επίπεδο της ταξινόμησης κειμένου, στη βιβλιογραφία συναντάμε πολλούς και διαφορετικούς αλγορίθμους. Ξεκινώντας από παραδοσιακούς αλγορίθμους κατηγοριοποίησης, ένας από αυτούς είναι ο αλγόριθμος Rocchio [11] που βασίζεται σε μία μέθοδο ανατροφοδότησης σχετικότητας (relevance feedback). Προχωρώντας σε μεθόδους συνόλου (ensemble-based), υπάρχουν αλγόριθμοι όπως ο boosting (τόσο οι αρχικές εκδοχές του αλλά και ο πιο σύγχρονος αλγόριθμος AdaBoost [12]) και ο bagging [13]. Στην στατιστική και στη μηχανική μάθηση, οι μέθοδοι συνόλου είναι μέθοδοι που χρησιμοποιούν πολλαπλούς αλγορίθμους ώστε οι επιδόσεις στις προβλέψεις να είναι καλύτερες από όπως θα ήταν με μόνο

έναν αλγόριθμο από αυτούς που τις απαρτίζουν. Από τους πιο απλούς αλγορίθμους εποπτευόμενης μάθησης είναι επίσης ο αλγόριθμος Logistic Regression [14], ενώ ένας αλγόριθμος που δεν απαιτεί πολλούς υπολογιστικούς πόρους είναι ο Naïve Bayes [15]. Επιπρόσθετα, προτείνονται αλγόριθμοι όπως ο K-Nearest Neighbor (kNN) [16] που είναι μη παραμετρικός, ο αλγόριθμος SVM (Support Vector Machine) [17] που αντιστοιχεί δεδομένα σε έναν πολυδιάστατο χώρο, αλλά και ταξινομητές βασιζόμενοι σε δένδρα (Decision Tree [18], Random Forest [19]). Πιο σύγχρονες τεχνολογίες αποτελούν αλγόριθμοι βαθιάς μάθησης – deep learning που χρησιμοποιούν νευρωνικά δίκτυα (Convolutional Neural Network – CNN [20], Recurrent Neural Network – RNN [21], Long Short Term Memory Network - LSTM [22], Deep Belief Network – DBN [23]) ή ακόμα και αλγόριθμοι που βασίζονται σε μετασχηματιστές (transformers) [24]. Οι αλγόριθμοι που θα μας απασχολήσουν περισσότερο και θα αναλύσουμε είναι οι εξής: Naïve Bayes, K-Nearest Neighbor (kNN), Decision Tree, Random Forest, Bagging, και από την κατηγορία των μετασχηματιστών το μοντέλο BERT (Bidirectional Encoder Representations from Transformers) [25].

2.3.1 Naïve Bayes

Ο **Naïve Bayes** είναι ένας αλγόριθμος ταξινόμησης που βασίζεται στο θεώρημα Bayes. Ο χαρακτηρισμός naïve, που σημαίνει αφελής στα Ελληνικά, δίδεται στον αλγόριθμο διότι υποθέτει ότι τα χαρακτηριστικά είναι ανεξάρτητα το ένα από το άλλο, υπόθεση που μπορεί να μην είναι αληθής σε προβλήματα του πραγματικού κόσμου. Ωστόσο, παρά την απλουστευτική αυτή υπόθεση, ο αλγόριθμος αποτελεί δημοφιλή επιλογή για πολλά προβλήματα κατηγοριοποίησης λόγω της απλότητας και της υψηλής ακρίβειάς του. Οι τρεις τύποι ταξινομητών Naïve Bayes που χρησιμοποιούνται κατά κύριο λόγο είναι οι εξής:

- Gaussian Naïve Bayes
- Multinomial Naïve Bayes
- Bernoulli Naïve Bayes

Η υποπερίπτωση που θα αναλύσουμε είναι ο **Multinomial Naïve Bayes**, ένας αλγόριθμος που χρησιμοποιείται για κατηγοριοποίηση εγγράφων ή κειμένου σε πολλαπλές κλάσεις. Παίζει πρωτεύοντα ρόλο σε εφαρμογές NLP που έχουν να κάνουν με ανίχνευση ανεπιθύμητων μηνυμάτων (spam detection) και ανάλυση συναισθημάτων (sentiment analysis). Ο Multinomial Naïve Bayes είναι ένας πιθανολογικός αλγόριθμος που υποθέτει την ανεξαρτησία των χαρακτηριστικών (δηλαδή των λέξεων) στα δεδομένα εισόδου. Χρησιμοποιεί το θεώρημα Bayes για να υπολογίσει την πιθανότητα κάθε κλάσης δεδομένων των χαρακτηριστικών εισόδου. Το **θεώρημα Bayes** ορίζει ότι:

$$P(y|x) = \frac{P(x|y) * P(y)}{P(x)}$$

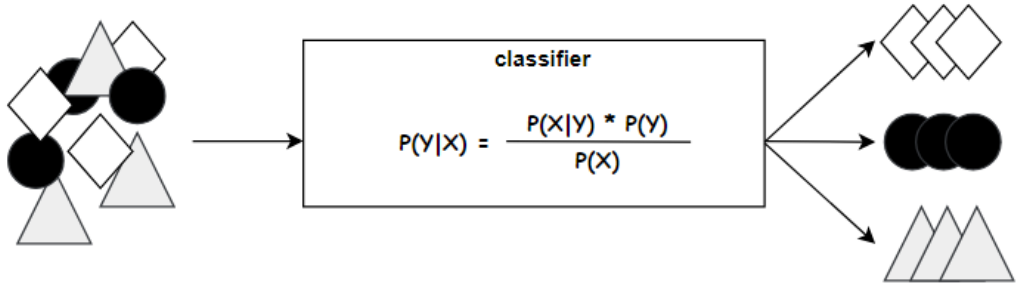
Εξίσωση 2.4: Θεώρημα Bayes

Όπου:

- $P(y|x)$ είναι η πιθανότητα της κλάσης y δοθείσας της εισόδου x

- $P(x|y)$ είναι η πιθανότητα της εισόδου x δοθείσας της κλάσης y
- $P(y)$ είναι η εκ των προτέρων (prior) πιθανότητα της κλάσης y
- $P(x)$ είναι η πιθανότητα της εισόδου x

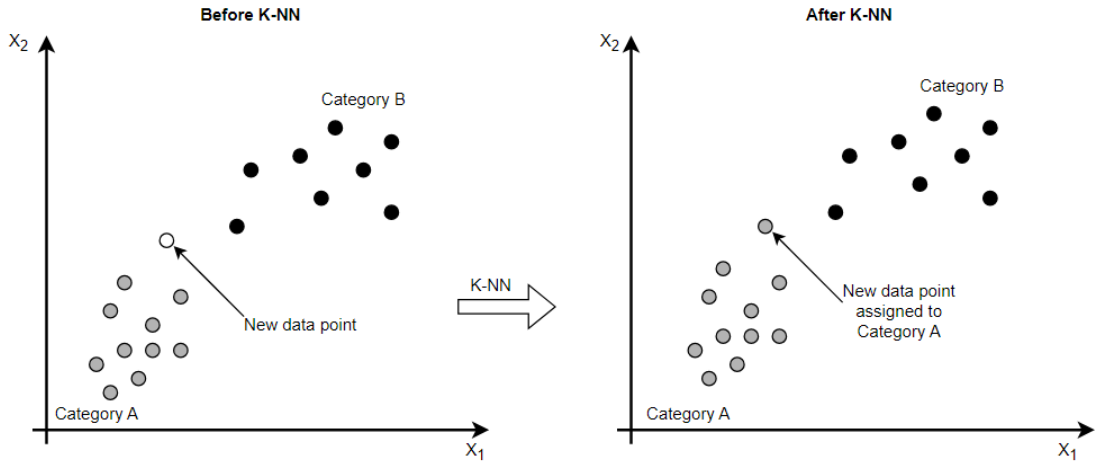
Στην κατηγοριοποίηση κειμένου, η είσοδος x είναι ένα έγγραφο ή κείμενο και οι κλάσεις y είναι οι διαφορετικές κατηγορίες ή ετικέτες στις οποίες επιθυμούμε να κατηγοριοποιήσουμε το κείμενο. Ο στόχος είναι να βρεθεί η κλάση y που έχει την υψηλότερη πιθανότητα δοθείσας της εισόδου x. Ο αλγόριθμος χρησιμοποιεί το μοντέλο bag-of-words για την αναπαράσταση του κειμένου εισόδου σαν διάνυσμα συχνοτήτων λέξεων. Κάθε λέξη στο κείμενο εισόδου αντιμετωπίζεται σαν χαρακτηριστικό, και η συχνότητά της στο έγγραφο είναι η τιμή αυτού του χαρακτηριστικού. Για τον υπολογισμό της πιθανότητας κάθε κλάσης δοθέντων των χαρακτηριστικών εισόδου, ο αλγόριθμος πρώτα υπολογίζει την εκ των προτέρων πιθανότητα κάθε κλάσης. Αυτή είναι η πιθανότητα κάθε κλάσης με βάση την συχνότητα των ετικετών των κλάσεων στα δεδομένα εκπαίδευσης. Έπειτα, ο αλγόριθμος υπολογίζει την πιθανοφάνεια κάθε χαρακτηριστικού δοθείσας κάθε κλάσης. Αυτή είναι η πιθανότητα κάθε λέξης που εμφανίζεται στα δεδομένα εκπαίδευσης για κάθε κλάση. Τέλος, ο αλγόριθμος συνδυάζει την εκ των προτέρων πιθανότητα και την πιθανοφάνεια για να υπολογίσει την πιθανότητα κάθε κλάσης δοθέντων των χαρακτηριστικών εισόδου με χρήση του θεωρήματος Bayes.



Εικόνα 2.2: Ταξινομητής Naïve Bayes

2.3.2 K-Nearest Neighbor (kNN)

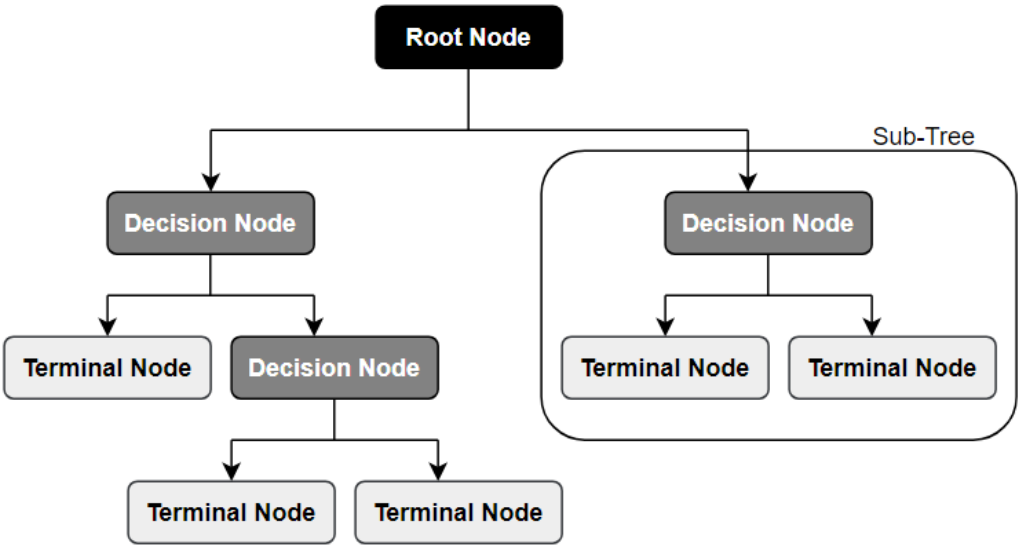
Ο αλγόριθμος **K-Nearest Neighbor (kNN)** είναι ένας από τους πιο απλούς αλγορίθμους εποπτευόμενης μάθησης (supervised learning). Είναι μη παραμετρικός (non-parametric), υπό την έννοια ότι δεν κάνει κάποια υπόθεση σε υποκείμενα δεδομένα (underlying data), και πολλές φορές αναφέρεται ως "Lazy-Learner Algorithm" διότι δεν μαθαίνει από το training set άμεσα, αλλά αποθηκεύει το dataset και την στιγμή της κατηγοριοποίησης εκτελεί κάποιες ενέργειες σε αυτό. Ο τρόπος με τον οποίο δουλεύει είναι ο εξής:
Ας υποθέσουμε ότι έχουμε 2 κατηγορίες, την κατηγορία A και την κατηγορία B, και οτι έχουμε ένα νέο σημείο δεδομένων x1 για το οποίο θέλουμε να αποφανθούμε σε ποια από τις 2 κατηγορίες ανήκει. Αρχικά πρέπει να γίνει η επιλογή του αριθμού k των γειτόνων (neighbors). Ύστερα υπολογίζεται η Ευκλείδεια απόσταση μεταξύ του x1 και όλων των ήδη υπάρχοντων training data points και επιλέγονται οι k πιο κοντινοί γείτονες μέσω της απόστασης αυτής. Το επόμενο βήμα είναι ανάμεσα σε αυτούς τους k γείτονες, να μετρηθεί ο αριθμός των σημείων δεδομένων σε κάθε κατηγορία και τέλος ο αλγόριθμος ολοκληρώνεται αναθέτοντας το σημείο στην κατηγορία με τον μεγαλύτερο αριθμό γειτόνων.



Εικόνα 2.3: Αλγόριθμος kNN – K-Nearest Neighbor

2.3.3 Decision Tree

Ένα **δένδρο απόφασης** είναι μία δενδρική δομή όμοια με ένα διάγραμμα ροής, όπου ένας εσωτερικός κόμβος αναπαριστά ένα χαρακτηριστικό, ένα κλαδί αναπαριστά έναν κανόνα απόφασης, και κάθε κόμβος – φύλλο αναπαριστά το αποτέλεσμα. Ο κόμβος που βρίσκεται στην κορυφή σε ένα δένδρο απόφασης είναι γνωστός ως κόμβος – ρίζα (root node). Μαθαίνει να διαχωρίζει το δένδρο με αναδρομικό τρόπο με βάση την τιμή του χαρακτηριστικού.



Εικόνα 2.4: Δένδρο Απόφασης

Οι **Decision Tree** ταξινομητές χρησιμοποιούνται σε πολλά και διαφορετικά προβλήματα κατηγοριοποίησης. Ανήκουν σε εκείνο τον τύπο αλγορίθμων μηχανικής μάθησης που θεωρούνται white box, διότι μοιράζονται την εσωτερική λογική λήψης αποφάσεων που δεν είναι διαθέσιμη σε αλγορίθμους τύπου black box όπως τα νευρωνικά δίκτυα. Σε σύγκριση με τα νευρωνικά δίκτυα, ο χρόνος εκπαίδευσης ενός decision tree ταξινομητή είναι πιο μικρός. Η χρονική πολυπλοκότητα του είναι μία συνάρτηση του αριθμού των εγγραφών (records) και των χαρακτηριστικών στα δεδομένα που δίνονται ενώ πρόκειται για μη παραμετρική μέθοδο που δεν εξαρτάται από υποθέσεις κατανομής πιθανοτήτων.
Τα βήματα που ακολουθεί κάθε αλγόριθμος δένδρων απόφασης είναι τα εξής:

1. Διάλεξε το καλύτερο χαρακτηριστικό (attribute) κάνοντας χρήση των Μέτρων Επιλογής Χαρακτηριστικού (Attribute Selection Measures - ASM) για να γίνει διαχωρισμός των εγγραφών
2. Κάνε αυτό το χαρακτηριστικό έναν κόμβο απόφασης και χώρισε το dataset σε μικρότερα υποσύνολα
3. Ξεκίνα το χτίσιμο των δένδρων επαναλαμβάνοντας αυτή την διαδικασία αναδρομικά για κάθε παιδί έως ότου μία από τις παρακάτω συνθήκες να ικανοποιείται:
    - Όλες οι πλειάδες ανήκουν στην ίδια τιμή χαρακτηριστικού
    - Δεν υπάρχουν άλλα εναπομείναντα χαρακτηριστικά
    - Δεν υπάρχουν άλλα στιγμιότυπα

Τα **Attribute Selection Measures (ASM)** είναι μία ευριστική προσέγγιση για την επιλογή του κριτηρίου διαχωρισμού που διαμερίζει τα δεδομένα με τον καλύτερο πιθανό τρόπο. Τα πιο δημοφιλή ASM είναι τα εξής: **Information Gain**, **Gain Ratio**, και **Gini Index**. Θα αναλύσουμε κάθε ένα από αυτά χωριστά.

Information Gain: Ο Claude Shannon εφηύρε την έννοια της εντροπίας, που μετράει το ποσό της "ακαθαρσίας" (impurity) του συνόλου εισόδου. Στη φυσική και τα μαθηματικά, η εντροπία αναφέρεται ως η τυχαιότητα ενός συστήματος. Στην πληροφορική, αναφέρεται στην "ακαθαρσία" σε ένα σύνολο από παραδείγματα. Η Information Gain είναι η μείωση της εντροπίας. Υπολογίζει την διαφορά μεταξύ της εντροπίας πριν από τον διαχωρισμό και της μέσης εντροπίας μετά τον διαχωρισμό του dataset με βάση τις τιμές χαρακτηριστικών που δίνονται. Ο αλγόριθμος δένδρων απόφασης ID3 (Iterative Dichotomizer) χρησιμοποιεί την Information Gain. Οι εξισώσεις που ισχύουν είναι οι εξής:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2 p_i$$

Εξίσωση 2.5: μέσο ποσό πληροφορίας

Όπου p_i είναι η πιθανότητα μία αυθαίρετη πλειάδα στο D να ανήκει στην κλάση C_i

$$Info_A(D) = \sum_{j=1}^{V} \frac{|D_j|}{|D|} * Info(D_j)$$

Εξίσωση 2.6: αναμενόμενη πληροφορία

$$Gain(A) = Info(D) - Info_A(D)$$

Εξίσωση 2.7: κέρδος πληροφορίας

Όπου:

- $Info(D)$ είναι το μέσο ποσό πληροφορίας που χρειάζεται για να αναγνωρισθεί η ετικέτα κλάσης μιας πλειάδας στο D
- Το $\frac{|D_j|}{|D|}$ δρα ως το βάρος της διαμέρισης με αριθμό j
- $Info_A(D)$ είναι η αναμενόμενη πληροφορία που απαιτείται για την ταξινόμηση μιας πλειάδας από το D με βάση την διαμέριση από το A

Το χαρακτηριστικό A με την μεγαλύτερη Information gain, $Gain(A)$, επιλέγεται ως το χαρακτηριστικό διαχωρισμού στον κόμβο N().

Gain Ratio: Η Information gain προτιμά το χαρακτηριστικό με μεγάλο αριθμό διακριτών τιμών. Για παράδειγμα, ας θεωρήσουμε ένα χαρακτηριστικό με ένα μοναδικό αναγνωριστικό, όπως π.χ το customer_ID, που έχει μηδενική $Info(D)$ λόγω καθαρής διαμέρισης. Αυτό μεγιστοποιεί την Information gain και δημιουργεί ανωφελή διαμέριση.
Ο αλγόριθμος C4.5, ο οποίος αποτελεί βελτιωμένη έκδοση του ID3, χρησιμοποιεί μία επέκταση της Information gain γνωστή και ως Gain Ratio. Το Gain Ratio αντιμετωπίζει το πρόβλημα που αναφέραμε παραπάνω κανονικοποιώντας το κέρδος πληροφορίας με χρήση του Split Info.

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} * \log_2(\frac{|D_j|}{|D|})$$

Εξίσωση 2.8: Split Info

Όπου:

- Το $\frac{|D_j|}{|D|}$ δρα ως το βάρος της διαμέρισης με αριθμό j
- Το v είναι ο αριθμός των διακριτών τιμών στο χαρακτηριστικό A

Το Gain Ratio μπορεί να οριστεί ως εξής:

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo_A(D)}$$

Εξίσωση 2.9: Gain Ratio

Το χαρακτηριστικό με το μεγαλύτερο Gain Ratio επιλέγεται ως το χαρακτηριστικό διαχωρισμού.

Gini Index: Είναι ένα μέτρο της καθαρότητας ή της "ακαθαρσίας" όταν δημιουργείται ένα δένδρο απόφασης. Υπολογίζεται αφαιρώντας το άθροισμα των τετραγώνων των πιθανοτήτων κάθε κλάσης από την μονάδα. Είναι το ίδιο με την εντροπία αλλά υπολογίζεται γρηγορότερα σε σχέση με την εντροπία. Ο αλγόριθμος δένδρων απόφασης CART (Classification and Regression Tree) χρησιμοποιεί το Gini Index σαν Attribute Selection Measure για την επιλογή του καλύτερου χαρακτηριστικού διαχωρισμού. Το χαρακτηριστικό με το χαμηλότερο Gini Index χρησιμοποιείται ως το καλύτερο χαρακτηριστικό για διαχωρισμό. Η εξίσωση που ισχύει είναι:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

Εξίσωση 2.10: Gini

Όπου p_i είναι η πιθανότητα μία πλειάδα στο D να ανήκει στην κλάση C_i.

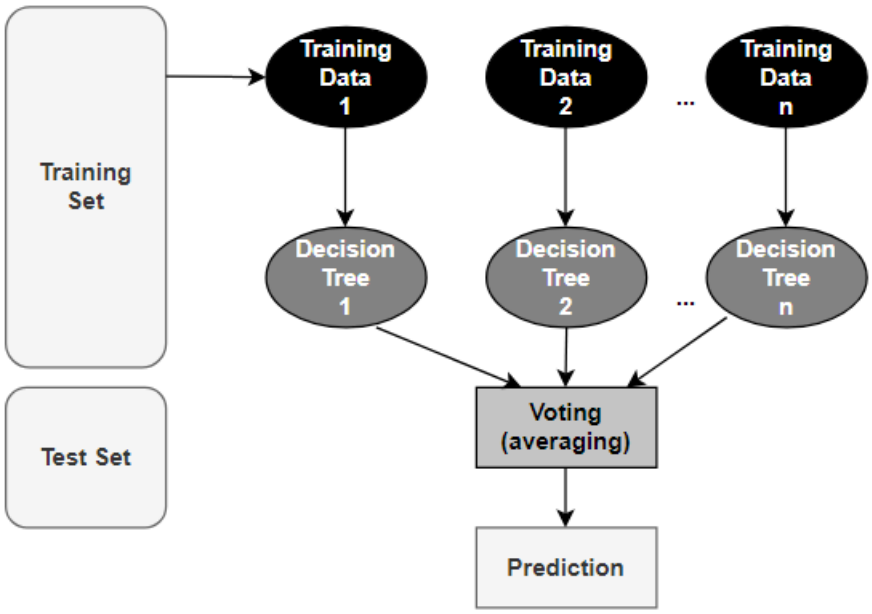
2.3.4 Random Forest

Ο αλγόριθμος **Random Forest** είναι ένας αλγόριθμος εκμάθησης με επίβλεψη που χρησιμοποιείται (κυρίως) για προβλήματα ταξινόμησης αλλά και για προβλήματα παλινδρόμησης (Regression). Ο αλγόριθμος δημιουργεί δένδρα απόφασης σε δείγματα δεδομένων, λαμβάνει τις προβλέψεις από το καθένα, και τέλος διαλέγει την καλύτερη λύση με βάση την **ψήφο (voting)**. Πρόκειται για μία μέθοδο συνόλου (ensemble learning method) που είναι καλύτερη από ένα μόνο δένδρο απόφασης διότι μειώνει την υπερπροσαρμογή

(overfitting) παίρνοντας τον μέσο όρο για το αποτέλεσμα. Τα βήματα του αλγορίθμου είναι τα εξής:

1. Αρχικά, ξεκίνα με την επιλογή τυχαίων δειγμάτων από ένα δοθέν dataset
2. Ύστερα, κατασκεύασε ένα δένδρο απόφασης για κάθε δείγμα. Λάβε το αποτέλεσμα - πρόβλεψη από κάθε δένδρο απόφασης
3. Εφάρμοσε την τεχνική της ψήφου (voting) για κάθε αποτέλεσμα – πρόβλεψη
4. Τέλος, επίλεξε ως τελικό αποτέλεσμα την πρόβλεψη με τις περισσότερες ψήφους

Το παρακάτω διάγραμα οπτικοποιεί την παραπάνω διαδικασία:

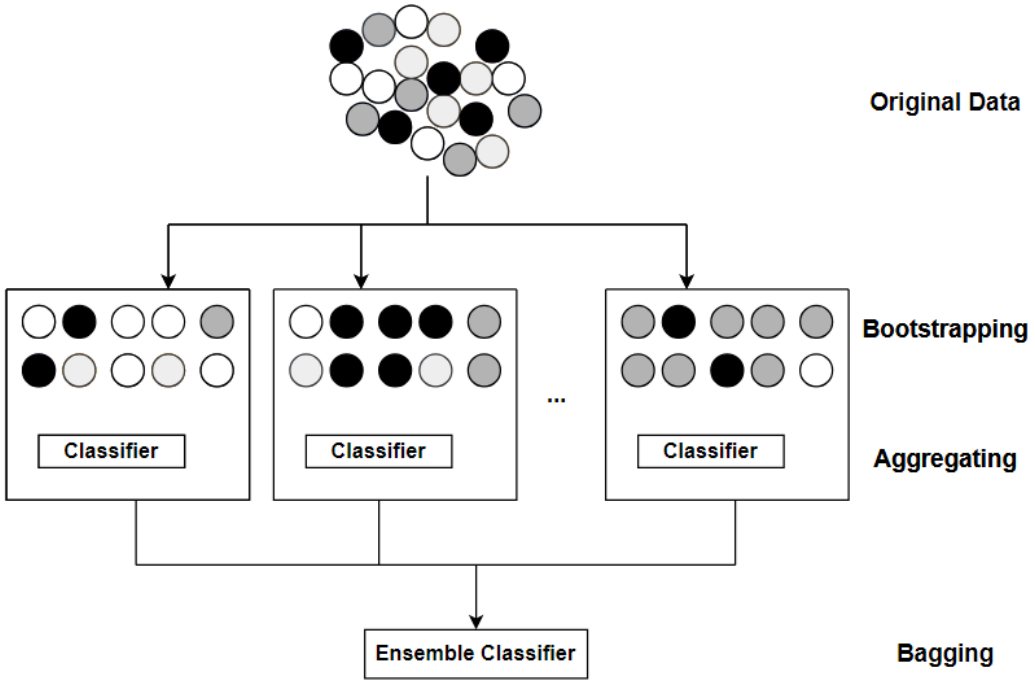


Εικόνα 2.5: Αλγόριθμος Random Forest

2.3.5 Bagging

Μέθοδοι όπως τα δένδρα απόφασης (Decision Trees) είναι επιρρεπείς σε υπερπροσαρμογή (overfitting) του μοντέλου στο training set, το οποίο σημαίνει πως το μοντέλο δίνει ακριβείς προβλέψεις για δεδομένα που ανήκουν στο σύνολο αυτό, αλλά όχι για νέα δεδομένα.
Η τεχνική **Bootstrap Aggregation (Bagging)** είναι μία μέθοδος συνόλου που προσπαθεί να επιλύσει το πρόβλημα της υπερπροσαρμογής για προβλήματα κατηγοριοποίησης (classification) ή παλινδρόμησης (regression). Στοχεύει στην βελτίωση της ακρίβειας και της επίδοσης των αλγορίθμων μηχανικής μάθησης. Ο τρόπος με τον οποίο αυτό επιτυγχάνεται είναι παίρνοντας τυχαία υποσύνολα του αρχικού dataset, με αντικατάσταση, και εφαρμόζοντας είτε έναν ταξινομητή (για προβλήματα κατηγοριοποίησης) είτε έναν παλινδρομητή – regressor (για προβλήματα παλινδρόμησης) σε κάθε υποσύνολο. Οι προβλέψεις για κάθε υποσύνολο ύστερα συναθροίζονται (Aggregating) μέσω πλειοψηφίας (majority vote) όταν έχουμε

κατηγοριοποίηση ή με χρήση μέσου όρου όταν έχουμε παλινδρόμηση, αυξάνοντας έτσι την ακρίβεια μιας πρόβλεψης.



Εικόνα 2.6: Ταξινομητής Bagging

Αν και η τεχνική Bagging αποτελεί ουσιαστικά μία γενικού σκοπού διαδικασία για τη μείωση της διασποράς μιας στατιστικής μεθόδου εκμάθησης, ο ταξινομητής – base estimator που εφαρμόζεται συνήθως στην περίπτωση της κατηγοριοποίησης είναι ο Decision Tree. Η διασπορά του μειώνεται αφού εισάγεται η τυχαιότητα στην διαδικασία κατασκευής του και μετά χτίζεται το σύνολο από αυτόν.
Όταν τυχαία υποσύνολα του dataset επιλέγονται ως τυχαία υποσύνολα των δειγμάτων, τότε ο αλγόριθμος ονομάζεται Pasting [26]. Όταν τυχαία υποσύνολα του dataset επιλέγονται ως τυχαία υποσύνολα των χαρακτηριστικών, τότε ο αλγόριθμος ονομάζεται Random Subspaces [27]. Τέλος, όταν οι ταξινομητές – base estimators χτίζονται σε υποσύνολα τόσο δειγμάτων όσο και χαρακτηριστικών, ο αλγόριθμος ονομάζεται Random Patches [28].

2.3.6 BERT

Το μοντέλο **BERT (Bidirectional Encoder Representations from Transformers)** είναι ένα μοντέλο αναπαράστασης γλώσσας το οποίο βασίζεται πάνω σε πρόσφατα έργα στην προ-εκπαίδευση αναπαραστάσεων συμφραζομένων, που περιλαμβάνουν τα εξής:

- Semi-supervised Sequence Learning [29]
- Generative Pre-Training [30]
- Embeddings from Language Models - ELMo [31]
- Universal Language Model Fine-tuning - ULMFit [32]

Ωστόσο, σε αντίθεση με τα προηγούμενα αυτά μοντέλα, το BERT είναι το πρώτο **βαθιά αμφίδρομο** (deeply bidirectional), μη εποπτευόμενο (unsupervised) μοντέλο αναπαράστασης γλώσσας, προεκπαιδευμένο (pre-trained) χρησιμοποιώντας μόνο ένα σώμα απλού κειμένου: την Wikipedia

Γενικά, οι προ-εκπαιδευμένες αναπαραστάσεις μπορούν να είναι είτε **χωρίς συμφραζόμενα (context-free)** ή **με συμφραζόμενα (contextual)**:

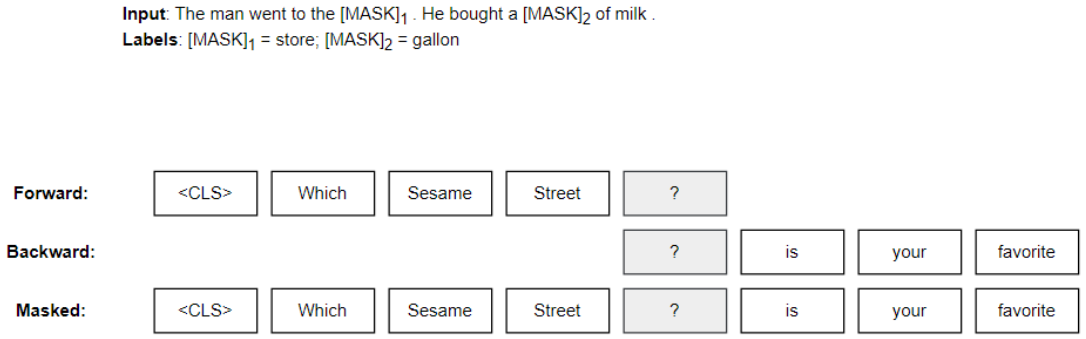
Μοντέλα χωρίς συμφραζόμενα όπως το word2vec ή το GloVe δημιουργούν μία μόνο διανυσματική αναπαράσταση λέξεων για κάθε λέξη στο λεξιλόγιο. Για παράδειγμα, η λέξη "γράμμα" θα είχε την ίδια χωρίς συμφραζόμενα αναπαράσταση τόσο στην φράση "γράμμα της Ελληνικής αλφαβήτου" όσο και στην φράση "γράμμα προς τον καθηγητή".

Αντίθετα, τα **μοντέλα με συμφραζόμενα** δημιουργούν μία αναπαράσταση κάθε λέξης που βασίζεται στις υπόλοιπες λέξεις της πρότασης. Οι αναπαραστάσεις με συμφραζόμενα μπορούν επίσης να είναι μονής κατεύθυνσης ή αμφίδρομες. Για παράδειγμα, στην πρόταση "Το άλφα είναι το πρώτο γράμμα της Ελληνικής αλφαβήτου", ένα μοντέλο με συμφραζόμενα μονής κατεύθυνσης θα αναπαριστούσε την λέξη "γράμμα" με βάση το κομμάτι "Το άλφα είναι το πρώτο" αλλά όχι το κομμάτι "της Ελληνικής αλφαβήτου". Παρόλαυτά, το μοντέλο BERT αναπαριστά την λέξη "γράμμα" χρησιμοποιώντας τόσο τα προηγούμενα όσο και τα επόμενα συμφραζόμενά της – "Το άλφα είναι το πρώτο ... της Ελληνικής αλφαβήτου" – αρχίζοντας από το πιο κάτω σημείο από ένα βαθύ νευρωνικό δίκτυο, καθιστώντας το μοντέλο βαθιά αμφίδρομο.

Τα μοντέλα γλώσσας που βασίζονται σε αμφίδρομα δίκτυα μακράς βραχύχρονης μνήμης (Long Short-term Memory - LSTM) εκπαιδεύουν ένα κοινό μοντέλο γλώσσας "από αριστερά προς τα δεξιά" και επίσης εκπαιδεύουν το αντίστροφο μοντέλο γλώσσας, δηλαδή "από δεξιά προς τα αριστερά" το οποίο προβλέπει προηγούμενες λέξεις από επόμενες όπως στην τεχνική ELMo. Στην τεχνική ELMo, υπάρχει ένα μόνο δίκτυο LSTM για το "προς τα μπρος" μοντέλο γλώσσας και το "προς τα πίσω" μοντέλο. Η μεγάλη διαφορά είναι ότι κανένα LSTM δεν λαμβάνει υπόψη και τα προηγούμενα και τα επόμενα tokens την ίδια στιγμή.

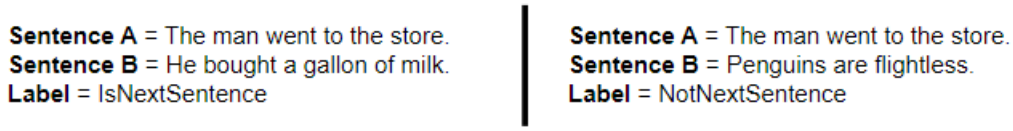
Ένα βαθιά αμφίδρομο μοντέλο είναι αυστηρά πιο ισχυρό από ένα μοντέλο που κάνει αναπαραστάσεις συμφραζομένων από "αριστερά προς τα δεξιά" ή από την ένωση ενός μοντέλου "αριστερά προς τα δεξιά" και ενός μοντέλου "δεξιά προς τα αριστερά". Δυστυχώς, τα τυπικά μοντέλα γλώσσας υπό όρους (conditional language models) μπορούν μόνο να εκπαιδευτούν από αριστερά προς τα δεξιά ή από δεξιά προς τα αριστερά, αφού η αμφίδρομη ρύθμιση θα επέτρεπε σε κάθε λέξη να "βλέπει τον εαυτό της" έμμεσα σε ένα περιβάλλον πολλαπλών επιπέδων.

Για να λύσει αυτό το πρόβλημα, το μοντέλο BERT χρησιμοποιεί την τεχνική του **masking** για να "κρύψει" κάποιες από τις λέξεις στην είσοδο και ύστερα να ρυθμίσει κάθε λέξη αμφίδρομα για να κάνει προβλέψεις για αυτές. Ένα παράδειγμα είναι το παρακάτω:



Εικόνα 2.7: Παράδειγμα της τεχνικής masking στο μοντέλο BERT

Επίσης, το μοντέλο BERT μαθαίνει να μοντελοποιεί σχέσεις μεταξύ προτάσεων με την προ-εκπαίδευση σε μία πολύ απλή "άσκηση" που μπορεί να παραχθεί από οποιοδήποτε σώμα κειμένου: "Δοθέντων 2 προτάσεων Α και Β, είναι η Β πράγματι η επόμενη πρόταση μετά την Α στο σώμα κειμένου, ή απλά μια τυχαία πρόταση;" Για παράδειγμα:



Εικόνα 2.8: Παράδειγμα προβλήματος απόφασης επόμενης πρότασης στο μοντέλο BERT

Αξίζει να αναφερθεί ότι το BERT έχει ανώτατο όριο **512** σε **λέξεις – tokens** που μπορεί να επεξεργαστεί, με εξαίρεση κάποια μοντέλα όπως το Reformer [33] που μπορεί να επεξεργαστεί 64.000 tokens ή το Longformer [34] που μπορεί να επεξεργαστεί 4096 tokens. Άλλες παραλλαγές του BERT που συναντώνται στη βιβλιογραφία είναι τα μοντέλα RoBERTa [35], ALBERT [36], ELECTRA [37], XLNet [38], DistilBERT [39], SpanBERT [40], BERTSUM [41] και TinyBERT [42].

3

Σχετικές εργασίες

Στο κεφάλαιο αυτό θα παρουσιάσουμε εργασίες σχετικές με τη διπλωματική. Οι σχετικές περιοχές είναι το ευρύτερο πεδίο της κατηγοριοποίησης κειμένων, υποπεριοχή του οποίου είναι η κατηγοριοποίηση νομικών κειμένων και ειδικότερα η κατηγοριοποίηση πολλαπλών ετικετών νομικών κειμένων.

3.1 Κατηγοριοποίηση κειμένων

Το πρόβλημα της κατηγοριοποίησης κειμένων αναφέρεται στην διαδικασία κατάταξης εγγράφων κειμένου σε προκαθορισμένες ομάδες. Η επίλυση προβλημάτων αυτού του τύπου απαιτεί την βαθιά κατανόηση μεθόδων μηχανικής μάθησης. Ενώ ήδη υπάρχοντες αλγόριθμοι της φυσικής επεξεργασίας γλώσσας είναι επιτυχείς στο εν λόγω ζήτημα, η εύρεση κατάλληλων δομών, αρχιτεκτονικών και τεχνικών για την κατηγοριοποίηση κειμένου αποτελεί πρόκληση για τους ερευνητές.
Στο [43] συνοψίζονται οι ερευνητικές τάσεις στο κομμάτι της κατηγοριοποίησης κειμένου. Τα περισσότερα συστήματα κατηγοριοποίησης κειμένων αποτελούνται από τέσσερα στάδια:

- Εξαγωγή Χαρακτηριστικών (Feature Extraction): Το στάδιο αυτό μετατρέπει τα έγγραφα κειμένου τα οποία στην αρχική τους μορφή είναι αδόμητες ακολουθίες σε έναν οργανωμένο χώρο χαρακτηριστικών. Αρχικά γίνεται καθαρισμός του κειμένου μέσω αφαίρεσης χαρακτήρων και λέξεων που δεν προσδίδουν αξία στο κείμενο και στη συνέχεια γίνεται η μαθηματική μοντελοποίηση για να δοθεί το κείμενο ως είσοδο σε έναν ταξινομητή. Τεχνικές που χρησιμοποιούνται είναι η Term Frequency-Inverse document frequency (TF-IDF), term frequency (TF), και οι ενσωματώσεις λέξεων (Word2Vec, αναπαραστάσεις λέξεων με συμφραζόμενα, Global Vectors for Word Representation (GloVe), και FastText).
- Μείωση Διαστάσεων (Dimensionality Reduction): Καθώς ένα κείμενο ενδέχεται να περιέχει πολλές μοναδικές λέξεις, υπάρχει κίνδυνος καθυστέρησης των βημάτων προεπεξεργασίας λόγω υψηλής χρονικής και χωρικής πολυπλοκότητας. Αν και μία λύση σε αυτό το ζήτημα είναι η χρήση υπολογιστικά φθηνών αλγορίθμων, πολλές φορές αυτοί οι αλγόριθμοι δεν έχουν καλές επιδόσεις. Η πιο αποδοτική λύση

προκειμένου να επιτευχθεί ελάττωση της πολυπλοκότητας αποφεύγοντας την πτώση των επιδόσεων είναι η μείωση διαστάσεων. Σημαντικές μέθοδοι είναι οι principal component analysis (PCA), linear discriminant analysis (LDA), non-negative matrix factorization (NMF), random projection, Autoencoder, και t-distributed Stochastic Neighbor Embedding (t-SNE).

- Τεχνικές Κατηγοριοποίησης (Classification Techniques): Η επιλογή του καλύτερου ταξινομητή στην κατηγοριοποίηση κειμένου αποτελεί το πιο σημαντικό βήμα. Χρειάζεται βαθιά κατανόηση του τρόπου λειτουργίας κάθε αλγορίθμου προκειμένου να γίνει η σωστή επιλογή. Χρησιμοποιούνται οι αλγόριθμοι Rocchio, bagging and boosting, logistic regression (LR), Naïve Bayes Classifier (NBC), k-nearest Neighbor (KNN), Support Vector Machine (SVM), decision tree classifier (DTC), random forest, conditional random field (CRF), καθώς και αλγόριθμοι βαθιάς μάθησης.
- Αξιολόγηση (Evaluation): Το τελευταίο στάδιο της κατηγοριοποίησης κειμένου αποτελεί η αξιολόγηση. Η κατανόηση του πως ένα μοντέλο αποδίδει είναι κρίσιμη για την χρήση και την ανάπτυξη μεθόδων κατηγοριοποίησης κειμένου. Μέθοδοι αξιολόγησης αποτελούν οι accuracy, F_β, Matthew correlation coefficient (MCC), receiver operating characteristics (ROC), και area under curve (AUC).

3.2 Κατηγοριοποίηση νομικών κειμένων

Η κατηγοριοποίηση νομικών κειμένων είναι μία συγκεκριμένη εφαρμογή κατηγοριοποίησης κειμένου που αφορά την ταξινόμηση και οργάνωση νομικών εγγράφων, όπως συμβάσεις, δικαστικές αποφάσεις, καταστατικά κλπ. Αν και οι βασικές αρχές της ταξινόμησης κειμένου παραμένουν αμετάβλητες, η σημαντικότερη διαφορά που διακατέχει την κατηγοριοποίηση νομικών κειμένων είναι η νομική ορολογία και η γλώσσα. Τα νομικά κείμενα είναι γραμμένα σε ειδική, επίσημη γλώσσα που περιλαμβάνει πολύπλοκη ορολογία. Αυτό μπορεί να αποτελέσει πρόκληση για μοντέλα μηχανικής μάθησης τα οποία έχουν μεν μία καλή κατανόηση μιας συγκεκριμένης γλώσσας λόγω προυπάρχουσας εκπαίδευσης αλλά δεν έχουν εκπαιδευθεί συγκεκριμένα σε κείμενα νομικού τύπου.

Στην εργασία [44] αναπτύσσεται ένα σύστημα χωρίς επίβλεψη το οποίο βασίζεται στη μοντελοποίηση θεμάτων (topic modeling) για την κατηγοριοποίηση δικαστικών αποφάσεων. Το dataset που χρησιμοποιείται είναι ένα σύνολο από 14.783 αποφάσεις. Το σύστημα αποτελείται από ένα κομμάτι προεπεξεργασίας κειμένου, ένα κομμάτι μοντελοποίησης θεμάτων όπου χρησιμοποιείται ένα μοντέλο LDA (Latent Dirichlet Allocation) και ένα κομμάτι annotation. Σύμφωνα με την μετρική f1-score τα αποτελέσματα είναι παρόμοια με προηγούμενα όμοια συστήματα με επίβλεψη ενώ ταξινομείται σωστά το 67% των εγγράφων. Η κατηγοριοποίηση όσον αφορά τις γενικές κατηγορίες είναι πιο ακριβής από εκείνη των ειδικών κατηγοριών σύμφωνα με τις μετρικές precision και f1-score. Η μέση τιμή για την μετρική recall του συστήματος είναι 0.5 και η μετρική precision είναι 0.37 για τις ειδικές κατηγορίες.

Η εργασία [45] ασχολείται με κατηγοριοποίηση νομικών εγγράφων στο πρόβλημα της Πρόβλεψης Σχετικότητας. Το dataset που χρησιμοποιείται αποτελείται από 20 υποθέσεις – 263 έγγραφα συνολικά. Χρησιμοποιούνται ενσωματώσεις λέξεων (word embeddings), μεταξύ των οποίων και λεξικό word2vec. Αν και δοκιμάζονται πολλοί αλγόριθμοι και με διαφορετικές

τιμές υπερπαραμέτρων (Νευρωνικά Δίκτυα, KNeighbors, Gaussian Process, AdaBoost, Random Forest, Support Vector Classification - SVC, Gaussian Naïve Bayes και Decision Tree), την καλύτερη επίδοση έχει ένα Νευρωνικό δίκτυο με σιγμοειδή συνάρτηση ενεργοποίησης, learning rate = 0.02, batch size = 20, epochs = 150, αριθμό νευρώνων στο πρώτο layer ίσο με 16 και αντίστοιχα στο δεύτερο ίσο με 8. Επιτυγχάνεται τιμή accuracy ίση με 0.835.

Η εργασία [46] παρουσιάζει ένα νέο dataset, διαθέσιμο στο ευρύ κοινό, με όνομα RAPTARCHIS47k και αξιολογεί διάφορους αλγορίθμους κατηγοριοποίησης με βάση αυτό. Το σύνολο αυτό αποτελείται από περισσότερους από 47 χιλιάδες επίσημους, ταξινομημένους πόρους Ελληνικής νομοθεσίας. Χρησιμοποιείται από παραδοσιακή μηχανική μάθηση και επαναλαμβανόμενα μοντέλα μέχρι μοντέλα μεταφοράς μάθησης. Μέσω της αξιολόγησης, φαίνεται πως αν και ταξινομητές παραδοσιακής μηχανικής μάθησης όπως οι Support Vector Machines θέτουν γερά θεμέλια για τα περισσότερα προβλήματα, υστερούν σε σύγκριση με πιο πολύπλοκες μεθόδους. Επαναλαμβανόμενες αρχιτεκτονικές βασιζόμενες σε Bidirectional Gated Recurrent Units οι οποίες έχουν περάσει από την τεχνική του fine-tuning παρέχουν βελτιωμένη συνολική απόδοση και ανταγωνίζονται ακόμα και πολυγλωσσικά μοντέλα που βασίζονται σε μετασχηματιστές, όπως το XLM-RoBERTa. Ένας κρίσιμος παράγοντας για την βελτιωμένη αυτή απόδοση είναι οι προ-εκπαιδευμένες, σχετικές με το πεδίο αυτό διανυσματικές παραστάσεις λέξεων, οι οποίες έχουν γνώση για τα συμφραζόμενα και συγκρατούν την σημασιολογική ομοιότητα των λέξεων. Αποδεικνύεται πως τα πιο προηγμένα τεχνολογικά, μονογλωσσικά και πολυγλωσσικά, μοντέλα βασισμένα στους μετασχηματιστές δίνουν αδιαμφισβήτητα τα καλύτερα αποτελέσματα. Πιο συγκεκριμένα, το BERT-BASE-ML κυριαρχεί στα προβλήματα επιπέδου Chapter και Subject, ενώ το GREEK-BERT αποδίδει καλύτερα από όλους τους άλλους ταξινομητές στο πρόβλημα κατηγοριοποίησης σε επίπεδο Volume.

3.3 Κατηγοριοποίηση Πολλαπλών Ετικετών νομικών κειμένων

Η κατηγοριοποίηση πολλαπλών ετικετών νομικών κειμένων είναι μία υποπεριοχή της κατηγοριοποίησης νομικών κειμένων που διαφέρει από την ευρύτερη περιοχή υπό την έννοια ότι επιτρέπει σε ένα νομικό κείμενο να σχετίζεται με πολλαπλές ετικέτες – κατηγορίες ταυτόχρονα, αντί να ανατίθεται σε μία μόνο κατηγορία. Αν και τα προβλήματα που ανήκουν σε αυτήν την υποπεριοχή δεν είναι εγγενώς πιο δύσκολα, η πολυπλοκότητα διαχείρισης πολλαπλών ετικετών, η ύπαρξη εξαρτήσεων μεταξύ τους και η πιθανή ανισορροπία των δεδομένων μπορούν να αποτελέσουν σημαντικές προκλήσεις.

Στο dataset RAPTARCHIS47k, σε ένα πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών ανατίθενται μία ή περισσότερες ετικέτες από ένα σύνολο περισσότερων από δύο χιλιάδες ετικέτες οργανωμένες με ιεραρχικό τρόπο [47]. Το πρόβλημα κατατάσσεται στην κατηγορία προβλημάτων Κατηγοριοποίησης Πολλαπλών Ετικετών Μεγάλης Κλίμακας (Large-scale Multi-Label Text Classification - LMTC). Εφόσον υπάρχει μεγάλη ανισορροπία στην συχνότητα εμφάνισης διαφορετικών κλάσεων, δημιουργείται η ανάγκη διαχωρισμού σε τρία επίπεδα συχνότητας εμφάνισης των ετικετών: α) all-labels, δηλαδή όλες οι ετικέτες β) frequent, δηλαδή συχνές ετικέτες και γ) few-case, δηλαδή πιο σπάνιες ετικέτες

Σε αυτά τα τρία επίπεδα, αξιολογούνται διάφοροι αλγόριθμοι με διαφορετικές παραμετροποιήσεις και μάλιστα κάθε αλγόριθμος διαχειρίζεται κείμενο προεπεξεργασμένο με διαφορετικό τρόπο. Πιο συγκεκριμένα, αξιολογούνται τρία είδη αλγορίθμων:
i) Μέθοδοι Πιθανοτικών Δένδρων Ετικέτας (Hierarchical PLT – Probabilistic Label Tree). Αναλυτικότερα, οι δύο αλγόριθμοι που ανήκουν σε αυτήν την κατηγορία είναι ο Parabel και ο Bonsai. Σε επίπεδο προεπεξεργασίας, εδώ χρησιμοποιείται η τεχνική TF-IDF, μία τεχνική που ανήκει στις BOW (Bag Of Words) προσεγγίσεις. Επίσης, αφαιρούνται οι τόνοι από το κείμενο και μετατρέπονται όλα τα γράμματα σε πεζά (lowercase)
ii) Υβριδικές μέθοδοι πιθανοτικών δένδρων (PLT and Attention Aware Networks Hybrid). Σε αυτή την κατηγορία ανήκει η μέθοδος AttentionXML η οποία δέχεται word embeddings ως είσοδο. Για αυτόν τον λόγο, χρησιμοποιείται ένα σύνολο από προεκπαιδευμένα embeddings Ελληνικών λέξεων και σε επίπεδο προεπεξεργασίας τα γράμματα μετατρέπονται σε κεφαλαία (uppercase), αφαιρούνται οι τόνοι, και κάθε ψηφίο αντικαθίσταται από τον χαρακτήρα “D”.
iii) Μέθοδοι βασιζόμενες σε μετασχηματιστές (Transformers). Σε αυτήν την κατηγορία ανήκει το μοντέλο GREEK-BERT και το GreekLegalBERT. Η προεπεξεργασία που γίνεται και για τα δύο αυτά μοντέλα είναι η αφαίρεση των τόνων από το κείμενο, και η μετατροπή των γραμμάτων σε πεζά.
Τα αποτελέσματα στα οποία καταλήγει η εργασία αυτή, δεδομένου ότι η σύγκριση των αλγορίθμων γίνεται σε 3 επίπεδα συχνότητας εμφάνισης των ετικετών όπως αναφέρθηκε παραπάνω (όλες οι ετικέτες, συχνές ετικέτες, σπάνιες ετικέτες), είναι τα εξής:
Σε γενικές γραμμές, δεν υπάρχει αυστηρός κανόνας για το ποιος αλγόριθμος πρέπει να προτιμάται πάντα διότι διαφορετικές επιλογές έδωσαν τα καλύτερα αποτελέσματα στα διαφορετικά επίπεδα συχνότητας. Πιο αναλυτικά, για τις συχνές ετικέτες τα μοντέλα που βασίζονται σε μετασχηματιστές (Transformer-based) είναι τα καλύτερα, με το GreekLegalBERT να αποδίδει λίγο καλύτερα από το GREEK-BERT. Για σπάνιες ετικέτες, ο αλγόριθμος Bonsai (ανήκει στην κατηγορία Πιθανοτικών Δένδρων Ετικέτας) αποδίδει καλύτερα. Ωστόσο αν κανείς πρέπει να επιλέξει τον καλύτερο αλγόριθμο στην γενική περίπτωση, αυτός είναι το μοντέλο GreekLegalBERT.

3.4 Εργαλεία κατηγοριοποίησης μέσω του θησαυρού Eurovoc

Εδώ θα παρουσιάσουμε δύο εργαλεία τα οποία κάνουν κατηγοριοποίηση εγγράφων με βάση τον θησαυρό Eurovoc. Το Eurovoc είναι ένας πολύγλωσσος, διεπιστημονικός θησαυρός – εργαλείο ευρετηρίασης που αρχικά είχε συνταχθεί ειδικά για την επεξεργασία των τεκμηριακών πληροφοριών των θεσμικών οργάνων της Ευρωπαϊκής Ένωσης (Ε.Ε.). Ωστόσο, καλύπτει τομείς αρκετά ευρείς ώστε να περιλαμβάνουν όχι μόνον τις πτυχές της Ε.Ε. αλλά και τις αντίστοιχες εθνικές, με μία έμφαση στην κοινοβουλευτική δραστηριότητα. Τα εργαλεία λαμβάνουν ως είσοδο ένα έγγραφο και δίνουν ως έξοδο ετικέτες – descriptors του θησαυρού.

3.4.1 Jex JRC Eurovoc Indexer

Το εργαλείο JEX (JRC Eurovoc Indexer) είναι ένα πακέτο λογισμικού υλοποιημένο σε Java, που αναπτύχθηκε στο Κοινό Ερευνητικό Κέντρο της Ευρωπαϊκής Επιτροπής (Joint Research Centre - JRC) με σκοπό την αυτόματη κατηγοριοποίηση εγγράφων με ετικέτες – descriptors

από τον θησαυρό Eurovoc [48]. Μπορεί επίσης να χρησιμοποιηθεί από προχωρημένους χρήστες για ερευνητικούς σκοπούς, με χρήση της training μεθόδου που έχει αναπτυχθεί στο JRC, για εκπαίδευση του συστήματος με κάποιο διαφορετικό τύπο οντολογιών ή απλώς για την αύξηση της επίδοσής του. Έχει εκπαιδευτεί στο να κατηγοριοποιεί έγγραφα σε 22 γλώσσες: Βουλγαρικά, Τσέχικα, Δανικά, Ολλανδικά, Αγγλικά, Εσθνοικά, Φινλανδικά, Γαλλικά, Γερμανικά, Ελληνικά, Ουγγρικά, Ιταλικά, Λετονικά, Λιθουανικά, Μαλτέζικα, Πολωνικά, Πορτογαλικά, Ρουμάνικα, Σλοβακικά, Σλοβενικά, Ισπανικά και Σουηδικά.

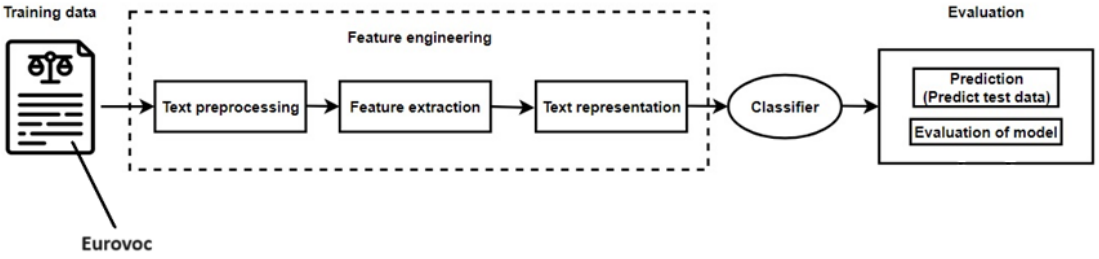
3.4.2 PyEuroVoc

Το PyEuroVoc είναι ένα εργαλείο κατηγοριοποίησης νομικών κειμένων με ετικέτες – descriptors από τον θησαυρό Eurovoc υλοποιημένο σε Python με τον κώδικα διαθέσιμο στο ευρύ κοινό (Open Source) [49]. Χρησιμοποιεί ποικίλες μορφές του μοντέλου BERT (Bidirectional Encoder from Transformers), ανάλογα την εκάστοτε γλώσσα, από 22 συνολικά γλώσσες που μπορούν να βρεθούν στα έγγραφα JRC-Acquis και OPOCE. Με πολλαπλούς διαχωρισμούς (splits) που έγιναν στα δεδομένα, διαπιστώθηκε ότι έχει πολύ καλύτερες επιδόσεις από το JEX.

4

Αυτόματη Κατηγοριοποίηση Νομοθεσίας

Στο κεφάλαιο αυτό περιγράφουμε το πρώτο πείραμα με το οποίο ασχοληθήκαμε το οποίο αφορά κατηγοριοποίηση κειμένων Ελληνικής Νομοθεσίας με βάση τον θησαυρό Eurovoc. Στα στάδια της κατηγοριοποίησης, είσοδος είναι ένα κείμενο νομοθεσίας το οποίο περνάει από το κομμάτι της μηχανικής χαρακτηριστικών (Feature engineering). Το κομμάτι αυτό περιλαμβάνει την προεπεξεργασία κειμένου (Text preprocessing), για την αφαίρεση μη χρήσιμης πληροφορίας και θορύβου από το κείμενο, και ακολούθως την εξαγωγή χαρακτηριστικών (Feature extraction) ώστε να μετατραπούν οι αρχικές αδόμητες ακολουθίες κειμένου σε έναν δομημένο χώρο χαρακτηριστικών. Η αριθμητική πλέον αναπαράσταση του κειμένου δίνεται ως είσοδος σε έναν ταξινομητή (Classifier), ο οποίος αξιολογείται ανάλογα με τις προβλέψεις στο test set.



Εικόνα 4.1: Στάδια Κατηγοριοποίησης Νομοθεσίας

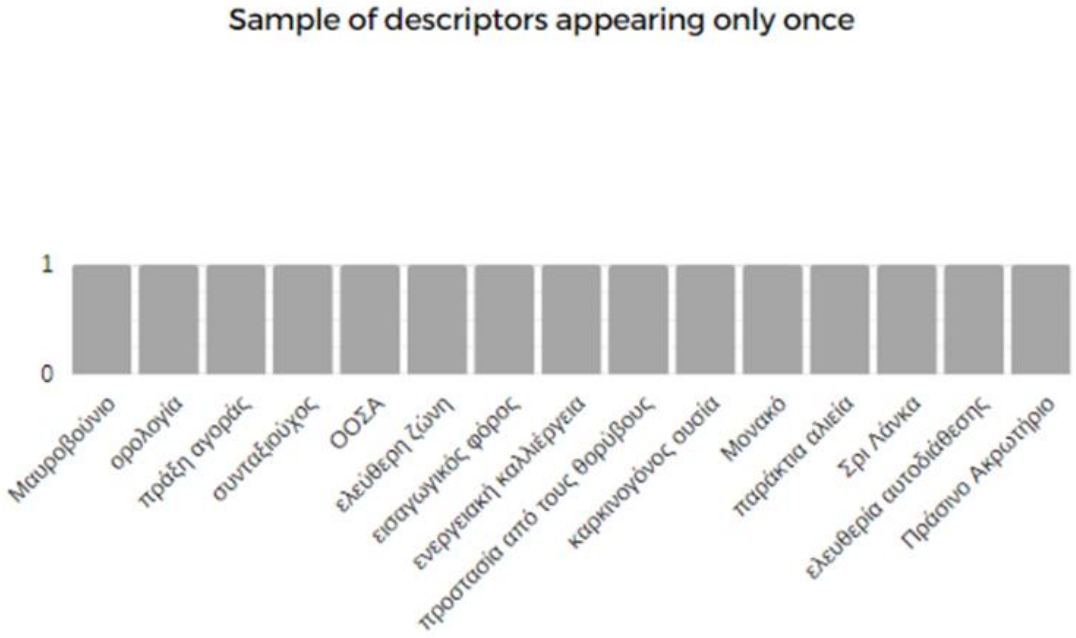
4.1 Σώμα Κειμένων – Επισημασμένο σύνολο δεδομένων αληθείας

Το αρχικό dataset που χρησιμοποιήθηκε αποτελείτο από 447 Ελληνικούς νόμους σε μορφή txt. Ένα παράδειγμα νόμου στην μορφή αυτή φαίνεται παρακάτω:

```
ΝΟΜΟΣ ΥΠ' ΑΡΙΘ. 3814
Κύρωση Πράξης Νομοθετικού Περιεχομένου
και άλλες διατάξεις.
Ο ΠΡΟΕΔΡΟΣ
ΤΗΣ ΕΛΛΗΝΙΚΗΣ ΔΗΜΟΚΡΑΤΙΑΣ
Εκδίδομε τον ακόλουθο νόμο που ψήφισε η Βουλή:
Άρθρο πρώτο
Κυρώνεται και έχει ισχύ νόμου από τη δημοσίευσή της στην Εφημερίδα της Κυβερνήσεως η από
16.9.2009 Πράξη Νομοθετικού Περιεχομένου «Ρύθμιση θεμάτων Φ.Π.Α., επισφαλών απαιτήσεων,
ληξιπρόθεσμων χρεών προς το Δημόσιο, αναπροσαρμογή ποσού οφειλών προς ασφαλιστικά ταμεία και
αναστολή πλειστηριασμών από πιστωτικά ιδρύματα», που δημοσιεύθηκε στο υπ' αριθ. 181 Φύλλο της
Εφημερίδας της Κυβερνήσεως (τεύχος Α΄), που έχει ως εξής:
«ΠΡΑΞΗ
ΝΟΜΟΘΕΤΙΚΟΥ ΠΕΡΙΕΧΟΜΕΝΟΥ
Ρύθμιση θεμάτων Φ.Π.Α., επισφαλών απαιτήσεων, ληξιπρόθεσμων χρεών προς το Δημόσιο,
αναπροσαρμογή ποσού οφειλών προς ασφαλιστικά ταμεία και αναστολή πλειστηριασμών από πιστωτικά
ιδρύματα.»
Ο ΠΡΟΕΔΡΟΣ
ΤΗΣ ΕΛΛΗΝΙΚΗΣ ΔΗΜΟΚΡΑΤΙΑΣ
Έχοντας υπόψη:
1. Τη διάταξη του άρθρου 44 παρ. 1 του Συντάγματος,
2. την έκτακτη περίπτωση εξαιρετικά επείγουσας και απρόβλεπτης ανάγκης:
α) να ολοκληρωθεί η προσαρμογή της φορολογικής νομοθεσίας πριν από την λήξη του τρέχοντος
έτους, προκειμένου να εκτελεσθεί απρόσκοπτα ο κρατικός προϋπολογισμός του έτους 2010 και να
ρυθμισθούν θέματα επισφαλών απαιτήσεων και
β) να προστατευθούν άμεσα, ενόψει της οικονομικής κρίσεως, οι πολίτες που αντιμετωπίζουν
δυσχέρειες για την έγκαιρη εξόφληση των χρεών τους προς το Δημόσιο, οι μικροοφειλέτες που
υπόκεινται σε ποινικές διώξεις και οι δανειολήπτες που δεν δύνανται να αντιμετωπίσουν τις
δανειακές τους υποχρεώσεις.
```

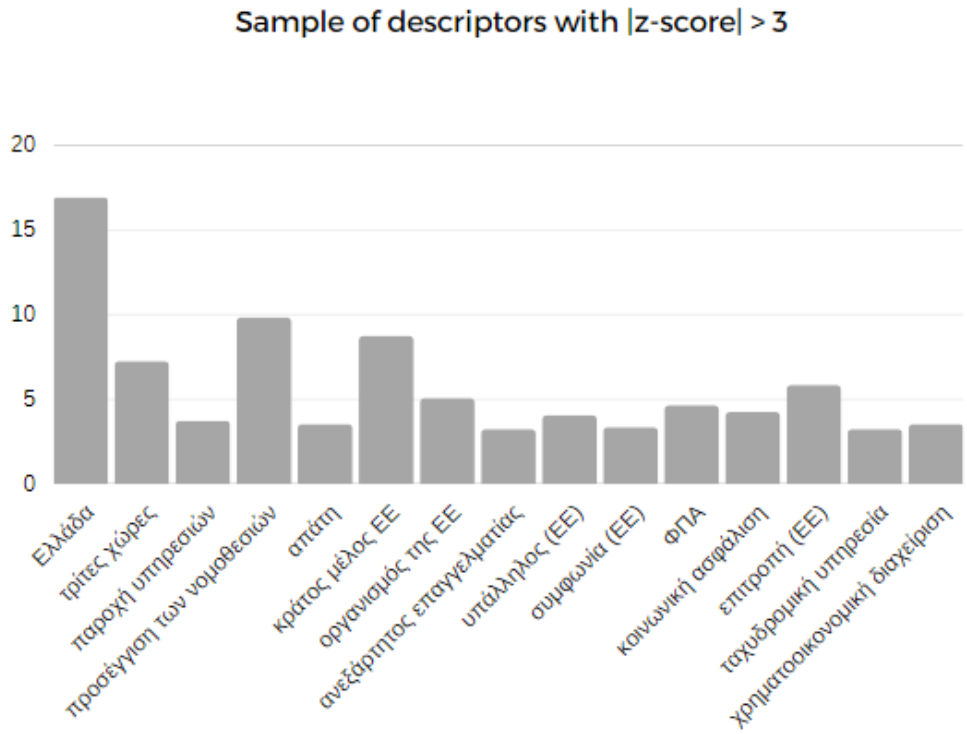
Εικόνα 4.2: Παράδειγμα νόμου σε μορφή txt

Εφόσον ένας από τους ταξινομητές (το μοντέλο BERT) που χρησιμοποιήσαμε και αναλύουμε σε επόμενη ενότητα είχε όριο λέξεων επεξεργασίας (512 tokens), αναγκαστήκαμε να διαχωρίσουμε τον κάθε νόμο στα άρθρα του ώστε να δίνεται το κάθε άρθρο ως είσοδος για τον συγκεκριμένο ταξινομητή. Οι 447 αυτοί νόμοι αποτελούνται συνολικά από 8909 άρθρα. Επίσης, από τη στιγμή που είχαμε να αντιμετωπίσουμε ένα πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών με τις ετικέτες (descriptors) του θησαυρού Eurovoc, δημιουργήθηκε η ανάγκη να έχουμε ένα ήδη κατηγοριοποιημένο dataset με βάση τα υφιστάμενα εργαλεία που υπάρχουν, το οποίο θα θεωρήσουμε ως **Ground Truth**, δηλαδή αντιπροσωπεύει τις «σωστές» ή «αληθείς» κατηγορίες για κάθε κείμενο. Από τα δύο εργαλεία που αναφέρουμε στο Κεφάλαιο 3, δηλαδή το JEX (JRC Eurovoc Indexer) και το PyEuroVoc, χρησιμοποιήσαμε το **PyEuroVoc** για να αναθέσουμε descriptors στα κείμενα. Δεδομένου ότι το εργαλείο PyEuroVoc βασίζεται και αυτό στο μοντέλο BERT, ερχόμενοι πάλι αντιμέτωποι με το ζήτημα του ορίου των 512 λέξεων – tokens, για τη δημιουργία του dataset εκτελέσαμε το εργαλείο για κάθε άρθρο από κάθε νόμο. Η ιδιαιτερότητα που παρατηρήσαμε υλοποιώντας το δικό μας μοντέλο BERT είναι ότι δεν αποδίδει καλά όταν το dataset έχει ανισορροπίες (imbalances), και στο dataset μας παρατηρήσαμε το εξής: Ενώ συνολικά έχουμε 8909 άρθρα, το 80% των οποίων είναι 7127 άρθρα, υπάρχουν descriptors που δεν εμφανίζονται ούτε στο 1% αυτών των άρθρων (1% του 7127 = 71 άρθρα). Οι descriptors αυτοί είναι 1568. Ως παράδειγμα, μόνο οι descriptors οι οποίοι εμφανίζονται μόνο μία φορά σε ολόκληρο το dataset είναι 338. Βλέπουμε μερικούς από αυτούς παρακάτω:



Εικόνα 4.3: Παράδειγμα descriptors που εμφανίζονται μόνο μία φορά στο dataset

Επίσης, με χρήση της στατιστικής μετρικής z-score [50] η οποία δείχνει πόσες τυπικές αποκλίσεις μακριά είναι μία τιμή από τον μέσον όρο, αφαιρέσαμε όσους descriptors είχαν την μετρική αυτή μεγαλύτερη από 3 κατ'απόλυτη τιμή. Συνολικά οι descriptors αυτοί ήταν 35. Στο παρακάτω σχήμα, ένα δείγμα αυτών παρουσιάζονται στον άξονα x ενώ το αντίστοιχο z-score τους φαίνεται στον άξονα y:



Εικόνα 4.4: Descriptors με |z-score|>3

Με αυτή την αφαίρεση, προκύπτουν κάποια κείμενα χωρίς καθόλου descriptors. Διαγράφοντάς τα, τα άρθρα είναι πλέον 8182 σε αριθμό και αντιστοιχούν σε 442 νόμους. Από τους descriptors που έμειναν για κάθε άρθρο κρατήσαμε μόνο τον πρώτο descriptor (αυτόν με την μεγαλύτερη πιθανότητα). Με αυτόν τον τρόπο, ο τελικός αριθμός των descriptors – κλάσεων ήταν 157. Ενώνοντας τους descriptors για όλα τα άρθρα ενός νόμου (και αφαιρώντας ενδεχόμενα duplicates) μπορούμε πλέον να ισχυριστούμε πως έχουμε μια σωστή κατηγοριοποίηση για κάθε νόμο. Η τελική μορφή είναι δύο αρχεία CSV με τα πεδία lawNumber, δηλαδή ο αριθμός του νόμου (στην εικόνα 4.4 επαναλαμβάνεται για κάθε άρθρο που ανήκει σε έναν νόμο), text, δηλαδή το κείμενο του άρθρου ή του νόμου, και descriptors. Η διαδικασία αυτή της αντιστοίχησης – mapping φαίνεται παρακάτω για την δημιουργία του dataset (η ίδια διαδικασία ακολουθείται και για την ένωση των αποτελεσμάτων που προβλέπει το δικό μας μοντέλο BERT):

	lawNumber	text	descriptors
0	3814_article.json	κυρωνεται ισχυ νομου δημοσιευση εφημεριδα κυβερνησεως πραξη νομοθετικου περιεχομενου ρυθμιση θεματων επισφαλων απαιτησεων ληξιπροθεσμων χρεων δημοσιο αναπροσαρμογη οφειλων ασφαλιστικα ταμεια αναστολη πλειστηριασμων πιστωτικα ιδρυματα δημοσιευθηκε αριθ φυλλο εφημεριδας κυβερνησεως τευχος πραξη νομοθετικου περιεχομενου ρυθμιση θεματων επισφαλων απαιτησεων ληξιπροθεσμων χρεων δημοσιο αναπροσαρμογη οφειλων ασφαλιστικα ταμεια αναστολη πλειστηριασμων πιστωτικα προεδρος ελληνικης δημοκρατιας εχοντας διαταξη αρθρου συνταγματος εκτακτη περιπτωση εξαιρετικα επειγουσας απροβλεπτης αναγκης ολοκληρωθει προσαρμογη φορολογικης νομοθεσιας ληξη τρεχοντος ετους εκτελεσθει απροσκοπτα κρατικος προυπολογισμος ετους ρυθμισθουν θεματα επισφαλων απαιτησεων προστατευθουν ενοψει οικονομικης κρισεως πολιτες αντιμετωπιζουν δυσχερειες εγκαιρη εξοφληση χρεων δημοσιο μικροοφειλετες υποκεινται ποινικες διωξεις δανειοληπτες δυνανται αντιμετωπισουν δανειακες υποχρεωσεις γεγονος ενοψει οικονομικης κρισης καθιστανται ...	απαίτηση
1	3814_article.json	παραγραφος αρθρου πραξης νομοθετικου περιεχομενου κυρωνεται αρθρο πρωτο νομου αντικαθισταται ακολουθως υποκειμενους φορο κανουν χρηση ευχερειας παρεχεται παραγραφο συγκεκριμενη φορολογικη περιοδο υποβαλλουν ανακριβη δηλωση υποβαλλουν ανακριβη υποβαλλουν περιοδικη δηλωση επομενων φορολογικων περιοδων διαχειριστικης περιοδου επιβαλλεται προσθετος φορος διαφορας φορου οφειλεται ποσοστο τεσσεραμισι εκατο περιπτωση ανακριβους δηλωσης πεντε εκατο περιπτωση υποβολης δηλωσης μηνα ανωτατο οριο διακοσια εκατο περιπτωση διοικητικης επιλυσης διαφορας δικαστικου συμβιβασμου εφαρμοζονται διαταξεις αρθρου ισχυουν διαταξεις παρουσας παραγραφου ισχυουν περιοδικες δηλωσεις προθεσμια υποβολης ληγει αρθρο πραξης νομοθετικου περιεχομενου κυρωνεται αρθρο πρωτο νομου καταργειται πλειστηριασμοι αρθρου πραξης νομοθετικου περιεχομενου αναστελλονται ιουνιου ισχυς νομου αρχιζει δημοσιευση εφημεριδα κυβερνησεως	φορολογία

Εικόνα 4.5: Ο νόμος υπ' αριθμόν 3814 πριν από την διαδικασία mapping

	lawNumber	text	descriptors
0	3814_article.json	κυρωνεται ισχυ νομου δημοσιευση εφημεριδα κυβερνησεως πραξη νομοθετικου περιεχομενου ρυθμιση θεματων επισφαλων απαιτησεων ληξιπροθεσμων χρεων δημοσιο αναπροσαρμογη οφειλων ασφαλιστικα ταμεια αναστολη πλειστηριασμων πιστωτικα ιδρυματα δημοσιευθηκε αριθ φυλλο εφημεριδας κυβερνησεως τευχος πραξη νομοθετικου περιεχομενου ρυθμιση θεματων επισφαλων απαιτησεων ληξιπροθεσμων χρεων δημοσιο αναπροσαρμογη οφειλων ασφαλιστικα ταμεια αναστολη πλειστηριασμων πιστωτικα προεδρος ελληνικης δημοκρατιας εχοντας διαταξη αρθρου συνταγματος εκτακτη περιπτωση εξαιρετικα επειγουσας απροβλεπτης αναγκης ολοκληρωθει προσαρμογη φορολογικης νομοθεσιας ληξη τρεχοντος ετους εκτελεσθει απροσκοπτα κρατικος προυπολογισμος ετους ρυθμισθουν θεματα επισφαλων απαιτησεων προστατευθουν ενοψει οικονομικης κρισεως πολιτες αντιμετωπιζουν δυσχερειες εγκαιρη εξοφληση χρεων δημοσιο μικροοφειλετες υποκεινται ποινικες διωξεις δανειοληπτες δυνανται αντιμετωπισουν δανειακες υποχρεωσεις γεγονος ενοψει οικονομικης κρισης καθιστανται ...	απαίτηση, φορολογία

Εικόνα 4.6: Ο νόμος υπ' αριθμόν 3814 μετά από την διαδικασία mapping

Το dataset της πρώτης μορφής χρησιμοποιήθηκε για να εκπαιδευτεί το δικό μας μοντέλο BERT (για να μην παραβιαστεί το όριο των 512 tokens) ενώ το dataset της δεύτερης μορφής χρησιμοποιήθηκε για την εκπαίδευση των υπόλοιπων αλγορίθμων (οι οποίοι δεν έχουν όριο στο πόσες λέξεις επεξεργάζονται, άρα μπορούν να διαβάσουν ολόκληρο τον νόμο) όπως επίσης και για τη σύγκριση των αποτελεσμάτων κάθε αλγορίθμου με τις σωστές κατηγορίες (μέσω του test set - 15 νόμοι ή αλλιώς 819 άρθρα).
Συγκεντρωτικά, παρακάτω παρουσιάζουμε τα έγγραφα που είχαμε αρχικά, αυτά που αφαιρέσαμε και αυτά που τελικά κρατήσαμε:

Έγγραφα	
Αρχικός αριθμός εγγράφων:	447 νόμοι ή 8909 άρθρα
Έγγραφα που απορρίφθηκαν:	5 νόμοι ή 727 άρθρα
Έγγραφα πειράματος:	442 νόμοι ή 8182 άρθρα

Πίνακας 4.1: Πείραμα Νομοθεσίας - Έγγραφα

Στον παρακάτω πίνακα παρουσιάζουμε τα έγγραφα που χρησιμοποιήθηκαν για training, validation, testing:

Train Validation Test Split	
Αριθμός εγγράφων στο training set:	412 νόμοι ή 6544 άρθρα
Αριθμός εγγράφων στο validation set:	15 νόμοι ή 819 άρθρα
Αριθμός εγγράφων στο test set:	15 νόμοι ή 819 άρθρα

Πίνακας 4.2: Πείραμα Νομοθεσίας – Train Validation Test Split

Μία δεύτερη εκδοχή της **Ground truth** που μας απασχόλησε ήταν σύμφωνα με τις ετικέτες που έχουν ανατεθεί από το **Εθνικό Μητρώο Διοικητικών Διαδικασιών - Μίτος**, το επίσημο μητρώο διαδικασιών του Ελληνικού Δημοσίου. Από τους 15 νόμους που αποτελούσαν το test set μας, οι 8 έχουν ήδη κατηγοριοποιηθεί από το σύστημα Μίτος, ενώ στους υπόλοιπους νόμους αφήσαμε τις ετικέτες από το PyEuroVoc. Παρακάτω βλέπουμε 10 από τα 15 κείμενα του test set (εδώ δεν δείχνουμε το κείμενο κάθε νόμου για να είναι πιο ευδιάκριτες οι ετικέτες):

	LawNumber	MITOS
0	4249_article.json	Απασχόληση στο δημόσιο τομέα,Θέματα εκπαίδευσης,Εκπαίδευση,Άσκηση επαγγελμάτων,Εργασία και ασφάλιση
1	4250_article.json	Εργασία και ασφάλιση,Απασχόληση στο δημόσιο τομέα
2	4251_article.json	Άτομα με αναπηρίες και χρόνιες παθήσεις,Πολίτες τρίτων χωρών (Πολίτες άλλων κρατών),Επιδόματα,Πολίτες άλλων κρατών,Ίδρυση και λειτουργία επιχειρήσεων,Πολίτες κράτους μέλους της ΕΕ
3	4252_article.json	οικονομικό έτος
4	4253_article.json	οικονομικό έτος
5	4254_article.json	Άσκηση επαγγελμάτων,Ελεύθεροι επαγγελματίες,Διασυνοριακή παροχή υπηρεσιών,Διαχείριση ακίνητης περιουσίας,Μεταφορές,Λειτουργία εγκαταστάσεων εξυπηρέτησης οχημάτων,Λιανεμπόριο -Χονδρικό εμπόριο,Αδειοδοτήσεις και συμμόρφωση,Επιχειρηματική δραστηριότητα,Μητρώο
6	4255_article.json	Δικαστήριο της Ευρωπαϊκής Ένωσης,Ευρωπαϊκή Ένωση,Ευρωπαϊκό Κοινοβούλιο,έντυπο,αποζημίωση και απόδοση εξόδων,γενικός γραμματέας του οργάνου,δημόσια διοίκηση της Κοινότητας,εισαγωγή (ΕΕ),εξέλεγξη λογαριασμών,επιτροπολογία,θεσμική αρμοδιότητα (ΕΕ),κοινοποίηση των δεδομένων,κοινωνική παροχή,περιφερειακές ενισχύσεις,πολιτική απασχόλησης,προϊόν καταγωγής,τεχνικό πρότυπο,τραπεζική δραστηριότητα,φαρμακευτικό προϊόν,χρηματοδοτικό μέσο της ΕΕ
7	4256_article.json	Ναυτιλία,Τουρισμός,Αλιευτικά σκάφη
8	4257_article.json	Άτομα με αναπηρίες και χρόνιες παθήσεις,Ακίνητα,Δημόσια Περιουσία,Δημόσια Περιουσία (Δημόσια περιουσία και εθνικά κληροδοτήματα / κοινωφελείς περιουσίες),Τέλη και ειδικοί φόροι,Αδειοδοτήσεις και συμμόρφωση,Οχήματα
9	4258_article.json	Φυσικές καταστροφές,Νομοθεσία και αποφάσεις,Κτηματολόγιο

Εικόνα 4.7: Νόμοι του test set με ετικέτες από το σύστημα MITOS

Το επόμενο βήμα ήταν να βρούμε μία αντιστοιχία των ετικετών του συστήματος Μίτος με εκείνες από τον θησαυρό Eurovoc ώστε και τα 15 κείμενα να είναι κατηγοριοποιημένα με βάση τον θησαυρό, αλλά οι κατηγορίες να "συμπίπτουν" όσο καλύτερα γίνεται σημασιολογικά. Το ίδιο δείγμα των 10 κειμένων με αυτή την τροποποίηση παρουσιάζεται παρακάτω:

	LawNumber	MITOStoEurovoc
0	4249_article.json	δημόσιος υπάλληλος,εργασία πλήρους απασχόλησης,εργασία μερικής απασχόλησης,εκπαίδευση,επαγγελματική σταδιοδρομία,σχέσεις εκπαίδευσης-επαγγελματικής ζωής,εργασία,ασφάλιση
1	4250_article.json	εργασία,ασφάλιση,δημόσιος υπάλληλος,εργασία πλήρους απασχόλησης,εργασία μερικής απασχόλησης
2	4251_article.json	άτομο με αναπηρία,χρόνια νόσος,τρίτες χώρες,επίδομα μέριμνας,οικογενειακό επίδομα,επικουρικό επίδομα,επίδομα στουδών,επίδομα λόγω θανάτου,επίδομα μητρότητας,αλλοδαπός,δημιουργία επιχείρησης,κράτος μέλος ΕΕ
3	4252_article.json	οικονομικό έτος
4	4253_article.json	οικονομικό έτος
5	4254_article.json	επαγγελματική σταδιοδρομία,σχέσεις εκπαίδευσης-επαγγελματικής ζωής,ανεξάρτητος επαγγελματίας,παροχή υπηρεσιών,διασυνοριακές μεταφορές,σύνορα,ακίνητη περιουσία,διαχείριση,χρηματοοικονομική διαχείριση,εναέριες μεταφορές,επίγειες μεταφορές,όχημα,λιανικό εμπόριο,χονδρικό εμπόριο,ειδική άδεια,δραστηριότητα της επιχείρησης,επιχείρηση,εγγραφή εταιρείας στα μητρώα,επίσημο έγγραφο
6	4255_article.json	Δικαστήριο της Ευρωπαϊκής Ένωσης,Ευρωπαϊκή Ένωση,Ευρωπαϊκό Κοινοβούλιο,έντυπο,αποζημίωση και απόδοση εξόδων,γενικός γραμματέας του οργάνου,δημόσια διοίκηση της Κοινότητας,εισαγωγή (ΕΕ),εξέλεγξη λογαριασμών,επιτροπολογία,θεσμική αρμοδιότητα (ΕΕ),κοινοποίηση των δεδομένων,κοινωνική παροχή,περιφερειακές ενισχύσεις,πολιτική απασχόλησης,προϊόν καταγωγής,τεχνικό πρότυπο,τραπεζική δραστηριότητα,φαρμακευτικό προϊόν,χρηματοδοτικό μέσο της ΕΕ
7	4256_article.json	ναυτιλιακή πολιτική,Διεθνής Ναυτιλιακός Οργανισμός,τουρισμός,αλιευτικό πλοίο
8	4257_article.json	άτομο με αναπηρία,χρόνια νόσος,ακίνητη περιουσία,δημόσια περιουσία,κληροδότημα,ειδικοί φόροι κατανάλωσης,ειδική άδεια,όχημα
9	4258_article.json	φυσική καταστροφή,νομοθεσία,λήψη απόφασης,δημόσιο κτηματολόγιο

Εικόνα 4.8: Αντιστοίχηση MITOS - Eurovoc

4.2 Προεπεξεργασία

Text preprocessing: Η διαδικασία προεπεξεργασίας κειμένου είναι η εξής:

1. Μετατροπή των γραμμάτων σε πεζά
2. Διατήρηση μόνο των συμβολοσειρών με αλφαβητικό περιεχόμενο (χαρακτήρες α-ω)
3. Αφαίρεση μικρών λέξεων – διατήρηση μόνο αυτών με μήκος μεγαλύτερο του 2
4. Αφαίρεση των stopwords

Feature Extraction: Με την εξαίρεση του μοντέλου BERT, το σώμα κειμένου της στήλης text μετατρέπεται σε αριθμητική μορφή μέσω της τεχνικής TF-IDF, όπου έχουμε ορίσει τις παραμέτρους max_df = 0.8 και min_df = 5. Με αυτόν τον τρόπο, τις λέξεις οι οποίες εμφανίζονται σε περισσότερο από 80% των εγγράφων (max_df = 0.8) αλλά και τις λέξεις οι οποίες εμφανίζονται συνολικά λιγότερο από 5 φορές (min_df = 5) δεν τις λαμβάνουμε υπόψη.

4.3 Αλγόριθμοι

Οι αλγόριθμοι οι οποίοι χρησιμοποιήθηκαν για την κατηγοριοποίηση κειμένων νομοθεσίας ήταν οι εξής: **Naïve Bayes**, **K-Nearest Neighbor (kNN), Decision Tree, Random Forest**, **Bagging** και **BERT**. Εξαιρουμένου του μοντέλου BERT, οι υπόλοιποι ήταν διαθέσιμοι από την βιβλιοθήκη scikit learn (sklearn) της Python και συνδυάστηκαν με την τεχνική Label Powerset για το πρόβλημά μας. Για όλους χρησιμοποιήθηκε η μέθοδος predict_proba η οποία υπολογίζει πιθανότητες κατηγοριοποίησης για κάθε κλάση.
Για κάθε έναν από αυτούς είχαμε διαφορετική συνθήκη όσον αφορά την ελάχιστη τιμή πιθανότητας πάνω από την οποία θα κρατήσουμε τις προβλεπόμενες κλάσεις για ένα κείμενο. Τις πιθανότητες που ήταν κάτω από αυτήν την τιμή τις εξισώσαμε με μηδέν, οπότε οι αντίστοιχες κλάσεις δεν εμφανίζονται. Ο λόγος για τις διαφορετικές συνθήκες ανά αλγόριθμο είναι οτι απέδιδαν καλύτερα για διαφορετικές τιμές.
Θα αναλύσουμε αυτές τις συνθήκες μαζί με την επιλογή υπερπαραμέτρων χωριστά για κάθε αλγόριθμο, και επίσης θα παραθέσουμε και μία οπτικοποίηση των αποτελεσμάτων συγκρίνοντας τις ετικέτες που προέβλεψε κάθε αλγόριθμος σε σχέση με αυτές που προέρχονται από το PyEuroVoc (τα ακριβή αποτελέσματα παρουσιάζονται στην ενότητα 4.5). Η μέθοδος που ακολουθήσαμε για την ρύθμιση υπερπαραμέτρων είναι η Manual Search, στην οποία ο μηχανικός ορίζει ένα σύνολο πιθανών τιμών για κάθε υπερπαράμετρο και ύστερα επιλέγει και προσαρμόζει τις τιμές χειροκίνητα έως ότου οι επιδόσεις του μοντέλου να είναι ικανοποιητικές. Παρακάτω παραθέτουμε έναν συγκεντρωτικό πίνακα με κάθε αλγόριθμο και τις αντίστοιχες τιμές υπερπαραμέτρων με τις οποίες εκτελέστηκε:

Αλγόριθμος	Υπερπαράμετροι	Τιμές
Naïve Bayes	α	0.001
K-Nearest Neighbor	n_neighbors	10
Decision Tree	Χρησιμοποιήθηκαν οι default τιμές	
Random Forest	n_jobs	-1
	criterion	“entropy”
	max_depth	8
Bagging	n_jobs	-1
BERT	N_EPOCHS	50
	BATCH_SIZE	32
	MAX_LEN	256
	LR	3e-5

Πίνακας 4.3: Αλγόριθμοι και υπερπαράμετροι – Πείραμα Νομοθεσίας

4.3.1 Naïve Bayes

Όσον αφορά τις υπερπαραμέτρους, στον αλγόριθμο Naïve Bayes μας ενδιαφέρει κυρίως η **παράμετρος εξομάλυνσης α ≥ 0** της οποίας κύριος λόγος ύπαρξης είναι η αντιμετώπιση της έλλειψης κάποιων χαρακτηριστικών στα δείγματα εκμάθησης όπως επίσης και εμποδίζει την εμφάνιση μηδενικών πιθανοτήτων σε μεταγενέστερους υπολογισμούς. Η ανάθεση α = 1 ονομάζεται εξομάλυνση Laplace (Laplace smoothing) ενώ αν α < 1 έχουμε εξομάλυνση Lidstone (Lidstone smoothing). Ο αλγόριθμος εκτελέστηκε θέτοντας **α = 0.001**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **20%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο. Στην οπτικοποίηση των αποτελεσμάτων παρακάτω, κάθε γραμμή αντιστοιχεί σε ένα κείμενο του test set ενώ το πεδίο Actual Tags αντιστοιχεί στις Ground truth ετικέτες και το πεδίο Predicted Tags αντιστοιχεί στις ετικέτες που προέβλεψε ο αλγόριθμος (το κείμενο κάθε νόμου δεν το δείξαμε για να είναι πιο ευδιάκριτες οι ετικέτες):

	Actual Tags	Predicted Tags
10	(εμπορική συμφωνία, συμφωνία συνεργασίας (ΕΕ))	(εμπορική συμφωνία,)
7	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπόλ, αποζημίωση και απόδοση εξόδων, ασφάλεια στην εργασία, δημόσια ασφάλεια, διαβίβαση δεδομένων, διακινούμενος εργαζόμενος, διαρθρωτικά ταμεία, διοίκηση προσωπικού, διοικητικό συμβούλιο, επισήμανση, θαλάσσια ασφάλεια, θαλάσσια μεταφορά, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, λιμενικές εγκαταστάσεις, μεταφορά επιβατών, μισθός, πλοίο, πολιτική συνεργασί...	(Ευρωπαϊκό Κοινοβούλιο, Ευρωπόλ, Κάτω Χώρες, δίκαιο της ΕΕ, δημόσια ασφάλεια, διακινούμενος εργαζόμενος, επισήμανση, επιτροπολογία, θαλάσσια ασφάλεια, θαλάσσια μεταφορά, κρατικές ενισχύσεις, λιμενικές εγκαταστάσεις, μεταφορά εμπορευμάτων, μεταφορά επιβατών, πιστωτικό ίδρυμα, πλοίο, πολιτική του ανταγωνισμού, προσωπικό πληρώματος, στρατός, συμβουλευτική επιτροπή (ΕΕ), συνθήκες εργασίας, τηλεπικ...
0	(Ευρωπαϊκό Κοινοβούλιο, Ευρωπόλ, αστυνομική συνεργασία της ΕΕ, ασφάλεια των μεταφορών, γενικός γραμματέας του οργάνου, γενικός προϋπολογισμός (ΕΕ), δίκαιο της ΕΕ, δίκτυο διαβίβασης, δίκτυο πληροφόρησης, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διαχείριση, δικαστική συνεργασία της ΕΕ σε ποινικές υποθέσεις, διοίκηση προσωπικού, διοικητική συνεργασία, διορισμός των μελών, είσοδος αλλοδα...	(Κάτω Χώρες, έλεγχος της ΕΕ, έντυπο, αερολιμένας, αναγνώριση διπλωμάτων, ανταλλαγή πληροφοριών, αποζημίωση και απόδοση εξόδων, αστυνομική συνεργασία της ΕΕ, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, δημόσια υγεία, διοικητική διατύπωση, είσοδος αλλοδαπών, επαγγελματικά προσόντα, θαλάσσια ασφάλεια, θαλάσσια μεταφορά, λιμενικές εγκαταστάσεις, μισθός, πλοίο, πολιτική για την υγεία, πολιτι...

Εικόνα 4.9: Οπτικοποίηση του Naïve Bayes - Νομοθεσία

4.3.2 K-Nearest Neighbor (kNN)

Ο αλγόριθμος εκτελέστηκε θέτοντας την παράμετρο **n_neighbors**, που εκφράζει τον αριθμό των γειτόνων, **ίση με 10** (η προκαθορισμένη τιμή είναι 5). Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **40%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο. Παρακάτω μία οπτικοποίηση των αποτελεσμάτων:

	Actual Tags	Predicted Tags
8	(Δικαστήριο της Ευρωπαϊκής Ένωσης, Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκό Κοινοβούλιο, έλεγχος της ΕΕ, αερολιμένας, βούτυρο, γενικός γραμματέας του οργάνου, γενικός προϋπολογισμός (ΕΕ), δίκτυο διαβίβασης, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διαβίβαση δεδομένων, διαρθρωτικά ταμεία, διαχείριση, διαχείριση των αποβλήτων, διοίκηση προσωπικού, διοικητική οργάνωση, διορισμός των μελών,...	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπόλ, έλεγχος της ΕΕ, ανταλλαγή πληροφοριών, απαίτηση, αποζημίωση και απόδοση εξόδων, γενικός γραμματέας του οργάνου, γενικός προϋπολογισμός (ΕΕ), δημόσια διοίκηση της Κοινότητας, διαβίβαση δεδομένων, διακινούμενος εργαζόμενος, διαρθρωτικά ταμεία, διαχείριση, δικαίωμα παραμονής, διοίκηση προσωπικού, εισαγωγή (ΕΕ), εκτέλεση του προϋπ...
9	(Ευρωπαϊκή Ένωση, Ευρωπόλ, αερολιμένας, δημόσια ασφάλεια, διαχείριση, διαχείριση των αποβλήτων, δράση της ΕΕ, επισήμανση, ηλεκτρική ενέργεια, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, οικοδομικά υλικά, περιφερειακές ενισχύσεις, πλοίο, σύμβαση έργων, υποδομή μεταφορών, χρηματοδοτικό μέσο της ΕΕ, όχημα δημοσίας χρήσεως)	(έντυπο, αερολιμένας, δημόσια διοίκηση της Κοινότητας, διαρθρωτικά ταμεία, διαχείριση των αποβλήτων, διοικητική διατύπωση, διορισμός των μελών, εισαγωγή (ΕΕ), ελεύθερη κυκλοφορία των κεφαλαίων, εμπορία, ενεργειακή πολιτική, ηλεκτρική ενέργεια, καθεστώς ενισχύσεων, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, λιμενικές εγκαταστάσεις, οικοδομικά υλικά, περιβαλλοντική πολιτική, περιφερειακές ε...
4	(οικονομικό έτος,)	(δίκαιο της ΕΕ, εισαγωγή (ΕΕ), οικονομικό έτος)

Εικόνα 4.10: Οπτικοποίηση του K-Nearest Neighbor - Νομοθεσία

4.3.3 Decision Tree

Ο αλγόριθμος εκτελέστηκε **χωρίς ρύθμιση κάποιας υπερπαράμετρου**, δηλαδή με τις default τιμές που παρέχει η βιβλιοθήκη sklearn. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **20%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
8	(Δικαστήριο της Ευρωπαϊκής Ένωσης, Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκό Κοινοβούλιο, έλεγχος της ΕΕ, αερολιμένας, βούτυρο, γενικός γραμματέας του οργάνου, γενικός προϋπολογισμός (ΕΕ), δίκτυο διαβίβασης, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διαβίβαση δεδομένων, διαρθρωτικά ταμεία, διαχείριση, διαχείριση των αποβλήτων, διοίκηση προσωπικού, διοικητική οργάνωση, διορισμός των μελών, δράση της ΕΕ, εκπαιδευτικό ίδρυμα, εκτέλεση του προϋπολογισμού, επισήμανση, εταιρικό δίκαιο, θεσμική αρμοδιότητα (ΕΕ), καθεστώς ενισχύσεων, κρατικές ενισχύσεις, μισθός, οικοδομικά υλικά, οικονομικό έτος, περιφερειακές ενισχύσεις, πληροφόρηση των εργαζομένων, πολιτική μεταφορών, πολιτιστική πολιτική, σύμβαση, σύμβαση έργων, τηλεπικοινωνία)	(δημόσια διοίκηση της Κοινότητας, διαχείριση των αποβλήτων, διοικητική διατύπωση, εισαγωγή (ΕΕ), εκπαιδευτικό σύστημα, περιβαλλοντική πολιτική, πολιτική συνεργασίας, πρόστιμο, συμβουλευτική επιτροπή (ΕΕ))
13	(γενικός προϋπολογισμός (ΕΕ),)	(εισαγωγή (ΕΕ),)
9	(Ευρωπαϊκή Ένωση, Ευρωπόλ, αερολιμένας, δημόσια ασφάλεια, διαχείριση, διαχείριση των αποβλήτων, δράση της ΕΕ, επισήμανση, ηλεκτρική ενέργεια, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, οικοδομικά υλικά, περιφερειακές ενισχύσεις, πλοίο, σύμβαση έργων, υποδομή μεταφορών, χρηματοδοτικό μέσο της ΕΕ, όχημα δημοσίας χρήσεως)	(Ευρωπόλ, αποζημίωση και απόδοση εξόδων, γενικός γραμματέας του οργάνου, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διεύρυνση της ΕΕ, διοίκηση προσωπικού, διοικητική οργάνωση, διοικητικό συμβούλιο, δράση της ΕΕ, εκπαιδευτική πολιτική, εκτέλεση του προϋπολογισμού, ελεύθερη κυκλοφορία των κεφαλαίων, εμπορική συμφωνία, επαγγελματικά προσόντα, εταιρική συμμετοχή, ευρωπαϊκή κοινωνική πολιτική, κοινοτικός προϋπολογισμός, κρατικές ενισχύσεις, ποινική διαδικασία, ποινική κύρωση, πολιτιστική πολιτική, πρόστιμο, συμβουλευτική επιτροπή (ΕΕ), συνθήκες εργασίας, φορολογία, φορολογική απαλλαγή, χρηματοπιστωτικές ρυθμίσεις)

Εικόνα 4.11: Οπτικοποίηση του Decision Tree – Νομοθεσία

4.3.4 Random Forest

Όσον αφορά τις υπερπαραμέτρους, ρυθμίσαμε την παράμετρο **n_jobs**, που εκφράζει τον αριθμό των εργασιών που γίνονται παράλληλα, **ίση με -1** για χρήση όλων των επεξεργαστών. Οι άλλες 2 παράμετροι που ρυθμίσαμε είναι η παράμετρος **criterion**, που εκφράζει την συνάρτηση που μετράει την ποιότητα ενός διαχωρισμού (split) και θέσαμε ίση με **"entropy"** (για το Κέρδος Πληροφορίας Shannon) και η παράμετρος **max_depth**, που εκφράζει το μέγιστο βάθος του δένδρου και θέσαμε **ίση με 8**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **50%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
12	(Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπαϊκό Κοινοβούλιο, Ευρωπόλ, έλεγχος της ΕΕ, έντυπο, απαίτηση, ασφαλιστική εταιρεία, δίκαιο του ανταγωνισμού, δίκτυο διαβίβασης, δημόσια διοίκηση της Κοινότητας, διοικητική διατύπωση, διοικητική οργάνωση, εισαγωγή (ΕΕ), εκτέλεση του προϋπολογισμού, εναρμόνιση κοινωνικών ασφαλίσεων, εξέλεγξη λογαριασμών, επένδυση, επαγγελματικά προσόντα, επεξεργασία πληροφοριών...	(Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπόλ, έντυπο, ανταλλαγή πληροφοριών, αποζημίωση και απόδοση εξόδων, γενικός γραμματέας του οργάνου, δίκαιο της ΕΕ, δημόσια διοίκηση της Κοινότητας, διαχείριση των αποβλήτων, διοίκηση προσωπικού, διοικητική διατύπωση, διοικητική δικονομία, διοικητικό συμβούλιο, εισαγωγή (ΕΕ), εξέλεγξη λογαριασμών, επισήμανση, εταιρικό δίκαιο, καθεστώς ενισχύσεων, κοινοποίηση τω...
13	(γενικός προϋπολογισμός (ΕΕ),)	(εισαγωγή (ΕΕ), εμπορική συμφωνία, οικονομικό έτος)
1	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπαϊκό Κοινοβούλιο, Κάτω Χώρες, έλεγχος της ΕΕ, αναγνώριση διπλωμάτων, αποζημίωση και απόδοση εξόδων, δημόσια διοίκηση της Κοινότητας, διαχείριση, διαχείριση των αποβλήτων, διοίκηση προσωπικού, διορισμός των μελών, εμπορία, ενιαία αγορά, εταιρικό δίκαιο, ευεργέτημα πενίας, καθεστώς ενισχύσεων, κοινοποίηση των δεδομένων, κοινωνική πα...	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπόλ, έντυπο, ανταλλαγή πληροφοριών, δημόσια διοίκηση της Κοινότητας, διοίκηση προσωπικού, διοικητική διατύπωση, διοικητική δικονομία, διοικητικό συμβούλιο, εισαγωγή (ΕΕ), καθεστώς ενισχύσεων, κρατικές ενισχύσεις, μισθός, πρόγραμμα της ΕΕ, συμβουλευτική επιτροπή (ΕΕ))

Εικόνα 4.12: Οπτικοποίηση του Random Forest - Νομοθεσία

4.3.5 Bagging

Ο αλγόριθμος εκτελέστηκε θέτοντας την παράμετρο **n_jobs**, που εκφράζει τον αριθμό των εργασιών που γίνονται παράλληλα, **ίση με -1**, το οποίο σημαίνει ότι όλοι οι επεξεργαστές

χρησιμοποιούνται. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **60%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
9	(Ευρωπαϊκή Ένωση, Ευρωπόλ, αερολιμένας, δημόσια ασφάλεια, διαχείριση, διαχείριση των αποβλήτων, δράση της ΕΕ, επισήμανση, ηλεκτρική ενέργεια, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, οικοδομικά υλικά, περιφερειακές ενισχύσεις, πλοίο, σύμβαση έργων, υποδομή μεταφορών, χρηματοδοτικό μέσο της ΕΕ, όχημα δημοσίας χρήσεως)	(έντυπο, εισαγωγή (ΕΕ), μισθός, συμβουλευτική επιτροπή (ΕΕ))
1	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπαϊκό Κοινοβούλιο, Κάτω Χώρες, έλεγχος της ΕΕ, αναγνώριση διπλωμάτων, αποζημίωση και απόδοση εξόδων, δημόσια διοίκηση της Κοινότητας, διαχείριση, διαχείριση των αποβλήτων, διοίκηση προσωπικού, διορισμός των μελών, εμπορία, ενιαία αγορά, εταιρικό δίκαιο, ευεργέτημα πενίας, καθεστώς ενισχύσεων, κοινοποίηση των δεδομένων, κοινωνική πα...	(Ευρωπόλ, έντυπο, δημόσια διοίκηση της Κοινότητας, εταιρική συμμετοχή, μισθός, οικονομικό έτος, πληροφόρηση των εργαζομένων)
5	(απαίτηση, εισαγωγή (ΕΕ), ευρωπαϊκή κοινωνική πολιτική, πιστωτικό ίδρυμα)	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπόλ, ανταλλαγή πληροφοριών, ασφαλιστική εταιρεία, δίκαιο του ανταγωνισμού, εισαγωγή (ΕΕ), εξέλεγξη λογαριασμών, ηλεκτρονικό χρήμα, πίστη, πιστωτική αγορά, πιστωτικό ίδρυμα, πρόστιμο, συμβουλευτική επιτροπή (ΕΕ), τραπεζική δραστηριότητα, φορολογία, χρηματοπιστωτικές ρυθμίσεις)

Εικόνα 4.13:Οπτικοποίηση του Bagging - Νομοθεσία

4.3.6 BERT

Όσον αφορά το BERT, η υλοποίηση του κώδικα ήταν αρκετά πιο πολύπλοκη και μακροσκελής. Χρησιμοποιήθηκαν οι βιβλιοθήκες Hugging Face Transformers (για το μοντέλο BERT και τον Tokenizer), PyTorch (framework για βαθιά μάθηση αλλά και για προετοιμασία του dataset), PyTorch Lightning (ορισμός του μοντέλου και εκπαίδευση) και η sklearn για τον διαχωρισμό του dataset και για τις μετρικές. Χρησιμοποιήθηκε η ελληνική έκδοση του BERT, δηλαδή το GreekBERT (nlpaueb/bert-base-greek-uncased-v1) [51] του οποίου η προεκπαίδευση (pretraining) έχει γίνει με τα εξής κείμενα:

- Το Ελληνικό μέρος της Wikipedia
- Το Ελληνικό μέρος του European Parliament Proceedings Parallel Corpus
- Το Ελληνικό μέρος του OSCAR, μία καθαρισμένη έκδοση του Common Crawl

Κατ'αυτόν τον τρόπο, το μοντέλο έχει μία καλή κατανόηση της Ελληνικής γλώσσας, αλλά πιθανώς πολλές νομικές λέξεις που περιλαμβάνονται στα κείμενά μας να μην τις έχει δει στην φάση της προεκπαίδευσης. Έτσι, χρειάζεται να τελειοποιήσουμε (fine-tune) το μοντέλο μας στο δικό μας dataset για να το κατανοήσει και να γίνει καλύτερο στο πρόβλημα της κατηγοριοποίησης. Ο τρόπος για να γίνει αυτό, είναι προσθέτοντας ένα classification layer πάνω από τον πυρήνα του μοντέλου και μετά εκπαιδεύοντας όλο το μοντέλο με το dataset μας. Το μοντέλο βγάζει ως έξοδο ένα διάνυσμα μήκους 768 για κάθε λέξη (token) και επίσης για το pooled output (CLS). Το pooled output στο τέλος του κύκλου εκπαίδευσης του μοντέλου έχει συγκεντρώσει αρκετές πληροφορίες για το πρόβλημα και βοηθάει στο να γίνουν οι προβλέψεις. Ανάλογα με το πόσες ετικέτες έχουμε στο πρόβλημά μας, βάζουμε ένα γραμμικό layer με τον ίδιο αριθμό εξόδων πάνω απο τις 768 εξόδους από το BERT. Τέλος, εφόσον είχαμε να αντιμετωπίσουμε ένα πρόβλημα πολλαπλών ετικετών, αρχικά σκεφτήκαμε να χρησιμοποιήσουμε μία σιγμοειδή συνάρτηση ενεργοποίησης για την τελική έξοδο και μία συνάρτηση κόστους Binary Cross-Entropy. Ωστόσο, το documentation του Pytorch, το οποίο και χρησιμοποιήσαμε, προτείνει την χρήση της συνάρτησης BCEWithLogitsLoss() η οποία συνδυάζει ένα σιγμοειδές layer και την BCELoss σε μία μόνο κλάση.
Για την εκπαίδευση χρησιμοποιήθηκαν οι GPUs που μας παρείχε το Google Colab, ενώ χρησιμοποιήθηκε η λογική του checkpointing, όπου αποθηκεύουμε ένα checkpoint του μοντέλου μας για μετέπειτα συνέχιση της εκπαίδευσης, καθότι η χρήση GPUs έχει χρονικά όρια στο περιβάλλον του Colab.

Στην περίπτωση του BERT, υπερπαράμετροι είναι ο αριθμός των epochs (**N_EPOCHS**) για τον οποίο εκπαιδεύεται το μοντέλο, το μέγεθος της "παρτίδας" (**BATCH_SIZE**), το μέγιστο μήκος σε λέξεις/tokens (**MAX_LEN**) και τέλος ο ρυθμός με τον οποίο μαθαίνει το μοντέλο (**learning rate - LR**). Εκπαιδεύσαμε το μοντέλο μας για **50 epochs** συνολικά, κρατώντας το μοντέλο με την μικρότερη **validation loss**. Το καλύτερο μοντέλο επιτεύχθηκε στο **epoch 45** με **val_loss = 0.03**. Όσον αφορά τις υπόλοιπες παραμέτρους, θέσαμε **BATCH_SIZE = 32**, **MAX_LEN = 256** (για οποιαδήποτε τιμή μεγαλύτερη από 300 είχαμε προβλήματα με την μνήμη) και τέλος **LR = 3e-5**. Για το BERT, κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από **20%**.

	Actual Tags	Predicted Tags
9	(Ευρωπαϊκή Ένωση, Ευρωπόλ, αερολιμένας, δημόσια ασφάλεια, διαχείριση, διαχείριση των αποβλήτων, δράση της ΕΕ, επισήμανση, ηλεκτρική ενέργεια, κοινοποίηση των δεδομένων, κρατικές ενισχύσεις, οικοδομικά υλικά, περιφερειακές ενισχύσεις, πλοίο, σύμβαση έργων, υποδομή μεταφορών, χρηματοδοτικό μέσο της ΕΕ, όχημα δημοσίας χρήσεως)	(έντυπο, ανταλλαγή πληροφοριών, δίκαιο της ΕΕ, επισήμανση, καθεστώς ενισχύσεων, κρατικές ενισχύσεις, οικοδομικά υλικά, περιβαλλοντική πολιτική, περιφερειακές ενισχύσεις, υποδομή μεταφορών)
11	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, έλεγχος της ΕΕ, αερολιμένας, ανταλλαγή πληροφοριών, αποζημίωση και απόδοση εξόδων, ασφαλιστική εταιρεία, βούτυρο, γενικός γραμματέας του οργάνου, δίκαιο της ΕΕ, διακινούμενος εργαζόμενος, διοίκηση προσωπικού, διοικητική δικονομία, διοικητική οργάνωση, διοικητικό συμβούλιο, ελεύθερη παροχή υπηρεσιών, εξέλεγξη λογαριασμών, επένδυση, επιστροφή κατά την εξαγωγή, εταιρεία επενδύσεων, εταιρικό δίκαιο, κεντρική τράπεζα, κινητές αξίες, κοινοποίηση των δεδομένων, κοινοτικός προϋπολογισμός, κρατικές ενισχύσεις, μισθός, νομικό καθεστώς, οικονομικό έτος, πίστη, παράδοση, πιστωτική αγορά, πιστωτικό ίδρυμα, πολιτική άμυνα, πρόστιμο, συμβουλευτική επιτροπή (ΕΕ), τραπεζική δραστηριότητα, φάρμακο, χρηματοπιστωτικές ρυθμίσεις)	(Ευρωπαϊκή Κεντρική Τράπεζα, Ευρωπαϊκή Τράπεζα Επενδύσεων, ανταλλαγή πληροφοριών, αποζημίωση και απόδοση εξόδων, ασφαλιστική εταιρεία, δίκαιο της ΕΕ, δημόσια διοίκηση της Κοινότητας, ελεύθερη παροχή υπηρεσιών, εξέλεγξη λογαριασμών, εταιρική συμμετοχή, εταιρικό δίκαιο, ευρωπαϊκή κοινωνική πολιτική, ηλεκτρονικό χρήμα, καθεστώς ενισχύσεων, κινητές αξίες, κοινοποίηση των δεδομένων, μισθός, νομικό καθεστώς, πίστη, πιστωτικό ίδρυμα, πρόστιμο, συμβουλευτική επιτροπή (ΕΕ), τραπεζική δραστηριότητα, φάρμακο, φορολογία)
0	(Ευρωπαϊκό Κοινοβούλιο, Ευρωπόλ, αστυνομική συνεργασία της ΕΕ, ασφάλεια των μεταφορών, γενικός γραμματέας του οργάνου, γενικός προϋπολογισμός (ΕΕ), δίκαιο της ΕΕ, δίκτυο διαβίβασης, δίκτυο πληροφόρησης, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διαχείριση, δικαστική συνεργασία της ΕΕ σε ποινικές υποθέσεις, διοίκηση προσωπικού, διοικητική συνεργασία, διορισμός των μελών, είσοδος αλλοδαπών, εξέλεγξη λογαριασμών, επένδυση, επιτροπολογία, κρατικές ενισχύσεις, μισθός, οδικές μεταφορές, πλοίο, ποινική κύρωση, πολιτική άμυνα, προστασία δεδομένων, προσωπικό πληρώματος, προϊόν διατροφής, σιδηροδρομικές μεταφορές, στρατιωτικό προσωπικό, στρατός, σύμβαση προμηθειών, τραπεζική δραστηριότητα, φαρμακευτικό προϊόν)	(Ευρωπαϊκή Τράπεζα Επενδύσεων, Ευρωπόλ, αλλοδαπός, δίκαιο της ΕΕ, δίκτυο διαβίβασης, δημόσια ασφάλεια, δημόσια διοίκηση της Κοινότητας, διαχείριση, εξέλεγξη λογαριασμών, επισήμανση, κοινωνική παροχή, μισθός, ποινική κύρωση, πολιτική άμυνα, προστασία δεδομένων, πρόστιμο, σιδηροδρομικές μεταφορές, στρατιωτικό προσωπικό, στρατός, σύμβαση έργων, φάρμακο, φαρμακευτικό προϊόν)

Εικόνα 4.14: Οπτικοποίηση του BERT - Νομοθεσία

4.4 Παράμετροι Αξιολόγησης

Ο τρόπος αξιολόγησης που ακολουθείται στην βιβλιογραφία είναι η εστίαση στις μετρικές **precision**, **recall**, και **F1 score** για καθεμία από τις εξής τρεις κατηγορίες: **Micro**, **Macro** και **Weighted macro** [52]. Πρωτού αναλύσουμε κάθε μετρική χωριστά, θα εισάγουμε κάποιες έννοιες. Σε ένα πρόβλημα Κατηγοριοποίησης Πολλαπλών Ετικετών, όπου είσοδος είναι ένα κείμενο και ένας ταξινομητής καλείται να αναθέσει ετικέτες, υπάρχουν δύο είδη σφαλμάτων που μπορεί να κάνει αυτός ο ταξινομητής:

1. Τα **false positives** (FP), γνωστά και ως σφάλματα τύπου I. Τέτοιου είδους σφάλματα συναντάμε όταν ο ταξινομητής προβλέπει μία ετικέτα που δεν υπάρχει στην Ground truth της εισόδου.
2. Τα **false negatives** (FN), γνωστά και ως σφάλματα τύπου II. Τέτοιου είδους σφάλματα συναντάμε όταν ο ταξινομητής αποτυγχάνει να προβλέψει μία ετικέτα που υπάρχει στην Ground truth της εισόδου.

Ομοίως, υπάρχουν δύο τρόποι που οι προβλέψεις του ταξινομητή μπορούν να είναι σωστές:

1. Τα **true positives** (TP), που συναντώνται όταν ο ταξινομητής προβλέπει ορθά την ύπαρξη μιας ετικέτας.
2. Τα **true negatives** (TN), που συναντώνται όταν ο ταξινομητής προβλέπει ορθά την ανυπαρξία μιας ετικέτας.

Η μετρική **precision** εκφράζει την αναλογία των σωστών προβλέψεων ανάμεσα σε όλες τις προβλέψεις μίας συγκεκριμένης κλάσης. Με άλλα λόγια, είναι ο λόγος των true positives προς όλες τις positive (false + true) προβλέψεις.

$$Precision = \frac{TP}{FP + TP}$$

Εξίσωση 4.1: Μετρική precision

Η μετρική **recall** εκφράζει την αναλογία παραδειγμάτων μιας συγκεκριμένης κλάσης που έχουν προβλεφθεί από το μοντέλο πως ανήκουν σε αυτή την κλάση. Με άλλα λόγια, είναι η αναλογία των true positives ανάμεσα σε όλα τα αληθή παραδείγματα.

$$Recall = \frac{TP}{FN + TP}$$

Εξίσωση 4.2: Μετρική recall

Η μετρική **F1 score** για μια συγκεκριμένη κλάση είναι ο αρμονικός μέσος της μετρικής precision και της μετρικής recall.

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Εξίσωση 4.3: Μετρική F1 score

Έχοντας ορίσει τις μετρικές precision, recall και F1 score για κάθε κλάση του προβλήματος, δημιουργείται η ανάγκη για την μετάβαση σε έναν "συγκεντρωτικό" δείκτη για κάθε μετρική (ώστε ένας ταξινομητής να χαρακτηρίζεται τελικά από μία τιμή precision, μία τιμή recall, και μία τιμή F1 score). Η μετάβαση – συνάθροιση αυτή γίνεται με την βοήθεια της παραμέτρου average που παρέχει η βιβλιοθήκη sklearn για κάθε μετρική. Όταν η παράμετρος average έχει τιμή "micro", οι μετρικές υπολογίζονται καθολικά μετρώντας τα συνολικά true positives, false negatives, και false positives. Όταν έχει τιμή "macro", υπολογίζονται οι μετρικές για κάθε ετικέτα, και γίνεται εύρεση του αστάθμητου μέσου τους. Σε αυτή την περίπτωση δεν λαμβάνονται υπόψη ανισορροπίες των ετικετών. Τέλος όταν η παράμετρος average έχει τιμή "weighted", υπολογίζονται οι μετρικές για κάθε ετικέτα, και γίνεται εύρεση του σταθμισμένου μέσου όρου τους με βάση τον αριθμό των αληθινών στιγμιοτύπων για κάθε ετικέτα (ο αριθμός αυτός λέγεται αλλιώς και support). Με αυτόν τον τρόπο εξαλείφεται η αδυναμία της "macro" σχετικά με τις ανισορροπίες των ετικετών και υπάρχει πιθανότητα η τελική τιμή F-score να μην είναι ανάμεσα στις precision και recall.

4.5 Αποτελέσματα

Συγκρίνοντας τα αποτελέσματα των αλγορίθμων μας σε σχέση με εκείνα του εργαλείου **PyEuroVoc** στο test set που απαρτίζεται από 15 κείμενα, προκύπτει ο παρακάτω πίνακας με βάση τις μετρικές:

(Οι τιμές είναι στρογγυλοποιημένες στο τρίτο ψηφίο και η σύμβαση που ακολουθείται είναι η εξής: DT = Decision Tree, NB = Naïve Bayes, RF = Random Forest, kNN = K-Nearest Neighbor, Prec. = Precision, Rec. = Recall)

	Micro scores			Macro scores			Weighted macro scores		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
DT	0.353	0.173	0.232	0.188	0.139	0.142	0.303	0.173	0.197
Bagging	0.269	0.137	0.182	0.132	0.106	0.104	0.199	0.137	0.144
NB	0.301	**0.425**	0.352	0.253	**0.357**	0.271	0.364	**0.425**	0.363
RF	0.384	0.271	0.318	0.128	0.161	0.124	0.221	0.271	0.218
kNN	0.31	0.392	0.346	0.239	0.3	0.236	0.348	0.392	0.325
BERT	**0.611**	0.405	**0.487**	**0.371**	0.321	**0.314**	**0.531**	0.405	**0.422**

Πίνακας 4.4: Αποτελέσματα μετρικών – Πείραμα Νομοθεσίας (PyEuroVoc)

Θα αναλύσουμε τα αποτελέσματα τόσο σε επίπεδο μετρικών (κάθετα στον πίνακα) όσο και σε επίπεδο αλγορίθμων (οριζόντια στον πίνακα). Ξεκινώντας σε επίπεδο μετρικών, έχουμε τα εξής:

- Μετρική **micro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.611). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Random Forest (0.384), Decision Tree (0.353), K-Nearest Neighbor (0.31) και Naive Bayes (0.301) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.269).
- Μετρική **micro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.425). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.405), K-Nearest Neighbor (0.392), Random Forest (0.271), Decision Tree (0.173) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.137)
- Μετρική **micro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.487). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.352), K-Nearest Neighbor (0.346), Random Forest (0.318), Decision Tree (0.232) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.182).
- Μετρική **macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.371). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.253), K-Nearest Neighbor (0.239), Decision Tree (0.188), Bagging (0.132) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Random Forest (0.128).
- Μετρική **macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.357). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.321), K-Nearest Neighbor (0.3), Random Forest (0.161), Decision Tree (0.139) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.106).
- Μετρική **macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.314). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.271), K-Nearest Neighbor (0.236), Decision Tree (0.142), Random Forest (0.124) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.104).
- Μετρική **weighted macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.531). Ακολουθούν κατά φθίνουσα

σειρά οι αλγόριθμοι Naïve Bayes (0.364), K-Nearest Neighbor (0.348), Decision Tree (0.303), Random Forest (0.221) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.199).

- Μετρική **weighted macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.425). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.405), K-Nearest Neighbor (0.392), Random Forest (0.271), Decision Tree (0.173) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.137).
- Μετρική **weighted macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.422). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.363), K-Nearest Neighbor (0.325), Random Forest (0.218), Decision Tree (0.197) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.144).

Προχωρώντας σε επίπεδο αλγορίθμων, έχουμε τα εξής:

- Αλγόριθμος **Decision Tree**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro precision (0.353). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.303), micro F1 score (0.232), weighted macro F1 score (0.197), macro precision (0.188), micro recall και weighted macro recall (0.173), macro F1 score (0.142) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro recall (0.139)
- Αλγόριθμος **Bagging**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro precision (0.269). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.199), micro F1 score (0.182), weighted macro F1 score (0.144), micro recall και weighted macro recall (0.137), macro precision (0.132), macro recall (0.106) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro F1 score (0.104)
- Αλγόριθμος **Naïve Bayes**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.425). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.364), weighted macro F1 score (0.363), macro recall (0.357), micro F1 score (0.352), micro precision (0.301), macro F1 score (0.271) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.253)
- Αλγόριθμος **Random Forest**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro precision (0.384). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές micro F1 score (0.318), micro recall και weighted macro recall (0.271), weighted macro precision (0.221), weighted macro F1 score (0.218), macro recall (0.161), macro precision (0.128) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro F1 score (0.124)
- Αλγόριθμος **K-Nearest Neighbor**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.392). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.348), micro F1 score (0.346), weighted macro F1 score (0.325), micro precision (0.31),

macro recall (0.3), macro precision (0.239) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro F1 score (0.236).

- Αλγόριθμος **BERT**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro precision (0.611). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.531), micro F1 score (0.487), weighted macro F1 score (0.422), micro recall και weighted macro recall (0.405), macro precision (0.371), macro recall (0.321) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro F1 score (0.314)

Για την σύγκριση με τις κατηγορίες που είναι ορισμένες από το **σύστημα Μίτος**, στο πλαίσιο του ίδιου πειράματος, τα αποτελέσματα με βάση τις μετρικές είναι τα εξής:

	Micro scores			Macro scores			Weighted macro scores		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
DT	0.173	0.289	0.217	0.102	0.139	0.11	0.221	0.289	0.234
Bagging	0.122	0.211	0.154	0.045	0.096	0.057	0.104	0.211	0.129
NB	0.102	**0.489**	0.169	0.085	**0.228**	0.118	0.196	**0.489**	0.267
RF	0.148	0.356	0.209	0.065	0.142	0.074	0.173	0.356	0.195
kNN	0.101	0.433	0.164	0.09	0.194	0.11	0.198	0.433	0.246
BERT	**0.187**	0.422	**0.259**	**0.112**	0.186	**0.128**	**0.272**	0.422	**0.303**

Πίνακας 4.5: Αποτελέσματα μετρικών – Πείραμα Νομοθεσίας (MITOS)

Εδώ παρατηρούμε ότι κάποιες τιμές είναι αρκετά πιο χαμηλά σε σχέση με πριν, το οποίο είναι λογικό αφού η εκπαίδευση των μοντέλων δεν έχει γίνει με βάση κείμενα κατηγοριοποιημένα από το σύστημα Μίτος αλλά απλώς χρησιμοποιούμε κάποια από αυτά στο πλαίσιο της αξιολόγησης. Σε επίπεδο μετρικών, έχουμε τα εξής:

- Μετρική **micro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.187). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Decision Tree (0.173), Random Forest (0.148), Bagging (0.122) και Naive Bayes (0.102) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.101).
- Μετρική **micro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.489). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι K-Nearest Neighbor (0.433), BERT (0.422), Random Forest (0.356), Decision Tree (0.289) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.211)
- Μετρική **micro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.259). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Decision Tree (0.217), Random Forest (0.209), Naïve Bayes (0.169), K-Nearest Neighbor (0.164) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.154).
- Μετρική **macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.112). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Decision Tree (0.102), K-Nearest Neighbor (0.09), Naïve Bayes (0.085), Random Forest (0.065) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.045).

- Μετρική **macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.228). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι K-Nearest Neighbor (0.194), BERT (0.186), Random Forest (0.142), Decision Tree (0.139) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.096).
- Μετρική **macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.128). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.118), K-Nearest Neighbor και Decision Tree (0.11), Random Forest (0.074) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.057).
- Μετρική **weighted macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.272). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Decision Tree (0.221), K-Nearest Neighbor (0.198), Naïve Bayes (0.196), Random Forest (0.173) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.104).
- Μετρική **weighted macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος Naïve Bayes (0.489). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.405), K-Nearest Neighbor (0.433), BERT (0.422), Random Forest (0.356) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.211).
- Μετρική **weighted macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.303). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.267), K-Nearest Neighbor (0.246), Decision Tree (0.234), Random Forest (0.195) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Bagging (0.129).

Σε επίπεδο αλγορίθμων, έχουμε τα εξής:

- Αλγόριθμος **Decision Tree**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.289). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.234), weighted macro precision (0.221), micro F1 score (0.217), micro precision (0.173), macro recall (0.139), macro F1 score (0.11) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.102)
- Αλγόριθμος **Bagging**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.211). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές micro F1 score (0.154), weighted micro F1 score (0.129), micro precision (0.122), weighted micro precision (0.104), macro recall (0.096), macro F1 score (0.057) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.045)
- Αλγόριθμος **Naïve Bayes**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.489). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.267), macro recall (0.228), weighted micro precision (0.196), micro F1 score (0.169), macro F1 score (0.118), micro precision (0.102) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.085)

- Αλγόριθμος **Random Forest**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.356). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές micro F1 score (0.209), weighted macro F1 score (0.195), weighted macro precision (0.173), micro precision (0.148), macro recall (0.142), macro F1 score (0.074) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.065)
- Αλγόριθμος **K-Nearest Neighbor**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.433). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.246), weighted macro precision (0.198), macro recall (0.194), micro F1 score (0.164), macro F1 score (0.11), micro precision (0.101) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.09).
- Αλγόριθμος **BERT**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.422). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.303), weighted macro precision (0.272), micro F1 score (0.259), micro precision (0.187), macro recall (0.186), macro F1 score (0.128) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.112)

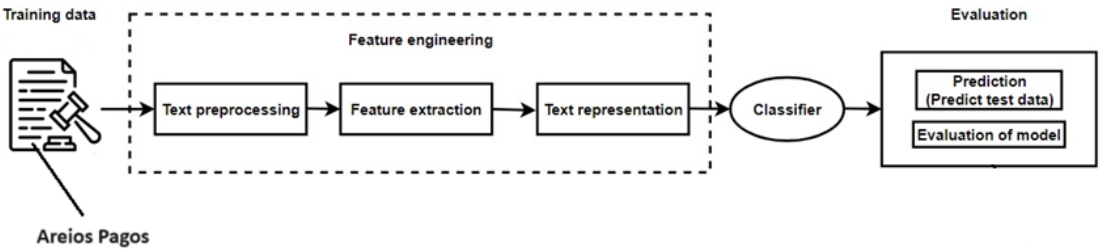
4.6 Σύνοψη συμπερασμάτων αξιολόγησης

Συνοψίζοντας, συγκρίνοντας τα αποτελέσματα των αλγορίθμων μας σε σχέση με τις ετικέτες από το εργαλείο PyEuroVoc, σε επίπεδο μετρικών, σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, macro F1 score, weighted macro precision, και weighted macro F1 score ο καλύτερος αλγόριθμος είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, και weighted macro recall ο καλύτερος αλγόριθμος είναι ο Naïve Bayes. Σε επίπεδο αλγορίθμων, οι αλγόριθμοι Decision Tree, Bagging, Random Forest, BERT παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιείται η μετρική micro precision, ενώ οι αλγόριθμοι Naïve Bayes και K-Nearest Neighbor παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall. Στην σύγκριση με τις κατηγορίες που είναι ορισμένες από το σύστημα Μίτος, σε επίπεδο μετρικών, ομοίως με πριν σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, macro F1 score, weighted macro precision και weighted macro F1 score ο καλύτερος αλγόριθμος είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, και weighted macro recall ο καλύτερος αλγόριθμος είναι ο Naïve Bayes. Σε επίπεδο αλγορίθμων, και οι 6 αλγόριθμοι παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall. Οι αλγόριθμοι που προτείνονται λοιπόν τελικά για την κατηγοριοποίηση της Νομοθεσίας είναι ο BERT και ο Naïve Bayes.

5

Αυτόματη Κατηγοριοποίηση Νομολογίας

Στο κεφάλαιο αυτό περιγράφουμε το δεύτερο πείραμα με το οποίο ασχοληθήκαμε το οποίο αφορά κατηγοριοποίηση κειμένων Ελληνικής Νομολογίας. Εδώ έχουμε να κάνουμε πάλι με ένα πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών αλλά οι ετικέτες προέρχονται από τον Άρειο Πάγο και όχι από τον θησαυρό Eurovoc. Τα στάδια της γενικότερης διαδικασίας κατηγοριοποίησης παρουσιάζονται παρακάτω όπως και στο πείραμα Νομοθεσίας, με την διαφορά ότι σε αυτή την περίπτωση είσοδος είναι ένα κείμενο Νομολογίας.

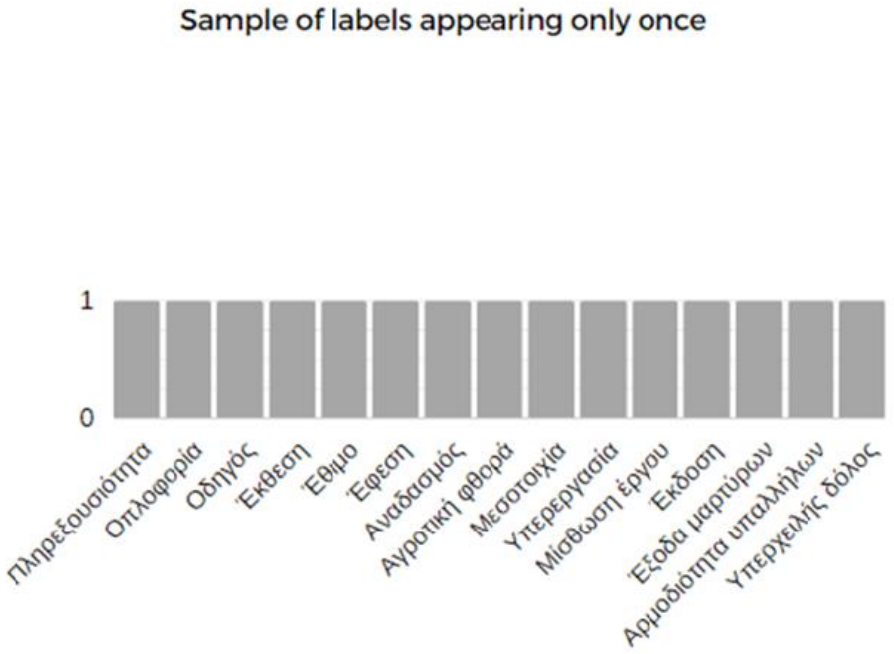


Εικόνα 5.1: Στάδια Κατηγοριοποίησης Νομολογίας

5.1 Σώμα Κειμένων – Επισημασμένο σύνολο δεδομένων αληθείας

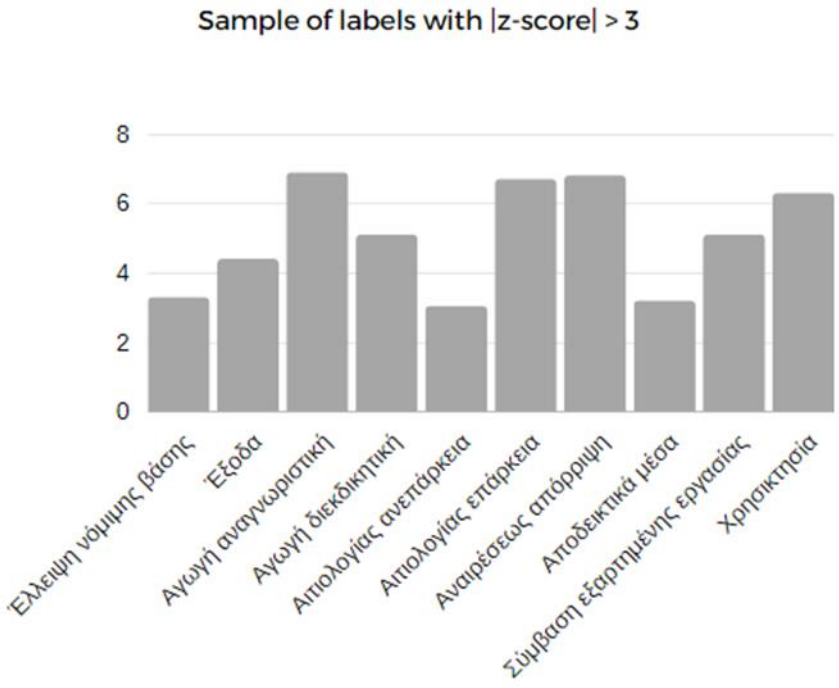
Το αρχικό dataset ήταν χίλιες (1.000) σε πλήθος αποφάσεις του Δικαστηρίου του Αρείου Πάγου από το σύνολο δεδομένων που δημιουργήθηκε στα πλαίσια της [53]. Η εργασία αυτή ασχολείται με Αυτόματη Περίληψη Δικαστικών Αποφάσεων και στο σύνολο δεδομένων που δημιουργείται παράγει μία περίληψη που αντιστοιχεί σε κάθε δικαστική απόφαση. Οι δικαστικές αποφάσεις του dataset μας είναι ήδη κατηγοριοποιημένες από τους νομικούς συντάκτες του Αρείου Πάγου. Σε αντίθεση με το προηγούμενο πείραμα, αντί να χωρίσουμε μία δικαστική απόφαση σε κομμάτια, τροφοδοτούσαμε το μοντέλο BERT με την περίληψη κάθε απόφασης. Όμοια με το πείραμα νομοθεσίας, βλέπουμε το εξής:

Ενώ συνολικά έχουμε 1.000 κείμενα, το 80% των οποίων είναι 800 κείμενα, υπάρχουν ετικέτες που δεν εμφανίζονται ούτε στο 1% αυτών των κειμένων (1% του 800 = 8 κείμενα). Οι ετικέτες αυτές είναι 343. Ως παράδειγμα, μόνο οι ετικέτες οι οποίες εμφανίζονται μόνο μία φορά σε ολόκληρο το dataset είναι 157. Βλέπουμε μερικές από αυτές παρακάτω:



Εικόνα 5.2: Παράδειγμα ετικετών που εμφανίζονται μόνο μία φορά στο dataset

Όμοια, αφαιρούμε τις ετικέτες με απόλυτη τιμή μετρικής z-score μεγαλύτερη από 3 (10 στο σύνολο). Στο παρακάτω σχήμα, οι ετικέτες αυτές βρίσκονται στον άξονα x ενώ το αντίστοιχο z-score τους φαίνεται στον άξονα y:



Εικόνα 5.3: Ετικέτες με |z-score|>3

Ο αριθμός των εναπομεινασών ετικετών του Αρείου Πάγου είναι 71. Με αυτή την αφαίρεση, προκύπτουν κάποια κείμενα χωρίς καθόλου ετικέτες. Διαγράφοντάς τα, τα κείμενα Νομολογίας είναι πλέον 649 σε αριθμό. Συγκεντρωτικά, παρακάτω παρουσιάζουμε τα έγγραφα που είχαμε αρχικά, αυτά που αφαιρέσαμε και αυτά που τελικά κρατήσαμε:

Έγγραφα	
Αρχικός αριθμός εγγράφων:	1000
Έγγραφα που απορρίφθηκαν:	351
Έγγραφα πειράματος:	649

Πίνακας 5.1: Πείραμα Νομολογίας – Έγγραφα

Στον παρακάτω πίνακα παρουσιάζουμε τα έγγραφα που χρησιμοποιήθηκαν για training, validation, testing:

Train Validation Test Split	
Αριθμός εγγράφων στο training set:	519
Αριθμός εγγράφων στο validation set:	65
Αριθμός εγγράφων στο test set:	65

Πίνακας 5.2: Πείραμα Νομολογίας – Train Validation Test Split

5.2 Προεπεξεργασία

Text preprocessing: Η διαδικασία προεπεξεργασίας όπως και στο πείραμα Νομοθεσίας είναι η εξής:

1. Μετατροπή των γραμμάτων σε πεζά
2. Διατήρηση μόνο των συμβολοσειρών με αλφαβητικό περιεχόμενο (χαρακτήρες α-ω)
3. Αφαίρεση μικρών λέξεων – διατήρηση μόνο αυτών με μήκος μεγαλύτερο του 2
4. Αφαίρεση των stopwords

Feature Extraction: Με την εξαίρεση του μοντέλου BERT, το σώμα κειμένου της στήλης text μετατρέπεται σε αριθμητική μορφή μέσω της τεχνικής TF-IDF, όπου έχουμε ορίσει τις παραμέτρους max_df = 0.8 και min_df = 5. Με αυτόν τον τρόπο, τις λέξεις οι οποίες εμφανίζονται σε περισσότερο από 80% των εγγράφων (max_df = 0.8) αλλά και τις λέξεις οι οποίες εμφανίζονται συνολικά λιγότερο από 5 φορές (min_df = 5) δεν τις λαμβάνουμε υπόψη.

5.3 Αλγόριθμοι

Όμοια με το πείραμα Νομοθεσίας, οι αλγόριθμοι οι οποίοι χρησιμοποιήθηκαν για κατηγοριοποίηση κειμένου ήταν οι **Naïve Bayes**, **K-Nearest Neighbor (kNN), Decision Tree,**

Random Forest, **Bagging** και **BERT**. Οι τιμές των υπερπαραμέτρων που χρησιμοποιήθηκαν είναι **ίδιες με το πείραμα νομοθεσίας** με την διαφορά ότι εκπαιδεύσαμε το μοντέλο **BERT** για **200 epochs**, και το μοντέλο με την χαμηλότερη **validation loss** επετεύχθη στην **epoch 103**. Παρακάτω παραθέτουμε έναν συγκεντρωτικό πίνακα με κάθε αλγόριθμο και τις αντίστοιχες τιμές υπερπαραμέτρων με τις οποίες εκτελέστηκε:

Αλγόριθμος	**Υπερπαράμετροι**	**Τιμές**
Naïve Bayes	α	0.001
K-Nearest Neighbor	n_neighbors	10
Decision Tree	Χρησιμοποιήθηκαν οι default τιμές	
Random Forest	n_jobs	-1
	criterion	“entropy”
	max_depth	8
Bagging	n_jobs	-1
BERT	N_EPOCHS	200
	BATCH_SIZE	32
	MAX_LEN	256
	LR	3e-5

Πίνακας 5.3: Αλγόριθμοι και υπερπαράμετροι – Πείραμα Νομολογίας

5.3.1 Naïve Bayes

Ο αλγόριθμος εκτελέστηκε θέτοντας **α = 0.001**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **20%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο. Στην οπτικοποίηση των αποτελεσμάτων παρακάτω, κάθε γραμμή αντιστοιχεί σε ένα έγγραφο του test set ενώ το πεδίο Actual Tags αντιστοιχεί στις Ground truth ετικέτες και το πεδίο Predicted Tags αντιστοιχεί στις ετικέτες που προέβλεψε ο αλγόριθμος (το περιεχόμενο κάθε εγγράφου νομολογίας δεν παρουσιάζεται για να είναι πιο ευδιάκριτες οι ετικέτες):

	Actual Tags	Predicted Tags
30	(Ακυρότητα απόλυτη, Τραπεζική επιταγή ακάλυπτη)	(Αναβολής αίτημα, Υπέρβαση εξουσίας)
34	(Ανθρωποκτονία από αμέλεια,)	(Ανθρωποκτονία από αμέλεια,)
39	(Κλήτευση ,)	(Απαράδεκτη συζήτηση,)
49	(Αναιρέσεως παραδοχή,)	(Αβάσιμοι λόγοι, Ακροάσεως έλλειψη, Ακυρότητα απόλυτη, Αναίρεση μερική, Αναβολής αίτημα, Αναιρέσεως λόγοι, Απάτη, Εισαγγελέας Αρείου Πάγου, Ελαφρυντικές περιστάσεις, Επιεικέστερος νόμος, Ισχυρισμός αυτοτελής, Νόμου εφαρμογή και ερμηνεία, Παραγραφή, Πραγματογνωμοσύνη, Υπέρβαση εξουσίας, Υπεξαίρεση, Υπεξαίρεση στην υπηρεσία)
62	(Έγγραφα, Ακυρότητα απόλυτη, Απάτη, Ηθική αυτουργία, Ψευδορκία μάρτυρα)	(Απάτη,)
42	(Γαίες, Νομή)	(Κυριότητα,)
13	(Ακυρότητα απόλυτη, ΕΣΔΑ, Νόμου εφαρμογή και ερμηνεία)	(Απαράδεκτο αναιρέσεως, ΕΣΔΑ)

Εικόνα 5.4: Οπτικοποίηση του Naïve Bayes - Νομολογία

5.3.2 *K-Nearest Neighbor (kNN)*

Ο αλγόριθμος εκτελέστηκε θέτοντας **n_neighbors = 10**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **40%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο. Παρακάτω μία οπτικοποίηση των αποτελεσμάτων:

	Actual Tags	Predicted Tags
9	(Ισχυρισμός αυτοτελής, Πλαστογραφία)	(Αβάσιμοι λόγοι, Αναιρέσεως λόγοι, Απάτη, Πλαστογραφία, Υπέρβαση εξουσίας)
30	(Ακυρότητα απόλυτη, Τραπεζική επιταγή ακάλυπτη)	(Ακυρότητα απόλυτη, Αναβολής αίτημα, Αναιρέσεως ανυποστήρικτο, Υπέρβαση εξουσίας)
39	(Κλήτευση ,)	(Απαράδεκτη συζήτηση,)
2	(Αοριστία αγωγής, Δημόσιο)	(Έγγραφα, Έλλειψη αιτιολογίας, Αοριστία αγωγής, Δεδικασμένο, Καταχρηστική άσκηση δικαιώματος)
17	(Σωματική βλάβη από αμέλεια,)	(Ανθρωποκτονία από αμέλεια, Νόμου εφαρμογή και ερμηνεία, Σωματική βλάβη από αμέλεια)
10	(Ακυρότητα απόλυτη, Νόμου εφαρμογή και ερμηνεία)	(Ακυρότητα απόλυτη, Αναιρέσεως λόγοι, Απάτη, Αποζημίωση, Απορρίπτει αναίρεση, Βεβαίωση ένορκη, Δεδικασμένο, Ελαφρυντικές περιστάσεις, Επίδομα αδείας, Επίδομα εορτών, Νόμου εφαρμογή και ερμηνεία, Υπεξαίρεση στην υπηρεσία)

Εικόνα 5.5: Οπτικοποίηση του K-Nearest Neighbor - Νομολογία

5.3.3 *Decision Tree*

Ο αλγόριθμος εκτελέστηκε **χωρίς ρύθμιση κάποιας υπερπαράμετρου**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **20%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
53	(Ένδικο μέσο, Δημόσιο , Προθεσμία)	(Διαθήκη,)
60	(Απάτη,)	(Πραγματογνωμοσύνη, Σωματική βλάβη από αμέλεια)
0	(Αοριστία αγωγής,)	(Αοριστία αγωγής,)
45	(Νομή,)	(Έλλειψη αιτιολογίας, Αοριστία αγωγής, Καταχρηστική άσκηση δικαιώματος, Νομή)
5	(Ένδικο μέσο,)	(Ακυρότητα απόλυτη, Απάτη, Νόμου εφαρμογή και ερμηνεία, Πλαστογραφία)
61	(Διαθήκης ακύρωση,)	(Διαθήκης ακύρωση,)
16	(Ένδικο μέσο, Δεδικασμένο, Καταχρηστική άσκηση δικαιώματος)	(Καταχρηστική άσκηση δικαιώματος,)
12	(Έγγραφα, Ακυρότητα απόλυτη, Αναίρεση μερική, ΕΣΔΑ, Υπέρβαση εξουσίας, Ψευδής καταμήνυση)	(Έλλειψη ειδικής και εμπεριστατωμένης αιτιολογίας, Δυσφήμηση συκοφαντική)

Εικόνα 5.6: Οπτικοποίηση του Decision Tree - Νομολογία

5.3.4 *Random Forest*

Ο αλγόριθμος εκτελέστηκε θέτοντας **n_jobs = -1**, **criterion = "entropy"**, και **max_depth=8**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **50%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
42	(Γαίες, Νομή)	(Έλλειψη αιτιολογίας, Βεβαίωση ένορκη, Βυζαντινορωμαϊκό Δίκαιο, Γαίες, Δεδικασμένο, Δημόσια κτήματα, Κυριότητα, Νομή)
49	(Αναιρέσεως παραδοχή,)	(Έλλειψη ειδικής και εμπεριστατωμένης αιτιολογίας, Ακυρότητα απόλυτη, Αναιρέσεως παραδοχή, Ελαφρυντικές περιστάσεις, Νόμου εφαρμογή και ερμηνεία, Παραγραφή, Υπέρβαση εξουσίας)
19	(Αναιρέσεως απαράδεκτο,)	(Αναιρέσεως απαράδεκτο,)
53	(Ένδικο μέσο, Δημόσιο , Προθεσμία)	(Απαράδεκτη συζήτηση,)
56	(Ακυρότητα απόλυτη, Ελαφρυντικές περιστάσεις, Ισχυρισμός αυτοτελής)	(Έλλειψη ειδικής και εμπεριστατωμένης αιτιολογίας, Ακροάσεως έλλειψη, Ακυρότητα απόλυτη, Αναιρέσεως απαράδεκτο, Ανθρωποκτονία από αμέλεια, Απάτη, Ελαφρυντικές περιστάσεις, Ισχυρισμός αυτοτελής, Νόμου εφαρμογή και ερμηνεία, Παραγραφή, Υπέρβαση εξουσίας, Υπεξαίρεση)
13	(Ακυρότητα απόλυτη, ΕΣΔΑ, Νόμου εφαρμογή και ερμηνεία)	(Ακυρότητα απόλυτη, Αναιρέσεως απαράδεκτο, Απαράδεκτο αναιρέσεως, ΕΣΔΑ, Προθεσμία, Υπέρβαση εξουσίας)
4	(Ναρκωτικά,)	(Αναιρέσεως λόγοι,)

Εικόνα 5.7: Οπτικοποίηση του Random Forest - Νομολογία

5.3.5 Bagging

Ο αλγόριθμος εκτελέστηκε θέτοντας **n_jobs = -1**. Κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από το **60%** της μεγαλύτερης πιθανότητας για αυτόν τον αλγόριθμο.

	Actual Tags	Predicted Tags
40	(Ένδικο μέσο, Κλήτευση , Κυριότητα)	(Απαράδεκτη συζήτηση,)
57	(Αναβολής αίτημα, Απάτη, Ισχυρισμός αυτοτελής)	(Ακυρότητα απόλυτη, Απάτη, Ελαφρυντικές περιστάσεις)
18	(Ένδικο μέσο, Αοριστία αγωγής, Καταχρηστική άσκηση δικαιώματος)	(Αοριστία αγωγής, Δεδικασμένο)
15	(Ένδικο μέσο, Νομή)	(Κυριότητα,)
43	(Ανέλεγκτη η ουσιαστική εκτίμηση,)	(Ένδικο μέσο, Αποδοχές μισθωτού, Βυζαντινορωμαϊκό Δίκαιο, Δημόσια κτήματα, Νομή, Χρησικτησία έκτακτη)
39	(Κλήτευση ,)	(Απαράδεκτη συζήτηση,)
45	(Νομή,)	(Έγγραφα, Νομή, Παραγραφή)
32	(Αναίρεση μερική,)	(Ελαφρυντικές περιστάσεις,)

Εικόνα 5.8: Οπτικοποίηση του Bagging - Νομολογία

5.3.6 BERT

Ο αλγόριθμος εκτελέστηκε θέτοντας **N_EPOCHS = 200, BATCH_SIZE = 32**, **MAX_LEN = 256,** και **learning rate = 3e-5**. Για τον αλγόριθμο αυτό, κρατήσαμε όσες ετικέτες είχαν πιθανότητα μεγαλύτερη ή ίση από **20%**.

	Actual Tags	Predicted Tags
0	(Αοριστία αγωγής,)	(Αοριστία αγωγής,)
1	(Δημόσιο , Νομή)	(Νομή,)
2	(Αοριστία αγωγής, Δημόσιο)	(Έλλειψη αιτιολογίας, Αοριστία αγωγής)
3	(Ανθρωποκτονία από αμέλεια,)	(Ανθρωποκτονία από αμέλεια,)
4	(Ναρκωτικά,)	(Αναιρέσεως λόγοι,)
5	(Ένδικο μέσο,)	()
6	(Ένδικο μέσο,)	()
7	(Νόμου εφαρμογή και ερμηνεία,)	(Ακυρότητα απόλυτη, Παραγραφή, Υπέρβαση εξουσίας)
8	(Αναίρεση μερική, Ανθρωποκτονία από αμέλεια)	(Ανθρωποκτονία από αμέλεια,)
9	(Ισχυρισμός αυτοτελής, Πλαστογραφία)	(Νόμου εφαρμογή και ερμηνεία,)

Εικόνα 5.9: Οπτικοποίηση του BERT – Νομολογία

5.4 Παράμετροι Αξιολόγησης

Ομοίως με το πείραμα νομοθεσίας, η εστίαση γίνεται στις μετρικές precision, recall, και F1 score για καθεμία από τις εξής τρεις κατηγορίες: Micro, Macro και Weighted macro.

5.5 Αποτελέσματα

Τα αποτελέσματα με βάση τις μετρικές είναι τα εξής:

	Micro scores			Macro scores			Weighted macro scores		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
DT	0.194	0.156	0.173	0.104	0.083	0.085	0.207	0.156	0.161
Bagging	0.196	0.287	0.233	0.136	0.176	0.146	0.239	0.287	0.251
NB	0.255	0.385	0.307	0.201	0.264	0.198	0.343	0.385	0.327
RF	0.163	0.443	0.238	0.134	0.272	0.151	0.231	0.443	0.261
kNN	0.206	**0.557**	0.301	0.194	**0.342**	**0.221**	0.368	**0.557**	**0.407**
BERT	**0.616**	0.369	**0.462**	**0.253**	0.203	0.217	**0.469**	0.369	0.4

Πίνακας 5.4: Αποτελέσματα μετρικών – Πείραμα Νομολογίας

Αναλύοντας τα αποτελέσματα, σε επίπεδο μετρικών παρατηρούμε τα εξής:

- Μετρική **micro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.616). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.255), K-Nearest Neighbor (0.206), Bagging (0.196), Decision Tree (0.194) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Random Forest (0.163).
- Μετρική **micro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.557). Ακολουθούν κατά φθίνουσα

σειρά οι αλγόριθμοι Random Forest (0.443), Naïve Bayes (0.385), BERT (0.369), Bagging (0.287) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.156)

- Μετρική **micro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.462). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.307), K-Nearest Neighbor (0.301), Random Forest (0.238), Bagging (0.233) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.173).
- Μετρική **macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.253). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Naïve Bayes (0.201), K-Nearest Neighbor (0.194), Bagging (0.136), Random Forest (0.134), ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.104).
- Μετρική **macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.342). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Random Forest (0.272), Naïve Bayes (0.264), BERT (0.203), Bagging (0.176) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.083).
- Μετρική **macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.221). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.217), Naïve Bayes (0.198), Random Forest (0.151), Bagging (0.146) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.085).
- Μετρική **weighted macro precision**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος BERT (0.469). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι K-Nearest Neighbor (0.368), Naïve Bayes (0.343), Bagging (0.239), Random Forest (0.231) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.207).
- Μετρική **weighted macro recall**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.557). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι Random Forest (0.443), Naïve Bayes (0.385), BERT (0.369), Bagging (0.287) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.156).
- Μετρική **weighted macro F1 score**: Σε αυτή την περίπτωση, παρατηρούμε ότι την καλύτερη επίδοση έχει ο αλγόριθμος K-Nearest Neighbor (0.407). Ακολουθούν κατά φθίνουσα σειρά οι αλγόριθμοι BERT (0.4), Naïve Bayes (0.327), Random Forest (0.261), Bagging (0.251) ενώ την χειρότερη επίδοση έχει ο αλγόριθμος Decision Tree (0.161).

Σε επίπεδο αλγορίθμων, έχουμε τα εξής:

- Αλγόριθμος **Decision Tree**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική weighted macro precision (0.207). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές micro precision (0.194), micro F1 score (0.173), weighted macro F1 score (0.161), micro recall και weighted macro recall (0.156), macro precision (0.104), macro F1 score (0.085) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro recall (0.083)
- Αλγόριθμος **Bagging**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.287).

Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.251), weighted macro precision (0.239), micro F1 score (0.233), micro precision (0.196), macro recall (0.176), macro F1 score (0.146) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.136)

- Αλγόριθμος **Naïve Bayes**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.385). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.343), weighted macro F1 score (0.327), micro F1 score (0.307), macro recall (0.264), micro precision (0.255), macro precision (0.201) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro F1 score (0.198)
- Αλγόριθμος **Random Forest**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.443). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές macro recall (0.272), weighted macro F1 score (0.261), micro F1 score (0.238), weighted macro precision (0.231), micro precision (0.163), macro F1 score (0.151) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.134)
- Αλγόριθμος **K-Nearest Neighbor**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro recall και weighted macro recall (0.557). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro F1 score (0.407), weighted macro precision (0.368), macro recall (0.342), micro F1 score (0.301), macro F1 score (0.221), micro precision (0.206) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro precision (0.194).
- Αλγόριθμος **BERT**: Για τον αλγόριθμο αυτό, οι καλύτερες επιδόσεις παρουσιάζονται με βάση την μετρική micro precision (0.616). Ακολουθούν κατά φθίνουσα σειρά οι μετρικές weighted macro precision (0.469), micro F1 score (0.462), weighted macro F1 score (0.4), micro recall και weighted macro recall (0.369), macro precision (0.253), macro F1 score (0.217) ενώ οι χειρότερες επιδόσεις παρουσιάζονται με βάση την μετρική macro recall (0.203)

5.6 Σύνοψη συμπερασμάτων αξιολόγησης

Συνοψίζοντας, σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, weighted macro precision ο καλύτερος αλγόριθμος είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, macro F1 score, weighted macro recall, και weighted macro F1 score ο καλύτερος αλγόριθμος είναι ο K-Nearest Neighbor. Σε επίπεδο αλγορίθμων, οι αλγόριθμοι Bagging, Naïve Bayes, Random Forest και K-Nearest Neighbor παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall, ο αλγόριθμος Decision Tree παρουσιάζει τις καλύτερες επιδόσεις του όταν χρησιμοποιείται η μετρική weighted macro precision, ενώ ο αλγόριθμος BERT παρουσιάζει τις καλύτερες επιδόσεις του όταν χρησιμοποιείται η μετρική micro precision. Οι αλγόριθμοι που προτείνονται λοιπόν τελικά για την κατηγοριοποίηση της Νομολογίας είναι ο BERT και ο K-Nearest Neighbor.

6

Επίλογος

6.1 Σύνοψη και συμπεράσματα

Συνοψίζοντας, στην διπλωματική επιχειρούμε να συγκρίνουμε αλγορίθμους μηχανικής μάθησης στο πρόβλημα της κατηγοριοποίησης πολλαπλών ετικετών σε Ελληνικά νομικά κείμενα. Για να πετύχουμε αυτό τον στόχο, εκτελέσαμε ένα πείραμα σε κείμενα νομοθεσίας με ετικέτες από τον θησαυρό Eurovoc που προέκυψαν από το εργαλείο PyEuroVoc και το σύστημα MITOS και ένα πείραμα σε ήδη κατηγοριοποιημένα κείμενα νομολογίας με ετικέτες από τον Άρειο Πάγο. Οι αλγόριθμοι οι οποίοι χρησιμοποιήθηκαν είναι οι Naïve Bayes, K-Nearest Neighbor (kNN), Decision Tree, Random Forest, Bagging και BERT. Η αξιολόγηση έγινε με βάση τις μετρικές precision, recall, και F1 score για καθεμία από τις εξής τρεις κατηγορίες: Micro, Macro και Weighted macro. Τα συμπεράσματα είναι τα εξής:
Για το πείραμα νομοθεσίας όπου συγκρίνονται τα αποτελέσματα των αλγορίθμων μας σε σχέση με τις ετικέτες από το εργαλείο PyEuroVoc, σε επίπεδο μετρικών, σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, macro F1 score, weighted macro precision, και weighted macro F1 score ο καλύτερος αλγόριθμος είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, και weighted macro recall ο καλύτερος αλγόριθμος είναι ο Naïve Bayes. Σε επίπεδο αλγορίθμων, οι αλγόριθμοι Decision Tree, Bagging, Random Forest, BERT παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιείται η μετρική micro precision, ενώ οι αλγόριθμοι Naïve Bayes και K-Nearest Neighbor παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall. Στην σύγκριση με τις κατηγορίες που είναι ορισμένες από το σύστημα Μίτος, σε επίπεδο μετρικών, ομοίως με πριν σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, macro F1 score, weighted macro precision και weighted macro F1 score ο καλύτερος αλγόριθμος είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, και weighted macro recall ο καλύτερος αλγόριθμος είναι ο Naïve Bayes. Σε επίπεδο αλγορίθμων, και οι 6 αλγόριθμοι παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall. Για το πείραμα νομολογίας όπου οι ετικέτες προέρχονται από τον Άρειο Πάγο, σε επίπεδο μετρικών, σύμφωνα με τις μετρικές micro precision, micro F1 score, macro precision, weighted macro precision ο καλύτερος αλγόριθμος

είναι το BERT, ενώ σύμφωνα με τις μετρικές micro recall, macro recall, macro F1 score, weighted macro recall, και weighted macro F1 score ο καλύτερος αλγόριθμος είναι ο K-Nearest Neighbor. Σε επίπεδο αλγορίθμων, οι αλγόριθμοι Bagging, Naïve Bayes, Random Forest και K-Nearest Neighbor παρουσιάζουν τις καλύτερες επιδόσεις τους όταν χρησιμοποιούνται οι μετρικές micro recall και weighted macro recall, ο αλγόριθμος Decision Tree παρουσιάζει τις καλύτερες επιδόσεις του όταν χρησιμοποιείται η μετρική weighted macro precision, ενώ ο αλγόριθμος BERT παρουσιάζει τις καλύτερες επιδόσεις του όταν χρησιμοποιείται η μετρική micro precision.

6.2 Μελλοντικές επεκτάσεις

Μελλοντικές επεκτάσεις που θα είχε ενδιαφέρον να γίνουν στο κομμάτι της κατηγοριοποίησης Ελληνικών νομικών κειμένων και ενδεχομένως σε ένα παρόμοιο πρόβλημα κατηγοριοποίησης πολλαπλών ετικετών, είναι να εξεταστεί η επίδοση μοντέλων μετασχηματιστών που δεν έχουν όριο λέξεων επεξεργασίας (όπως το όριο των 512 λέξεων του BERT). Δύο τέτοια μοντέλα είναι εκείνα που αναφέραμε στο τέλος του κεφαλαίου 2: Reformer και Longformer. Θα είχε ουσία να εξετάσει κανείς κατά πόσο καλύτερα αποδίδει ένα τέτοιο μοντέλο από τη στιγμή που μπορεί να “δέχεται” περισσότερη πληροφορία από ένα κείμενο και επίσης πόσος χρόνος χρειάζεται για να εκπαιδευτεί καθώς και τι υπολογιστικοί πόροι χρειάζονται. Ανεξάρτητα από το ζήτημα του ορίου λέξεων, χρήσιμο θα ήταν να εξετάσει κανείς και άλλες αρχιτεκτονικές βαθιάς μάθησης εκτός από το μοντέλο BERT. Προτείνεται η εξερεύνηση και άλλων προ-εκπαιδευμένων μοντέλων όπως το GPT-3, το RoBERTa ή το XLNet προκειμένου να συγκριθούν οι επιδόσεις τους με εκείνες του BERT.
Στο πλαίσιο των αλγορίθμων που ήδη εκτελέσαμε στα πειράματα, θα είχε αξία να διεξαχθεί μία πιο εκτενής διαδικασία ρύθμισης υπερπαραμέτρων. Αυτό μπορεί να βοηθήσει στην περαιτέρω βελτίωση των ήδη υπαρχόντων μοντέλων και δυνητικά να έχουμε καλύτερα αποτελέσματα. Η μέθοδος που ακολουθήσαμε για την ρύθμιση υπερπαραμέτρων είναι η Manual Search, στην οποία ο μηχανικός ορίζει ένα σύνολο πιθανών τιμών για κάθε υπερπαράμετρο και ύστερα επιλέγει και προσαρμόζει τις τιμές χειροκίνητα έως ότου οι επιδόσεις του μοντέλου να είναι ικανοποιητικές. Αυτή η μέθοδος χρησιμοποιείται συχνά όταν ο αριθμός των υπερπαραμέτρων είναι σχετικά μικρός και το μοντέλο είναι απλό, αλλά μπορεί να απαιτεί αρκετό χρόνο και δοκιμές για να βρεθεί ο βέλτιστος συνδυασμός υπερπαραμέτρων. Τέσσερις άλλες μέθοδοι που μπορούν να δοκιμαστούν είναι η Grid Search (που περιλαμβάνει εκπαίδευση μοντέλου για κάθε πιθανό συνδυασμό υπερπαραμέτρων σε ένα προκαθορισμένο σύνολο), η Random Search (που περιλαμβάνει την τυχαία επιλογή ενός συνδυασμού υπερπαραμέτρων από ένα προκαθορισμένο σύνολο και την εκπαίδευση μοντέλου με βάση αυτόν), η Bayesian Optimization (που χρησιμοποιεί Μπεϋζιανή βελτιστοποίηση για την εύρεση του βέλτιστου συνδυασμού υπερπαραμέτρων), και η Hyperband (που χρησιμοποιεί μία bandit-based προσέγγιση για να ψάξει με αποδοτικό τρόπο στον χώρο υπερπαραμέτρων).
Όσον αφορά ελαφρώς διαφορετικού τύπου προβλήματα με τα οποία μπορεί κανείς να ασχοληθεί στο μέλλον, οι νομικές υποθέσεις συχνά δεν περιλαμβάνουν μόνο κείμενο αλλά και εικόνες, ακουστικό υλικό ή βίντεο. Η εργασία μπορεί να επεκταθεί με δυνατότητα διαχείρησης πολλών τύπων δεδομένων (Multimodal data) προκειμένου να εξερευνήσει κανείς πως δεδομένα κειμένου και μη μπορούν να συνδυασθούν για καλύτερη ταξινόμηση. Η άλλη

δυνατότητα επέκτασης είναι η υλοποίηση ενός συστήματος που να μπορεί να κατηγοριοποιεί νομικά κείμενα σε πραγματικό χρόνο μαζί με ένα φιλικό προς τον χρήστη περιβάλλον διεπαφής.

7

Βιβλιογραφία

[1] Georgieva-Trifonova, T., & Duraku, M. (2021). Research on N-grams feature selection methods for text classification. *IOP Conference Series: Materials Science and Engineering*, *1031*(1), 012048. https://doi.org/10.1088/1757-899X/1031/1/012048

[2] V M, N., & Kumar R, Dr. A. (2019). Implementation on Text Classification Using Bag of Words Model. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3507923

[3] Ammar Kadhim. (2018). An Evaluation of Preprocessing Techniques for Text Classification. *International Journal of Computer Science and Information Security*, *16*, 22–32.

[4] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. https://doi.org/10.48550/ARXIV.1301.3781

[5] Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical*

Methods in Natural Language Processing (EMNLP), 1532–1543. https://doi.org/10.3115/v1/D14-1162

[6] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146. https://doi.org/10.1162/tacl_a_00051

[7] Melamud, O., Goldberger, J., & Dagan, I. (2016). context2vec: Learning Generic Context Embedding with Bidirectional LSTM. *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 51–61. https://doi.org/10.18653/v1/K16-1006

[8] Luaces, O., Díez, J., Barranquero, J., Del Coz, J. J., & Bahamonde, A. (2012). Binary relevance efficacy for multilabel classification. *Progress in Artificial Intelligence*, *1*(4), 303–313. https://doi.org/10.1007/s13748-012-0030-x

[9] Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2009). Classifier Chains for Multi-label Classification. In W. Buntine, M. Grobelnik, D. Mladenić, & J. Shawe-Taylor (Eds.), *Machine Learning and Knowledge Discovery in Databases* (Vol. 5782, pp. 254–269). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-04174-7_17

[10] Tsoumakas, G., Katakis, I., & Vlahavas, I. (2011). Random k-Labelsets for Multilabel Classification. *IEEE Transactions on Knowledge and Data Engineering*, *23*(7), 1079–1089. https://doi.org/10.1109/TKDE.2010.164

[11] Zeng, A., & Huang, Y. (2011). A Text Classification Algorithm Based on Rocchio and Hierarchical Clustering. In D.-S. Huang, Y. Gan, V. Bevilacqua,

& J. C. Figueroa (Eds.), *Advanced Intelligent Computing* (Vol. 6838, pp. 432–439). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-24728-6_59

[12] Yoav Freund & Robert E. Schapire. (1999). A Short Introduction to Boosting. *Journal of Japanese Society for Artificial Intelligence*, *14*(5), 771–780.

[13] Li, X., Shi, T., Li, P., & Zhou, W. (2019). Application of Bagging Ensemble Classifier based on Genetic Algorithm in the Text Classification of Railway Fault Hazards. *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, 286–290. https://doi.org/10.1109/ICAIBD.2019.8836988

[14] Pranckevicius, T., & Marcinkevicius, V. (2016). Application of Logistic Regression with part-of-the-speech tagging for multi-class text classification. *2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*, 1–5. https://doi.org/10.1109/AIEEE.2016.7821805

[15] Raschka, S. (2014). Naive Bayes and Text Classification I - Introduction and Theory. https://doi.org/10.48550/ARXIV.1410.5329

[16] Trstenjak, B., Mikac, S., & Donko, D. (2014). KNN with TF-IDF based Framework for Text Categorization. *Procedia Engineering*, *69*, 1356–1364. https://doi.org/10.1016/j.proeng.2014.03.129

[17] Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In C. Nédellec & C. Rouveirol (Eds.),

Machine Learning: ECML-98 (Vol. 1398, pp. 137–142). Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0026683

[18] Apté, C., & Weiss, S. (1997). Data mining with decision trees and decision rules. *Future Generation Computer Systems*, *13*(2–3), 197–210. https://doi.org/10.1016/S0167-739X(97)00021-6

[19] Luo, X. (2018). An improved text classifier based on random forest algorithm—Comparative studies on multiple text classifiers. *Proceedings of the 2017 4th International Conference on Machinery, Materials and Computer (MACMC 2017)*. 2017 4th International Conference on Machinery, Materials and Computer (MACMC 2017), Xi'an, China. https://doi.org/10.2991/macmc-17.2018.39

[20] Lydia, A. A., & Francis, F. S. (2020). Multi-Label Classification using Deep Convolutional Neural Network. *2020 International Conference on Innovative Trends in Information Technology (ICITIIT)*, 1–6. https://doi.org/10.1109/ICITIIT49094.2020.9071539

[21] Chen, S.-F., Chen, Y.-C., Yeh, C.-K., & Wang, Y.-C. F. (2017). *Order-Free RNN with Visual Attention for Multi-Label Classification*. https://doi.org/10.48550/ARXIV.1707.05495

[22] Feng, S., Li, H., & Qiao, J. (2022). Hierarchical multi-label classification based on LSTM network and Bayesian decision theory for LncRNA function prediction. *Scientific Reports*, *12*(1), 5819. https://doi.org/10.1038/s41598-022-09672-1

[23] Fitriawan, A., Wasito, I., Syafiandini, A. F., Amien, M., & Yanuar, A. (2016). Multi-label classification using deep belief networks for virtual screening of multi-target drug. *2016 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 102–107. https://doi.org/10.1109/IC3INA.2016.7863032

[24] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*. https://doi.org/10.48550/ARXIV.1706.03762

[25] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[26] Breiman, L. (1999). Pasting Small Votes for Classification in Large Databases and On-Line. *Machine Learning*, *36*(1/2), 85–103. https://doi.org/10.1023/A:1007563306331

[27] Tin Kam Ho. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(8), 832–844. https://doi.org/10.1109/34.709601

[28] Louppe, G., & Geurts, P. (2012). Ensembles on Random Patches. In P. A. Flach, T. De Bie, & N. Cristianini (Eds.), *Machine Learning and Knowledge Discovery in Databases* (Vol. 7523, pp. 346–361). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33460-3_28

[29] Dai, A. M., & Le, Q. V. (2015). Semi-supervised Sequence Learning. https://doi.org/10.48550/ARXIV.1511.01432

[30] A. Radford, K. Narasimhan, T. Salimans, & I. Sutskever. (2018). Improving language understanding by generative pre-training.

[31] Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2227–2237. https://doi.org/10.18653/v1/N18-1202

[32] Howard, J., & Ruder, S. (2018). Universal Language Model Fine-tuning for Text Classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 328–339. https://doi.org/10.18653/v1/P18-1031

[33] Kitaev, N., Kaiser, Ł., & Levskaya, A. (2020). *Reformer: The Efficient Transformer*. https://doi.org/10.48550/ARXIV.2001.04451

[34] Beltagy, I., Peters, M. E., & Cohan, A. (2020). *Longformer: The Long-Document Transformer*. https://doi.org/10.48550/ARXIV.2004.05150

[35] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. https://doi.org/10.48550/ARXIV.1907.11692

[36] Louppe, G., & Geurts, P. (2012). Ensembles on Random Patches. In P. A. Flach, T. De Bie, & N. Cristianini (Eds.), *Machine Learning and Knowledge*

Discovery in Databases (Vol. 7523, pp. 346–361). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33460-3_28

[37] Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*. https://doi.org/10.48550/ARXIV.2003.10555

[38] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). *XLNet: Generalized Autoregressive Pretraining for Language Understanding*. https://doi.org/10.48550/ARXIV.1906.08237

[39] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter*. https://doi.org/10.48550/ARXIV.1910.01108

[40] Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., & Levy, O. (2020). SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Transactions of the Association for Computational Linguistics*, *8*, 64–77. https://doi.org/10.1162/tacl_a_00300

[41] Liu, Y. (2019). *Fine-tune BERT for Extractive Summarization*. https://doi.org/10.48550/ARXIV.1903.10318

[42] Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2019). *TinyBERT: Distilling BERT for Natural Language Understanding*. https://doi.org/10.48550/ARXIV.1909.10351

[43] Kowsari, Jafari Meimandi, Heidarysafa, Mendu, Barnes, & Brown. (2019). Text Classification Algorithms: A Survey. *Information*, *10*(4), 150. https://doi.org/10.3390/info10040150

[44] H. Carlsson & T. Lindgren. (2021). Classification of Legal Documents—A Topic Modeling Approach.

[45] Bc. Jiří Mauritz. (2018). Automatic Classification of Legal Documents.

[46] Christos N. Papaloukas & Ilias Chalkidis. (2020). Legal Text Classification based on Greek Legislation.

[47] Panagiota G. Kampili. (2022). Large-Scale Multi-label Classification of Greek legislation.

[48] Steinberger, R., Ebrahim, M., & Turchi, M. (2013). *JRC EuroVoc Indexer JEX - A freely available multi-label categorisation tool*. https://doi.org/10.48550/ARXIV.1309.5223

[49] Research Institute for Artificial Intelligence, Romanian Academy Bucharest, Romania, Avram, A.-M., Păiş, V., Research Institute for Artificial Intelligence, Romanian Academy Bucharest, Romania, Tufiş, D., & Research Institute for Artificial Intelligence, Romanian Academy Bucharest, Romania. (2021). PyEuroVoc: A Tool for Multilingual Legal Document Classification with EuroVoc Descriptors. *Proceedings of the Conference Recent Advances in Natural Language Processing - Deep Learning for Natural Language Processing Methods and Applications*, 92–101. https://doi.org/10.26615/978-954-452-072-4_012

[50] Research Schholar, Dept of CSE, Acharya Nagarjuna University, NagarjunaNagar, India., Venkataanusha*, P., Anuradha, Ch., Assistant professor, Dept of CSE, VRSEC, Vijayawada, Chandra Murty, Dr. P. S. R.,

Research Supervisor,Dept of CSE, Acharya Nagarajuna University., Chebrolu, Dr. S. K., & Associate Professor, Dept of CSE, NRI Institute of technology. (2019). Detecting Outliers in High Dimensional Data Sets Using Z-Score Methodology. *International Journal of Innovative Technology and Exploring Engineering*, *9*(1), 48–53. https://doi.org/10.35940/ijitee.A3910.119119

[51] Koutsikakis, J., Chalkidis, I., Malakasiotis, P., & Androutsopoulos, I. (2020). *GREEK-BERT: The Greeks visiting Sesame Street*. https://doi.org/10.48550/ARXIV.2008.12014

[52] Song, D., Vold, A., Madan, K., & Schilder, F. (2022). Multi-label legal document classification: A deep learning-based approach with label-attention and domain-specific pre-training. *Information Systems*, *106*, 101718. https://doi.org/10.1016/j.is.2021.101718

[53] Koniaris, M., Galanis, D., Giannini, E., & Tsanakas, P. (2023). Evaluation of Automatic Legal Text Summarization Techniques for Greek Case Law. *Information*, *14*(4), 250. https://doi.org/10.3390/info14040250