NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

BSc THESIS

Comp-BERT-ition: Which BERT model is better for Greek legal text classification?

Efstratios G. Vamvourellis

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

ATHENS

August 2021

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ποιο BERT μοντέλο είναι καλύτερο στην κατηγοριοποίηση Ελληνικών νομικών εγγράφων;

Ευστράτιος Γ. Βαμβουρέλλης

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

ΑΘΗΝΑ

Αύγουστος 2021

BSc THESIS

Comp-BERT-ition: Which BERT model is better for Greek legal text classification?

Efstratios G. Vamvourellis

S.N.: 1115201600014

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

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ποιο BERT μοντέλο είναι καλύτερο στην κατηγοριοποίηση Ελληνικών νομικών εγγράφων;

Ευστράτιος Γ. Βαμβουρέλλης

Α.Μ.: 1115201600014

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

ABSTRACT

Deep Neural Networks (DNN) is a very hot subfield of Artificial Intelligence (AI). Experts believe that it may be the future of Computer Science. Natural Language Processing (NLP) is the area of AI and linguistics concerned with the interactions between computers and human language, in particular how to program computers to process and analyze natural language data. With the creation of BERT [5], a large DNN tasked with the understanding of the English Language, in 2018, and its integration into the Google search algorithm, the field of NLP took a big leap forward. Since then, only a few models have managed to surpass BERT by a bit. Until recently, in 2020, BERT and its variants, were considered state of the art. The current thesis examines different variations of the BERT model, trained on different datasets and their ability to classify Greek legal documents. It also discusses ways to further improve our fine-tuned models for legal domain tasks, like domain specific adaptation and vocabulary expansion. We use the RAPTARCHIS [3] dataset which provides Greek legal documents for three classification tasks. Our fine-tuned Greek-only models have very similar performance, while the multilingual model falls behind. We conclude that using domain and task adaptive pre-training, the performance will surly improve across all models. We also hypothesize that, based on known heuristics, described in chapter 5, the multilingual model could surpass the others. The metrics we use are precision (P), recall (R) and F1 score. We chose these metrics to have a direct comparison with the prior models evaluated on the same dataset.

SUBJECT AREA: Artificial Intelligence

KEYWORDS: BERT, Neural Networks, Natural Language Processing,Legal Documents

ΠΕΡΙΛΗΨΗ

Τα βαθιά νευρωνικά δίκτυα είναι ένας σύγχρονος τομέας της Τεχνητής Νοημοσύνης. Πολλοί επιστήμονες πιστεύουν ότι μπορεί να είναι το μέλλον των υπολογιστών. Η επεξεργασία φυσικής γλώσσας είναι μια περιοχή της Τεχνητής Νοημοσύνης και της Γλωσσολογίας που εξετάζει την αλληλεπίδραση των υπολογιστών με την ανθρώπινη γλώσσα, και πιο συγκεκριμένα, πώς θα μάθουμε σε ένα πρόγραμμα να επεξεργάζεται και να καταλαβαίνει δεδομένα φυσικής γλώσσας. Με τη δημιουργία του μοντέλου BERT [5], ενός μεγάλου βαθιού νευρωνικού δικτύου που κατασκευάστηκε για να καταλαβαίνει την Αγγλική γλώσσα, το 2019, και την ενσωμάτωσή του στη μηχανή αναζήτησης της Google, ο τομέας της εξεργασίας φυσικής γλώσσας έκανε ένα άλμα μπροστά. Από τότε, μόνο λίγα μοντέλα έχουν καταφέρει να ξεπεράσουν το BERT κατά ελάχιστο. Αυτή η πτυχιακή εξετάζει διαφορετικές εκδοχές του BERT, που έχουν εκπαιδευτεί σε διαφορετικά δεδομένα, και την ικανότητά τους να κατηγοριοποιήσουν Ελληνικά νομικά έγγραφα. Επίσης σχολιάζει τους τρόπους που μπορούμε να βελτιώσουμε τα μοντέλα μας, προσαρμόζοντάς τα σε ένα συγκεκριμένο γλωσσικό τομέα και επεκτείνοντας το λεξιλόγιό τους. Χρησιμοποιούμε τη συλλογή δεδομένων RAPTARCHIS [3] που περιέχει Ελληνικά νομικά έγγραφα διαθέσιμα για τρία διαφορετικά προβλήματα κατηγοριοποίησης. Τα τελικά Ελληνικά μοντέλα πετυχαίνουν πολύ παρόμοια απόδοση, ενώ το πολυγλωσσικό μοντέλο υστερεί. Καταλήγουμε ότι προσαρμόζοντας τα μοντέλα μας στο συγκεκριμένο γλωσσικό τομέα των νομικών κειμένων, σίγουρα θα βελτιώσουμε την απόδοση τους. Επίσης, βασισμένοι σε γνωστές ευρετικές μεθόδους, που περιγράφονται στο κεφάλαιο 5, υποθέτουμε ότι το πολυγλωσσικό μοντέλο θα μπορούσε να ξεπεράσει τα άλλα. Οι μετρικές αποτίμησης της αποτελεσματικότητας που χρησιμοποιούμε είναι η ακρίβεια (precision), η ανάκληση (recall) και η μετρική F1. Διαλέξαμε αυτές τις μετρικές αποτίμησης για να έχουμε απευθείας σύγκριση αποτελεσματικότητας σε σχέση με προηγμενα μοντέλα που αξιολογήθηκαν στο ίδιο σύνολο δεδομένων.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Θεματική Περιοχή

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: BERT, Νευρωνικά Δίκτυα, Επεξεργασία Φυσικής Γλώσσας, Νομικά ’Εγγραφα

Ευχαριστώ τους γονείς μου για την ανιδιοτελή τους αγάπη, δεν θα βρισκόμουν εδώ χωρίς αυτούς.

ACKNOWLEDGEMENTS

I would like to sincerely thank my supervisors, Manolis Koumparakis and Despina-Athanasia Pantazi for their invaluable support and guidance throughout the planning and development of this Thesis. I couldn't ask for better guides to introduce me to the world of research. Our collaboration was flawless, they gave me the freedom to explore this topic as I saw fit and at my own pace.

CONTENTS

LIST OF FIGURES

PREFACE

I began programming at the age of 10, so I was always into computers. Until my first year in college I was unsure of which field of computer science I liked the most. That was when I was introduced to AI, with a small introductory project in genetic programming. From that moment I was sure I wanted to work in AI. I had courses in many different areas: Machine Learning, Linear algorithms, Reinforcement Learning, but what intrigued me the most was neural networks. That was why I chose this subject for my thesis. I never expected to work with such a large model, as BERT, so soon in my academic journey. The experience was more than rewarding. I will do my best to walk you through our thought process and share our findings.

1. INTRODUCTION

In this thesis we will try to find the best performing BERT model on the RAPTARCHIS dataset. We will also try to find ways to increase the performance of BERT models on this dataset and extract some statistics and metrics.

The RAPTARCHIS dataset contains Greek legislation from the Official Government Gazette. This dataset embeds three classification tasks, each one with more classes, hence more difficult, than the other. To complete these tasks we chose the BERT model. Bidirectional Encoder Representations from Transformers (BERT) is the name of a large language model introduced in 2018 by the Google AI team. BERT achieved state of the art performance in a wide variety of NLP tasks. We chose to use Greek-BERT [8], Greek-Legal-BERT [7] and Multilingual-BERT (M-BERT) [4].

Our task can be done using a variety of methods. We chose to use a pretrained statistical language model, or language model for short. A language model is a probability distribution over sequences of words. Given such a sequence, say of length m, it assigns a probability $P(w_1, ..., w_m)$ to the whole sequence [24].

Our dataset consists solely of legal documents. The legal language is unique and can be very different from the every day spoken word. This difference is obvious especially in Greek, where up until 1977 all the official legal documents had to be written in Katharevousa, an older form of Greek, very different from Demotic-Modern Greek [16]. For this reason, we briefly review the literature for techniques to expand or alter BERT's vocabulary. This in theory and in practice can help our model 'understand' the legal domain language better.

Another technique that can help our model do better on datasets containing domain specific language is domain adaptive pre-training (DAPT) and task adaptive pre-training (TAPT) [6]. This is a very new technique, introduced in 2020, that doesn't alter BERT's vocabulary.

Our fine-tuned models showcase the effectiveness of pre-training a model on a general corpus, and then fine-tuning it on a domain specific dataset. We compute some known heuristics that can tell us if TAPT and/or DAPT improve the results on the RAPTARCHIS dataset, and more generally, on the Greek legal domain. Lastly, we experiment with M-BERT and conclude that it can be powerful not only for low resource languages but for high resource languages as well.

This thesis is divided in the following five chapters:

- In chapter 2, we provide background information about the BERT model and other variations of it. Moreover, we introduce the RAPTARCHIS dataset in more detail.
- In chapter 3, we review the literature on ways to improve our suggested models for the legal domain using vocabulary alteration or continuing the pre-training.
- In chapter 4, we fine-tune Greek-BERT, Greek-Legal-BERT and M-BERT on the RAPTARCHIS dataset and showcase the results. We also explain intuitively why M-BERT may achieve such good results. Lastly, we experiment with removing all the numbers from the dataset and try to explain the results.
- In chapter 5, we calculate some statistics for our fine-tuned models based on the used legal datasets. We also calculate some previously known heuristics that may showcase the benefits of TAPT and DAPT.

- In chapter 6, we reach some conclusions and try to predict the effectiveness of domain and task adaptive pre-training in comparison to the vocabulary alteration or the expansion on the Greek legal domain.

2. BACKGROUND AND RELATED WORK

2.1 BERT

In this chapter we will analyze the original BERT model introduced in Devlin et al [5] in 2018. We will also understand the dataset we will be using, RAPTARCHIS.

2.1.1 Pre-trained language models and fine-tuning

BERT is a pre-trained language model. This means that it was firstly trained on a general task on a very large unannotated dataset. This pre-training procedure can not be done by everyone, as it requires large amounts of training data and computational resources. BERT's weights were made available to the public, so everyone can download and train the model a little bit more, for a specific NLP task. This is called fine-tuning.

Starting from the exact same pre-trained BERT model, using a relatively small dataset and relatively few resources, we can create our own model for any given downstream task. All we have to do is remove the last layer of BERT and add our own last layer depending on our task. Then, starting with the original pre-trained weights (and random ones for the new last layer) we train BERT on our specific task. This is fine-tuning and it's a step of transfer learning. This works because the pre-trained BERT already 'understands' the language it was pre-trained on and does not need to start from scratch, the model already 'packs' a lot of knowledge that can help with every downstream NLP task in that language.

2.1.2 Architecture

BERT's model architecture is a multi-layer bidirectional Transformer encoder based on the original implementation [21]. BERT base is the model we will be describing and using in this thesis. There is also a BERT large, a larger model with more layers. BERT base has 12 staked encoders, each one with 768 hidden size and 12 attention heads.

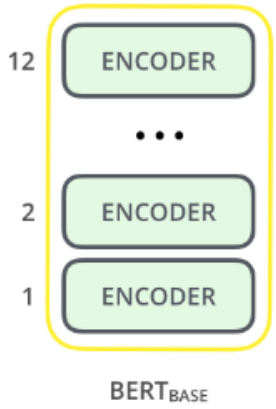

Figure 2.1: BERT architecture overview

2.1.3 Embeddings

BERT's vocabulary uses WordPiece embeddings [25] with a 30,000 token vocabulary. This helps BERT represent every word in existence, by combining learned tokens. Every input must start with the [CLS] token, which is used for classification tasks. When there are two sentences, we separate them with the [SEP] token. Those don't have to be real grammatically correct sentences, simple chunks of text are enough. For example, in the question answering task, we separate the question from the text using the [SEP] token. There is also the [PAD] token, which is used to pad smaller sentences to the maximum model input size, and the [MASK] token which is usually used for pre-training.

The word embeddings are calculated as shown in the following image.

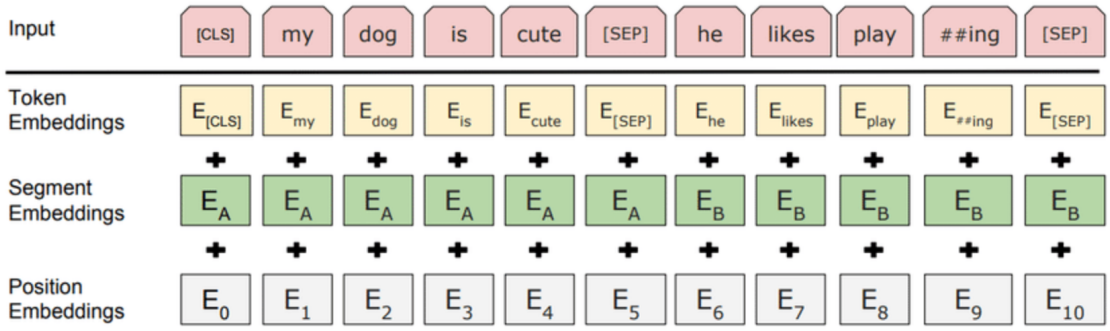

Figure 2.2: BERT embeddings

A token embedding is the learned embedding corresponding to each token in the sentence. Token embeddings are meant to represent the meaning of the token. A segment embedding is 0 for the first sentence, the one before the [SEP] token, and 1 for the second one. Segment embeddings are meant to help the model distinguish the tokens belonging to the first and the second virtual sentence. A position embedding is a fixed embedding, depending on the position of the token in the sentence. For example, the first position embedding will be the same for every sentence. These embeddings are the same as the ones introduced in the original transformer paper. They exist to give the model a way to understand how far a word is from another, since attention is calculated for all words, simultaneously. For example, if these embeddings did not exist, the model would not be able to tell the difference between the first and the second 'I' in the sentence 'I think, therefore I am'. All these embeddings are added together to produce the input of BERT.

2.1.4 Pre-training

Even thought other language models like ELMo [15] and GPT [17][18] are pre-trained from left-to-right or right-to-left, BERT is trained with a totally different method. A deep bidirectional model is strictly more powerful than either a left-to-right model or the shallow concatenation of a left-to-right and a right-to-left model. Unfortunately, standard conditional language models can only be trained left-to-right or right-to-left, since bidirectional conditioning would allow each token to indirectly "see itself". In order to train a deep bidirectional representation, BERT uses two new tasks for pre-training, Masked Language Modeling (MLM) and Next Sentence Prediction (NSP).

Masked Language Modeling [20] is the task where we mask some tokens in a sentence and ask our model to predict these tokens. For example, in the sentence 'John went to

the supermarket to buy some milk' we can mask the word-token 'supermarket'. Then, if we give as input to our model the sentence 'John went to the [MASK] to buy some milk', we expect it to predict the word 'supermarket' with a high probability. In our case we pass the outputs of BERT, which are the final hidden vectors, through a softmax over the vocabulary. In the original BERT, 15% of the tokens were masked in each sentence at random. A downside of this procedure is that the [MASK] token does not appear in the downstream task. To mitigate this, during pre-training we do not always replace masked words with the actual [MASK] token. The training data generator chooses 15% of the token positions at random for prediction. If a token is chosen, we replace this token with the [MASK] token 80% of the time, a random token 10% of the time, or we leave the token unchanged 10% of the time.

Next Sentence Prediction is the task where we give two virtual sentences, A and B, to the model and ask it to predict whether or not the second sentence can logically follow the first one. In our case, 50% of the time B actually follows A and 50% of the time B is random sentence from the corpus. During this classification task we only use the [CLS] token output of BERT.

We can understand that both these tasks are unsupervised, since they can be done on a corpus without the need for human labeling.

2.1.5 Training data

The original BERT model was pre-trained on BooksCorpus (800M words) [27] and English Wikipedia (2,500M words). From Wikipedia, only text passages were used, while lists, tables, headers were ignored.

2.2 Problem definition - RAPTARCHIS dataset

The "Permanent Greek Legislation Code - Raptarchis3" contains Greek legislation until 2015, since the creation of the Greek state in 1834. It includes laws, decrees, regulations and decisions with their respective amendments such as replacements, modifications and deletions, while its only source of information is the Official Government Gazette. It consists of 47 legislative volumes and each volume corresponds to a main thematic topic. Inside each volume is divided into thematic subcategories which are called chapters and subsequently, each chapter breaks down to subjects which contain the legal resources. The total number of chapters is 389 while the total number of subjects is 2285. Each legal document-resource is a json file whose structure is best explained by figure 2.4. It contains several fields, but for our purposes we only use the fields chapter, volume, subject, header and articles. For the input, we concatenate the header and all the articles into one large sentence. As for the label, we use the fields chapter, volume or subject, corresponding to the classification task at hand.

The dataset contains three types of legal documents-resources for each thematic level, based on their label frequency in the training and testing set: Frequent, few-shot and zero-shot. Frequent classes occur more than 10 times in training documents. Few-shot classes appear in 1 to 10 training documents. Zero-shot classes appear in the development and/or test set, but not in the training set.

The following tables and images are taken directly from the paper introducing RAPT-

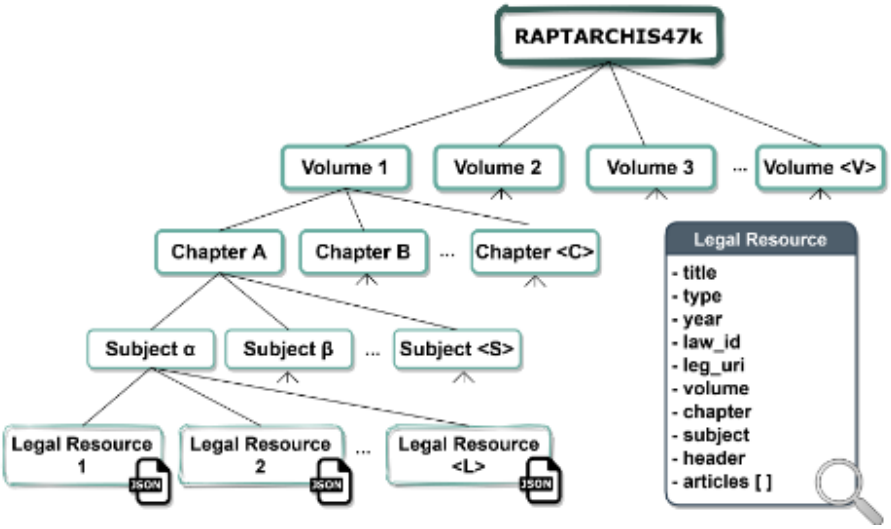

Figure 2.3: RAPTARCHIS dataset structure


```
{
  "title": "17. ΠΡΟΕΔΡΙΚΟ ΔΙΑΤΑΓΜΑ υπ΄αριθ. 234",
  "type": "ΠΡΟΕΔΡΙΚΟ ΔΙΑΤΑΓΜΑ",
  "year": "1996",
  "law_id": "234",
  "leg_uri": "http://legislation.di.uoa.gr/eli/pd/1996/234",
  "volume": "ΒΙΟΜΗΧΑΝΙΚΗ ΝΟΜΟΘΕΣΙΑ",
  "chapter": "ΔΙΑΦΟΡΕΣ ΒΙΟΜΗΧΑΝΙΕΣ",
  "subject": "ΔΗΜΟΣΙΑ ΕΠΙΧΕΙΡΗΣΗ ΠΕΤΡΕΛΑΙΟΥ",
  "header": "Αύξηση κατά δεκαπέντε [...] Α.Ε.».",
  "articles": [
    "\nΕγκρίνεται η από 22 [...] δραχμών η κάθε μία.",
    "\nΤο Διάταγμα αυτό ισχύει από την [...] διατάγματος."
  ]
}
```


Figure 2.4: RAPTARCHIS dataset legal document-resource example

ARCHIS [3] and will help us understand the dataset's composition. As shown in Table 2.2 and 2.3, many classes are under-represented, especially in the subjects level, causing the appearance of few-shot and zero-shot categories. This makes the subjects problem more difficult than the volume and subject one, not only because it has more classes in general, but also because it has more zero-shot classes. Figure 2.5 shows the token distribution of the documents, in other words how many tokens-words each document contains.

Table 2.1: RAPTARCHIS characteristics (1)

Subset	Docss	Mean # of tokens per doc	Short docs (<100 tokens)
Train (60%)	28536	600	15412 (54%)
Dev (20%)	9511	574	5175 (54,4%)
Test (20%)	9516	595	5075 (53,3%)
Total	47563	594	25662 (54%)

Table 2.2: RAPTARCHIS characteristics (2)

	Total classes	Frequent	Few-shot (<10 occ.)	Zero-shot
Volume	47	47 (100%)	0	0
Chapter	389	333 (85,6%)	53 (13,6%)	3 (0,7%)
Subject	2285	712 (31,2%)	1431 (62,6%)	142 (6,2%)

Table 2.3: RAPTARCHIS characteristics (3)

	Total docs	Frequent	Few-shot (<10 occ.)	Zero-shot
Volume	47563	47563 (100%)	0	0
Chapter	47563	47108 (99%)	445 (0,9%)	10 (<0,1%)
Subject	47563	38475 (80,9%)	8870 (18,6%)	218 (0,5%)

To summarize, our goal is to fine-tune BERT for three classification tasks: volume (47 classes), chapter (389 classes) and subject (2285 classes), using the RAPTARCHIS Greek legal dataset. The more the classes are, the more difficult the problem becomes.

2.3 Other BERT models

Our corpus is in Greek, so we chose three BERT models that can 'understand' Greek. These models are Greek-BERT, Greek-Legal-BERT and Multilingual-BERT. All of these have an uncased vocabulary.

2.3.1 Greek-BERT

Greek-BERT is a model that has the same original BERT architecture, trained on the Greek language. It was trained on 29GB of Greek corpus. Specifically, the Greek part of Wikipedia (0,73GB, 0.08 billion tokens), the Greek part of the European Parliament Proceedings Parallel Corpus (Europarl) (0,38GB, 0,04 billion tokens), and the Greek part of OSCAR [14], a clean version of Common Crawl (27GB, 2,92 billion tokens). It has a 35000 tokens vocabulary.

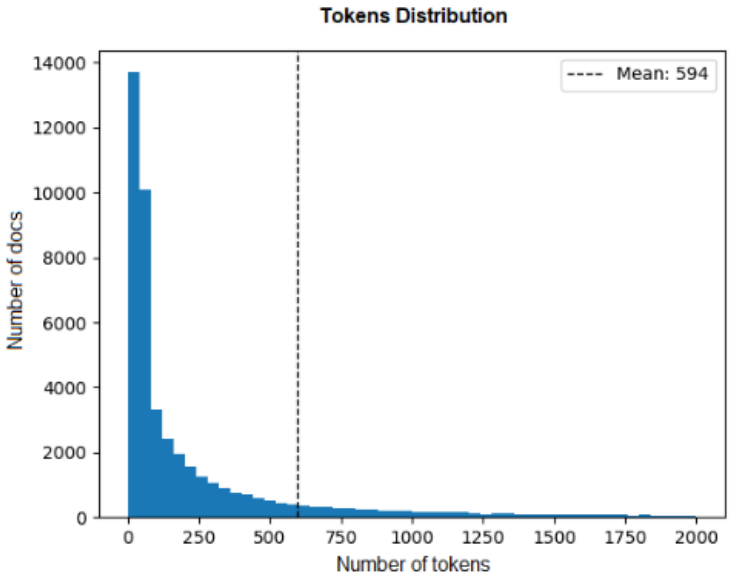

Figure 2.5: Dataset's token distribution over docs

2.3.2 Greek-Legal-BERT

Greek-Legal-BERT is a model following the same original BERT architecture, trained on the Greek legal language. It was trained on 4.5 GB of Greek legal corpus. Specifically the entire corpus of Greek Legislation available in Nomothesi@ platform [2]. This corpus consists of numerous laws, announcements and resolutions in the Greek Language. It has a 35100 tokens vocabulary. Note that of the three models we use, Greek-Legal-BERT has the smallest pre-training dataset.

2.3.3 M-BERT

M-BERT is a model that has the same original BERT architecture, trained on the Wikipedia pages of the top 100 languages with the largest Wikipedias. It was trained on 0,35GB of Greek Wikipedia corpus and an extremely large Wikipedia mixed corpus. It has a 110000 mixed tokens vocabulary. Of those only around 1200 are for the Greek language.

3. DOMAIN SPECIFIC ADAPTATIONS

Our dataset belongs to the legal domain and has legal domain specific language. This can make it difficult for BERT to understand our documents. Instead of just fine-tuning Greek-BERT, we could also try to expand-alter its vocabulary. This is most likely not needed for Greek-Legal-BERT since it was originally pre-trained on the legal domain.

3.1 Background and Related work

There are not many papers that try to adapt BERT's vocabulary to be more domain specific. The main are BioBERT v1.0 [10] and Don't Stop pre-training [6] which basically follow the same technique. Most authors chose to pre-train a BERT model from scratch with a new domain specific vocabulary, like Greek-Legal-BERT. There are, however, papers that try to modify multilingual BERT's vocabulary to create a version of BERT more suitable for certain low resource languages, and report good results. We can assume that for a multilingual BERT, a specific low recourse language's vocabulary is considered domain-specific vocabulary. Under this assumption, we can use the same techniques to modify Greek-BERT's vocabulary to the legal domain.

In all of the papers that try to alter the original BERT vocabulary to better suit a specific domain (except exBERT's [19]), the original vocabulary size remains the same. The new words are added with the following two ways:

- Unused tokens: BERT's vocabulary has 1000 [UNUSEDxxx] tokens that are meant to help us expand it without getting into the trouble of changing its size, as that would also mean changing the model's weight dimensions. Greek-BERT only has 59 of those tokens.
- Removing 'unpopular' tokens: BERT's vocabulary usually has words that do not appear in some domains. In our case, we can run the Greek-BERT tokenizer over the whole dataset and flag the tokens that are not used even once. Those tokens exist for sure. For instance, with a quick revision of Greek-BERT vocabulary, it contains a lot of English words and abbreviations that are certainly absent from Greek legal documents.

The newly added tokens won't be recognized by BERT, as they won't have trained embeddings. There are different techniques that appear in the literature, for initializing and training those embeddings. I will present an overview of some of those below.

3.1.1 German-Legal-BERT

In the master thesis named "Effects of Inserting Domain Vocabulary and Fine-tuning BERT for German Legal Language" [26], they try to do the same task with us, but for a German legal dataset. They created their own slightly larger, than the original, German-Legal vocabulary. They expected results similar to SciBERT [1] +0.6% on F1 score but found no significant improvements when comparing the German-BERT with the expanded vocabulary one. They suggest there is no improvement because "*the original vocabulary already includes 90% of the legal vocabulary. This is one of the characteristics that renders the*

legal domain especially hard to grasp for both humans and machines, many of the day-to-day words are imbued with a different meaning when used in the legal context. SciBERT, the pre-trained BERT model for scientific texts, improves around 0.60% on F1 across all the datasets but their original-scientific vocabulary overlap is just 47%.". We believe this is not the case with the Greek language as many of the laws are outdated and written before 1977.

3.1.2 Patch-BERT

Patch-BERT [12] introduces several ways to expand or alter BERT's vocabulary. The best results reported are +<1.2%. The methods described in the paper are ways to introduce new subwords in the vocabulary. I will briefly describe some of the methods presented in the paper:

- Add a new subword X. Find the closest subword Y already in BERT's vocabulary, using character distance. Subword X and Y share weights during fine-tuning.
- Find a subword X that is not used in the downstream task and replace it with a new one Y. Y's subword weights are initialized with X's, but are free to change during fine-tuning.
- Add a new subword and initialize its weights randomly.
- Lets say we want to add subword X. Find a similar subword, to the new subword, using BERT's mask word prediction. The new subword can share weights or have the same initial weights.

We will not get into details for which methods were used to expand the vocabulary and which to alter it, as all could be used for both. These methods could also be used for adding-modifying whole words and not just subwords.

3.1.3 exBERT

ExBERT is a modified version of BERT to support an extension of plus 18k tokens to the original vocabulary. In contrast to SciBERT, it isn't just pre-trained from scratch with a new, bigger, domain specific vocabulary. The goal is to change the vocabulary size without changing the model's weight dimensions, as this would mean that the entire model will have to be pre-trained from scratch.

To achieve that, exBERT has an 'expansion module' to the vocabulary-to-embedding layer and each hidden layer. The 'expansion module', added to each hidden BERT layer, uses the same transformer-based architecture as BERT, with much smaller sizes. The 'vocabulary-to-embedding expansion' is exactly the same as the original vocabulary-to-embedding layer, with the sole purpose of supporting the new vocabulary tokens. For a better understanding of the architecture see the images below.

During pre-training all original BERT weights are frozen and just the 'expansion modules' are trained. ExBERT has a +1-3% F1 score, on the Named Entity Recognition task, against BioBERT. While exBERT has more parameters (153M compared to 110M of BioBERT), when comparing its performance for the same time of pre-training, exBERT outperforms BioBERT (note that in the same time of pre-training, for example 24hrs, exBERT

goes through less data when compared to BioBERT, 24% vs 34%, but it still manages to outperform it).

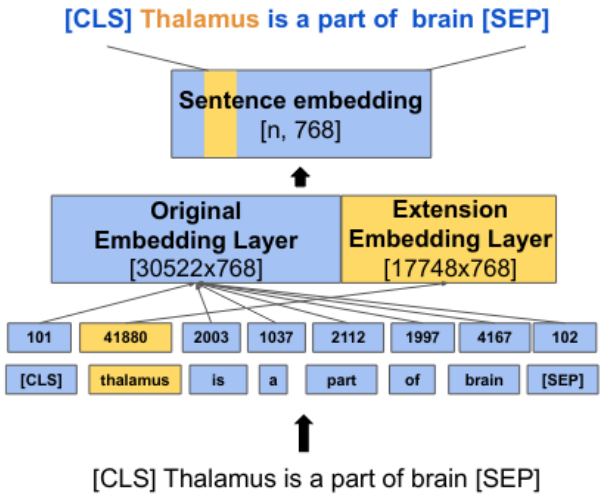

Figure 3.1: Derivation of the sentence embeddings based on both the original and extension vocabulary.

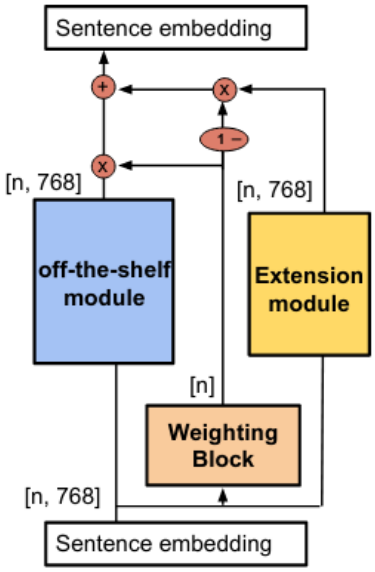

Figure 3.2: Each input sentence consists of n 768-dimensional embedding vectors, where n is 128. The output embedding is a component-wise weighted (computed by the weighted block) sum of outputs from the two modules.

3.1.4 Domain and task adaptive pre-training

Domain and task adaptive pre-training is a new technique for adapting a language model better to a specific domain or task. The technique is simple, just continue pre-training your model on a more domain specific corpus (DAPT). If such corpus is not available, continue the pre-training on the task specific dataset the model will be fine-tuned on (TAPT). Continuing pre-training means that the model will perform the NSP and MLM tasks on the chosen corpus. The domain or task specific datasets will most likely be smaller than the general purpose dataset the model was originally trained on. This technique intuitively

works because pre-training on a domain specific corpus will nudge the model's weights in the right direction. Another way to think of it, for the task adaptive pre-training, is that the model will perform more epochs on the dataset we want to fine-tune it on, but it will not overfit because the pre-training tasks are different than the downstream task. The best results come from performing TAPT after DAPT.

4. FINE-TUNING ON RAPTARCHIS

To evaluate our fine-tuned models we follow the RAPTARCHIS paper, in order to have a measure of comparison. The following scores are reported: Recall (R), Precision (P), F1 score.

$$R = \frac{True\ positive}{True\ positive\ +\ False\ negative}$$
$$P = \frac{True\ positive}{True\ positive\ +\ False\ Positive}$$
$$F1 = 2 * \frac{Precision\ +\ Recall}{Precision\ *\ Recall}$$

We report scores separately for all the labels, frequent and few-shot.

4.1 BERT fine-tuning parameters

For the fine-tuning parameters we mostly followed the Greek-BERT suggestions, which in turn follow the original BERT's paper suggestions.

For the model architecture we also followed the original BERT's paper suggestions for fine-tuning. We just replaced the last layer with a classification layer that takes as input the CLS token from the last layer's embedding.

For the model input we concatenated each legal resource's header and articles. If the max length of 512 was reached, we truncated to 512. Otherwise if the length was less than the maximum sentence length of the current batch, we padded to that maximum. Using this padding technique we saved up to 20% off the training time. There is also this smart batching technique [11] that could potentially save even more time but we chose not to implement it.

We also think there is a chance this helps our model with few-shot classes. The padding as we know has little to no effect on the results. But for the few-shot classes with less than 10 examples, every little bit helps. We may confuse a little bit our model if more than half the sentence is padding. This is also hypothesized in the last paragraph of [11] "The difference in accuracy ... is interesting ... It makes me curious to look at the attention mask implementation more–perhaps the [PAD] tokens are still having some small influence on the results."

In the following table are the implementation details.

Table 4.1: Fine-tuning implementation details

Parameter	value
Learning rate	0.00002
Batch size	8
Gradient clipping	NONE
Patience	2
Scheduler	optim.lr_scheduler.ReduceLROnPlateau(mode='min', factor=0.5, patience=1)

4.2 Training environment

We used the free GPUs offered by Google Colab. Colab offers mainly 3 types of GPUs in order of descending performance: Tesla P100, Tesla T4 and Tesla K80. The next table shows the approximate time per epoch on the RAPTARCHIS dataset per GPU. M-BERT took approximately 10 more minutes per epoch.

Table 4.2: Time per fine tuning epoch

GPU	time per epoch
Tesla P100	25 min
Tesla P4	52 min
Tesla K80	60 min

4.3 Greek-BERT results

We run our own Greek-BERT fine-tuning procedure and reported very similar results to the RAPTARCHIS paper. We used early stopping with patience equals 2. Volume, Chapter, Subject took 5, 9, 14 epochs respectively.

Table 4.3: Greek-BERT results (from RAPTARCHIS paper)

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	88,2	88,2	88,2	88,2	88,2	88,2	-	-	-
Chapter	81,4	81,4	81,4	81,4	81,8	81,6	81,3	40,2	53,8
Subject	79,3	79,3	79,3	80,8	83,4	82,1	73,3	68,7	70,9

Table 4.4: Greek-BERT results (our tests)

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	88,36	88,36	88,36	88,36	88,36	88,36	-	-	-
Chapter	83,27	83,27	83,27	83,61	83,75	83,68	52,63	98,04	68,49
Subject	79,27	79,27	79,27	84,43	86,32	85,36	62,84	84,1	71,93

4.4 Greek-Legal-BERT results

We used the exact same parameters as in Greek-BERT fine-tuning. Volume, Chapter, Subject took 4, 7, 14 epochs respectively.

It is worth noting that the scores were really high from the first few epochs. The last epochs changed them only a bit. The scores are slightly better than the the ones we got using Greek-BERT, with a 0,6% - 0,8% increase.

Table 4.5: Greek-Legal-BERT results

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	89,07	89,07	89,07	89,07	89,07	89,07	-	-	-
Chapter	83,98	83,98	83,98	84,42	84,57	84,49	44,21	95,45	60,43
Subject	80,04	80,04	80,04	84,5	86,93	85,7	66,48	82,9	73,79

4.5 M-BERT

Here we will discuss why the M-BERT model might be worth considering even thought there are Greek and Greek-Legal BERT models also available.

4.5.1 Greek with transfer learning

As shown in Greek-BERT the average difference between Greek-BERT's and M-BERT's (uncased) F1 score on the 2 easy (less language specific) tasks Part of Speech tagging (PoS), Named Entity Recognition (NER) is around 1.04%. On the difficult Natural Language Inference (NLI) (more language specific) it is around 6%. The difference in fragmentation ratio between Greek-BERT and M-BERT in all tasks is approximately 1. But this does not affect the results as much as we would expect.

In our case Greek-BERT's fragmentation ratio on the cleaned RAPTARCHIS training dataset is 1.561 and M-BERT's is 2.932. The difference is around 1.4. Our classification task is closer to difficulty to PoS and NER, so we could expect M-BERT to have similar results to Greek-BERT in RAPTARCHIS.

It is true that M-BERT has the smallest Greek pre-training dataset, but the larger overall-mixed pre-training dataset. Its dataset consists of 100 different languages. Some of those could help it 'understand' Greek better through transfer learning.

From a linguistics perspective the Greek language is part of a larger family named Indo-European, this contains Albanian, Armenian, Balto-Slavic, Celtic, Germanic, Hellenic, Indo-Iranian and Italic [23]. Those languages could, theoretically, help our model learn Greek easier. The Greek language is also part of a smaller group named Hellenic languages, but the only modern language of that group is modern Greek [22].

Hindi, Russian, Urdu, Bulgarian, Arabic help BERT learn the Greek language faster [13] [9]. The following image is taken directly from the paper, Greek is on the 4rth row annotated 'el'. The rows indicate the target language BERT tries to learn. The columns indicate the auxiliary languages used in addition to the target language, in the hope that they can help BERT 'understand' the target language better. The table shows the performance difference of accuracy scores for zero-shot learning on some dataset. The larger the difference, the more the auxiliary language helps the target.

We observe that all languages tested, except Swedish and English, help more or less the Greek language. M-BERT was trained on all of those languages. We could expect that could help the model.

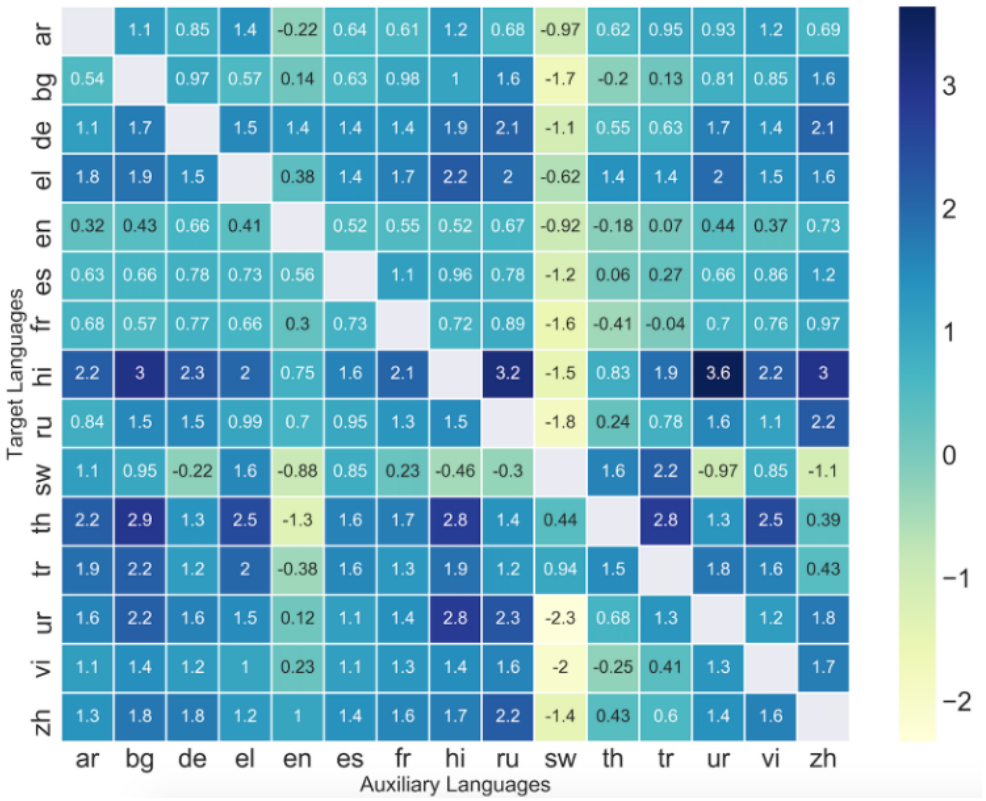

Figure 4.1: BERT zero-shot accuracy difference from crosslingual transfer learning

4.5.2 Results

We used the exact same parameters as in Greek-BERT fine tuning. Volume, Chapter, Subject took 7, 10, 16 epochs respectively. M-BERT did remarkably well, considering that it was trained on a really small Greek dataset and has only about 1/30 of the vocabulary, compared to the other tow models. Here the scores are considerably worse that the other two models, with a 0,3%-4,2% difference.

Table 4.6: M-BERT results

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	87,63	87,63	87,63	87,63	87,63	87,63	-	-	-
Chapter	80,77	80,77	80,77	81,15	81,23	81,19	46,32	95,65	62,41
Subject	75,08	75,08	75,08	83,12	84,96	84,03	46,45	71,92	56,44

4.6 Numbers free dataset - Cleaning

We noticed a lot of numeric values in the RAPTARCHIS dataset that, we hypothesize, are not helping the model predict the correct class. We trained Greek-BERT on the dataset after removing all the numbers using the following procedure. Just deleting the numeric values, this is the simplest way possible and it definitely isn't perfect. For example the sentence "Σύμφωνα με τον νόμο 1234/2006" will become "Σύμφωνα με τον νόμο /", the '/' will not be deleted. We also tried replacing all the numbers with #, so as to give the model

the information that some number was there instead of totally removing that number. The above example, then, becomes “Σύμφωνα με τον νόμο #/#”. The # symbol is not in Greek-BERT’s vocabulary so we replaced the number 5 with #. The newly introduced symbol’s embedding is the embedding of 5 (it is free to change during fine-tuning).

4.6.1 Fine-tuning results

Fine-tuning for numbers replaced with # took 6, 4, 10 epochs for Volume, Chapter, Subject respectively.

Table 4.7: Greek-BERT results with numbers replaced with #

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	88,17	88,17	88,17	88,17	88,17	88,17	-	-	-
Chapter	82,96	82,96	82,96	83,35	83,41	83,38	47,37	91,845	62,5
Subject	74,91	74,91	74,91	80,8	83,12	81,94	55,01	85,91	67,08

Fine-tuning for number free dataset took 4, 6, 9 epochs for Volume, Chapter, Subject, respectively.

Table 4.8: Greek-BERT results with numbers free dataset

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Volume	89,4	89,4	89,4	89,4	89,4	89,4	-	-	-
Chapter	81,33	81,33	81,33	82,0	82,03	82,02	17,89	94,44	30,09
Subject	77,09	77,09	77,09	81,24	84,56	82,86	64,26	89,3	74,75

In the volume and chapter classes, removing the numbers from the dataset was clearly a better solution, as we reached the same accuracy with almost half as many epochs. When considering the subject classes, none of the cleaning methods managed to reach the desired accuracy.

4.6.2 Conclusions

The most difficult classification problem is the one about the subject classes out of the three. Not only does it have a greater number of classes, but it also has a high percentage of few and zero-shot classes. Our initial assumption, that the numeric values in the dataset are not helping the model, was obviously incorrect.

There are two main probable reasons that can explain these results:

- The numbers in the dataset provide no information at all, but act as noise that helps our model generalize. This is not likely, because if it was true, the model without the numbers probably wouldn’t outperform the ‘#’ one.
- numbers in the dataset provide information about the class. For this to happen, some numbers must appear in the dataset a considerable amount of times.

To test the first hypothesis we fine-tuned a Greek-BERT model on RAPTARCHIS and replaced all the numbers in the dataset with a random number of the same length. For example the sentence “Σύμφωνα με τον νόμο 1234/2006” will become “Σύμφωνα με τον νόμο 8932/8352”. These are the results after epoch number 11 on the Subject classification problem, which was the one the other models fell short.

Table 4.9: Greek-BERT results with Numbers replaced by random numbers

	All labels			Frequent			Few-shot		
	R	P	F1	R	P	F1	R	P	F1
Subject	77,76	77,76	77,76	82,26	84,71	83,46	63,94	85,74	73,25

The model still didn’t reach the goal. That shows that the numbers must provide some information. This can only happen if some numbers are repeated a significant amount of times in the dataset. After testing we find that a lot of numbers (around 350) are repeated more than 100 times in the RAPTARCHIS dataset. Some of those numbers are, 1948, 1949, 195, 1950, 1951, 1952, 1953, 1954, 1955, 1956, 000, 01, 02, 08, 09, 10, 100, 115, 116, 117, 118, 119, 12, 120, 121. We conclude that the truth must be somewhere in between, some numbers provide meaningfull information and others act as noise.

4.7 Conclusions

The Greek-Legal-BERT model yields the best results, based on the F1 score, in almost all categories and classes, except the few-shot chapters. Second comes Greek-BERT with slightly worse scores. M-BERT has better scores than expected but still comes last.

Table 4.10: All fine-tuning F1 scores

		Greek-BERT	Greek-BERT (our tests)	Greek-Legal-BERT	M-BERT
All labels	Volume	88,2	88,36	**89,07**	87,63
	Chapter	81,4	83,27	**83,98**	80,77
	Subject	79,3	79,27	**80,04**	75,08
Frequent	Volume	88,2	88,36	**89,07**	87,63
	Chapter	81,6	83,68	**84,49**	81,19
	Subject	82,1	85,36	**85,7**	84,03
Few-shot	Volume	-	-	-	-
	Chapter	53,8	**68,49**	60,43	62,41
	Subject	70,9	71,93	**73,79**	56,44

5. DOMAIN AND TASK ADAPTIVE HEURISTICS

TAPT and DAPT are more expensive relatively to just fine-tuning. The more the domain and task specific the vocabulary and language are, the more gains we expect. We calculated some heuristics to evaluate if it is worth performing TAPT and DAPT for the RAPTARCHIS dataset and generally for the Greek legal domain.

5.1 RAPTARCHIS dataset exploration

Up until 1977 all Greek legal documents had to be written in Katharevousa. Katharevousa is very different than Demotics-Modern Greek. Greek-BERT is trained on Demotics. Greek-Legal-BERT's dataset, Nomothesi@, containers laws written after 1990. Raptarchis on the other hand contains 27807 (58.46%) laws before 1977 and only 19756 (41.54%) laws after 1977. That makes our dataset difficult, as it is mostly written on a different-older Greek dialect.

5.2 BERT vocabulary exploration

In the following table are the number of english words and numbers in Greek and Legal BERT vocabulary.

Table 5.1: BERT vocabulary composition

	Total words	Unused	English words	Numbers
Greek-BERT	35000	100(0,35%)	3847 (11,02%)	627 (1,79%)
Legal-BERT	35100	100(0,35%)	3000 (8,57%)	3363 (9.6%)

Greek-BERT has more English words in it's vocabulary. Legal-BERT's vocabulary has more English words than expected and contains more than 5 times more numbers.

5.3 BERT Vocabulary overlap

We calculated the overlap between the vocabulary of Greek-BERT, Greek-Legal-BERT and M-BERT. This is calculated based on the following formula:

$$overlap = \frac{|vocab1 \cup vocab2|}{|vocab1 \cap vocab2|}$$

In figures 5.1 and 5.2 we include the results, considering the whole M-BERT's vocabulary on the first case, and only the Greek words from M-BERT's vocabulary on the second case. The overlap between M-BERT and the other models is larger when we include all the words because there are a lot of numbers in M-BERT's vocabulary.

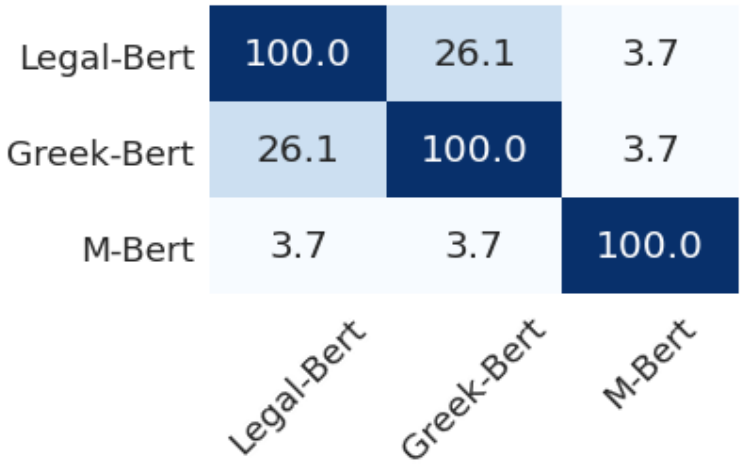

Figure 5.1: BERT vocabulary overlap (all)

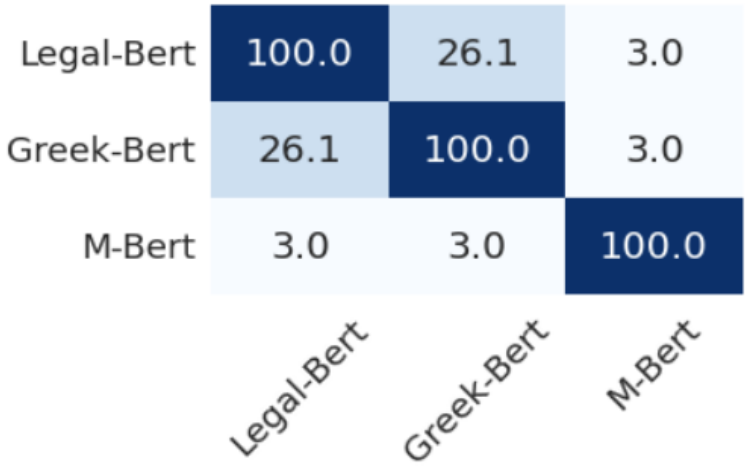

Figure 5.2: BERT vocabulary overlap (Greek words only)

5.4 Fragmentation ratio

The word fragmentation ratio is defined as the average number of sub-word tokens per word. In other words, we ran each BERT tokenizer on each dataset and measured in average, in how many subwords each BERT model breaks each word down. The results are shown in table 5.2.

Table 5.2: Fragmentation ratio

	RAPTARCHIS	Nomothesia	Europarl + OSCAR + Greek wiki
Greek-BERT	1,646	1,154	1,151
Legal-BERT	1,513	1,06	1,277
M-BERT	2,968	2,059	1,277

Obviously, Greek-Legal-BERT has the lowest fragmentation ratio on Nomothesia, the dataset it was trained on. The same is true for Greek-BERT and Europarl + OSCAR + Greek wiki. M-BERT has the worst fragmentation ratios in all three datasets since it has the smallest vocabulary.

5.5 Pre-training dataset overlap

In Gururangan et al [6] they use pre-training dataset vocabulary overlap to measure the similarity of pre-training datasets. The less similar the datasets are, the more a model will benefit from TAPT and DAPT. This is calculated by taking the top 10k most frequent words from each dataset and just like the vocabulary overlap, calculate the formula for overlap: $overlap = \frac{|vocab1 \cup vocab2|}{|vocab1 \cap vocab2|}$. In their paper they took a sample of around 50k documents from each dataset. We calculated this metric based on all the whole datasets (this takes up to 70GB of RAM).

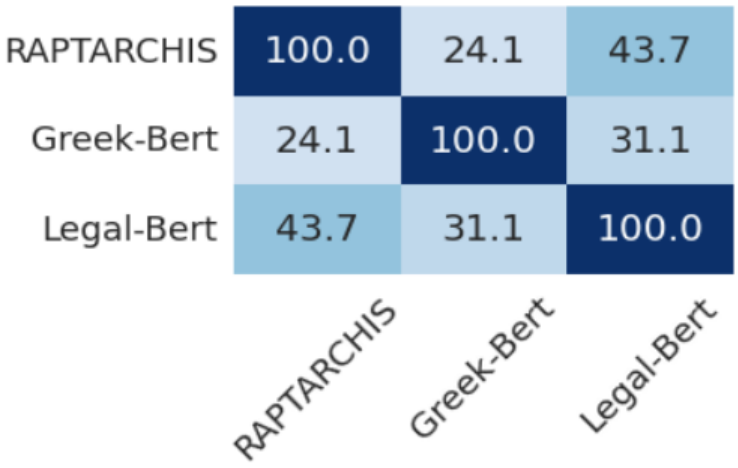

Figure 5.3: Pre-training dataset overlap between Greek-Legal-BERT, Greek-BERT and RAPTARCHIS

RAPTARCHIS has a greater overlap with Greek-Legal-BERT's pretraining dataset than Greek-BERT's, as expected. Although we don't have M-BERT's exact pretraining dataset, it certainly has the smallest overlap with RAPTARCHIS. All in all, M-BERT is expected to have greater gains from DAPT, followed by Greek-BERT.

6. CONCLUSIONS AND FUTURE WORK

At the beginning of this thesis, we explored the architecture of the BERT model and vocabulary. We also explored some Greek legal datasets. We fine-tuned three BERT models on the RAPTARCHIS dataset, on three classification tasks each. The multilingual model did considerably worse, but still exceeded our expectations, while the Greek-Legal model was undoubtedly the best. We also tried to find if the numbers in our dataset are useful. Finally, we compared techniques for making BERT's score better on a task with a more domain specific vocabulary and calculated some heuristics that can calculate the domain specificity of the corpus. We concluded that DAPT and TAPT are worth considering for the Greek-Legal domain. We hypothesize, based on those heuristics, that maybe with some vocabulary adjustments, like those introduced in Sangwhan et al [12], after DAPT and TAPT, the multilingual model could surpass both the Greek and Greek-Legal models.

For the future, we plan to apply TAPT and DAPT to all three models and also alter M-BERT's vocabulary to test our hypothesis.

ABBREVIATIONS - ACRONYMS

AI	Artificial Intelligence
NLP	Natural Language Processing
BERT	Bidirectional Encoder Representations from Transformers
TAPT	Task Adaptive pre-training
DAPT	Domain Adaptive pre-training
DNN	Deep Neural Networks
P	Precision
R	Recall
NER	Named Entity Recognition
PoS	Part of Speech tagging
NLI	Natural Language Inference

BIBLIOGRAPHY

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3613–3618. Association for Computational Linguistics, 2019.

[2] Ilias Chalkidis, Charalampos Nikolaou, Panagiotis Soursos, and Manolis Koubarakis. Modeling and querying greek legislation using semantic web technologies. In Eva Blomqvist, Diana Maynard, Aldo Gangemi, Rinke Hoekstra, Pascal Hitzler, and Olaf Hartig, editors, *The Semantic Web*, pages 591–606, Cham, 2017. Springer International Publishing.

[3] Manolis Koubarakis Christos Papaloukas, Ilias Chalkidis. Legal text classification for greek legislation. manuscript in preparation.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Slav Petrov Kristina Toutanova. Multilingual bert. `https://github.com/google-research/bert/blob/master/multilingual.md`, 2019.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[6] Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8342–8360. Association for Computational Linguistics, 2020.

[7] Athinaios Konstaninos. Named entity recognition using a novel linguistic model for greek legal corpora based on bert model, 2020. Dept. Informatics and Telecommunication, National and Kapodistrian University of Athens.

[8] John Koutsikakis, Ilias Chalkidis, Prodromos Malakasiotis, and Ion Androutsopoulos. GREEK-BERT: the greeks visiting sesame street. In Constantine D. Spyropoulos, Iraklis Varlamis, Ion Androutsopoulos, and Prodromos Malakasiotis, editors, *SETN 2020: 11th Hellenic Conference on Artificial Intelligence, Athens, Greece, September 2-4, 2020*, pages 110–117. ACM, 2020.

[9] Anne Lauscher, Vinit Ravishankar, Ivan Vulic, and Goran Glavas. From zero to hero: On the limitations of zero-shot cross-lingual transfer with multilingual transformers. *CoRR*, abs/2005.00633, 2020.

[10] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, abs/1901.08746, 2019.

[11] Chris McCormick. Smart batching tutorial - speed up bert training, Jul 2020. `https://mccormickml.com/2020/07/29/smart-batching-tutorial`.

[12] Sangwhan Moon and Naoaki Okazaki. Patchbert: Just-in-time, out-of-vocabulary patching. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7846–7852. Association for Computational Linguistics, 2020.

[13] Farhad Nooralahzadeh, Giannis Bekoulis, Johannes Bjerva, and Isabelle Augenstein. Zero-shot cross-lingual transfer with meta learning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4547–4562. Association for Computational Linguistics, 2020.

[14] Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online, July 2020. Association for Computational Linguistics.

[15] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018.

[16] Printzou. Current strategic trends and perspectives in public administration (central & local government) in greece and abroad, 2018. Dept. Geology and Geoenvironment, National and Kapodistrian University of Athens. Page 20.

[17] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[18] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2018.

[19] Wen Tai, H. T. Kung, Xin Dong, Marcus Z. Comiter, and Chang-Fu Kuo. exbert: Extending pre-trained models with domain-specific vocabulary under constrained training resources. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1433–1439. Association for Computational Linguistics, 2020.

[20] Wilson L. Taylor. "cloze procedure": A new tool for measuring readability. *Journalism Quarterly*, 30(4):415–433, 1953.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[22] Wikipedia. Hellenic languages — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Hellenic%20languages&oldid=1033038922`, 2021. [Online; accessed 30-July-2021].

[23] Wikipedia. Indo-European languages — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Indo-European%20languages&oldid=1036219751`, 2021. [Online; accessed 30-July-2021].

[24] Wikipedia. Language model — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Language%20model&oldid=1035600729`, 2021. [Online; accessed 30-July-2021].

[25] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[26] Chin Man Yeung. Effects of inserting domain vocabulary and fine-tuning bert for german legal language, November 2019.

[27] Yukun Zhu, Ryan Kiros, Richard S. 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 2015, Santiago, Chile, December 7-13, 2015*, pages 19–27. IEEE Computer Society, 2015.