NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

BSc THESIS

Large-Scale Multi-label Classification of Greek legislation

Panagiota G. Kampili

Supervisors: **Manolis Koubarakis,** Professor
Despina - Athanasia Pantazi, PhD Candidate

ATHENS

February 2022

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Κατηγοριοποίηση πολλαπλών ετικετών μεγάλης κλιμάκας σε κείμενα ελληνικής νομοθεσίας

Παναγιώτα Γ. Καμπύλη

Επιβλέποντες: **Μανόλης Κουμπαράκης,** Καθηγητής
Δέσποινα – Αθανασία Πανταζή, Υποψήφια Διδάκτωρ

ΑΘΗΝΑ

Φεβρουάριος 2022

BSc THESIS

Large-Scale Multi-label Classification of Greek legislation

Panagiota G. Kampili
S.N.: 1115201500060

SUPERVISORS: **Manolis Koubarakis,** Professor
Despina - Athanasia Pantazi, PhD Candidate

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Κατηγοριοποίηση πολλαπλών ετικετών μεγάλης κλιμάκας σε κείμενα ελληνικής νομοθεσίας

Παναγιώτα Γ. Καμπύλη
Α.Μ.: 1115201500060

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Μανόλης Κουμπαράκης,** Καθηγητής
Δέσποινα – Αθανασία Πανταζή, Υποψήφια Διδάκτωρ

ABSTRACT

Natural Language Processing is an area in Artificial Intelligence that is constantly attracting scientific interest and facilitates everyday tasks. We focus on a specific case of multi-label classification problem, which over time and with the constantly increasing volume of data, becomes more and more frequent. Large-scale Multi-label Text Classification is characterized by large label space typically organized in a hierarchical manner and unbalanced label distributions. Our area of interest is the legal domain and we chose to experiment with the Greek language and more specifically, ”RAPTARCHIS47k“, a dataset consisting of more than forty seven thousand Greek legal documents. Objective of this thesis constitutes the hands-on evaluation of multi-label approaches on Greek legal documents, the comparison of LMTC dedicated techniques to general state-of-the-art methods and the experimentation of learning to predict labels that rarely occur in the training set. We focus on some of the most well-known and promising hierarchical Probabilistic Label Tree methods, hybrid PLT-neural network methods, and we further experiment with transfer learning utilizing the latest transformer-based approaches. We evaluate these methods on three different levels of frequency (all-labels, frequent, few-case), and we investigate a multitude of configurations for every method separately. Our experiments showed that there is no rule of thumb about what method should be used, as different approaches gave the best performance in all three sub-tasks. Cutting edge technology Transformer-based models gave the best performance in sub-tasks, where the common labels dominate the hierarchy, while PLTs proved their supremacy on the task involving tail labels. As far as we know the scientific area of Large-scale Multi-label Text Classification is vastly understudied, especially for the Greek language, and we hope that this study will be a reference point for future research.

SUBJECT AREA: Natural Language Processing, Artificial Intelligence

KEYWORDS: Legal Documents, Multi-label Classification, Probabilistic Label Trees, Neural Networks

ΠΕΡΙΛΗΨΗ

Η επεξεργασία φυσικής γλώσσας είναι ένας τομεάς της Τεχνητής Νοημοσύνης που διαρκώς προσεγγίζει επιστημονικό ενδιαφέρον και διευκολύνει ανάγκες της καθημερινότητας. Θα επικεντρωθούμε σε μια συγκεκριμένη περίπτωση κατηγοριοποίησης πολλαπλών ετικετών, η οποία με την πάροδο του χρόνου και το διαρκώς αυξανόμενο όγκο δεδομένων, γίνεται όλο και πιο συχνή. Η Κατηγοριοποίηση Πολλαπλής Ετικέτας Μεγάλης Κλίμακας χαρακτηρίζεται απο μεγάλο χώρο ετικετών, οργανωμένες με ιεραρχικό τρόπο και ανισσοροπία στην κατανομή των ετικετών. Ο τομέας ενδιαφέροντός μας είναι η νομική επιστήμη και επιλέξαμε να ασχοληθούμε με την ελληνική γλώσσα, και πιο συγκεκριμένα με το σύνολο δεδομένων "RAPTARCHIS47K“, το οποίο αποτελείται απο πάνω απο 47 χιλιάδες νομικές πηγές. Στόχος αυτής της πτυχιακής είναι η πρακτική αξιολόγηση μεθόδων κατηγοριοποίησησς πάνω σε ελληνικά νομικά κείμενα, η σύγκριση μεθόδων ειδικά διαμορφωμένων για προβλήματα κατηγοιοποίησης πολλαπλών ετικετών μεγάλης κλίμακας με σύγχρονες τεχνολογίες αιχμής, καθώς και ο πειραματισμός στην εκμάθηση πρόβλεψης ετικετέων που εμφανίζονται σπάνια στο σύνολο εκμάθησης. Θα επικεντρωθούμε σε κάποιες απο τις πιο διαδεδομένες και υποσχόμενες μεθόδους πιθανοτικών δέντρων ετικέτας, υβριδικών μεθόδων πιθανοτικών δέντρων, και νευρωνικών δικτύων κάθως επίσης και σε τεχνικές διαδιδόμενης μάθησης που αξιοπούν τις σύγχρονες μεθόδους βασισμένες σε μετασχηματιστές (Transformers). Αξιολογούμε αυτές τις μεθόδους πάνω σε τρία διαφορετικά επίπεδα συχνότητας εμφάνισης ετικετών (όλες οι ετικές, οι πιο συχνές, οι πιο σπάνιες), και ερευνούμε μια πληθώρα παραμαετροποιήσεων για κάθε μέθοδο ξεχωριστά. Τα πειράματα μας έδειξαν ότι δεν υπάρχει κανόνας για το ποια μέθοδος πρέπει να προτιμάται πάντα καθώς διαφορετικές επιλογές έδωσαν τα καλύτερα αποτελέσματα στα διαφορετικά επιπεδα εξέτασης. Τα μοντέλα βασισμένα σε τελευταίας τεχνολογίας μετασχηματιστές έδωσαν καλύτερα αποτελέσματα στα προβλήματα όπου οι συχνές ετικέτες κυριαρχούσαν, ενώ οι μέθοδοι βασισμένες σε πιθανοτικά δέντρα έδειξαν την υπεροχή τους σε προβλήματα που υπήρχαν κυρίως σπάνιες ετικέτες. Από όσο γνωρίζουμε, η επιστημονική περιοχή της κατηγοριοποίησης πολλαπλών ετικετών μεγάλης κλίμακας είναι υπομελετημένη ειδικά στην περίπτωση της ελληνικής γλώσσας, και ελπίζουμε ότι αυτή η μελέτη θα αποτελέσει σημείο αναφοράς για μελλοντικές έρευνες.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Επεξεργασία Φυσικής Γλώσσας, Τεχνητή Νοημοσύνη

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Νομικά Κείμενα, Κατηγοριοποίηση πολλαπλών ετικετών,
Πιθανοτικά δέντρα ετικετών, Νευρωνικά Δίκτυα

Στην αγαπημένη μου γιαγιά Παναγιώτα

ACKNOWLEDGEMENTS

I would like to thank my supervisors: Manolis Koubarakis and Despina Athanasia Pantazi for their constant support and guidance throughout this process. They gave me the opportunity to work with such an interesting task and were always more than willing to help me with whatever obstacle I encountered. I also would like to thank my friends and family that support me and encourage me all these years.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

This research thesis was conducted in the context of acquiring the Bachelor’s degree in the Department of Informatics and Telecommunications of the National and Kapodistrian Unive-rsity of Athens.

1. INTRODUCTION

As technology evolves, the need for gathering, storing, and analyzing data grows too. In this data driven environment the area of Natural Language Processing (NLP) gives us tools to better understand and utilize the human language. This need is more visible in areas like legal science, where the data collected from this field is characterized by complexity, ambiguity, and nomenclature, properties that increase the difficulty for both human and machine processes. For these reasons the legal document processing is considered a flourishing area of Artificial Intelligence (AI) and is constantly attracting scientific interest. The legal domain combined with the Greek language, which in itself has its peculiarities and conceptual difficulties, is the field of research for this thesis.

We chose to experiment with the multi-label classification task, which actually aims to predict multiple mutually non-exclusive labels from a label set. More specifically, we focused on a variation of this task called: Large-Scale Multi-Label Text Classification. Our objective constitutes the hands-on evaluation of various methods ranging from Probabilistic Label Trees to state-of-the-art Transformer-based [24] methods on Greek legal documents. The methods we chose to present and experiment with are: the Parabel[23] algorithm that grows three deep narrow trees in order to predict the correct labels from a label set and its evolution, the Bonsai[13] algorithm, that tries to eliminate Parabel's propagation errors and constraints in hope for better results in few and zero shot cases. Following we worked with AttentionXML[26] that offers a combination between Probabilistic Label Tree methods and deep neural networks, consisting a competitive alternative especially in cases of extreme-scale label sets. Last but not least, we experimented with two variations of well-known and powerful BERT model adapted to the Greek language. The first one is GREEK-BERT[14], a monolingual model with great performance on tasks engaging Greek documents, and the second one is GreekLegalBERT[2], a domain specific model targeting tasks with legal content. We focused on the assessment of these methods in order to find the one that gives the best results for our problem. We are particularly interested in the comparison between these two different generations of approaches: the algorithmic ones that utilize simple linear classifiers along with the label space, and the most recent methods that use the computational heavy but extremely efficient Transformers. This thesis is structured as follows:

- Chapter 2: We provide background information about the LMTC and XMTC tasks and mention methods that address this classification problem with special reference to advanced transfer learning techniques. Moreover, we present information about the RAPTARCHIS47k dataset along with related research.
- Chapter 3: In this chapter we provide detailed information about the assignment of this thesis. We present all the assumptions we had to make, and we clearly define the context in which the experiments were conducted. Finally, we shed some light on the label hierarchy our dataset engages.
- Chapter 4: We deal with each method separately, by providing general information about their inner workings without associating them to our assignment. We provide a brief overview of the procedures they follow along with comments about their origin and performance.
- Chapter 5: This chapter contains technical information about the environment the experiments were conducted. It also contains information about the different pre-

processing steps each method required based on the different label representation, and a brief presentation of the hyperparameter exploration and fine-tuning.

- Chapter 6: In this chapter we provide the output of our experiments. The results are categorised based on the frequency level of the labels. At the end of this chapter, there is also a final summary table that better depicts the comparison between the methods.
- Chapter 7: In this last chapter, there is a short summary describing the process that led us to our results. We state our conclusions and suggest methods that can further help deal with such tasks in Greek NLP.

2. BACKGROUND AND RELATED WORK

In this chapter, we define the LMTC and XMTC problems, provide a preview of well-known approaches, and introduce the dataset we will work with, RAPTARCHIS47k, along with the related research.

2.1 Large and Extreme Multi-label Text Classification

Large-Scale Multi-Label Text Classification (LMTC) is a specific variation of the multi-label classification task containing a label set that is considerably large (typically thousands) and is characterized by skewed label distributions and usually hierarchical connections between the labels. The label space can also be visualised as a graph and in some cases as a tree. In cases where the classification task involves hundreds of thousands or even millions of labels, it is transformed into Extreme-Scale Multi-Label Text Classification (XMTC). Both of these classification tasks attract the researchers' interest and as a result many efficient methods have been developed and continue to develop.

2.1.1 LMTC & XMTC Related Work

A common approach, the 1-vs-all approach [19], uses powerful classifiers that are dedicated to one label and train weighted vectors that can identify this specific label. A well-known method is DISMEC (Distributed Sparse Machines for Extreme Multi- label Classification) [3], which is a state-of-the-art model utilizing this type of classifiers coupled with capacity control in order to solve classification problems. Even though it is considered a reliable approach, its training complexity and low prediction speed led to a demand for further methods such as PPDsparse (Parallel Primal-Dual Sparse) [25] which is a variation of the original PDsparce (Primal-Dual Sparse) [10] algorithm that utilizes parallelization by introducing separable loss functions. In this approach, the complexity is sub-linear to the number of classes, making it more suitable for real time applications. The evolution of the algorithms mentioned above constitutes the Probabilistic Label Tree methods, some of which we will further discuss on this thesis, that managed to combine the powerful, but computationally inefficient, one versus all classifiers with methods that utilize the label space and form tree structures resulting into logarithmic complexity to the number of labels.

An extensive search on this topic is the one that Chalkidis et al. present in their paper ”An Empirical Study on Large-Scale Multi-Label Text Classification Including Few and Zero-Shot Labels”[7] that also constitutes a strong influence for this thesis. In this research they evaluate a plethora of methods ranging from vanilla RNNs to Probabilistic Label Trees to Transformer-based methods on three well known LMTC benchmarks from different domains and propose new approaches that can exploit the label hierarchy in order to improve few and zero shot learning. They conclude that hierarchical PLT-based methods are definitely worth considering, and at the same time AttentionXML proved to be adequate across all datasets, while the transfer learning techniques gave the best performance in general. They also emphasize on the fact that the success of such approaches strongly depends on the document's length, something that is restrictive especially in the legal domain where the texts tend to be long. Finally, they suggest that the best way to use the label hierarchy in neural methods depends on the proximity of the label assignments in each dataset and emphasizes on the fact that there is no general-principle about the method that should be used, as it is strongly affected by the applied domain, the document length, the language and the percentage of frequent, few and zero shot classes.

2.1.2 BERT in LMTC tasks

BERT model may not have been developed especially for LMTC or XMTC tasks, but is a cutting-edge-technology model that has proven to be more than sufficient for many modern difficult tasks in the Natural Language Processing area. This model was firstly introduced by Jacob Devin et.al.[9] and as the authors mention is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. Its' structure is consisted of stacked powerful Transformers[24], with each node's output passing to the next one. This approach faces a severe limitation as it can process up to 512 tokens (chunks of text).

BERT was trained on two fundamental NLP tasks (Masked Language Modeling and Next Sentence Prediction) and fed with BooksCorpus (Zhu et. al. [27]) of total size 800 million words and English Wikipedia corpus of total size 2500 million words. The computational complexity and the resources needed for training such a deep neural network with this amount of data makes it inefficient in most cases to train the model from the beginning. If we add layers on top of the pretrained BERT, we can efficiently deal with almost every downstream task just by fine-tuning the extra layers.

Later on more extensions of BERT were developed and trained, like Multilingual BERT that supports more than one hundred languages including Greek and is suitable for tasks like question answering and sequence classification. However, in our case, where we are specifically interested in the Greek language GREEK-BERT[14] may seem to be a more appropriate choice to solve the above tasks.

A really interesting part in the LMTC task, apart from the large label space, is the hierarchical connections the labels may form. Many researches support that, exploiting the label hierarchy leads to better performance. On this path Manginas et al.[16] propose four different ways to fine-tune BERT in a structured manner by guiding specific layers to predict specific hierarchy levels. The first method suggests that the last layers should predict the hierarchy levels with the top layer of BERT predicting the most specific level. In the second approach, they utilize the full depth of the model by guiding one layer to predict a specific level and skip the next one. Another way to utilize the full depth is by grouping the layers in pairs and using the output token for the prediction. The last proposed method is a hybrid method that combines the first approach with the grouping in pairs approach. The figure 2.1 depicts the four suggested methods along with the flat one for comparison.

The first suggested method yields the best results. Intuitively, this approach embraces the idea that the first layers learn the syntax and the more general context, and the last layers are more task-oriented. Even though it seems like a promising approach, we will not experiment with this guiding technique, as it doesn't serve the purpose of this thesis.

2.2 Greek Legal Document Classification

Legal document classification is a flourishing field in NLP, as the data from this domain has usually complicated context with special terminology along with large documents that

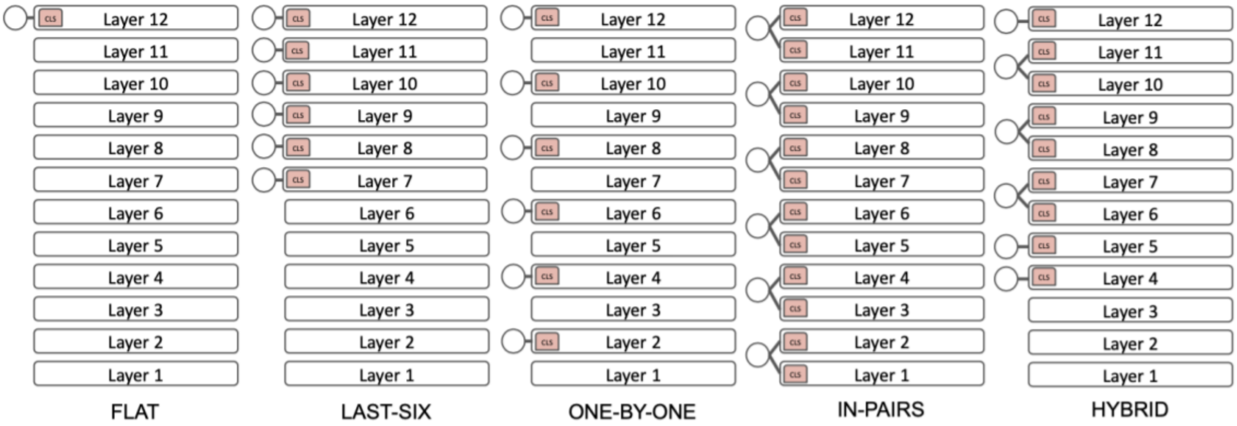


Figure 2.1: Variations of structured BERT

continuously mount up over the years. As far as Greek legislation is concerned, Greek Legislation Code (GLC) is publicly offered through the e-Themis portal, which is primarily focused on providing the most recent legislation and is addressed to not only legal entities, whose professional activity requires access to this type of content, but also every citizen that wants to be informed about the Greek legislation.

2.2.1 Introduction to RAPTARCHIS47k

RAPTARCHIS47k is a dataset consisting of more than forty seven thousand official categorized Greek legislation resources and was firstly introduced in the Papaloukas paper [18]. It contains Greek legislation codes from 1834 to 2015 with the most of them being published in the period 1960-2000. It includes laws, royal and presidential decrees, regulations and decisions. The only source of information is the Official Government Gazette[1].

The dataset was split into three subsets (training 60%, development 20%, and test 20%) after performing distribution in all the levels of hierarchy, maintaining the same level of partitioning from bottom to top. The label space is consisted of 47 legislative volumes that each of them is further divided into 389 chapters and subsequently, each chapter breaks down to 2285 subjects, which contain the legal resources forming a hierarchical organised label set.

In a more detailed and quantitative analysis of the dataset the authors split the label set into three different sets based on the appearance frequency of a label in the dataset as listed below.

- **Frequent**: In this category belong the labels that occur more than 10 times in the training set, and can be found in all three subsets (training, development, test). All of the volumes are characterised as “Frequent”, while 85.6% and 31.2% of chapter and subject respectively belong to this category.

- **Few Shot**: In this category belong the labels that appear at least one but no more than 10 times in the training set. The subject level has the most instances in this category with 62.6%, leaving behind the chapter level with 13.6% and the volume level with no instance at all.

[1] https://www.hellenicparliament.gr/en/Vouli-ton-Ellinon/I-Bibliothiki/Koinovouleftiki-Syllogiold/Efimeris-Tis-Kyverniseos-FEK/

- **Zero Shot**: Are the labels that first appear in the test and/or development set. Only a small subset of the label set is being tagged with this class with 0.7% and 6.2% belonging in the chapter and subject level respectively.

As we can observe, the dataset has a lot labels that are underrepresented, making it suitable for few and zero shot learning. Another interesting factor about the dataset is the legal resources distribution over classes in all the thematic levels. Moving from the broader thematic level of volumes with a mean of 1011 legal resources to the middle level chapter with 122 resources to the most specific level of subjects with only 20 legal resources on average, it is clearly noticeable that there is an imbalance in representation between the different hierarchy levels.

2.2.2 Related work on Greek Legislation

Groundwork on Greek legal text classification constitutes Christos N. Papaloukas' thesis [18] that experiments with a battery of NLP methods on the RAPTARCHIS47k dataset. He addresses this multi-label classification task like three different multi-class classification problems by predicting each level of the hierarchy separately, ignoring the connections between the different levels. The methods he experiments with range from simple Support Vector Machines (SVMs) to advanced transfer learning techniques utilizing different variations of BERT. The results showed that even though classic SVM satisfy the necessary preliminaries for the frequent classes, they are inadequate on more complicated cases. In contrast, deep neural network techniques that utilize and fine-tune bidirectional GRUs provide improved performance in all cases regardless of their label frequency. The best performance from the examined methods was given by Transformer-based approaches. GREEK-BERT outperformed every other classifier on volume level and Multilingual-BERT gave the best performance for chapter and subject level. Papaloukas also emphasizes on the small difference on performance between the two best models and raises the question of whether it is worth using so many resources to train and fine-tune a monolingual model when already established multilingual approaches have such good results.

A more focused preview on transfer learning approaches is being offered by Efstratios G. Vamvourellis' thesis[22] that experiments with different variations of pretrained BERT models including the domain specific GreekLegalBERT and tries to answer the question of which version is suitable for the Greek legal document classification task. His approach is similar to Papaloukas' as he also addresses the task like three different multi-class classification problems but he focuses only on BERT-based methods and discusses further improvements on already fine-tuned models on legal documents. The research showed that GreekLegalBERT outperformed every other configuration while Multilingual-BERT proved to be insufficient, although he suggested that Domain and Task adaptive pre-training could help multilingual BERT to surpass language specific models.

3. PROBLEM DEFINITION AND LABEL HIERARCHY

The area of interest for this thesis constitutes the specific variation of multi-label classification, LMTC, that was briefly described in chapter 2. The dataset we will work with is the RAPTARCHIS47k dataset that consists of more than forty-seven thousand Greek legislation documents. This dataset belongs to the LMTC task category, as its label set is consisted of more than two thousands labels organised in a hierarchical manner with highly visible imbalance in the frequency appearance between different classes. As far as we know, the only published researches on this dataset approach this task like three different flat multi-class classification problems. Our objective is to address this task like one multi-label classification problem.

As input we will use the text content of each document, consisting of the header along with one or more articles, and the targeting set will be the thematic chain consisting of the triplet: volume, chapter and subject, as shown in figure 3.1 originating from Papaloukas' et al.paper [17]. The evaluation and review of the examined methods will be performed over three different types of classes: “all labels”, “frequent” and “few cases”. The criteria based on which the labels are divided into the last two classes is the same as the ones Papaloukas followed in his research and are also described in section 2.2.1. In the case of “all labels” we will evaluate the chosen methods across the whole thematic chain, without excluding any subset based on the frequency of their appearance. This dataset also contains a small number of zero-shot cases, but do not constitute a field of study for this thesis, as the methods we chose to experiment with are not zero-shot capable and any result may be considered unsafe.

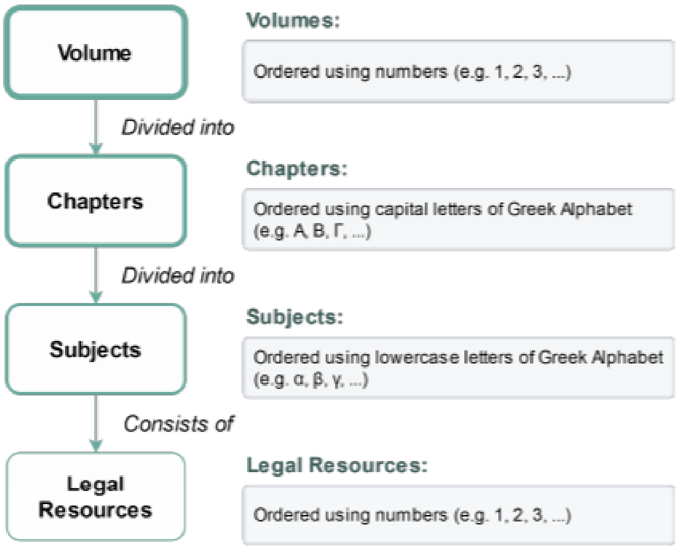
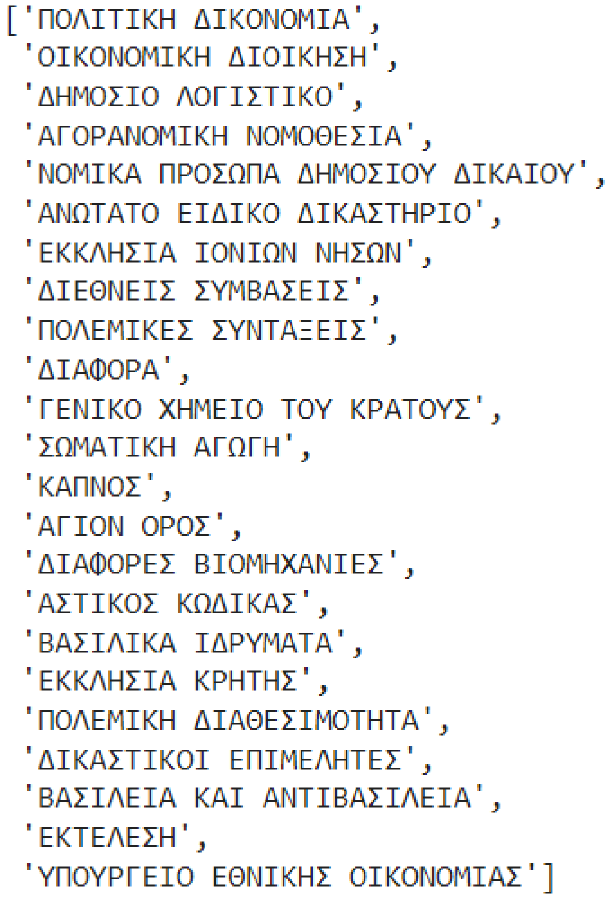

Volume
Divided into
Chapters
Divided into
Subjects
Consists of
Legal Resources
Volumes:
Ordered using numbers (e.g. 1, 2, 3, ...)
Chapters:
Ordered using capital letters of Greek Alphabet (e.g. A, B, Γ, ...)
Subjects:
Ordered using lowercase letters of Greek Alphabet (e.g. α, β, γ, ...)
Legal Resources:
Ordered using numbers (e.g. 1, 2, 3, ...)

Figure 3.1: Hierarchical thematic chain

Before proceeding, it is worth emphasising on the dataset's label hierarchy. We believe that the hierarchical connections the labels form between them is a really important piece of information that if used properly can help into accomplishing better performance especially in tail labels (labels that are underrepresented in the training set). The methods we chose to experiment with do not exploit the label hierarchy directly, but utilise it in a more indirect way. First by approaching the problem as multi-label classification task we do not ignore the relationship between the labels that belong in the same category, we essentially include this information in the training process and evaluate the models on their ability to predict the whole thematic chain. This applies to all the methods, but the Probabilistic Label Tree methods also utilize the label space by forming their own label hierarchy during constructing their trees. This phenomenon will be further discussed on chapter 4

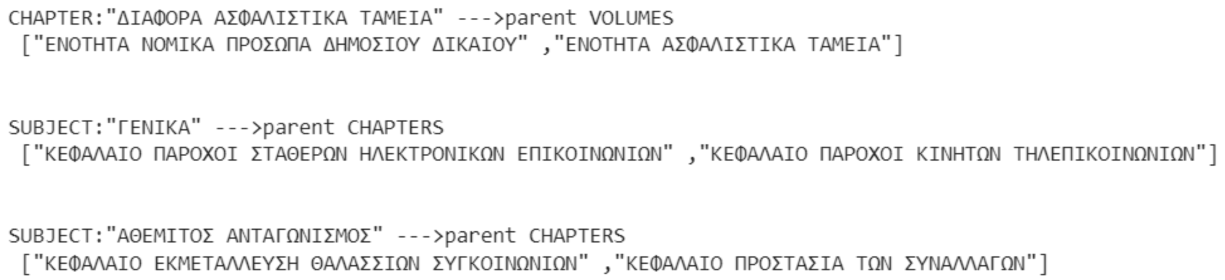
After some analysis on the dataset, we found out that the hierarchically organised labels cannot be depicted as a label tree, as we initially assumed, for two main reasons. Firstly, labels that belong to more than one level (volume, chapter or subject) exist. At the beginning we thought that a label can either be characterised as volume, chapter subject, but we found out that 23 out of the 2721 labels have this abnormality (figure 3.2). Secondly, labels with multiple parents exist. From the schema presented in figure 3.1 we would expect that one specific chapter would have a single volume as parent and respectively one subject would have only one chapter. Our analysis showed that 34 labels in total do not comply with this rule since 3 chapters and 31 subjects have multiple parents. In figure 3.3 we see some examples of such labels.



```
['ΠΟΛΙΤΙΚΗ ΔΙΚΟΝΟΜΙΑ',
 'ΟΙΚΟΝΟΜΙΚΗ ΔΙΟΙΚΗΣΗ',
 'ΔΗΜΟΣΙΟ ΛΟΓΙΣΤΙΚΟ',
 'ΑΓΟΡΑΝΟΜΙΚΗ ΝΟΜΟΘΕΣΙΑ',
 'ΝΟΜΙΚΑ ΠΡΟΣΩΠΑ ΔΗΜΟΣΙΟΥ ΔΙΚΑΙΟΥ',
 'ΑΝΩΤΑΤΟ ΕΙΔΙΚΟ ΔΙΚΑΣΤΗΡΙΟ',
 'ΕΚΚΛΗΣΙΑ ΙΟΝΙΩΝ ΝΗΣΩΝ',
 'ΔΙΕΘΝΕΙΣ ΣΥΜΒΑΣΕΙΣ',
 'ΠΟΛΕΜΙΚΕΣ ΣΥΝΤΑΞΕΙΣ',
 'ΔΙΑΦΟΡΑ',
 'ΓΕΝΙΚΟ ΧΗΜΕΙΟ ΤΟΥ ΚΡΑΤΟΥΣ',
 'ΣΩΜΑΤΙΚΗ ΑΓΩΓΗ',
 'ΚΑΠΝΟΣ',
 'ΑΓΙΟΝ ΟΡΟΣ',
 'ΔΙΑΦΟΡΕΣ ΒΙΟΜΗΧΑΝΙΕΣ',
 'ΑΣΤΙΚΟΣ ΚΩΔΙΚΑΣ',
 'ΒΑΣΙΛΙΚΑ ΙΔΡΥΜΑΤΑ',
 'ΕΚΚΛΗΣΙΑ ΚΡΗΤΗΣ',
 'ΠΟΛΕΜΙΚΗ ΔΙΑΘΕΣΙΜΟΤΗΤΑ',
 'ΔΙΚΑΣΤΙΚΟΙ ΕΠΙΜΕΛΗΤΕΣ',
 'ΒΑΣΙΛΕΙΑ ΚΑΙ ΑΝΤΙΒΑΣΙΛΕΙΑ',
 'ΕΚΤΕΛΕΣΗ',
 'ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΟΙΚΟΝΟΜΙΑΣ']
```

Figure 3.2: Labels that appear in more than one hierarchical level

At first, we worked with the abnormality of labels existing in multiple hierarchy levels, and tried to apply a greedy technique in order to eliminate it. Every label that appeared in more than one levels (volume, chapter, subject) was greedily assigned to the level that appeared most times. Every instance that was assigned with this label but on a different level than the one we chose with the greedy technique, was dropped. The removal of these so called, “problematic labels” , initially led to the removal of 24 labels from the label set. All the “problematic labels” were assigned in two levels apart from one that was assigned in three. This action further led to the elimination of 39 more labels that only



```
CHAPTER:"ΔΙΑΦΟΡΑ ΑΣΦΑΛΙΣΤΙΚΑ ΤΑΜΕΙΑ" --->parent VOLUMES
 ["ΕΝΟΤΗΤΑ ΝΟΜΙΚΑ ΠΡΟΣΩΠΑ ΔΗΜΟΣΙΟΥ ΔΙΚΑΙΟΥ" ,"ΕΝΟΤΗΤΑ ΑΣΦΑΛΙΣΤΙΚΑ ΤΑΜΕΙΑ"]


SUBJECT:"ΓΕΝΙΚΑ" --->parent CHAPTERS
 ["ΚΕΦΑΛΑΙΟ ΠΑΡΟΧΟΙ ΣΤΑΘΕΡΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ" ,"ΚΕΦΑΛΑΙΟ ΠΑΡΟΧΟΙ ΚΙΝΗΤΩΝ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ"]


SUBJECT:"ΑΘΕΜΙΤΟΣ ΑΝΤΑΓΩΝΙΣΜΟΣ" --->parent CHAPTERS
 ["ΚΕΦΑΛΑΙΟ ΕΚΜΕΤΑΛΛΕΥΣΗ ΘΑΛΑΣΣΙΩΝ ΣΥΓΚΟΙΝΩΝΙΩΝ" ,"ΚΕΦΑΛΑΙΟ ΠΡΟΣΤΑΣΙΑ ΤΩΝ ΣΥΝΑΛΛΑΓΩΝ"]
```

Figure 3.3: Example of labels that have multiple parents

appear in instances that were dropped and thus our label set size was decreased by 63 labels in total.

We quickly realized that this method of dropping instances with labels that existed in more than one levels was not the best choice and decided to make the assumption that the differentiation of two labels also depends on the hierarchy level. More specifically, if two labels have exactly the same string value but belong on different levels then they are considered to be two different labels. In order to clarify this assumption, we prepend a string designating the hierarchy rank as shown in figure 3.4.

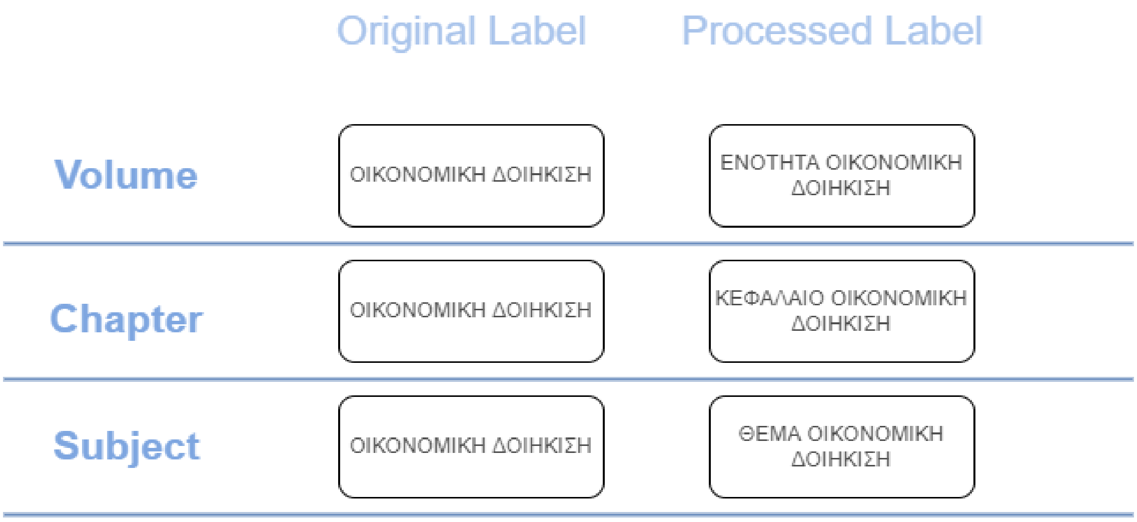


Figure 3.4: Example of including hierarchy rank in the label set

The first approach we followed in order to eliminate the second phenomenon of labels having multiple parents was to drop the instances of the dataset that had this abnormality. This resulted into removing 129 labels (16 from chapter level and 113 from subject level) and decreasing the total dataset size by almost 10%. As we can easily observe, this method greatly affects the original structure of RAPTARCHIS47k, so we experimented with a second approach. We tried to append the ancestors of a label into its string value. More specifically, for every label that belongs to the chapter level we concatenated the parent volume name with the chapter name, and for every subject we appended to its name, the name of its parent chapter and respectively chapter's parent volume name. This technique resulted in enlarging the label space as 3 labels were added in chapter level and 91 in subject level and consequently increased the computational complexity. This label set size of course is not considered to be extremely large for common LMTC and XMTC problems as these types of tasks usually involve hundreds of thousands labels and thus this addition of labels can be considered negligible.

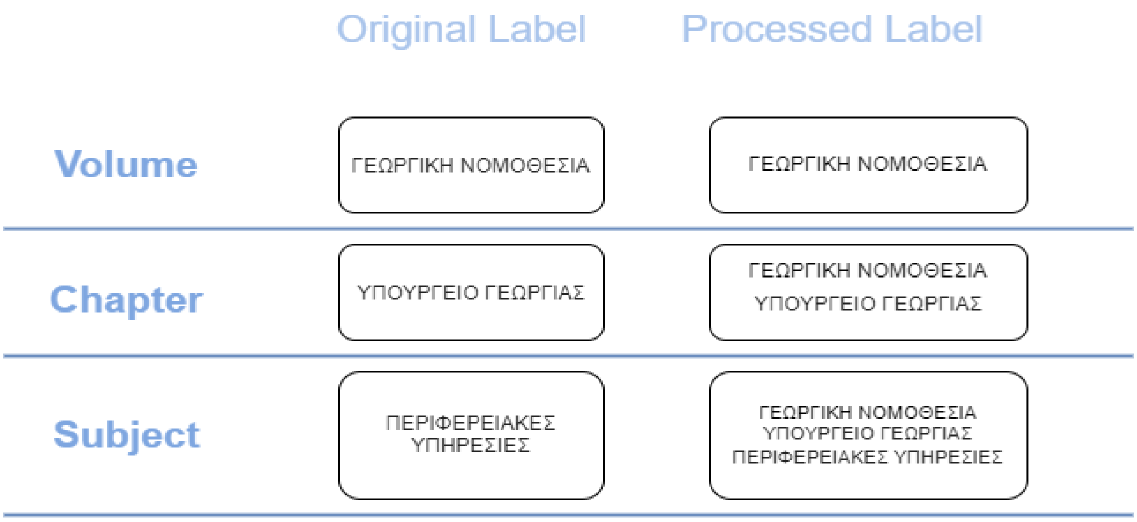


Figure 3.5: Example of including ancestor information in the label set

Both of the techniques we applied eliminated the two phenomenons that violate the tree constrains, as explained above, and may sound promising, but we didn't apply them during training and prediction of the examined methods. In this thesis we are not directly interested in utilizing the original label hierarchy, as our methods do not encode somehow this information. Nevertheless, we consider it to be a worth mentioning finding as it can be proven useful in future studies.

4. ALGORITHMS

In this chapter, we present the algorithms we experiment with along with some examples and comments about their performance.

4.1 Hierarchical PLT

The PLT model was firstly introduced in Jasinska et. al. [12] establishing groundwork for the development of further methods, some of which we discuss in this section. The research on the PLT method was driven by the need to decrease the training and prediction complexity of 1-vs-all approaches as mentioned in section 2.1.1, especially for extremely large datasets. As mentioned in the original paper [12] this model relies on the label tree approach [6, 5, 8] in which each leaf node corresponds to one label, and it was mainly designed for multi-label classification. The internal nodes of the tree contain classifiers that decide which child nodes the testing point should traverse forming a path from the root to a leaf node. When a testing point reaches a leaf, the final decision will be made about which label is relevant to this testing point.

4.1.1 Parabel

Parabel is a variation of a PLT approach, targeting the XMTC problem and was firstly introduced in Prabhu et al.[23]. Parabel is an evolution of the 1-vs-all approaches DISMEC [3] and PPDsparse [25] and addresses their computational complexity limitations by constructing a balanced tree over the labels, rather than the data points, such that similar labels end up together.

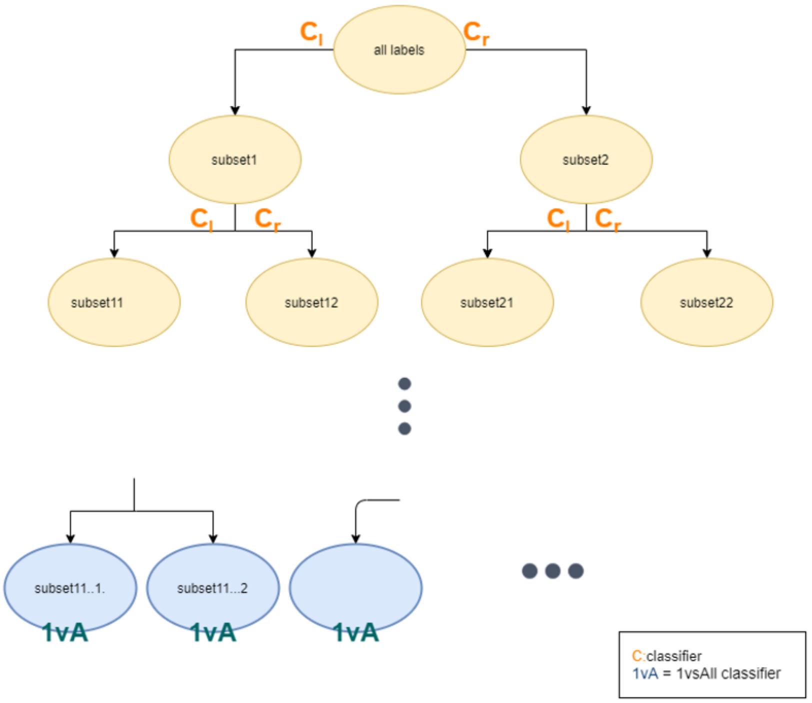


Figure 4.1: Parabel label tree structure, the blue nodes correspond to leaves and the orange to internal nodes.

Architecture

Based on the intuition that two labels are similar if they are assigned to similar training points, this method represents labels by the mean of the training points containing this label. Parabel learns a small ensemble of up to 3 deep narrow balanced trees with branching factor equal to two. The depth of the tree depends on the label set size along with the condition that distinguishes internal nodes from leaves. Every internal node has two linear classifiers and the leaf nodes are equipped with powerful 1-vs-all classifiers (figure 4.1). This method applies a constraint on the sizes of the child node's subsets. The label set size of nodes with the same parent node may differ by a maximum of one label. This constraint could be relaxed in case of imbalanced label sets.

Training and Prediction

During training, the root contains the whole label set and with the help of the linear classifiers the algorithm divides it into two smaller subsets. This partition procedure continues until a node contains less than a predefined number of labels (typically one hundred) and the node is converted into a leaf. For each testing point, the internal node classifiers decide whether the point should continue to the left, right or both children, forming multiple paths from the root to the leaves, as shown in figure 4.2. Essentially through this process the label space is divided into smaller parts resulting in forming a hierarchical relationship between the labels. Something worth mentioning is that this approach doesn't take into consideration the original label hierarchy, but only, through the constructing process, it creates its own hierarchy in order to better address the given task. The 1-vs-all classifiers in leaf nodes calculate the marginal probabilities of the testing point being associated with the corresponding labels. The final prediction is made by averaging these probabilities across the various trees of the ensemble.

Performance

Parabel is a tactful compromise between fast but low in performance tree approaches, and accurate but with high complexity 1-vs-all classifiers. It remains sub-optimal compared to the results of PPDsparse and DISMEC, but when large datasets are involved, it is one of the best choices, as the application of these algorithms is almost prohibitive. The disadvantages of this algorithm are mainly noticeable when the experiment dataset contains tail labels, something extremely common on LMTC and XMTC tasks. The strict constrain about the size of the child nodes force labels that are not particularly similar to end up together, and the small branching factor helps common labels dominate the hierarchy absorbing the tail labels. These constrains along with the propagation error due to the cascading effect of the deep trees, do not make the Parabel the best approach for few-case learning.

4.1.2 Bonsai

Bonsai[13] is a suit of algorithms based also on the PLT model. It was originally developed for XMTC tasks and is considered to be an evolution of Parabel.

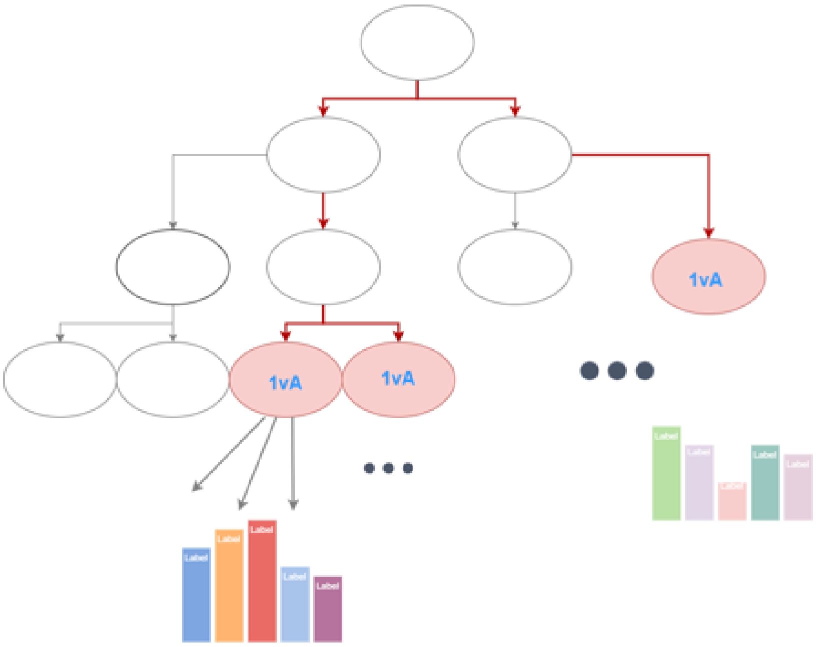


Figure 4.2: Parabel testing point traversing through a label tree.The red edges represent the path from the root to the leaves

Architecture

Bonsai has two main characteristics that identify its architecture. The first one is the general label representation it engages by providing three different label representation approaches:

- **Bonsai-i**: The labels are represented as the result of an aggregation function (e.g. average) on feature vectors of the documents they are assigned to. In this approach labels assigned to similar documents have similar representation.
- **Bonsai-o**: The representation of the labels depends on their co-occurrence with other labels. This is based on the idea that if two labels co-occur with a similar set of labels then there is a high probability they are related.
- **Bonsai-io**: Is a combination via concatenation of Bonsai-i, Bonsai-o.

The second characteristic of the Bonsai architecture is, the diverse and shallow trees this method constructs. The large branching factor, typically set to one hundred, doesn't allow Bonsai to grow deep trees with its common depth ranging between two or three levels. Also, every internal node of the tree has multiple linear classifiers (equal to the branching factor, 1 for every child) and all the leaves have 1-vs-all classifiers (figure 4.3), while no balancing constraints are applied.

Training and Prediction

The training procedure involves the non-leaf nodes in learning K linear classifiers separately, with each one corresponding to one child. These classifiers are responsible for

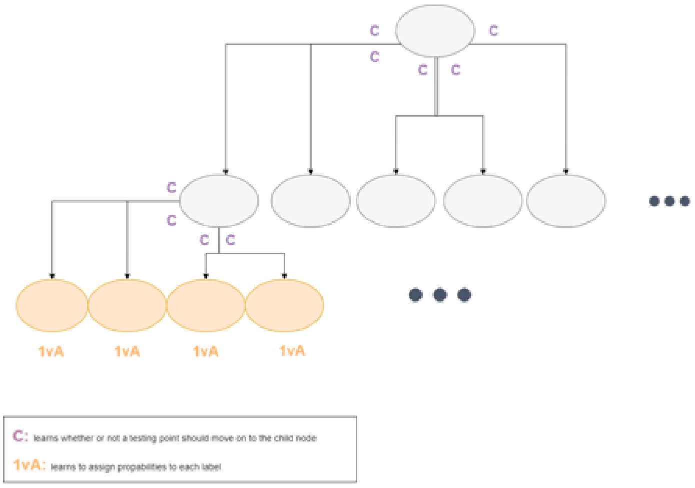

C
C
C
C
C
C
C
C
C
1vA
1vA
1vA
1vA
C: learns whether or not a testing point should move on to the child node
1vA: learns to assign propabilities to each label

Figure 4.3: Bonsai tree architecture

narrowing down the search space by deciding in which subset of child nodes the testing point should continue. When a testing point reaches one or more leaves, they predict the actual labels utilizing the 1-vs-all classifiers. Similar to Parabel, Bonsai utilises the label space by continuously partitioning the labels, and as a result a hierarchical relationship between the labels is created. Bonsai constructs three trees in total following the procedure described above, and averages their output in order to make the final decision.

Performance

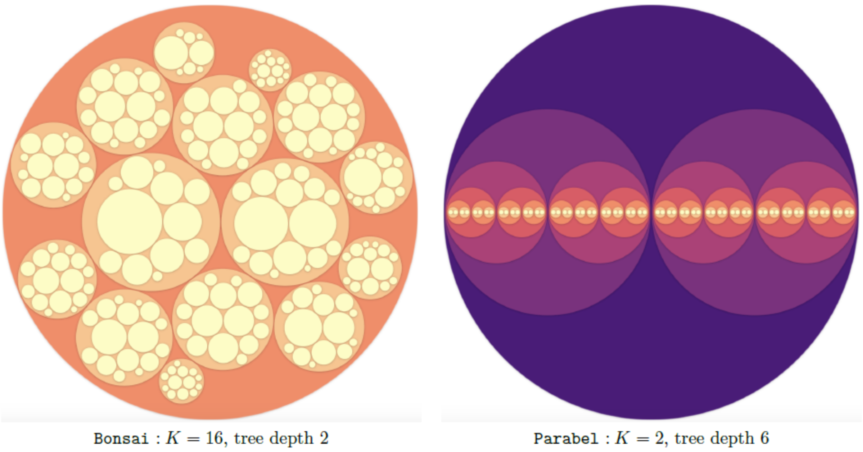
Bonsai uses a relatively big branching factor compared to Parabel, resulting into growing wide shallow trees instead of narrow deep ones, and hence avoids the propagation error Parabel suffers from. This family of algorithms doesn't apply any balancing constrains, ergo some tree nodes may contain quite more labels than others, helping similar labels ending in the same subset. For these reasons Bonsai is more suitable for tail labels. The comparison of the label space partitioning is better presented in figure 4.4 from the original paper [13]. A disadvantage Bonsai has compared to Parabel, is the high space complexity.

4.2 PLT and Attention Aware Networks Hybrid

Traditional LMTC and XMTC methods use Bag Of Word (BOW) representation for labels ignoring the deep semantic information of the data. The application of more advanced techniques based on deep neural networks , seemed almost prohibitive, as they could not scale up to extremely large label sets, due to their high time complexity. These two restrictions led to the research of a hybrid approach that combines the complexity efficiency of PLTs along with deep networks' ability to capture context.

4.2.1 AttentionXML

AttentionXML constitutes a hybrid PLT deep neural network approach and was firstly introduced by You et.al. [26].



Bonsai : $K = 16$, tree depth 2

Parabel : $K = 2$, tree depth 6

Figure 4.4: Comparison of partitioned label space by Bonsai and Parabel on EURLex-4K dataset. Each circle corresponds to one label partition (also a tree node), the size of circle indicates the number of labels in that partition and lighter color indicates larger node level. The largest circle is the whole label space.

Architecture

This method constructs a shallow and wide PLT similar to Bonsai and for every level l of a given tree with depth d trains a deep attention-aware mechanism, as shown in figure 4.5. It accepts raw text as input and with the help of a word representation layer, it converts it into deep semantic vectors that will be passed further to the next layers: first to a bidirectional LSTM layer[11], then to a Multi-Label Attention layer inspired by Lin et al. [15], and lastly to a fully connected linear layer that will make the final decision. The PLT of AttentionXML helps narrowing down the number of instances that are going to be given as input to the deep networks, which will eventually predict the target labels.

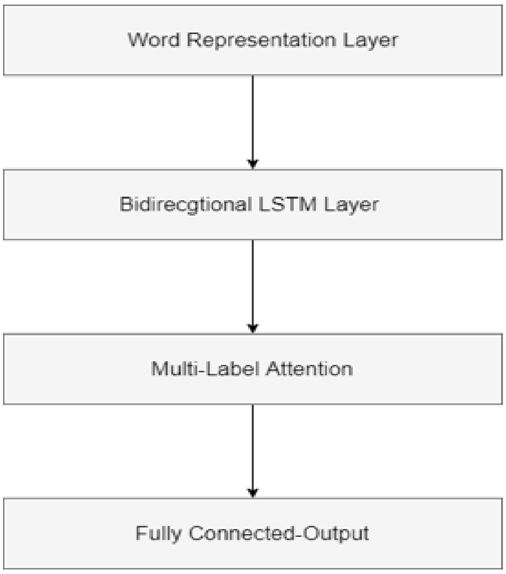


Figure 4.5: AttentionXML structure

Training and Prediction

If we define $AttentionXML_d$ as the deep neural network located in level d of a tree with max depth H, and a training point x, then the training procedure, as the authors of the original paper describe, involves the following steps:

1. Sort nodes of the $d-1$ level based on their scores predicted by $AttentionXML_{d-1}$ in descending order.
2. Keep only the first C nodes in the ordered collection provided by step 1, and choose their children as candidates $g(x)$.
3. The d-th level $(d > 1)$ is only trained by candidates $g(x)$

The training is conducted in a level-wise manner and the weights from $AttentionXML_{d-1}$ are passed down to $AttentionXML_d$ helping the next network converge faster. The steps provided above are also visualised in figure 4.6. Following the same idea during prediction, the testing point traverses the tree and for the d-th level ($d > 1$) it only predicts scores of nodes that belong to the $d-1$-th level top c candidates.

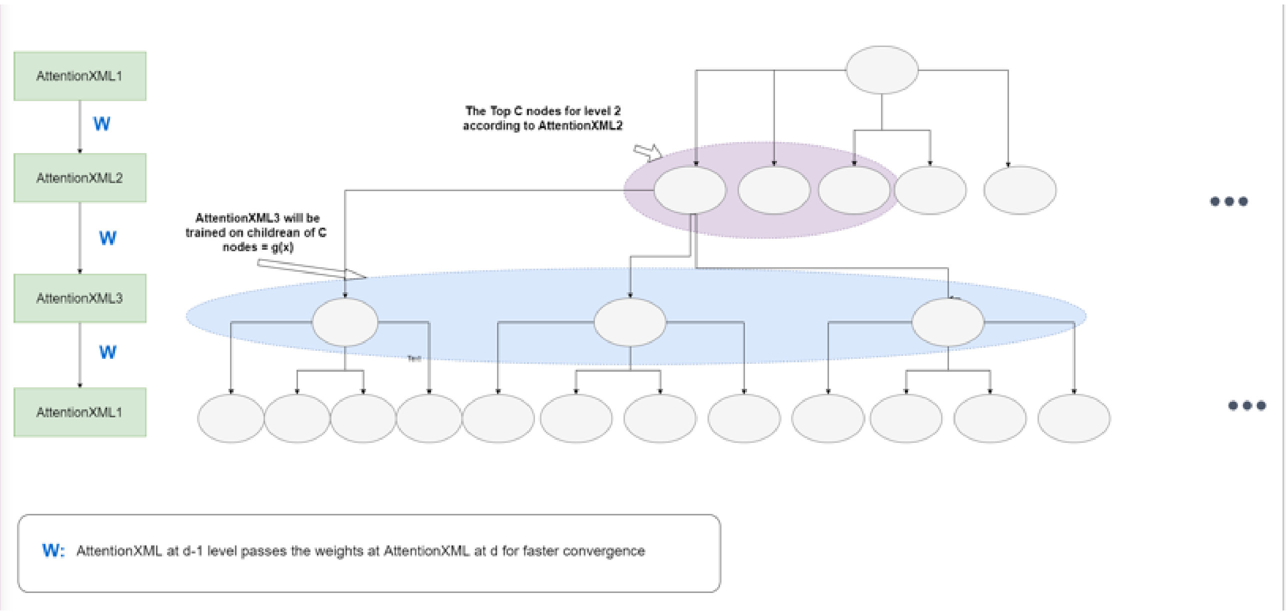

AttentionXML1
W
AttentionXML2
W
AttentionXML3
W
AttentionXML1
The Top C nodes for level 2 according to AttentionXML2
AttentionXML3 will be trained on childrean of C nodes = g(x)
Text
W: AttentionXML at d-1 level passes the weights at AttentionXML at d for faster convergence

Figure 4.6: AttentionXML training procedure

Performance

This combination of Probabilistic label Trees with attention-aware deep neural networks has proven to be very efficient on XMTC tasks and is considered to be one of the best methods concerning tail labels. Without the help of the PLT, the time and space complexity of the attention mechanism would make it impossible to use AttentionXML on XMTC tasks. Even though in this approach $H+1$ networks will be trained, the label size for each one is significantly smaller than the whole label set. As the authors of the original paper describe, the label size of $AttentionXML_1$ is only L/K^H, with L being the label set, K the branching factor and H the depth of the tree.

4.3 Transfer Learning

Transfer learning is a machine learning technique where a model is trained on one or more specific tasks and the knowledge gained from this procedure is used to solve another relative task that the model was not trained on. BERT constitutes a cornerstone of transfer learning in NLP tasks [20, 21, 4], as in most cases, fine-tuning a pretrained BERT model on a specific task, is more efficient than training a model from scratch.
Using the Greek language in NLP, is more difficult than using other more common languages like English, as the size of data in Greek cannot be compared to the the size and diversity of the data concerning the English corpus. Nevertheless, two variations of BERT for Greek language have been developed and offer exceptional results as part of many downstream tasks.

4.3.1 GREEK-BERT

GREEK-BERT is a monolingual BERT-based language model for modern Greek, and it was firstly developed and introduced by Koutsikakis et al. [14]. Greek-BERT is based on the BERT-BASE-UNCASED model introduced by Devin et. al. [9] that consists of twelve stacked Transformer layers (Figure 4.7). GREEK-BERT was trained on 29GB of Greek text originating from Wikipedia, Europarl and OSCAR. The processing required in order to use this corpus was to remove the Greek accents and convert the characters to lowercase as mentioned by the authors. It is now available as a pretrained model, and achieves top performance on NLP tasks like: Greek POS tagging, NER and natural language inference, outperforming state-of-the-art multilingual BERT-based models.

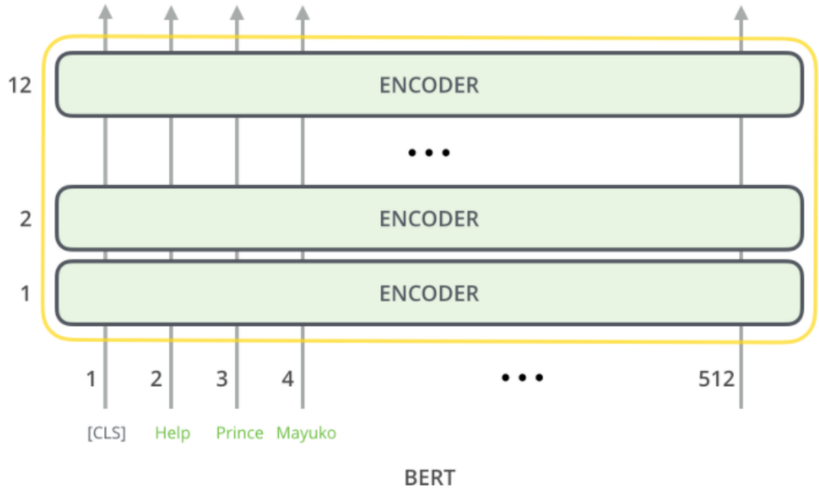


Figure 4.7: BERT-BASE-UNCASED structure

4.3.2 GreekLegalBERT

GreekLegalBERT is a monolingual domain specific pretrained BERT-based model that was firstly introduced as part of Athinaios K. thesis [2]. This model is also based on BERT-BASE-UNCASED, and it was trained on documents from Nomothesia platform, and contains mostly legal content including announcements, regulations and resolutions in the

Greek language. The total size of the input was only 4.5 GB of legal text due to the lack of resources, which is significantly smaller than the training input GREEK-BERT had. It is also used as a pretrained model and offers exceptional results in tasks like NER and multi-class classification.

5. EXPERIMENTS ON RAPTARCHIS47K

In this chapter we present the environment of our experiments on RAPTARCHIS47k, including the computational resources and the evaluation metrics we used. We also discuss the pre-processing steps we performed on the dataset for every method separately.

5.1 Corpus Pre-processing

The various methods we experimented with are based on different text representation approaches. In this section, we briefly describe the approach each method uses along with the pre-processing steps we needed to follow in order to utilize the most information from our corpus.

BOW Representation

The probabilistic label tree methods we experimented with (Parabel and Bonsai) use TF-IDF for label representation. This method belongs to the BOW approaches that handle text like a bag of words ignoring any context, grammatical or syntactical relationship between them. As a pre-processing step, we experimented with various common methods, like removing digits, removing punctuation, removing Greek stop words etc., but we chose to continue with removing accents and converting words to lowercase, as these two steps seemed to be the only ones that improved the performance. Something worth mentioning is that the Greek stop words we experimented with were relatively general containing words like “και”, “αλλα”, “ο”, etc., maybe more domain specific stop words would help improve further the performance.

Greek Word Embeddings

The AttentionXML method accepts deep semantic vectors (word embeddings) as input. This type of vectors captures syntactic relationship between words along with their context meaning. Our dataset is in Greek so we chose to experiment with a set of pretrained Greek word embeddings that was firstly introduced in [1]. This model was trained on Greek Legal resources and produced 100-dimensional vectors with a vocabulary of total 428.963 words. The pre-processing steps we performed were the ones described by the authors of this paper: convert letters to upper case, remove accents and replace digits with ’D’. In this way, we increased the probability of words from our corpus matching words in the pretrained model’s vocabulary and thus enhance the dataset representation.

GREEK-BERT Tokenizer

For the Transformer-based approaches we used two variations of BERT Tokenizer in order to encode the input corpus into something that the models can interpret. BERT-Tokenizer breaks input words into sub-words until it finds a match in its predefined vocabulary. If it is not possible to successfully split the words into sub-words it assigns to the whole word a special token (typically [UNK]). There are also three more special tokens: [CLS] which is a special classification token and is added at the beginning of the sentence, [SEP] token which separates sentence A from sentence B and is placed at the end of sentence A. If the input doesn't contain a second sentence, or its length is smaller than a predefined max length then the special [PAD] token is added until the sentence reaches the desirable size. For the GREEK-BERT model we used the GREEK-BERT-Tokenizer supplied by the [14], containing a general vocabulary with 35000 words from the Greek language. In the case of GreekLegalBERT, we used a more domain specific configuration containing a vocabulary with size 35100. For both models, we performed the pre-processing steps as suggested in the original GitHub repository, which is to the remove the Greek accents and convert the characters to lower case. Something worth mentioning is that after the pre-processing procedure, the mean tokens per instance were 542.10, where the maximum number of tokens BERT can process is 512. Every instance that exceeded this limit was forcibly truncated without the application of any more sophisticated strategy. It has been suggested in the past that this mandatory truncation of words will not harm the overall performance in cases where the vast majority of data can be adequately represented. This is also true in our case as more than the 80% of the dataset will remain intact, with the common sentences length ranging from 50 to 250 words on average.

5.2 Evaluation

The evaluation stage of the experiment is very important as the correct handling will shed light on the strengths and weaknesses of the methods we experimented with. Our assignment falls into the multi-label classification category so the metrics we chose to experiment with are the suggested ones for such classification tasks. These metrics are shown below:

- **Precision**: Measures how many of the marked as positive instances are indeed positive. It ignores the cases of instances that were indeed positive but were wrongly marked as negative.
- **Recall**: Measures how many of the true positive instances were marked as positive. This metric will score 100% in case of assigning all the instances as positive and should therefore be used in conjunction with precision.
- **F-score**: The harmonic mean of precision and recall. More specifically, we used the F1-score that weights evenly the two metrics.
- **nDCG score**: Is a measure of a model's ranking quality. It takes into consideration the sequence of the returned results penalizing highly relevant instances that appear after others with smaller relevance, and normalizes the results across the different queries based on the ideal result.

All these metrics take values that range from 0 to 1 with the second one indicating the best performance. For convenience, these values have been reduced to percentages.

5.3 Experiment set up

The PLT-based experiments run on the Colab Environment provided by Google, due to their small recourse requirements. In the case of the hybrid and BERT-based methods that engage deep recurrent networks, the computational requirements were significantly higher and thus we utilized a machine with an NVDIA Ge-Force GTX 1080ti GPU.

For each of the methods we examined in this thesis, we sought a hyper-parameter configuration that could lead to optimal performance. The search space we experimented with differs among the methods and mainly stems from the author's suggestions and former work on this topic.

The final configuration, with which the model was fined-tuned on the whole dataset, was chosen based on the best F1-score on the validation set.

In the case of the Parabel algorithm, we experimented with the number of trees that the algorithm grows along with the type of linear classifiers it uses. The difference in performance between growing three or five trees was negligible and the training/prediction time was significantly larger, so we continued with growing three trees. Our results confirmed that the suggested configuration by the bibliography (for both the number of trees and type of the classifiers) gave the best performance.

Table 5.1: Parabel parameter Grid Search

Parabel		
	Search Area	Best Configuration
Number of Trees	{1,2,3,5}	3
Linear Classifiers	{L2R_L2LOSS_SVC, L2R_LR}	L2R_L2LOSS_SVC

The experiments yield that the suggested by the bibliography Bonsai-i as the chosen label representation, along with growing three trees, gave also the best performance for Bonsai. Something worth mentioning is that if our label space was significantly larger (hundreds of thousands or even million labels) then Bonsai-io would be more suitable, as its extended representation capacity would be required. The experiments showed the default branching factor (100) was not the best choice and this can be easily explained by the fact that our label set is really small compared to the common XMTC label sets and thus a smaller branching factor was more appropriate.

Table 5.2: Bonsai parameter Grid Search

Bonsai		
	Search Area	Best Configuration
Number of Trees	{1,2,3,5}	3
Branching Factor	{20,50,100}	50
Label Representation	{Bonsai-i, Bonsai-o, Bonsai-io}	Bonsai-i

The AttentionXML grid search yield the best results were given by the default configuration, apart from the hidden size and the optimizer warm-up parameter. Something worth mentioning about AttentionXML is that in all the experiments we used dropout less than 0.5, the performance was dreadful, indicating that the deep neural networks really needed this regularization in order to generalize well.

In order to avail GREEK-BERT and GreekLegalBERT in our multi-label classification task, we used the classification token [CLS] as input to a linear layer, and then we applied the sigmoid function with the final output size being equal to the label space. This special token is explicitly targeting classification tasks and the approach we followed is the suggested one. Every output corresponds to the probability of assigning the relevant output label to the input instance. During running these two models, we used the Adam optimizer and the binary cross entropy loss as criterion. We also included a scheduler that reduces the learning rate ($factor = 0.1$) after a few rounds it sees no decrease ($threshold = 1e^{-4}$)

Table 5.3: AttentionXML parameter Grid Search

AttentionXML		
	Search Area	Best Configuration
Number of Epochs	{10,20, 30, 50}	30
Dropout	{0.2, 0.3, 0.5}	0.5
Hidden Size	{256, 512}	512
SWA WARMUP	{5,10,20}	5

in the validation loss. We also used an early stop mechanism and ran the model for 20 epochs, but in both cases the models needed 14-15 epochs in order to converge. The biggest training loss decline took place in the first 5 epochs, while in the last 10, the decreasing rate was small, resulting in the scheduler activation. The search area is the same for both method, due to their high resemblance, but the best configuration differs.

Table 5.4: GREEK-BERT parameter Grid Search

GREEK-BERT		
	Search Area	Best Configuration
Learning Rate	{0.00001, 0.00002, 0.0003, 0.00004, 0.0000}	0.000005
Dropout	{0.0, 0.1, 0.2}	0.1

Table 5.5: GreekLegalBERT parameter Grid Search

GreekLegalBERT		
	Search Area	Best Configuration
Learning Rate	{0.00001, 0.00002, 0.0003, 0.00004, 0.00005}	0.00004
Dropout	{0.0, 0.1, 0.2}	0.2

6. RESULTS

In this chapter we present the results of the examined methods. We decided to present them across the three different categories the label set forms based on their appearance frequency (all labels, frequent, few-shot), and we are more interested in the comparison between them rather than their performance as individual methods. We mainly focus on the results given by the F1-score and the nDCG score. Precision and Recall serve as auxiliary metrics that help us to further understand each method's behaviour

Before we start reporting the results, we consider it is worth mentioning the different running time, the various methods present. More specifically, Parabel and Bonsai proved to be the best concerning the training and prediction time being approximately 12 times faster than the AttentionXML, which in its turn was significantly faster (10 times) than the Transformer-based approaches. The difference in time complexity is mostly noticeable in the training procedure but also in the prediction phase, and can raise questions about which method should be used, especially in real time applications.

All labels

In this category we don't focus on any particular subsets of labels and we evaluate all the methods across the whole label set. As presented in table 6.1, Parabel achieves the worst performance among the methods examined. Something that probably contributed to this outcome is the propagation error it faces due to its deep and narrow trees. On the other hand, both GREEK-BERT and GreekLegalBERT offer excellent results, with GreekLegalBERT being slightly better. It doesn't surprise us at all, as Transformer-based methods proven to capture the syntactic and grammatical information with great precision .

Table 6.1: The results of the experiments for the whole dataset

ALL LABELS				
	P	**R**	**F1**	**nDCG**
Parabel	73.87	74.61	74.23	78.02
Bonsai	76.32	77.03	76.67	80.14
AttentionXML	78.02	78.51	78.26	81.37
GREEK-BERT	82.04	82.64	82.33	82.64
GreekLegalBERT	**83.07**	**83.68**	**83.37**	**83.68**

Frequent Labels

In this section we study the performance of the various methods only on the frequent classes (classes that appear more than 10 times on the training set). It is noticeable (table 6.2) that GREEK-BERT and GreekLegalBERT give approximately the same results, with the second one outperforming the first one by almost 1%. This difference in performance is not negligible and highlights the strengths of this domain specific model.

Table 6.2: The results of the experiments for the frequent labels

FREQUENT LABELS				
	P	**R**	**F1**	**nDCG**
Parabel	73.58	79.88	76.59	81.58
Bonsai	75.36	82.18	78.62	83.52
AttentionXML	77.93	81.32	79.59	83.24
GREEK-BERT	83.09	85.00	84.00	85.01
GreekLegalBERT	**83.81**	**86.22**	**84.95**	**86.22**

Few-shot Labels

This part of the evaluation is the most interesting one as it is considered to be the most difficult. We observe at the table 6.3, that Bonsai outperforms every other method, with Parabel following right after, giving comparable results. AttentionXML along with GREEK-BERT and GreekLegalBERT are far behind these hierarchical PLT methods and do not constitute a competitive choice for this subcategory.

Table 6.3: The results of the experiments for the few-shot labels

FEW SHOT LABELS				
	P	**R**	**F1**	**nDCG**
Parabel	**82.99**	59.64	69.41	59.50
Bonsai	81.67	**64.20**	**71.89**	**63.60**
AttentionXML	73.03	53.70	61.89	52.24
GREEK-BERT	63.06	58.01	58.27	58.01
GreekLegalBERT	65.58	58.02	59.35	58.02

Overall Comparison

If we are to compare all the results across the different levels of label frequency occurrence we will see that there is no one-size-fits-all method. Before proceeding to the analysis of the best methods, it is worth talking about the methods that gave the lowest scores.

Parabel did not succeed in taking the first place in any sub-task, but managed to keep a relatively steady behaviour across the different categories while also giving the second best result in the most difficult sub-task, which is to predict the “few-case” labels. Also, according to the bibliography, AttentionXML is supposed to be an evolution of Bonsai that achieves better results. In our case, AttentionXML performed just as well as Bonsai in the first two categories, but in the third category it under-performed with scoring only 61.89%. We believe that the reason behind this behaviour is the small label set size of

RAPTARCHIS47k. This model is a XMTC dedicated technique targeting tasks with hundreds of thousands or even million of labels while our label set size is just slightly larger than 2700 labels. GREEK-BERT and GreekLegalBERT fell sort against the third category, with GREEK-BERT monitoring the worst result. Apart from this sub-task, both approaches scored top results in the ”all labels” case; their exceptional ability to capture semantic context helped them to predict with high accuracy the most common labels, that constitute the majority of the dataset. Between these two Transformer-based methods we saw that GreekLegalBERT was the best, as its speciality on legal content proved more than useful for this classification task. On the other hand, Bonsai showed its ability to predict “few-case” labels by achieving the best result in this particular sub-task. Combined with Parabel’s result, we conclude that indeed the LMTC specific hierarchical PLT approaches are more suitable than the state-of-the-art Transformer-based methods on tasks that tail labels constitute the majority. It is quite difficult to name a method as the best for this classification task, but Bonsai is a strong candidate, since it scored 71.89% on the “few-case” label set and proved to be more than adequate across the other two sub-tasks. But then again such a conclusion would be general. The choice of method depends entirely on the purpose of the task and varies according to the targeting subset of labels.

Something worth mentioning is that the hierarchical PLTs offered these results, utilizing a lot fewer resources as they were 120 times faster than the Transformer-based and did not require the usage of a GPU or a TPU. Our label set size, as mentioned above, is considered to be small compared to common LMTC and XMTC tasks and thus makes it feasible to experiment with these Transformer-based methods. If a dataset with hundreds of thousands of labels was the study case then the use of such computational heavy approaches would be almost prohibitive.

Table 6.4: Overall results based on the F1-score metric

	All labels	**Frequent**	**Few-case**
Parabel	74.23	76.59	69.41
Bonsai	76.67	78.62	**71.89**
Attention XML	78.26	79.59	61.89
GREEK-BERT	82.33	84.00	58.27
GreekLegalBERT	**83.37**	**84.95**	59.35

7. CONCLUSION AND FUTURE WORK

In summary, we experimented with the Greek legal document classification task on the RAPTARCHIS47k dataset. It's essentially a multi-label classification problem belonging to the family of LMTC tasks. We focused on the skewed label distribution this dataset engages and tried to evaluate a battery of methods on different subsets of labels, by splitting them based on the frequency of their occurrences.
We experimented with various methods ranging from hierarchical PLTs to state-of-the-art Transformer-based models, and our experiments yield that there is no rule-of-thumb about the method that should be used, it strongly depends on the very nature of the problem along with the intended purpose. If we are more interested in the frequent classes then we most likely work with GreekLegalBERT or GREEK-BERT: both constitute state-of-the-art models that also achieved top-notch results on this particular sub-task. GreekLegalBERT was slightly better due to its domain specific vocabulary, but if the study case involves corpus from another domain then the GREEK-BERT constitutes a valid alternative. On the other hand, in the "few-shot" case, which is the most difficult to predict, Bonsai was the best, outperforming every other method. In the general case where we are interested in the overall good performance, GreekLegalBERT proved to be the best choice. This research showed that indeed the hierarchical PLT algorithms address the third and most difficult sub-task, which is to predict classes that are rare on the training set, with great success, and gave overall a decent performance. The Transformer-based methods offered top-notch results in the "frequent" labels but did not prove capable of tackling the ”few-case” problem as well as PLTs did.

A next step would be to actively include the actual label hierarchy information in the training process. Many researches have shown that including such information boosts performance especially in tail labels. Also, the layer wise training suggested in [16] could be used in GREEK-BERT and GreekLegalBERT in order to enhance their performance in such classification tasks. Of great scientific interest would be the further experimentation on methods that are capable to predict classes that have never been seen on the training set, and RAPTARCHIS47K could constitute the study case as it meets the criteria. The maintenance and enhancement of Greek datasets like this one assist and promote the scientific research concerning Greek NLP.

ABBREVIATIONS - ACRONYMS

AI	Artificial Intelligence
NLP	Natural Language Processing
BERT	Bidirectional Encoder Representations from Transformers
PLT	Probabilistic Label Tree
LMTC	Large Scale Multi-label Classification
XMTC	Extreme Scale Multi-label Classification
DISMEC	Distributed Sparse Machines for Extreme Multi-label Classification
PDsparse	Primal Dual sparse
PPDsparse	Parallel Primal Dual Sparse
BOW	Bag Of Words
BiLSTM	Bidirectional Long Short Term Memory
LSTM	Long Short Term Memory
NER	Named Entity Recognition
POS	Part Of Speech
TF-IDF	Term Frequency-Inverse Document Frequency
nDCG	Normalized Discounted Cumulative Gain
GPU	Graphics Processing Unit
TPU	Tensor Processing Unit
RNN	Recurrent Neural Network
GLC	Greek Legislation Code
Europarl	European Parliament Proceedings Parallel Corpus

BIBLIOGRAPHY

[1] Iosif Angelidis, Ilias Chalkidis, and Manolis Koubarakis. Named entity recognition, linking and generation for greek legislation, September 2018. http://cgi.di.uoa.gr/~koubarak/publications/2018/jurix2018.pdf.

[2] Konstantinos I. Athinaios, November 2020. https://pergamos.lib.uoa.gr/uoa/dl/frontend/el/browse/2927727#contents.

[3] Rohit Babbar and Bernhard Scholkopf. Dismec - distributed sparse machines for extreme multi-label classification, September 2016. https://arxiv.org/pdf/1609.02521.pdf.

[4] Xingce Bao and Qianqian Qiao. Transfer learning from pre-trained BERT for pronoun resolution. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 82–88, Florence, Italy, August 2019. Association for Computational Linguistics.

[5] Samy Bengio, Jason Westonand, and David Grangier. Label embedding trees for large multi-class tasks, 2010. https://papers.nips.cc/paper/2010/file/06138bc5af6023646ede0e1f7c1eac75-Paper.pdf.

[6] Alina Beygelzimer, John Langford, Yuri Lifshits, Gregory Sorkin, and Alex Strehl. Conditional probability tree estimation analysis and algorithms, 2009.

[7] Ilias Chalkidis, Manos Fergadioti, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Androutsopoulos. Ion. An empirical study on large-scale multi-label text classification including few and zero-shot labels, Octomber 2020. https://arxiv.org/pdf/2010.01653.

[8] Jia Deng, Sanjeev Satheesh, Alexander C. Berg3, and Li Fei-Fei. Fast and balanced: Efficient label tree learning for large scale object recognition, 2011. https://proceedings.neurips.cc/paper/2011/file/5a4b25aaed25c2ee1b74de72dc03c14e-Paper.pdf.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, May 2019. https://arxiv.org/pdf/1810.04805.pdf.

[10] Ian E.H. Yen Xiangru Huang Wei Dai Pradeep Ravikumar Inderjit Dhillon. Pd-sparse : A primal and dual sparse approach to extreme multiclass and multilabel classification, May 2017. http://people.csail.mit.edu/xrhuang/publications/PDSparse.pdf.

[11] HochreiterS., , and Schmidhuber. J. Long shortterm memory. neural computation, 1997.

[12] Kalina Jasinska, Krzysztof Dembczynski, Robert Busa-Fekete, Karlson Pfannschmidt, Timo Klerx, and Eyke Hullermeier. Extreme f-measure maximization using sparse probability estimates, September 2016. http://www.inf.u-szeged.hu/~busarobi/PDFs/Jasinska16.pdf.

[13] Sujay Khandagale, Han Xiao, and Rohit Babbar. Bonsai - diverse and shallow trees for extreme multi-label classification, August 2019. https://arxiv.org/pdf/1904.08249s.

[14] Jhon Koutsikakis, Ilias Chalkidis, Prodromos Malakasiotis, and Ion Androutsopoulos. Greek-bert: The greeks visiting sesame street, Sep 2020. https://arxiv.org/pdf/2008.12014.pdf.

[15] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding, March 2017. https://arxiv.org/pdf/1703.03130.pdf.

[16] Nikolaos Manginas, Ilias Chalkidis, and Prodromos Malakasiotis. Layer-wise guided training for bert: Learning incrementally refined document representations including few and zero-shot labels, Octomber 2020. https://arxiv.org/pdf/2010.05763.pdf.

[17] Christos Papaloukas, Ilias Chalkidis, Konstantinos Athinaios, Despina-Athanasia Pantazi, and Manolis Koubarakis. Multi-granular legal topic classification on greek legislation, September 2021. https://arxiv.org/pdf/2109.15298.pdf.

[18] Christos N. Papaloukas. Legal text classification based on greek legislation, DECEMBER 2020. https://pergamos.lib.uoa.gr/uoa/dl/frontend/el/browse/2931361#contents.

[19] Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification, March 2004. https://www.jmlr.org/papers/volume5/rifkin04a/rifkin04a.pdf.

[20] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification?, May 2019. https://arxiv.org/pdf/1905.05583.pdf.

[21] Vijay Srinivas Tida and Sonya Hsu. Universal spam detection using transfer learning of bert model. https://arxiv.org/ftp/arxiv/papers/2202/2202.03480.pdf.

[22] Efstratios G. Vamvourellis. Compbertition: Which bert model is better for greek legal text classification?, August 2021. https://pergamos.lib.uoa.gr/uoa/dl/frontend/el/browse/2960898#contents.

[23] Yashoteja Prabhu Anil Kag Shrutendra Harsola Manik Varma. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising, April 2018.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin and. Attention is all you need. in 31th annual conference on neural information processing systems usa., 2017. https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

[25] Ian E.H. Yen Xiangru Huang Wei Dai Pradeep Ravikumar Inderjit Dhillon Eric Xing. Ppdsparse: A parallel primal-dual sparse method for extreme classification, August 2017. https://www.cs.cmu.edu/~pradeepr/paperz/ppdsparse.pdf.

[26] Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification, November 2019. https://arxiv.org/pdf/1811.01727.pdf.

[27] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27, 2015.