

SHARED CONTEXT THROUGH MULTI-LEVEL ATTENTION TRANSFORMERS  
FOR TEXT CLASSIFICATION

A Thesis  
IN  
Computer Science

Presented to the Faculty of the University  
of Missouri–Kansas City in partial fulfillment of  
the requirements for the degree

MASTER OF SCIENCE

by  
CHARAN TEJ THOTA  
University of Missouri

Kansas City, Missouri  
2021

© 2021

CHARAN TEJ THOTA  
ALL RIGHTS RESERVED

SHARED CONTEXT THROUGH MULTI-LEVEL ATTENTION TRANSFORMERS  
FOR TEXT CLASSIFICATION

Charan Tej Thota, Candidate for the Master of Science Degree  
University of Missouri–Kansas City, 2021

ABSTRACT

Natural language processing (NLP) has seen recent explosive growth by creating artificial intelligence with human-level intelligence. Understanding the context using an attention mechanism could be further improved by fine-tuning their composition for classification, question answering, and topic modeling. Real-world datasets are much more complex and tend to require multi-fold models. Such models tend to be larger, deeper, more complicated; for example, BERT has 340 million parameters, Turing NLG is 17 billion parameters, and GPT-3 is about 175 billion parameters. Understanding their implications requires the immense computational ability to process the text corpus during both training and inferences.

This thesis proposes a novel deep learning architecture for scalable multi-fold text classification that is an extension of BERT by sharing context across abstraction levels of domains. Four types of deep learning models (BERT flat, BERT hierarchical, BERT

hierarchical tuned, BERT Feature extracted) are proposed for the multi-label attention transformers on the architecture. The proposed models provide a means to overcome computing limitations, training concurrently, and providing predictions for an extra level of classes simultaneously. Our work overcomes the limitations of knowledge distillation or transfer-learning, i.e., it is not scalable or sustainable, and it's also costly. We have performed experiments to validate the reliability model using both benchmark and real-world data (KCMO 311 data). Quantitative results confirm that the proposed models can enhance model performance in terms of computational requirements and provide competitive accuracy.

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Computing and Engineering, have examined a thesis titled “Shared Context through Multi-Level Attention Transformers for Text Classification,” presented by Charan Tej Thota, candidate for the Master of Science degree, and hereby certify that in their opinion it is worthy of acceptance.

Supervisory Committee

**Yugyung Lee, Ph.D.**, Committee Chair

Department Of Computer Science and Electrical Engineering,  
School of Computing and Engineering

**Ye Wang, Ph.D.**

College of Arts and Sciences,  
School of Journalism

**Brent Never, Ph.D.**

Public Affairs,  
Bloch School of Management

## CONTENTS

ABSTRACT . . . . .	iii
ILLUSTRATIONS . . . . .	viii
TABLES . . . . .	ix
ACKNOWLEDGEMENTS . . . . .	x
Chapter	
1 Introduction . . . . .	1
1.1 Overview . . . . .	1
1.2 Objectives . . . . .	2
1.3 Motivation . . . . .	4
1.4 Summary of Contributions . . . . .	5
2 Related Work . . . . .	7
2.1 Overview . . . . .	7
2.2 Attention Model . . . . .	7
2.3 Bidirectional Encoder Representations from Transformers . . . . .	10
2.4 Multi-Level Classification . . . . .	11
2.5 Hierarchical Models . . . . .	13
2.6 Limitations of Models . . . . .	14
3 Multi-Level Attention Transformers . . . . .	16
3.1 Model Design . . . . .	16

3.2	Model Architecture . . . . .	20
3.3	Base BERT . . . . .	21
3.4	Flat Multiple BERT . . . . .	22
3.5	Multi-Level Hierarchical BERT . . . . .	24
3.6	Multi-Level Hierarchical BERT Tuned . . . . .	24
3.7	Multi-Level Feature-Based BERT . . . . .	26
3.8	Summary of the Proposed Models . . . . .	28
4	Evaluation and Results . . . . .	31
4.1	Dataset . . . . .	31
4.2	Experiments . . . . .	54
4.3	Training Base BERT . . . . .	56
4.4	Multiple BERT Classification . . . . .	56
4.5	Multi-Level Hierarchical BERT . . . . .	56
4.6	Multi-Level Hierarchical BERT Tuned . . . . .	57
4.7	Multi-Level Feature-Based BERT . . . . .	58
4.8	Evaluation . . . . .	58
5	Conclusion and Future Work . . . . .	77
5.1	Conclusion . . . . .	77
5.2	Future Work . . . . .	77
	REFERENCE LIST . . . . .	79
	VITA . . . . .	80

## ILLUSTRATIONS

Figure	Page
1 Attention Visualization . . . . .	9
2 Transformer Model . . . . .	9
3 High-level BERT Architecture . . . . .	12
4 High Level Architecture for Multi-Level Attention Transformers . . . . .	18
5 High Level Architecture for Attention Transformers . . . . .	22
6 Multiple BERT Model Architecture . . . . .	23
7 Multi-Level BERT Model Architecture . . . . .	25
8 Multi-Level Tuned Model Architecture - Training . . . . .	27
9 Multi-Level Tuned Model Architecture - Inference . . . . .	27
10 Multi-Level BERT Feature Model - Training . . . . .	29
11 Multi-Level BERT Feature Model - Testing . . . . .	29
12 311 Kansas City Data analysis for Level 1 . . . . .	51
13 311 Kansas City Data Analysis for Level 2 . . . . .	52
14 Multiple BERT Accuracy vs. Epochs - Level 1 and Level 2 . . . . .	61
15 Multi-Level BERT Accuracy vs. Epochs - Level 1 and Level 2 . . . . .	61
16 Multi-Level Tuned BERT Accuracy vs. Epochs - Level 1 and Level 2 . . . . .	62
17 Multi-Level Feature Extracted BERT Accuracy vs. Epochs - Level 1 and Level 2 . . . . .	62



## TABLES

Tables		Page
1	Level 1 Classes for Data sets . . . . .	32
2	Level 2 Class Count - DBPedia and 311 Kansas City Dataset . . . . .	33
3	311 Kansas City Dataset Sample . . . . .	41
4	311 Kansas City Dataset Statistics . . . . .	45
5	311 Kansas City Dataset Level 1 and Level 2 Classes Mapping . . . . .	48
6	DBPedia Level 1 and Level 2 classes . . . . .	53
7	Sample of DBPedia Dataset . . . . .	55
8	Computational Requirements - 311 Kansas City Dataset . . . . .	59
9	Computational Requirements - DBPedia Dataset . . . . .	59
10	Accuracy Metrics - 311 Kansas City Dataset . . . . .	60
11	Accuracy Metrics - DBPedia Dataset . . . . .	60
12	Accuracy Metrics - DBPedia Dataset Increased Epochs . . . . .	60
13	Class Level 1 Accuracy - DBPedia Dataset . . . . .	63
14	Class Level 2 Accuracy - DBPedia Dataset . . . . .	63
16	Class Level 2 Accuracy - 31 Kansas City Dataset . . . . .	66
15	Class Level 1 Accuracy - 311 Kansas City Dataset . . . . .	76

## ACKNOWLEDGEMENTS

I want to thank my research advisor Dr. Yugyung Lee for all the support and guidance offered over the last two years. Dr. Lee's guidance helped me improve my knowledge in certain aspects of research and contribute to some of the challenging real-world problems. Deep learning is a new and growing field which have unlimited potential to solve many of the problems specific to automation. It is only with amusement and thanks to Dr. Lee to bring the new world technologies as courses with intuitive course learning helping students to be ready to solve some of the complex challenges. I'm consistently amazed by Dr. Lee's research drive and the kind of attention given to details despite handling several research problems at once.

I would like to take this opportunity to thank the individuals of my thesis committee, Dr. Yugyung Lee, Dr. Brent Never, and Dr. Ye Wang. I am fortunate to have worked on community research for Kansas City. It helped me to see a new perspective and conduct the research further to solve the problem. Thanks to the University of Missouri-Kansas City for providing the kind of support it did during tough times we have seen through 2020-21. The opportunities provided by UMKC in research space and teaching space helped me master new skills, and it's one of the finest decisions I made to pursue my education with UMKC.

I would like to thank my family for all the support and guidance I have received during the tough time. I extend great thanks to all friends and mentors who were part of my journey in these two years.

## CHAPTER 1

### INTRODUCTION

#### 1.1 Overview

Recent developments in Deep learning have established itself as a near ability of human-level intelligence on processing natural language data. Every day, text data contributes about 2.5 Quintilian bytes (2.5 e+9 GB). Availability of vast data and combination with Deep learning techniques made analysis and automated processing of text data easy to impact insights mining the data.

Traditionally different tasks can be addressed with natural language processing like question-answering, classification, topic modeling, language conversion, etc. Classification is a task with natural language processing to identify the class based on a group of sentences. A simple example of classification can be understood as sentiment analysis from the given description, and sentiment class can be positive, negative. In the case of two classes, classification can be called binary classification. In the same example, if a class of neutral is more than a binary classification called multi-label classification.

Considering real-world problems, classification is sometimes more complex and multi-fold than just identifying categorizing text corpus into a specific class. Considering the same example as earlier, once we understood the sentence is negative, there can be a definitive need to understand the class further, like the negative sentiment is out of concern, anger, or other means. Applying such classification can go to deeper levels, and

we can refer to such problems as Multi-Level classification.

The current proposed state-of-art models have established accuracy closer to 100%. However, The models are heavy and complex and do not address multi-fold problems like Multi-Level classifications. Multi-Level classification problems need to be addressed by running multiple state-of-art models. Considering the current Graphics Processing Units, the ability to run one state-of-art model is getting increasingly difficult. The ability to run multiple state-of-art models to solve Multi-Level classification problems is near impossible.

We propose a set of novel architectures to address Multi-Level classification by keeping the layers in the models fixed along with reduced training times. We consider the BERT model as a state-of-art technique to run our experiments. Attention mechanism capabilities include gathering knowledge from the context provided in the input features. The context being perceived as the critical parameter within natural language processing, we aim to re-use attention layers per domain to keep the model size fixed.

## **1.2 Objectives**

Although state-of-art models are working closer to complete accuracy levels, these models are not equipped to solve Multi-Level classification problems directly. State-of-art models are primarily enormous and computation-hungry models, which are difficult to experiment on regular graphic processing units. By its very nature of being complex and multifold, Multi-Level classification requires multiple models to adopt these state-of-art models directly. Considering a case of two levels of classification, running two models

to address this problem is near impossible when it's hardly possible to run one of the state-of-art models.

MIT research [1] concluded that the general problem faced by deep learning methodologies is the physical computing environment, not the deep learning methodologies. Despite being heavy in computations, state-of-art models have architecturally moved away from handling specific tasks like question-answering but produced a generalized model that understands context within the natural language and can be fine-tuned for different tasks like classification question-answering, etc. We consider reusing such attention layers within the base model to classify features as a designated class at multiple levels. Achieving re-usability of attention layers helps us to share the context between different classification levels.

The attention mechanism is at the core of the context understanding within the natural language, but the model is heavy with computing requirements. Using other attention layers for the deeper classification levels, we end up exhausting compute resources. We present an idea of utilizing the same attention layers by doing multiple training passes. After all, the attention mechanism is best placed for context gathering from provided input features. Classification level predictions can serve as a valuable feature in addition to text embeddings to narrow down the possibility of following classification level's classes to improve its performance.

### 1.3 Motivation

Natural language processing has seen exponential and explosive growth in the research community. While this resulted in establishing high benchmarks for accuracy, the downside of this research is that the models are highly computing hungry and cannot scale themselves down to day-to-day usage. While accuracy and techniques with the natural language processing are approached is essential, there is a definitive need to reduce the memory footprint to make them available for regular usage.

State of art models such as the Base BERT (110 million parameters), and the large BERT (340 million parameters) [4], Turing NLG (17 billion parameters) [8], GPT-3 (175 billion parameters) [12] are the leading models in natural language processing which are resource hungry and computing heavy models. Attention mechanism [12] have been the core of these models to parallelize training of the NLP models over the traditional LSTM models [5]. LSTM is a sequential model that operates by using memory-based, which can be understood as the current prediction function of the previous state and output along with current input.

Primarily, the state of art models has surfaced techniques like attention required for context parsing of natural language processing. It also has heavily increased the requirements of computing heavy graphic processing units, and their implications mean an increase in the training and the processing times. Secondly, state-of-art models demand substantial computational requirements, and there are physical limitations on how much we could grow with hardware. Infrastructure limitations will also restrict the significant change in the models, like adding new layers or additional feedback loops.

Considering the limitations, a task like Multi-Level classification becomes a problem by running a model with computation requirements. Having two or more state-of-art models running concurrently by a means to establish a shared context is a near-impossible task. While consideration of sequential training of one model after another gives out a train the models within infrastructure limitations. However, this will also significantly pose problems like longer training times, in-ability of valuable shared context passing between these two models, no collaborative learning, and fitting two models in one physical infrastructure at the time of inference.

#### **1.4 Summary of Contributions**

Our research focuses on adopting state-of-art models to run problems like Multi-Level classification without requiring additional layers. Our approach is to limit the size of the model finite despite the growth of the Multi-Level classification to deeper levels. Avoiding the other number of layers has implications for the reduction of the training times.

- Application of re-usability of attention layers which hold context in the natural language sequences for a Multi-Level classification task which is complex.
- Introduction of a novel technique and architecture to ensure model size doesn't grow with the depth of the classification, making it usable within the physical limitations.
- Introduction of a novel technique and architecture to ensure training time fixed

despite the number of classification levels being addressed.

- Provision of shared context passing between multiple classification levels to help the prediction being accurate.
- A novel boundary mapping algorithm has been devised to correct the data at the time of analysis to assign boundaries correctly for the records.



## CHAPTER 2

### RELATED WORK

#### 2.1 Overview

Natural language processing has taken a giant leap forward from the inception of the attention mechanism [12]. Natural language processing has been traditionally looked at in sequential models like LSTM [5] since the context is understood between the features. Sequential models require holding the previous state to process the current token, making the training a slow process not to utilize parallelism within our computing environments. This very nature of sequential models is also the reason for the vanishing gradient, where it loses the context of earlier stages in the extended corpus.

Attention mechanism has notable implementations such as BERT [4] to have established benchmarks in the natural language processing tasks. A simple demonstration of Attention can be explained in Figure 1. Attention typically aims at understanding the Attention in the input features keeping the focus pointed at the specific word.

#### 2.2 Attention Model

Attention techniques aim at having different embeddings to understand the sentences and their contexts within the sentences. This can be demonstrated by a simple example using an input sentence as below "The animal didn't cross the street because it was too tired." In comparison, predicting the value "it" in Figure 1 [3] while considering

the whole input in the sentence. The attention mechanism abstracts the input embeddings into vectors as queries, keys, and values. The vectors could be understood as search text, content text, and query text in the traditional searching mechanism. Attention could be represented as in Figure 2 [7]

- Input embedding: This embedding aims to represent each word with a unique token and represent that into the model for attention technique to perform context parsing.
- Segment embedding: This embedding aims to provide information about the words belonging to each sentence. So traditionally, two sentences are provided as input, where the segment embeddings for all the tokens in the first sentence are represented as one. The next sentence is represented as 2.
- Positional embedding: Positional embedding helps to understand the position of the tokens within the segments. So the positions would be tokenized with values starting 1 through the number of tokens in each sentence. So the positional embedding restarts from value 1 when starting a new sentence.

In the attention model, every input is evaluated into three vectors queries, keys, and values. The vectors represented as K, Q, and V have learned weights. The model built with an attention mechanism is called a transformer.

consider the (i) input features as  $x_i$

Query vector for  $q_i = W_q x_i$  Weighted vector for the query can be represented as  $W_q$

Key representation for token i for  $k_i = W_k x_i$  Weighted vector for the query can be represented as  $W_k$

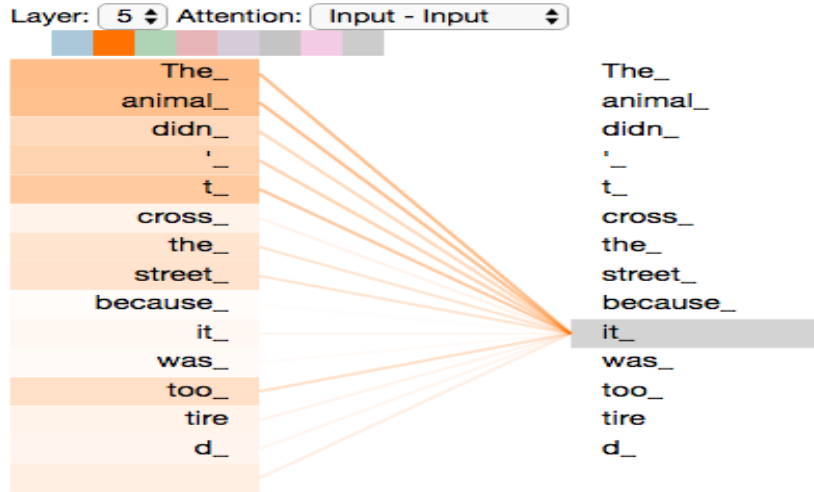


Figure 1: Attention Visualization

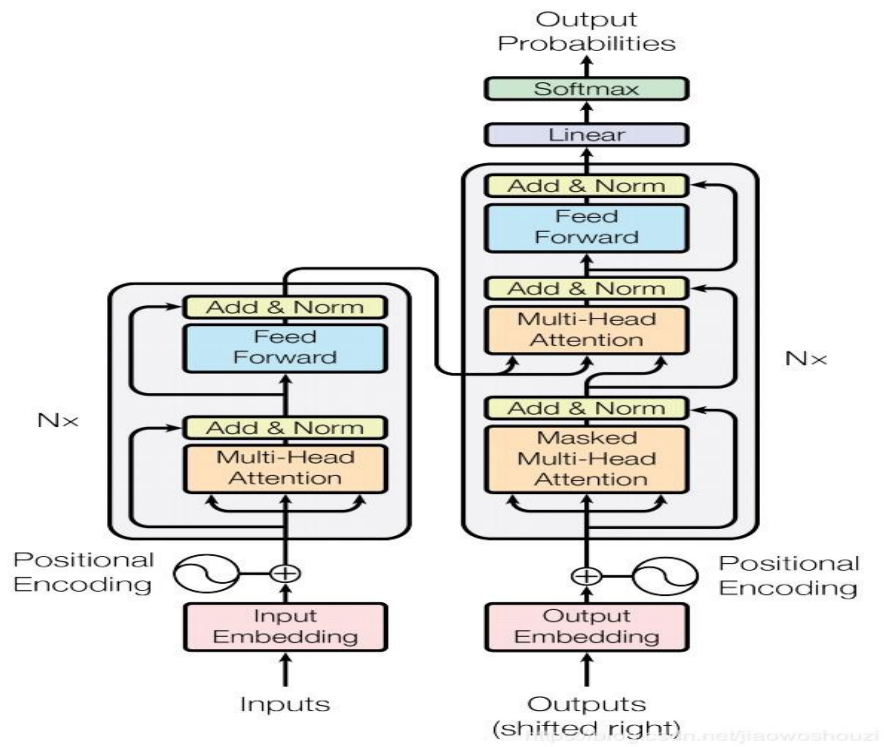


Figure 2: Transformer Model

Value representation for token  $i$  for  $v_i = W_v x_i$  Weighted vector for the query can be represented as  $W_q$

The transformer is the representative model for the attention mechanism implementation. BERT extended attention mechanism by understanding context traversing both directions in the features, i.e., left to right and right to left. BERT has implemented Base BERT with 12 x transformers and large BERT with 24 x transformers.

### **2.3 Bidirectional Encoder Representations from Transformers**

At the time of proposal, BERT is arguably the state-of-the-art model in almost all-natural language problems. BERT is also called Bidirectional Encoder representations from Transformers, relies on the attention mechanism. BERT is a group of transformers that help BERT understand the context of the natural language by looking at both directions. As an example, considering the sentence, 'I visited the bank.' If the sentence is followed by the word 'to deposit funds,' the context of the sentence helps us understand the term related to a 'financial institution,' given the scope of other words like 'blood bank' and 'river bank.'

BERT has been provided with two variants at the time of proposal, one being base BERT with 12x transformers and the other being large BERT with 24x transformers. Traversing both directions within the features helps BERT gather context and subsequently knowledge which could be fine-tuned for NLP tasks. Figure 3 [9] explains how sequence to sequence training is performed on BERT model.

The traditional problem with language model training is its aimed at specific learning tasks, something like classification or sequence-to-sequence translation. This limits the training and understanding of context. To overcome these problems, the training for BERT applies two strategies: the "Masked Language Modeling(MLM)" [4] and the "Next Sentence Prediction(NSP)" [4]. The two different training strategies applied to this model help the model to understand and review the context within provided domain knowledge.

MLM-based training involves replacing the input features to the BERT model with a unique token. It aims to replace 15% of inputs to be masked, and the same input features are identified at the end of the model. The loss functions consider the values predicted in the places of Masked token and calculates the loss with the difference between masked tokens and actual tokens. In the second training strategy, the following sentence prediction, BERT is provided with a couple of sentences which is separated by a unique token [CLS], the training data from the Wikipedia is fed to the model in a way that 50% of the data which has sentence followed by another sentence which is not adjacent sentences in the text corpus.

## **2.4 Multi-Level Classification**

Classification is a type of natural language problem that identifies the class-based text corpus, which can be comments, descriptions, or critiques. The number of classes the model classifies into it can be a binary classification or multi-label classification. Binary classification can be positive/negative. It can also be represented as 1 (representing positive) or 0 (representing negative) if another class called 'neutral' would make three

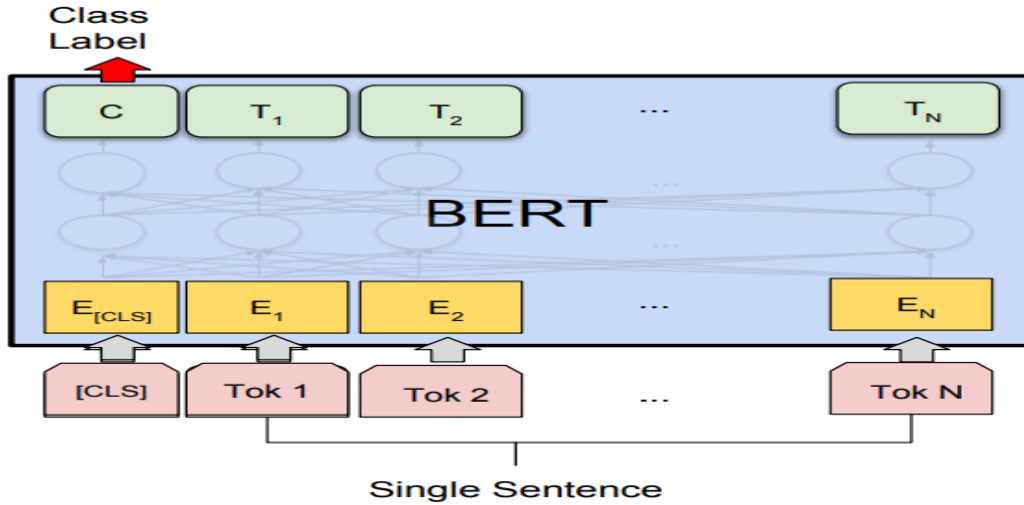


Figure 3: High-level BERT Architecture

classes and is called multi-label classification.

Most real-world problems need to go deeper into classification, and it requires us to go deeper with classification than just classifying a particular group of sentences. For example, if there is a neutral sentiment-based text category, there can be a need to identify whether it's information or a confirmation-based statement. It means the context of being classified as neutral is necessary to understand the different deeper classes as information or confirmation. In consideration of another example with multi-fold in nature, the negatively categorized statement can further be classified as the hurtful or concerned type of corpus to understand whether the negatively stated text corpus is classified as a malicious or concerned-based statement.

At the time of the model learning on the deeper classification level, the knowledge of having the previous level's classification serves as a valuable feature. Firstly, it helps

narrow down the deeper classes in consideration with the previous classification level's class. As an example from the dataset we are using for evaluation, we could observe this pattern of category and subcategory if the description read as "Openly daily burning of toxic materials at residence. Odor unbearable and usually burns between 5:30 -8:30 pm" from a customer representative, its empirical to identify the base problem and subcategory of it. Based on the description, the base problem can be categorized as "Air quality" and further deduced at the next layer as "Pollutants." As noted for reference, the first level of category supports the cause of second-level classification to add more context and meaning to the second level classification.

## **2.5 Hierarchical Models**

The hierarchical nature of deep learning models has been experimented with to solve problems about the length of the sentences and solve multitasks of NLP. However, to the best of our knowledge, there is no reference of similar research work conducted.

Hierarchical [2] nature has been evaluated to solve multiple NLP tasks at various levels of hierarchy within the same model. The paper aims at training different tasks like sentence-level and token-level functions by using various training strategies like MLM [4] and NSP [4]. The paper claims the effectiveness of different training strategies based on the type of tasks they solve.

Medical domain data with clinical notes has seen the implementation of the BERT models on hierarchical fashion [10] to solve extended text corpus. This work aims to apply classification tasks to clinical documents. The work claims about clinical records being

long in sentences will require further embedding over the traditional BERT embedding of 512 to solve the classification problem of clinical data by using various stage-based approaches. This paper, in its work, mentions the computational limits and resource constraints.

The Mobile BERT [11] is another research work that says about the resource limitations for this model and reduces the model's size by 4.3 times and 5.5 times in training times. This paper aims to address the problem of running the BERT on deficient computational devices like mobile. The paper claims to have used an inverted BERT model for training and distilled the knowledge for the reduced BERT version. However, it is essential to note the specific trade-off between accuracy and model size by reducing numbers.

Application of distilled BERT models [6] to solve sub-problems within the full BERT is an attempt to reduce the computing requirements. This approach does not address the Multi-Level classification problems directly. Considering this as the base model for the Multi-Level classification problem, we will still see growth with additional layers being added at deeper classification levels. Alternatively, applying our methodology to this distilled model poses a different trade-off for accuracy. In the distillation process, reducing the model size will result in trade-off problems.

## **2.6 Limitations of Models**

Primarily, the limitation of the deep learning models is the computational requirements. As the natural language processing domain research sees growth with accuracy as a primary interest, the models are growing more complex, deeper, and heavier. It is a



limitation on physical infrastructure like graphic processing units, which couldn't grow at the same speeds as the models.

State-of-art models usually measure performance for establishing benchmarks by looking at the cleaned standard datasets. However, the real-world problems are complex as it's difficult to have specific boundaries within the features and overlapping features between multiple classes. As we understand state-of-art models establishing new benchmarks, there is very little focus on applying these models to the data, which is multi-fold in nature. For example, in a two-level of Multi-Level classification, it's just not possible to hold two times larger models within a physical infrastructure. Although we could potentially train the models one after another, it poses multiple issues like a lack of shared context between models and collaborative learning.

The related work in hierarchical applications of BERT models has not aimed to solve Multi-Level classification. However, attempts in the related works have been made to reduce the computational requirements by reducing the layers of the BERT model through the distillation process. The distillation process has a trade-off with accuracy for reducing the attention layers, and applying a distilled version of models would make this problem only much worse. Application of its distilled model at each classification level will result again in the models' size, which will result in exceeding memory limitations.

## CHAPTER 3

### MULTI-LEVEL ATTENTION TRANSFORMERS

#### 3.1 Model Design

We propose a scalable model to solve the Multi-Level classification problem without exhausting the hardware limitation and a drastic increase in the training time with an increase in the depth of classification. The knowledge extraction for the Multi-Level classification is much complex in nature compared to the regular classification models. We believe the base BERT base model has enough knowledge and context within the pre-trained Wikipedia texts using two training strategies of Masked language modeling and next sentence prediction.

BERT is currently designed in two models based on the number of transformer layers it contains. the base BERT has 12 x transformers, and large BERT has 24 x transformers. We have considered base BERT for our experiments to prove our hypothesis and limiting the computation requirements. Our datasets are in the public domain, so we consider the pre-trained the base BERT model trained from Wikipedia texts. This model is further trained to bring Multi-Level classification design with sharing context with the only addition of custom classification layers. As part of the training strategy, the base BERT model needs to be trained separately using masked language modeling and next sentence prediction. The model gathers knowledge about the domain in the model is evaluated.

The proposed architecture itself is a technique that could be applied on the base BERT or large BERT. It is recommended the training of the base models are to be trained separately to gather knowledge about the domain-specific nature, i.e., medical domain, etc. The fact that the representations of the base model remain intact is proven by the adequate understanding of attention parameters that are represented from the base model. We believe the only layer that needs assistance with learning is the custom classifier layers. We further acknowledge that the context arrived by the current classification output level into subsequent layers as a sharable attribute that aids classification tasks. Our hypothesis has experimented on the model referred to in Figure 4.

Attention technique within the transformer model can be represented formally as below 3.1

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (3.1)$$

We have used BCE Loss as a loss function for individual models used for classification. BCE Loss is to calculate loss from each model. BCE Loss can be formally represented in Equations 3.2 and 3.3. A cumulative loss from individual BCE loss values from each classification level arrives at the final loss values. Representation of the cumulative loss function is defined in Equation3.4

$$l_N(x, y) = -w_n[y_n \cdot \log \sigma(x_n) + (1 - y_n) \cdot \log(1 - \sigma(x_n))] \quad (3.2)$$

$$BCE(x, y) = L = \{l_1, \dots, l_n\}^T \quad (3.3)$$

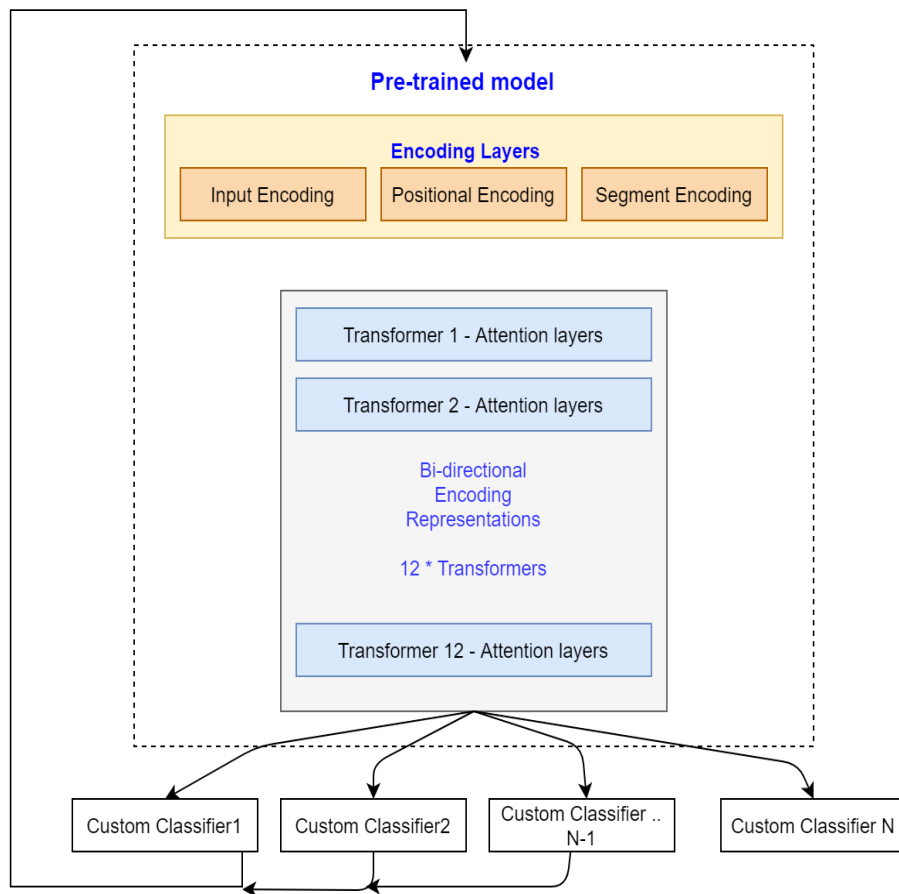


Figure 4: High Level Architecture for Multi-Level Attention Transformers

$$lf_{ml}(d) = \sum_0^d BCE(d - 1) \quad (3.4)$$

The core design of this model is based on the fact to it avoids the need for multiple attention layers as the context within the domain of the knowledge is tightly coupled. For example, if we apply the model towards a telecommunication dataset, looking at the text corpus relating to telecommunication remains the same across the models. If we are using this model for health-based data, the context within the health data remains the same across attention layers. To identify the definitive need to avoiding the repeated attention layers, which are responsible for understanding the context of the data, we propose a design to reuse the attention layers at multiple classification levels to retain the context alongside shared parameters.

The base BERT model facilitates training for a specific domain to understand the context within the domain. An opportunity can be viewed here to share the predicted parameters between classification levels to aid the predictions at deeper classification levels. As much depth as the model tends to get, and we can subsequently leave the trace of the shared parameters right from the start of the classifier. The context of the first level input from the full description of the data, as to the number of epochs increase the trend of training gets to see repeated instances of particular output from the classification being passed into description making that as sharable but value attribute to the deeper classification problem. This type of prediction could be represented as a mathematical function 3.5.

We represent  $x(i)$  as input features.  $y(d)$  is the prediction output at the depth  $d$ .

Predictions using BERT models could be represented as a function of  $bp$ .

$$bp(d) = concat(\sum_0^{d-1} y_d - 1) + x_i \quad (3.5)$$

### 3.2 Model Architecture

Python has been chosen as the language to build the models, PyTorch with CUDA toolkit and GPU capabilities are used to implement the built models. The PyTorch data loader library reads the dataset with a batch size of 4 for training and testing. Batch size is defined as 4 to save the GPU memory limitations and bring a fair comparison between different models designed and implemented. The case of a dataset having two levels of classifications is chosen to implement the proposed designs.

1. **Flat Multiple BERT:** Two classification levels require two BERT models to train each level of classification. The models are independently trained from each other.
2. **Multi-Level BERT:** The design for this model attempts to reuse the attention layers that are part of the BERT base model. The model design relies on feeding the predicted outputs from the current processing classification level to the following classification level as a shared context.
3. **Multi-Level BERT Tuned:** Multi-Level BERT depends on the shared context being passed from the predicted outputs from the current classification level. This means the predicted outputs do contain a certain amount of loss which is also transmitted to the second level of classification leading to depreciation in classification

accuracy at the following level is impacted. The training is made independent in this fined-tuned version by sending actual outputs rather than predicted outputs as the shared context between two classification levels.

4. **Multi-Level BERT Feature Tuned:** Both Multi-Level BERT and its tuned version of models have two training passes for two-level classification does not give a fixed training time with an increase in complexity. The feature extracted model aims to make one training pass with all the possible attributes at deeper level classification and then eventually map the different levels of classification at the time of training.

### 3.3 Base BERT

BERT model is built with several transformers layers which helps to implement the Attention mechanism. BERT is designed with two versions based on the number of transformers they contain at the time of the proposal. Namely, the base BERT has 12 x transformers with 110 million parameters, and the large BERT includes 24 x transformers with 340 million parameters. The base BERT model implemented as part of this experiment can be referred to 5.

The Wikipedia text corpus has primarily trained the base BERT model by applying training strategies as masked language modeling and following sentence predictions. Our experiments are based on public domain datasets, and thus we chose the public domain BERT, which is trained from Wikipedia. Considering the context retention happens within the BERT attention layers, it's essential to view that model's domain carefully. While applying the BERT model to different datasets, we believe it is empirical to train based

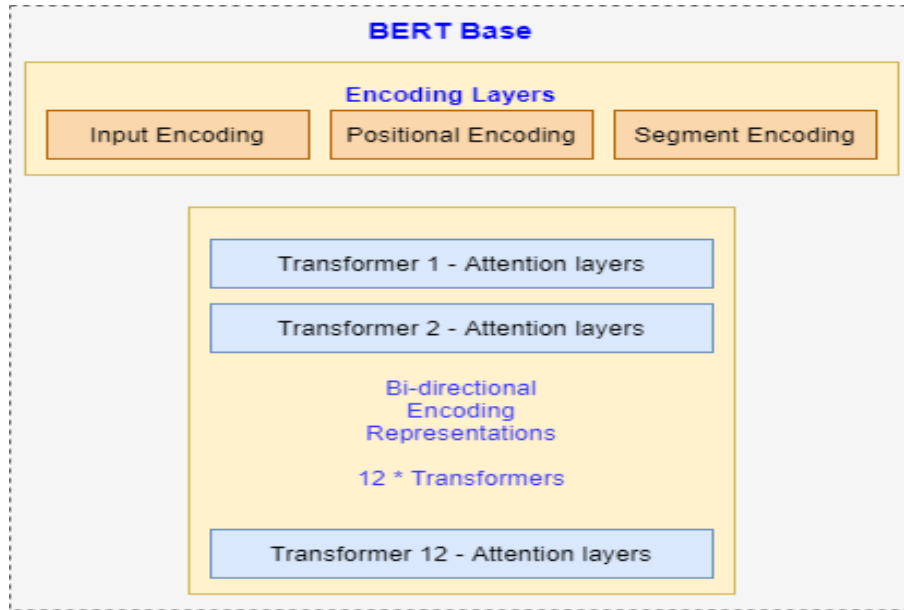


Figure 5: High Level Architecture for Attention Transformers

on the BERT model with training strategies like masked language modeling and next sentence prediction with that domain.

### 3.4 Flat Multiple BERT

Implementations for the Flat Multiple BERT have been implemented with Python3 and Pytorch for subsequent execution of the BERT. Before implementing the model, the data has been fed through a PyTorch-based custom dataset and data loader to run through records in the dataset in particular batch size. 12 x transformer model has been used to run this experiment, which consisted of about 110 million parameters. As part of the evaluation, we have considered two levels of classification. The BERT model has been tasked with classification at each level individually. This requires the exact implementation twice



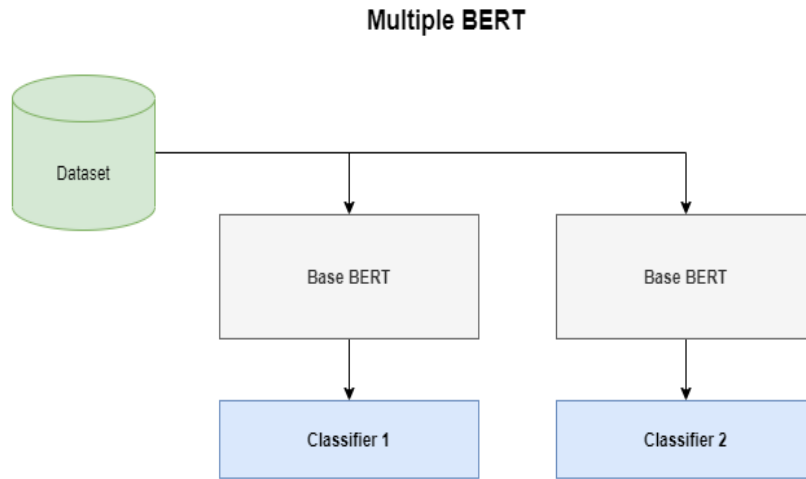


Figure 6: Multiple BERT Model Architecture

to apply classification on the base BERT model at different levels.

To implement the Flat Multiple BERT, we have build two models with classification outputs as Level 1 and Level 2. The flow diagram for this model can be demonstrated in Figure 6. These being two independent models, they are trained in a setting of one model at a time. The Graphic processing unit (referred to as GPU) is 11 GB, Ge force RTX 2080 TI. Considering each of the BERT models is in sheer size by itself, occupying 9.7 GB once loaded, the only way to train this model has been in a sequential order which is also the crux of our problem aiming to solve. Secondly, as the models are independently trained, there is no mutual sharing of the context of which class is being predicted at each classification level. The expected output can add valuable support to the text corpus to improve the classification of deeper levels.

### **3.5 Multi-Level Hierarchical BERT**

The Multi-Level BERT design spins around the hypothesis of reusing the attention layers. The implementation of BERT has remained the same between the earlier design. We propose the separate training of the domain expertise for the Base BERT model and then further trained for the deeper level tasks, which reuses the attention layers around all the level classifications with sharing of the parameter. A representation of this design can be observed in Figure 7.

The training is supposed to happen in two phases, training the base model and then further training for the fine-tuned tasks like classification. The dataset we currently deal with is related to the public domain. Hence we considered a pre-trained model with Wikipedia texts which have learned the context of natural language processing through training strategies of Masked Language modeling and following sentence predictions. Considering the two classifications, the base BERT, which is pre-trained with public domain and fine-tuned classification layers. The predicted outputs from the current classification level are shared with the following classification level as a shared context.

### **3.6 Multi-Level Hierarchical BERT Tuned**

The Multi-Level BERT performed within the limitations of hardware capabilities and showed competitive accuracy scores are achieved. However, there have been downsides observed in this type of learning since the training of the Level 2 classifier is majorly based on the predicted outputs of the Level 1 classifier. The accuracy and performance

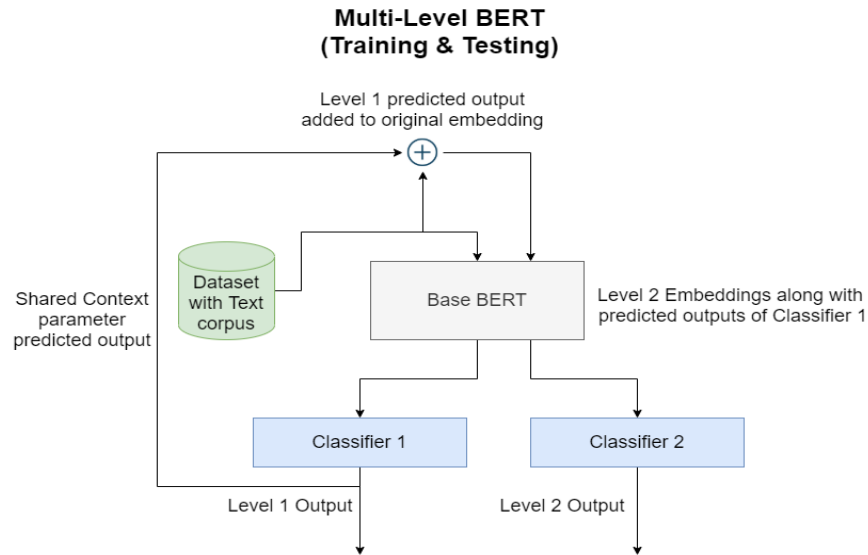


Figure 7: Multi-Level BERT Model Architecture

of Level 1 have been an essential dependency for the deeper classifications. It could easily be inferred that such dependencies will lead to a chained sequence of impact for the deeper classification level. The deeper classification layer's accuracy would be impacted with a chained sequence of classifiers passing the sharable context parameter wrong.

We proposed a tuned model over the Multi-Level BERT to make the training and inference independent. The proposed change is to pass the original values over predicted outputs in the training sequence to avoid model learning with the wrong shared context. This makes deeper classifier layers independent of the predicted outputs from the earlier classifiers. At the same time, it also solves the need to have shareable context parameters. Since the change is only apparent at the time of training and the evaluation remains the same, the implementation has to be detailed at the level of training and testing. The flow

diagram from the training could be referred to in Figure Fig. 9.

The text corpus for the first level of classification is added with embeddings of original outputs for the first level classification for the second level classification. As the model grows deeper, making it independent of the performance of the early levels of classifications. The tuned BERT implementation for the testing phase remains the same as the original Multi-Level BERT flow. During the evaluation, we no longer assume having the original outputs. The flow diagram can be referred in Fig. 9. The results have shown this technique has improved inaccuracy to the original Multi-Level BERT.

### **3.7 Multi-Level Feature-Based BERT**

The Multi-Level Hierarchical BERT tuned model has solved the problem with computational limitations. However, the model is involved in making two training passes for two levels of classification. The predicted/original outputs from the current classification level should be passed to the next classification level. Having reduced the number of layers, there is some amount of reduced training time, but the training time still depends on classification levels. We propose another lean version of our model to address this problem, which works on extracting the features from the base layers of BERT and applying them into different classification layers.

The training strategy is tuned further to extract the features from the base BERT model with all possible shared contexts till the deep classification level and then subject those features to the custom classifier layers. The strategy involves extracting the features from one training pass. The dependency of this model to run through multiple training

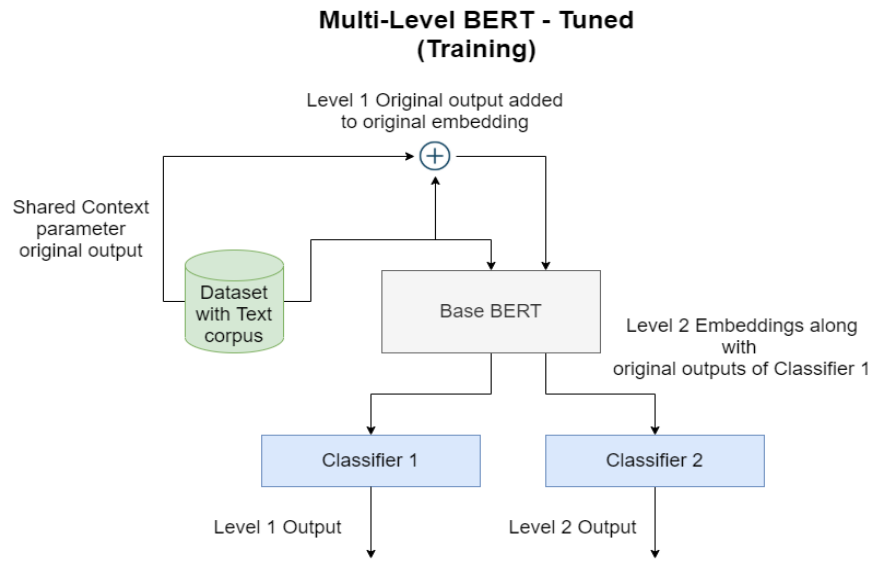


Figure 8: Multi-Level Tuned Model Architecture - Training

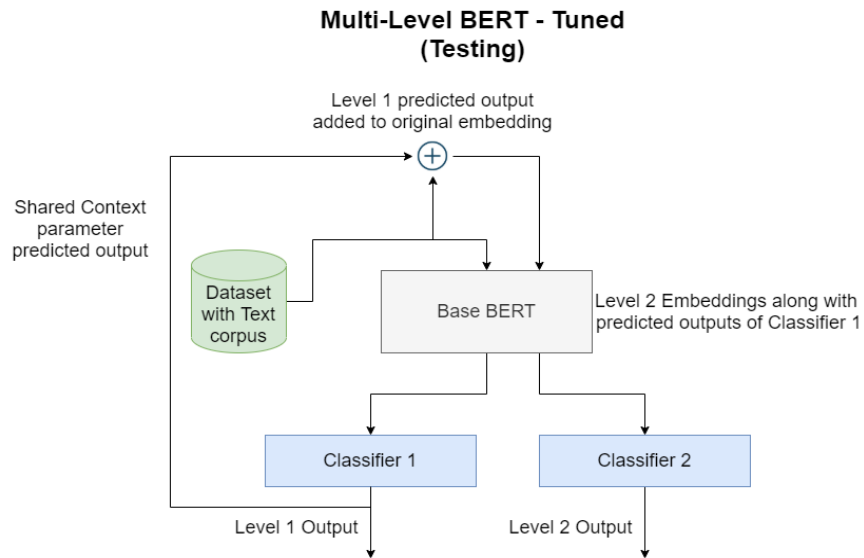


Figure 9: Multi-Level Tuned Model Architecture - Inference

passes has been removed. It is observed from the above implementations, and relatively the deeper classification layers suffered most of the accuracy trade-off. These models aiming at the deeper classification models also showed improvements in the convergence of deeper classification layers.

### **3.8 Summary of the Proposed Models**

The design outlines several models designed to solve memory footprint, model size, and training times. The design of the proposed models revolves around the usage of attention layers to reduce the model depth, memory footprint, and training times. An abstract of each models specialization is outlined in Table 3.8.

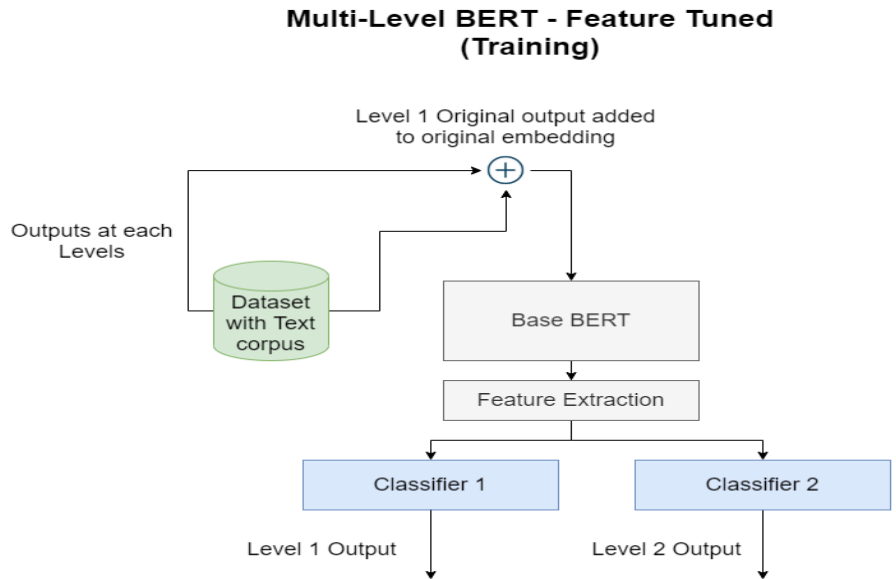


Figure 10: Multi-Level BERT Feature Model - Training

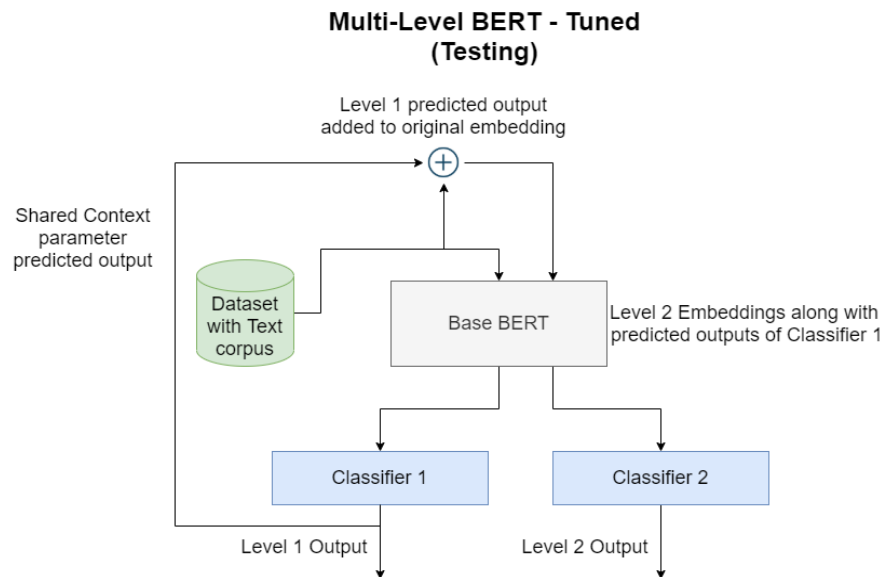


Figure 11: Multi-Level BERT Feature Model - Testing

<b>Model Name</b>	<b>Highlights</b>
Flat Multiple BERT	In order to solve two levels of classification problem, this model requires two full sized BERT models. The training is sequential in format of one after another. There is no possibility of sharing context between the two classification levels due to their independent nature of training the models.
Multi-Level Hierarchical BERT	Two-level classification problem can be solved by re-using attention layers of one BERT model. This effectively reduces the number of layers required to run the task. The context between the two classification levels is shared to aid the predictions. The shared context majorly contains the predicted outputs from the initial levels of classification. However, This model requires two training passes to accomplish the task.
Multi-Level Hierarchical BERT Tuned	This is a tuned model from the Multi-Level Hierarchical BERT, where the training and inference strategies are separated. At the time of training, the model is fed with actual inputs over predicted outputs from initial classification levels to ensure the loss from initial classification levels is not passed into the following classification levels.
Multi-Level Feature-Based BERT	Multi-Level Hierarchical BERT and its tuned version do two training pass for two classification levels. While the model size is regulated, the training time is dependent on the number of classification levels. The model is changed to extract features from one training pass with all possible context and feed the extracted features into custom classifier layers for classification. This makes the training time constant along with model size.



## CHAPTER 4

### EVALUATION AND RESULTS

#### **4.1 Dataset**

As part of community research, the model is conceived by reviewing the real-time data behaviors, which were multi-folded in nature. The model is built and evaluated on a real-time dataset to understand the performance of the model. The model has achieved satisfactory results in its performance, and the model is further evaluated with a research community accepted standard dataset DBPedia. The counts for each of the classes in the Level 1 and Level 2 for both 311 Kansas City data and DBPedia data can be found in Table 1 and 2.

##### 4.1.1 311 Kansas City Dataset

The idea of Multi-Level classification has been deemed a necessity from working on actual-world data. We have worked with 311 service requests for Kansas City and tried to identify the type of parent problem and then further down the sub-category of the problem. Most of the real-world problem like medical domain belongs in this type of arrangement. The first level of the class certainly gives contextual meaning to the later layers. The sample of the 311 Kansas City dataset with hierarchical relations can refer to in Table 3.

Table 1: Level 1 Classes for Data sets

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 1 Class</b>	<b>Count</b>	<b>Level 1 Class</b>	<b>Count</b>
Animals	34057	Agent	177341
City Facilities	189	Device	353
Data Not Available	131	Event	27059
Legal	1506	Place	65128
Lights/Signals/Signs	17478	Species	31149
Maintenance	57	SportsSeason	8307
Neighborhood	63	TopicalConcept	1115
Noise	64	UnitOfWork	2497
Other	18307	Work	29832
Parking	493		
Parks and Recreation	2654		
Property Violations	33254		
Public Health	6152		
Public Safety	19318		
Street/Sidewalks	33625		
Traffic	356		
Trash	81880		
Vehicles	12228		
Water	37770		

Table 2: Level 2 Class Count - DBPedia and 311 Kansas City Dataset

311 Kansas City Data		DBPedia Data	
Level 2 Class	Count	Level 2 Class	Count
311 Administration	55	Actor	1667
ADA	24	AmusementParkAttraction	675
Abandoned	390	Animal	21333
Abandoned On Street	7644	Artist	7091
Action Center	4	Athlete	44163
Administration	711	BodyOfWater	2695
Airports	22	Boxer	403
All	189	BritishRoyalty	685
Alley	208	Broadcaster	6549
Ambulance	4	Building	15266
Animal	692	Cartoon	2724
Athletics	22	CelestialBody	3514
Banners	4	Cleric	6420
Barking Dog	186	ClericalAdministrativeRegion	2693
Bite	680	Coach	2691
Bld / Pkwy	158	Comic	3034
Board Up	454	ComicsCharacter	203
Boulevards and Parkways	8	Company	11777
Broken Asphalt	699	Database	187
Brush	46	EducationalInstitution	6306
Building / Property	76	Engine	353
Bulky	186	Eukaryote	2698
Bulky Pick Up	3502	FictionalCharacter	3062

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
Capital Projects Office	16	FloweringPlant	346
Care	117	FootballLeagueSeason	2698
Catch Basin	376	Genre	1115
City	394	GridironFootballPlayer	2696
City Managers Office	49	Group	2659
City Property	10	Horse	2693
Cleaning	252	Infrastructure	5397
Commercial Parking	19	LegalCase	2497
Commercial Signs / Ads	508	MotorcycleRider	633
Commercial Vehicle	118	MusicalArtist	284
Communicable Disease	1	MusicalWork	9903
Communications	3	NaturalEvent	1088
Community Center	57	NaturalPlace	7766
Community Service	5	Olympics	2699
Construction Issue/Concern	1064	Organisation	10137
Construction Repair	3	OrganisationMember	553
Contractor	811	PeriodicalLiterature	8089
Contractor Restoration	198	Person	27892
Crack	252	Plant	4079
Cruelty or Neglect	3663	Politician	13514
Culvert	536	Presenter	318
Curb Box	28	Race	3016
Customer Service	68	RaceTrack	242
Cut / Permit	1492	RacingDriver	1593
Damage	196	RouteOfTransportation	8359
Damage / Dis-Repair	2864	Satellite	2224

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
Dangerous Building	4323	Scientist	824
Data Not Available	131	Settlement	5387
Dead Animal	7069	SocietalEvent	8608
Disabled/Unlicensed on Private Property	1611	Software	2699
Disease	21	Song	1000
Disease Control	1159	SportFacility	3251
Disrepair	120	SportsEvent	5766
Ditch	462	SportsLeague	3405
Dumping	9249	SportsManager	2695
Dumpsters	134	SportsTeam	7968
Early Set Out	93	SportsTeamSeason	5609
Elevator Inspection	59	Station	2073
Emerald Ash	242	Stream	3074
Emerald Ash Borer	23	Tournament	5882
Emergency Management	21	Tower	1788
Encroachment	184	Venue	724
Engineering	164	VolleyballPlayer	194
Facilities / Attractions	96	WinterSportPlayer	8972
Fire	114	Wrestler	425
Flood Barricade	34	Writer	1562
Food Establishment	1652	WrittenWork	2196
Graffiti	86		
Guardrail	124		
Hotel / Motel	143		
Hydrant Repair	32		

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
IT / Websites	7		
Icing	18		
Illegal Dumping	133		
Illegally Parked	1		
Improved Channel	100		
Industrial	1		
Injury or Cruelty involving an Animal	357		
Insect Problem	3		
Intersection	20		
Investigation	997		
Investigations	958		
KC Police	13		
Laboratory Service	28		
Land Bank	261		
Land Development	7		
Land Trust	33		
Land Use / Zoning Issue	1002		
Landlord Set Out	17		
Landscaping	24		
Law	1		
Lead Poisoning Prevention	1		
Leaf and Brush	119		
Leaf / Brush	6238		
Leak	9526		

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
MFS Referral (Meter Field Services)	565		
Maintenance	173		
Malfunction	380		
Manhole	599		
Markings / Paint	696		
Meter	314		
Missing	495		
Municipal Court	4		
NHS (Neighborhood Housing Services)	157		
Neighborhood Abatement Program	3		
New	1		
New Request	179		
No Dumping Sign	141		
No Water / Pressure	5576		
Noise	1140		
Not Provided By KCMO	1072		
Nuisance	10928		
Obstructed / Closure	545		
Other-Maintenance	1419		
Outage	301		
Outdoor Air Quality	260		
Owned or Stray at Large	1141		
PIAC	265		

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
Paper Street	20		
Park Maintenance	1916		
Park Property	8		
Parked on Unapproved Surface	2030		
Parking	16		
Parking Lot	27		
Parking Meter	12		
Parks	205		
Permit	98		
Permit / License	295		
Pipeline Referral	4326		
Pipeline Repair	66		
Pipeline Restoration Concerns	24		
Planning	3		
Planting	42		
Plate	480		
Police	805		
Pollutants	31		
Pool / Hot Tub	45		
Pothole	14106		
Priority	4		
Private / Commercial	11989		
Private Property	4948		
Property Maintenance	22656		



<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
Public Facilities	14		
Public Improvement Advisory Committee	16		
Public Property	961		
Public Restroom	26		
Public Works	296		
Quality	686		
Questionable Activity	733		
Rat Control and Treatment	170		
Rat Treatment	3754		
Recycle	468		
Recycling	14833		
Regulated Industries	116		
Removal	5292		
Repair	118		
Replacement	35		
Restoration	9		
Resurfacing	832		
Return Call to Citizen	72		
Right of Way	448		
Right of Way (ROW)	1503		
Scooter	18		
Services	12034		
Services (CPD)	75		
Services (NPD)	150		
Sewer	620		

<b>311 Kansas City Data</b>		<b>DBPedia Data</b>	
<b>Level 2 Class</b>	<b>Count</b>	<b>Level 2 Class</b>	<b>Count</b>
Sewer Backup / Leak	3845		
Sewer Odor	1474		
Sewer Referral	1626		
Sinkhole	537		
Sinkhole Referral	339		
Smoking / Tobacco	27		
Snow / Ice	4115		
Solid Waste Operations	7		
Speed Bump	346		
Storm Damage	3157		
Storm Drain / Catch Basin	3588		
Storm Water	247		
Storm Water Referral	420		
Stray	11801		
Stray Confined	193		
Street	1175		
Street Clean Up	2075		
Street Light	6438		
Street Name Signs	1371		
Street Preservation	23		
Street Services	739		
Street Sweeping	49		
Street and Traffic	53		
Stump Removal	51		
Ticket	2		
Tobacco	4		

311 Kansas City Data		DBPedia Data	
Level 2 Class	Count	Level 2 Class	Count
Tow Services	83		
Traffic Permit	27		
Traffic Sign	4917		
Traffic Signal	3663		
Traffic Signs	18		
Traffic Study	164		
Trails	131		
Trash	1		
Trash Cart	65		
Trash Collection	26650		
Treatment	2		
Trimming	5946		
Unapproved Objects	193		
Valve Repair	9		
Visibility	200		
Water	920		
Water Main Repair	6		
Wildlife	2825		

Table 3: 311 Kansas City Dataset Sample

Text corpus of service request	Category	Sub Category
The complainant stated his neighbor operates a scrap metal business and burns plastic in a barrel at his house instead of sending it to a landfill.	Public Health	Pollutants

<b>Text corpus of service request</b>	<b>Category</b>	<b>Sub Category</b>
The citizen is reporting excessive barking. The dog barks all day and all night. The dog is kept outside in the backyard. The dog is a German Shepherd.	Animals	Barking Dog
The nurse is calling from the Emergency room to report an animal bite. The dog is a brindle color and is a Shepherd. The dog bit the patient on the left thigh, and the skin is broken. The victim is an adult.	Animals	bite
The citizen reports a brown dog hanging by leach on the back fence of the property.	Animals	Injury or Cruelty involving an Animal
A citizen called to report this Senior Citizen home that has bed bugs.	Animals	Insect Problem
The citizen reports a pit bull in his backyard, black and white and very aggressive. The citizen states he has four kids in the home.	Animals	Owned or Stray at Large
A citizen called to report a confined aggressive black poodle. It is located in the back yard of his address. Call when in the area.	Animals	Stray Confined
The caller is reporting a sick raccoon on his back fence that will not leave his property. He states it appears to have rabies.	Animals	Wildlife
A citizen reported that the resident leaves the dog out all day and night. Citizen described no food and water or shelter. Citizen described that the dog cries all the time for being mistreated. Citizen described the dog as a German Shepherd. Dispatch: 22	Animals	Cruelty or Neglect

<b>Text corpus of service request</b>	<b>Category</b>	<b>Sub Category</b>
Citizen is reporting a dead deer on the side walk on the 48 street side of this house	Animals	Dead Animal
Citizen reporting a rooster at this address in a cage in the back yard. It was kept in the cage over night without food or water. They also have an unaltered Pitbull that is tan colored.Call taker 67	Animals	Permit / License
Citizen called and would like to know if the Byram's Ford bridge is being repaired or if it will be closed for good. Citizen received a letter saying it will be repaired, but her neighbor received a letter saying that it will be closed.	Maintenance	Repair
Caller reported sewer back up in the basement w/an odor located at 901 E. 76th Terr.	Water	Repair
Citizen calling to report she had snake and rooter come out to do some repairs and was told she need to contact the city because sewage may be going into the groundwater coming from the city side. They noticed mud at the tap. Snake and Rooter marked the street where it's located. Contact person Donna Wilson at 816xxxxxxx and she ran a camera on the East side of the property.	Water	Repair
Citizen is reporting a large hay bale blocking the northbound lanes	City Facilities	Airports
Citizen is requesting ROW mowing for 104th St from Prairie View to Amity along the airport properties	City Facilities	Airports

<b>Text corpus of service request</b>	<b>Category</b>	<b>Sub Category</b>
Citizen is wondering if the city would close the alley behind her home and give the property to the homeowners to build fences to completely fill it in. Citizen has concerns of safety in the neighborhood. There have been times that the homeless have been in the backyard and have walked up to numerous houses. The alley is not used at all	City Facilities	Building / Property
Citizen is reporting a television that has been dumped in the back of this property that is next to his building. He stated he has been calling in about this and it hasn't been removed. The Dumping is on the back of the entrance at the Gem theater	City Facilities	Building / Property
Citizen is reporting the south elevator in the garage are not working properly. One of the elevator is not working at all	City Facilities	Parking

#### **4.1.1.1 Data Issues**

311 KC dataset is a real-world dataset and human-generated, there are several problems like request descriptions being inaccurate, wrong assignments of geo-mapping, and problems with the classification mapped to the problem. Our research is to identify the text corpus within the dataset (description of the complaint) to classify the problem and its sub-problem. Description being features for our model to perform, we observe the overlap in the features between multiple classes at all the levels.

Table 4: 311 Kansas City Dataset Statistics

<b>311 Kansas City Stats</b>	<b>Counts</b>
Level 1 Class	19
Level 2 Class	214
Sampled Records	42800
Actual Records	299582

Considering an example from the 'trash' class, we see two sub-classes like 're-cycle' and 'recycling.' The descriptions for these classes talk about recycling the trash. The model views the features of these two classes are not providing clear boundaries to identify the class. Considering human intelligence, we could map these two classes into a single one for the model to perform. Still, having this a real-world dataset and not understanding its implications on having two different classes, we chose not to map them as a single class.

#### **4.1.1.2 Geo Mapping Issues**

Since the customer service center collects the data, the data does have human errors, primarily assigning the wrong category to the described problem. The presence of the automated solution doesn't exist for this problem as the descriptions can be very complex in explanation. The other problem, to perform statistical analysis, the geo-location tagged with the requests has been faulty, which is corrected utilizing the google geocode to address mapping. These geocode are also corrected by correctly mapping the data from the master data of boundaries shared by the concerned department. The boundary mapping algorithm has been designed to solve this problem at the of data analysis.

The boundary mapping algorithm maps the data to a particular record if all the

below conditions are met.

1. a geocode must be mapped at least as a point in the given boundaries which sits with lower latitude and greater longitude.
2. a geocode must be mapped at least as a point in the given boundaries which sits with lower latitude and lower longitude.
3. a geocode must be mapped as at least a point in the given boundaries that sits with greater latitude and greater longitude
4. a geocode must be mapped at least as a point in the given boundaries that sits with greater latitude and lower longitude.

The Kansas City 311 dataset comes with geo-locations-based address mapping. As part of exploratory data analysis, we found that about 42K records have been pointed to the exact geo-location, which also doesn't seem to be the part of the physical location of consideration. To recover these records, we have applied street address and zip codes to resolve the geo-location using google geocoding API and then resolved the right geo-mapping for the complaints raised. The resolved geo-mapping then applied against the neighborhood data of Kansas City and then mapped them accordingly to the parts of Kansas City.



#### 4.1.1.3 Imbalance Issue

The data is a public dataset that is constantly updated from 311 calls made in Kansas City. Real data of this nature is expected to repeat the problems repeatedly, specifically in the same neighborhood. It is always important to understand the nature of the data before exposing it to the model to bring out predictions. More over sampled classes will help the model to lean the predictions towards the over-sampled classes. The data is balanced by using random over-sampled and under-sampled to balance the data for Level 2 classes. This count of the records based on Level 2 classes can be found in Figure13. In the process, the reduced dataset is applied for the model as some of the classes are heavy in the count. It could be referred at Table 4

The data for level 1 is not as balanced as the case of Level 2. This is because the subproblem categories are shared across different categories. The reference for Level 1 can be found in Figure 12. The dataset has been normalized into balanced data concerning Level 2 classes. There are 229 Level 2 classes, but without applying the context from Level 1 classification, it is nearly impossible to maintain the performance of the Level 2 classification in a model that shares the attention layers for both levels of classification. Level 1 and Level 2 classes for the 311 Kansas City Dataset can referred in the Table 5.

Table 5: 311 Kansas City Dataset Level 1 and Level 2  
Classes Mapping

<b>Level 1 Classes</b>	<b>Level 2 Classes</b>
Animals (16)	Barking Dog, Bite, Cruelty or Neglect, Dead Animal, Injury or Cruelty involving an Animal, Insect Problem, Investigation, Owned or Stray at Large, Permit / License, Questionable Activity, Rat Treatment, Return Call to Citizen, Services, Stray, Stray Confined, Wildlife
City Facilities (11)	Action Center, Administration, Airports, Building / Property, Capital Projects Office, City, Emergency Management, KC Police, Land Development, Parking, Public Improvement Advisory Committee
Data Not Available (1)	Data Not Available
Legal (9)	311 Administration, ADA, City Managers Office, Communications, IT / Websites, Law, NHS (Neighborhood Housing Services), Not Provided By KCMO, Traffic Permit
Lights/Signals/Signs (9)	Banners, Commercial Signs / Ads, Damage, Malfunction, No Dumping Sign, Outage, Street Light, Street Name Signs, Traffic Sign, Traffic Signal
Maintenance (2)	Alley, Repair
Neighborhood (3)	Administration, Priority, Regulated Industries
Noise (1)	All
Other (17)	Administration, All, Alley, Blvd / Pkwy, Encroachment, PIAC, Paper Street, Parks, Private / Commercial, Public Property, Public Works, Right of Way (ROW), Sewer, Storm Water, Street and Traffic, Visibility, Water
Parking (4)	Maintenance, Parking Meter, Street, Ticket

<b>Level 1 Classes</b>	<b>Level 2 Classes</b>
Parks and Recreation (10)	Administration, Athletics, Community Center, Community Service, Facilities / Attractions, Graffiti, Landscaping, Park Maintenance, Planning, Trails
Property Violations (13)	Board Up, City Property, Construction Issue/Concern, Dangerous Building, Elevator Inspection, Land Use / Zoning Issue, Neighborhood Abatement Program, Parking Lot, Private Property, Property Maintenance, Services (CPD), Services (NPD), Traffic Signs
Public Health (17)	All, Boulevards and Parkways, Communicable Disease, Disease Control, Food Establishment, Hotel / Motel, Land Trust, Lead Poisoning Prevention, Noise, Outdoor Air Quality, Pollutants, Pool / Hot Tub, Public Facilities, Public Restroom, Rat Control and Treatment, Smoking / Tobacco, Tobacco
Public Safety (21)	All, Ambulance, Brush, Care, Customer Service, Disease, Emerald Ash, Emerald Ash Borer, Emergency Management, Fire, Intersection, Land Bank, Municipal Court, Permit, Planting, Police, Regulated Industries, Removal, Storm Damage, Stump Removal, Trimming
Street/Sidewalks (27)	Broken Asphalt, Crack, Cut / Permit, Damage, Damage / Disrepair, Disrepair, Ditch, Guardrail, Markings / Paint, Missing, New, New Request, Obstructed / Closure, Other-Maintenance, Outage, Park Property, Plate, Pothole, Replacement, Resurfacing, Sinkhole Referral, Snow / Ice, Speed Bump, Street Clean Up, Street Preservation, Street Services, Unapproved Objects
Traffic (2)	Damage, Traffic Study

<b>Level 1 Classes</b>	<b>Level 2 Classes</b>
Trash (21)	Animal, Bulky, Bulky Pick Up, City, Contractor, Customer Service, Dumping, Dumpsters, Early Set Out, Illegal Dumping, Landlord Set Out, Leaf and Brush, Leaf / Brush, Nuisance, Recycle, Recycling, Right of Way, Solid Waste Operations, Trash, Trash Cart, Trash Collection
Vehicles (10)	Abandoned, Abandoned On Street, Commercial Parking, Commercial Vehicle, Disabled/Unlicensed on Private Property, Illegally Parked, Meter, Parked on Unapproved Surface, Scooter, Tow Services
Water (34)	Catch Basin, Cleaning, Construction Repair, Contractor Restoration, Culvert, Curb Box, Engineering, Flood Barricade, Hydrant Repair, Icing, Improved Channel, Industrial, Investigations, Laboratory Service, Leak, MFS Referral (Meter Field Services), Manhole, No Water / Pressure, Pipeline Referral, Pipeline Repair, Pipeline Restoration Concerns, Quality, Repair, Restoration, Sewer Backup / Leak, Sewer Odor, Sewer Referral, Sinkhole, Storm Drain / Catch Basin, Storm Water Referral, Street Sweeping, Treatment, Valve Repair, Water Main Repair

#### 4.1.2 DBPedia Dataset

The DBPedia dataset is a standard dataset within the research community extracted from the Wikipedia dataset. There are about 6.0 million entities. Out of these entities, about 5.2 million are classified as inconsistent ontology, including 1.5 million

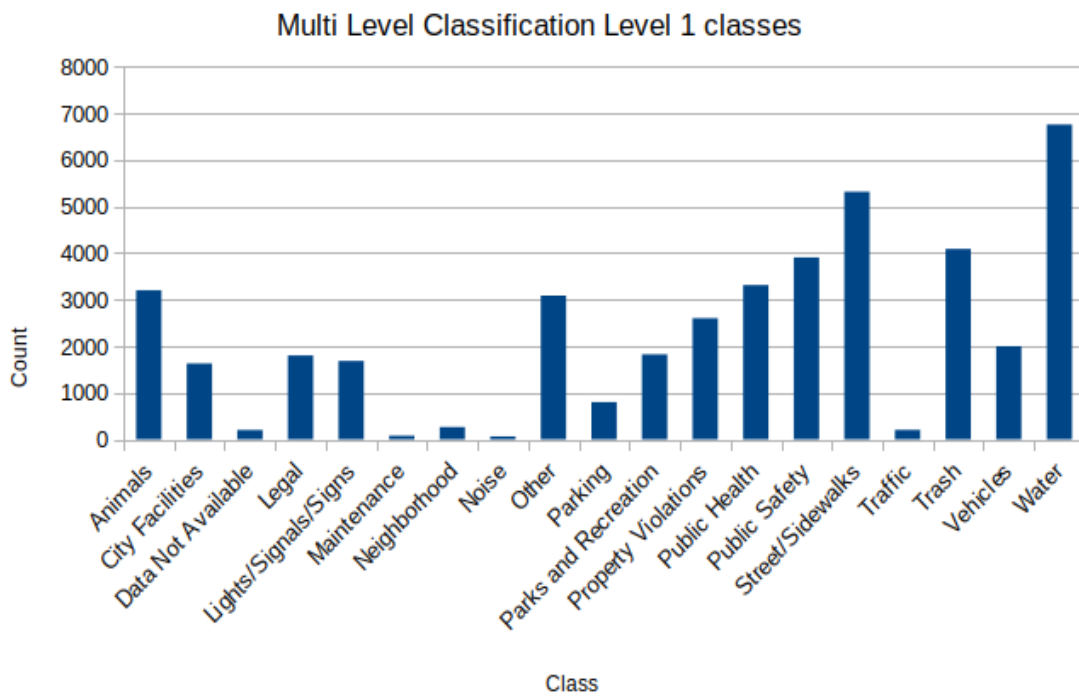


Figure 12: 311 Kansas City Data analysis for Level 1

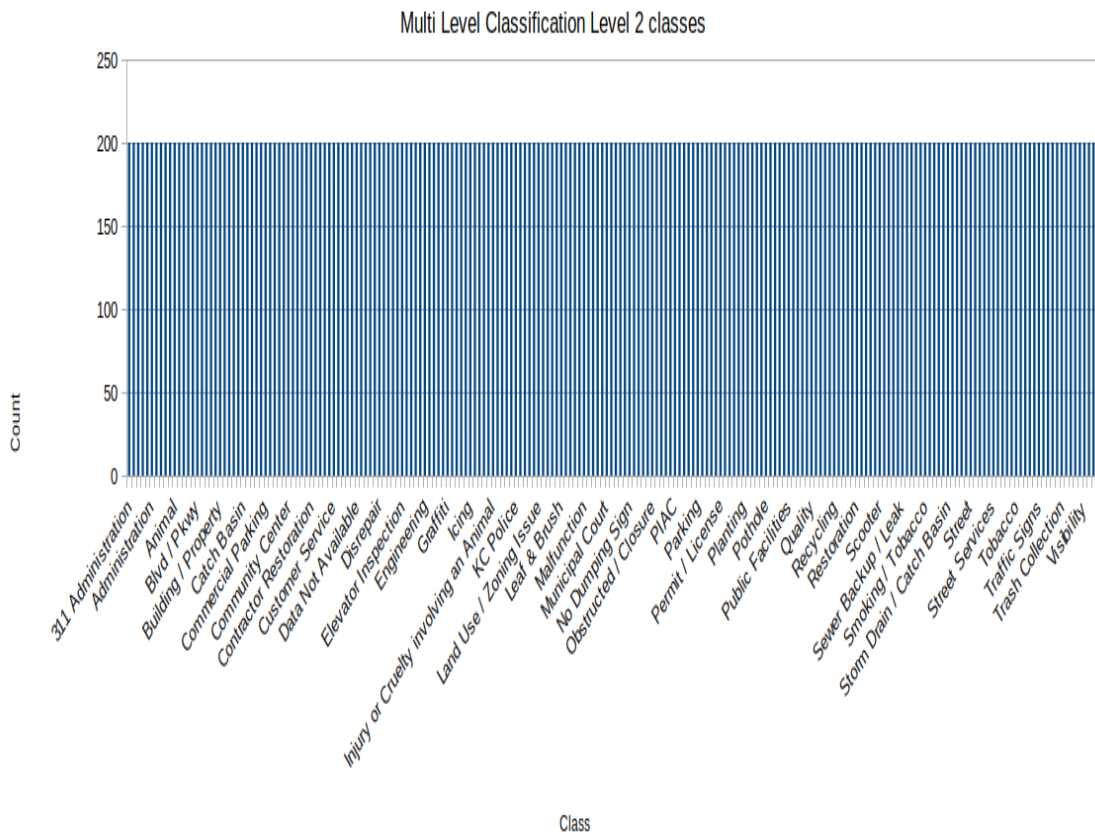


Figure 13: 311 Kansas City Data Analysis for Level 2

persons and 810K places, 135K music albums, 106K films, 20k video games, 275K organizations, 301K species, and 5K diseases.

#### 4.1.2.1 Data Imbalance Issues

DBPedia being a streamlined dataset with defined anthologies, we still see data imbalance issues within the dataset. In order to improve efficiency and balanced learning, the dataset is subjected to random over-sampling and random under-sampling techniques to resolve the data imbalances. The relations between Level 1 and Level 2 classes can be observed in Table 6. A sample of the data extracted from the dataset can be referred in Table 7.

Table 6: DBPedia Level 1 and Level 2 classes

Level 1 Classes	Level 2 Classes
Agent (30)	Actor, Artist, Athlete, Boxer, BritishRoyalty, Broadcaster, Cleric, Coach, ComicsCharacter, Company, EducationalInstitution, FictionalCharacter, GridironFootballPlayer, Group, MotorcycleRider, MusicalArtist, Organisation, OrganisationMember, Person, Politician, Presenter, RacingDriver, Scientist, SportsLeague, SportsManager, SportsTeam, VolleyballPlayer, WinterSportPlayer, Wrestler, Writer
Device (1)	Engine
Event (6)	NaturalEvent, Olympics, Race, SocietalEvent, SportsEvent, Tournament

Level 1 Classes	Level 2 Classes
Place (16)	AmusementParkAttraction, BodyOfWater, Building, CelestialBody, ClericalAdministrativeRegion, Infrastructure, NaturalPlace, RaceTrack, RouteOfTransportation, Satellite, Settlement, SportFacility, Station, Stream, Tower, Venue
Species (5)	Animal, Eukaryote, FloweringPlant, Horse, Plant
SportsSeason (2)	FootballLeagueSeason, SportsTeamSeason
TopicalConcept (1)	Genre
UnitOfWork (1)	LegalCase
Work (8)	Cartoon, Comic, Database, MusicalWork, PeriodicalLiterature, Software, Song, WrittenWork

## 4.2 Experiments

As we intend to prove the capability of Multi-Level BERT to reuse the attention layers. The goals of the experiments are directed towards holding a good performance in comparison with individually trained models over the Multi-Level-based architecture. With such an attempt, we also aim to prove the level of the hardware/GPU resources being limited from converting the models from the Multiple BERT to Multi-Level BERT architecture. In the case of the Multiple BERT, the training could only be done sequentially, one after another, due to the size of BERT models. Sequential training means the context is not shared between the multiple classification levels. Experiments are conducted in a GPU system like Ge Force RTX 2080 TI 11 GB and CPU powered by Intel-i9 processor with 32 GB RAM.



Table 7: Sample of DBPedia Dataset

<b>Description</b>	<b>Level 1 class</b>	<b>Level 2 class</b>
Olivia Saint is an American former actress.	Agent	Actor
Evaristo Baschenis was an Italian Baroque painter of the 17 century	Agent	Artist
Angel Espinosa Capo is a Cuban former amateur boxer best known to dominate the 1980s at junior middleweight and middleweight and capturing the 1986 World Amateur Boxing Championships. He never won an Olympic medal due to Cuba's boycott of the 1984 and 1988 Summer Olympics. A hard-hitting Cuban with a southpaw stance, one of the most feared in his era was known to have about 300 amateur fights.	Agent	Boxer
Sir Francis Haskins Eyles Stiles, 3rd Baronet (died 26 January 1762), formerly Eyles, was a British landowner.	Agent	BritishRoyalty
KBTL (88.1 FM) is a radio station broadcasting a College Radio format. Licensed to El Dorado, Kansas, USA, the station serves the Wichita area.	Agent	Broadcaster
John Francis Jack Meagher was an American football player, coach of football, basketball, and baseball, and college athletics administrator. Meagher played football for the University of Notre Dame in 1916, rising to a second-team end under then-assistant coach, Knute Rockne.	Agent	Coach
Seta, addressed as Sojiro Seta in the English-language anime dubs, is a fictional character from the Rurouni Kenshin universe created by Nobuhiro Watsuki. In the story, he is Shishio Makoto's right-hand man.	Agent	ComicsCharacter

### **4.3 Training Base BERT**

Our dataset is a public domain dataset, which requires us to have a model aware of the context within the public domain. We chose the pre-trained base BERT model with 12 x transformers. The base BERT model is trained with Wikipedia dataset to know the context within the public domain.

### **4.4 Multiple BERT Classification**

To perform training for two-level classification, we ran the BERT model individually as two separate instances. The context information is not shared across these models as they are treated independently. While utilizing GPU with 11 GB memory, the training is one at a time due to the hardware limitations. We have chosen Adam as the learning optimizer function with a learning rate of  $1e-05$ . The loss function is implemented with BCE With Logits Loss which has a Sigmoid function, and BCE loss given the task pertains to classification. The optimizer function and the loss function have been chosen as per the recommendations provided with BERT classifications.

### **4.5 Multi-Level Hierarchical BERT**

Application of base BERT model trained in Wikipedia text further adds a thin layer of a custom classifier. For evaluation of the methodology, we have considered two-level classification. Accordingly, two custom classifiers would be used predict values at Level 1 and the Level 2 classifiers. This model undergoes two training pass to predict Level 1 and Level 2 classification outputs. The predicted outputs from the level 1 classifier are shared

to the base BERT model and the original input embeddings. These input embeddings with shared context are further passed into base BERT to pass into the Level 2 classifier.

At each classifier level, BCE with logits is used as a loss function to calculate the loss. This loss calculated at Level 1 and the Level 2 is added up to pass to the Adam optimizer function to learn the weights and bias of the model. This limits the model to have additional attention layers for the Level 2 classifier. It is also important to note that the loss values from the Level 1 classifier are being passed into Level 2 as we share context from predicted outputs, which can be wrongly predicted in the first place.

#### **4.6 Multi-Level Hierarchical BERT Tuned**

The architecture itself is inherited from the Multi-Level Hierarchical BERT. The model is fine-tuned to ensure the training strategy caters to the fact that instead of sharing predicted outputs to the deeper classification levels, we share context using actual outputs. This helps to learn of the model without impacting the loss generated at the initial levels of classification and impacting the deeper classification levels.

It is also important to note, during inference, we cannot apply this strategy as we don't have the actual outputs for the prediction. The inference strategy remains unchanged for this model in comparison with the Multi-Level Hierarchical BERT model. Having arrived at architecture, the model is still limited by dependency on the training passes model has to perform at each level of classification. Despite having the training time reduced, it's still dependent on the number of classification levels of the task.

## 4.7 Multi-Level Feature-Based BERT

The Multi-Level BERT and its tuned version have addressed the growing model sizes and its memory footprint. However, the model limits with regard to training time depending on the number of classification levels the models are to be trained. Multi-Level Feature-Based BERT is designed to solve the multiple classification levels with one training pass. The impact on this training is potentially only at the time of the training. During inference is overlaps with the strategy being followed for other models due to lack of actual outputs.

As an alternative way of addressing the training strategy with one pass, the input embedding along with all the possible shared context at all levels are passed into base BERT models. The features are extracted from the base BERT model and fed into multiple classifiers. The loss is calculated at all levels, and the cumulative values are added as a loss at the Multi-Level classification level. The cumulative loss is passed into the Adam optimizer, so the weights and biases are learned.

## 4.8 Evaluation

The proposed models have been evaluated using a real-time dataset of 311 Kansas City data and a standard dataset as DBPedia. Our research is aimed to reduce the computing footprint by providing a reusable artifact as attention layers, so it's empirical to run comparisons against their resource consumption. Computational Metrics for 311 Kansas City dataset in Table 8 and DBPedia dataset in Table 9.

Analyzing further on the model's metrics, we could see the memory footprint

Table 8: Computational Requirements - 311 Kansas City Dataset

Model Architecture	GPU 2080 Ti(GB)	Model Parameters (Million)	GPU Memory (GB)	Training Mode	Training Time (Min)	Epochs
Multiple BERT	11	2 x 110	2 x 9.7	Sequential	2 x 1200	50
Multi-Level BERT	11	1 x 110	1 x 10	Collaborative	1 x 2200	50
Multi-Level BERT Tuned	11	1 x 110	1 x 10	Collaborative	1 x 2200	50
Multi-Level Feature Tuned	11	1 x 110	1 x 10	Collaborative	1 x 1200	50

Table 9: Computational Requirements - DBPedia Dataset

Model Architecture	GPU 2080 Ti(GB)	Model Parameters (Million)	GPU Memory (GB)	Training Mode	Training Time (Min)	Epochs
Multiple BERT	11	2 x 110	2 x 9.7	Sequential	2 x 750	10
Multi-Level BERT	11	1 x 110	1 x 10	Collaborative	1 x 1400	10
Multi-Level BERT Tuned	11	1 x 110	1 x 10	Collaborative	1 x 1400	10
Multi-Level Feature Tuned	11	1 x 110	1 x 10	Collaborative	1 x 750	10

is reduced and the training time has been reduced. Despite the reduction in memory footprint and computational requirements, accuracy is still deemed the main parameter for assessing the model’s performance. Metrics for accuracy for the 311 Kansas City dataset can be referred to in Table 10, and the DBPedia dataset can be referred at Table 11. In comparison, we have observed, while level 1 matched the accuracy in comparison with individual performance, the reduced accuracy is observed at Level 2, which is still competitive.

The feature-based Multi-Level BERT model has seen worse performance for the 311 Kansas City dataset, as the features are overlapped between several classes. Under the category ‘trash’, the sub-problem is having ‘recycle’ and ‘recycling’, which are confusing for this model. The same model has shown better performance in the streamlined dataset like DBPedia.

As referred to in Table 10 and Table 11, to make a fair comparison, we ran proposed models through the same number of epochs for DBPedia (10 epochs) and 311

Table 10: Accuracy Metrics - 311 Kansas City Dataset

<b>Model Architecture</b>	<b>Level 1-Accuracy</b>	<b>Level 2-Accuracy</b>	<b>Epochs</b>
Multiple BERT	83.40%	65.01%	50
Multi-Level BERT	81.79%	51.26%	50
Multi-Level BERT Tuned	82.06%	54.20%	50
Multi-Level Feature Tuned	30.64%	21.96%	50

Table 11: Accuracy Metrics - DBPedia Dataset

<b>Model Architecture</b>	<b>Level 1-Accuracy</b>	<b>Level 2-Accuracy</b>	<b>Epochs</b>
Multiple BERT	99.36%	98.125%	10
Multi-Level BERT	99.28%	79.88%	10
Multi-Level BERT Tuned	99.39%	83.05%	10
Multi-Level Feature Tuned	88.56%	84.70%	10

Kansas City dataset (50 epochs). However, as we hypothesized model convergence, the Multi-Level model is relatively slower than individual BERT models because the second classification level does not have its attention layers. To prove this, we ran the models further with 20 epochs for DBPedia datasets. Our hypothesis has been proved by results as referred to in Table 12.

Table 12: Accuracy Metrics - DBPedia Dataset Increased Epochs

<b>Model Architecture</b>	<b>Level 1-Accuracy</b>	<b>Level 2-Accuracy</b>	<b>Epochs</b>
Multiple BERT	99.36%	98.125%	20
Multi-Level BERT	99.31%	85.59%	20
Multi-Level BERT Tuned	99.40%	85.43%	20
Multi-Level Feature Tuned	88.56%	84.70%	20

Convergence of the model has been observed, looking at the accuracy measurement against the epochs. As we see trending accuracy improvement over epochs for

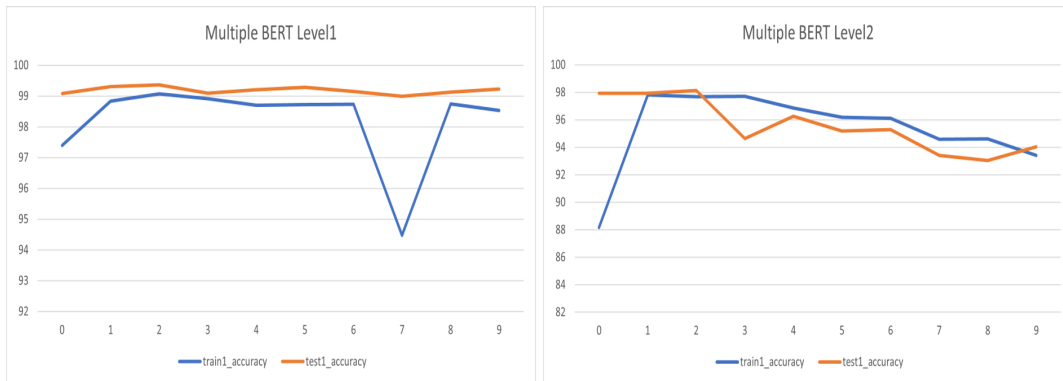


Figure 14: Multiple BERT Accuracy vs. Epochs - Level 1 and Level 2

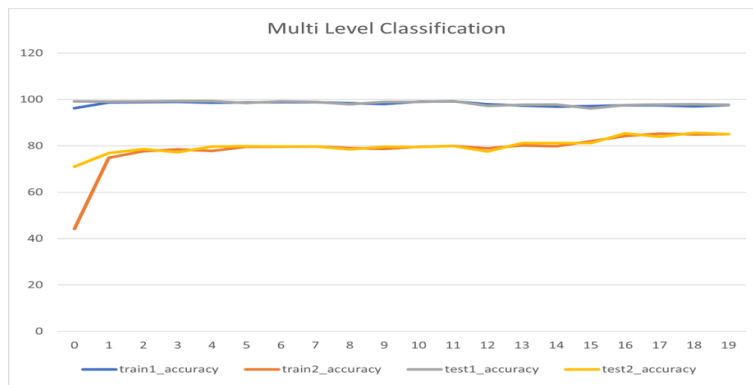


Figure 15: Multi-Level BERT Accuracy vs. Epochs - Level 1 and Level 2

Multi-Level BERT and its Tuned version, we executed the training for the Multiple BERT for ten epochs. Understanding the accuracy is not improved on further epochs. Still, for other proposed models like Multi-Level BERT, Multi-Level BERT tuned, and Multi-Level feature extracted BERT, we extended training to 20 epochs. Accuracy vs. Epochs plots has been derived for various proposed models, and the Multiple BERT can be referred to in Figure 14, Multi-Level BERT can be referred in Figure 15, Multi-Level tuned BERT in Figure 16 and Multi-Level feature extracted in Figure 17

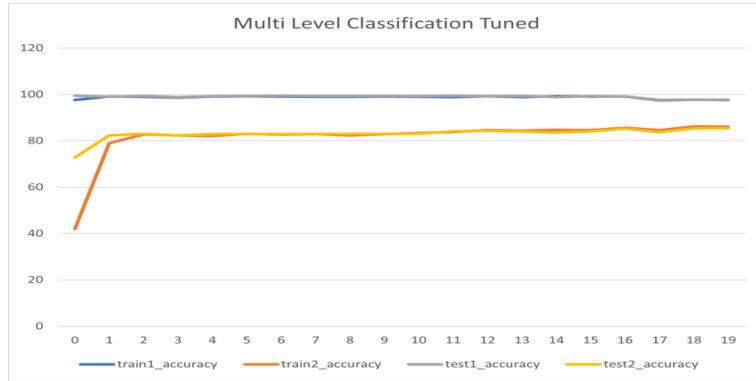


Figure 16: Multi-Level Tuned BERT Accuracy vs. Epochs - Level 1 and Level 2

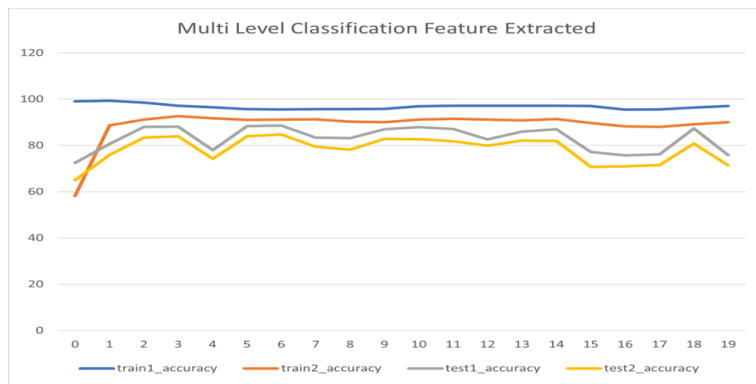


Figure 17: Multi-Level Feature Extracted BERT Accuracy vs. Epochs - Level 1 and Level 2



To understand the model’s performance at each class, we have provided the analysis for classes where the model is less confused and more confused with each specific methodology for the DBPedia dataset. Level 1 class-wise accuracy can be observed in Table 13 and the Level 2 class-wise accuracy can be observed in Table 14. Similar observations have been derived for the 311 Kansas City Dataset with Level 1 class-level accuracy in Table 15 and Level 2 class level accuracy in Table 16

Table 13: Class Level 1 Accuracy - DBPedia Dataset

Level 1 Classes	Flat Multiple BERT	Multi-Level BERT	Multi-level Tuned	Multi-Level Feature BERT
Agent	99.29	99.16	99.11	89.49
Device	100	100	0	100
Event	98.89	98.64	99.51	76.89
Place	99.84	99.76	99.68	99.51
Species	99.95	79.4	99.95	98.03
SportsSeason	96.04	98.39	97.53	49.07
TopicalConcept	96.54	97.28	97.53	0
UnitOfWork	100	100	100	0
Work	99.75	99.81	99.78	96.74

Table 14: Class Level 2 Accuracy - DBPedia Dataset

Level 2 Classes	Flat Multiple BERT	Multi-Level BERT	Multi-level Tuned	Multi-Level Feature BERT
Actor	95.34	94.3	96.63	92.49
AmusementParkAttraction	99.29	99.29	99.53	99.29
Animal	100	0	100	99.74
Artist	98.38	0	99.77	99.77
Athlete	100	100	100	100
BodyOfWater	98.95	97.38	96.59	98.16
Boxer	100	100	0	100

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
BritishRoyalty	99.75	99.75	0	0
Broadcaster	99.74	100	100	99.49
Building	99.74	0	99.74	99.74
Cartoon	87.24	97.66	95.31	89.58
CelestialBody	99.75	99.75	0	99.51
Cleric	98.76	99.01	98.51	97.03
ClericalAdministrativeRegion	99.73	100	100	96.54
Coach	92.04	96.52	93.28	92.79
Comic	95.4	0	92.25	91.04
ComicsCharacter	100	99.75	0	96.05
Company	97.29	97.54	95.81	92.61
Database	100	100	0	90.27
EducationalInstitution	98.56	98.8	98.8	0.96
Engine	100	100	0	100
Eukaryote	99.5	100	99.5	99.25
FictionalCharacter	99.76	99.76	99.76	98.55
FloweringPlant	100	100	100	97.84
FootballLeagueSeason	99.24	99.49	99.49	95.94
Genre	98.27	97.04	97.53	0
GridironFootballPlayer	98.51	95.04	99.01	98.76
Group	96.94	95.56	96.39	96.39
Horse	99.75	0	99.75	93.8
Infrastructure	99.48	0	99.22	99.22
LegalCase	100	100	100	0
MotorcycleRider	100	100	100	99.75

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
MusicalArtist	99.75	99.51	100	99.75
MusicalWork	97.66	98.7	98.18	93.49
NaturalEvent	100	100	100	99.77
NaturalPlace	98.99	98.73	98.99	97.72
Olympics	100	0	100	0
Organisation	86.47	88.41	91.3	36.71
OrganisationMember	79.53	81.35	79.79	83.68
PeriodicalLiterature	99.74	99.74	99.74	0
Person	96.28	91.81	0	83.87
Plant	99.21	99.21	100	98.94
Politician	88.73	93.9	95.07	94.13
Presenter	97.9	99.74	99.74	98.69
Race	99.75	0	98.53	10.57
RaceTrack	99.02	0	99.26	98.53
RacingDriver	99.5	98.75	100	99.75
RouteOfTransportation	98.97	99.74	98.2	99.49
Satellite	100	100	100	99.74
Scientist	95.09	98.45	98.71	91.99
Settlement	98.52	99.01	98.02	99.01
SocietalEvent	99.51	97.56	0	90
Software	99.27	99.27	99.27	97.34
Song	100	100	100	100
SportFacility	98.98	98.73	98.48	98.98
SportsEvent	97.63	98.42	99.47	87.07
SportsLeague	98.29	97.8	97.8	98.78

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
SportsManager	100	99.73	99.2	97.6
SportsTeam	99.02	99.51	99.27	95.61
SportsTeamSeason	97.59	97.11	95.18	3.86
Station	100	99.74	98.16	100
Stream	98.71	99.23	99.23	99.23
Tournament	99.75	99.24	99.75	73.92
Tower	99.31	99.31	99.31	99.54
Venue	99	99	99.5	99.5
VolleyballPlayer	100	100	0	100
WinterSportPlayer	99.77	99.77	100	100
Wrestler	100	100	100	100
Writer	95.71	97.14	97.43	95.71
WrittenWork	98.72	99.49	99.74	94.13

Table 16: Class Level 2 Accuracy - 31 Kansas City Dataset

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
311 Administration	84.62	82.05	82.05	0
ADA	93.33	0	95.56	0
Abandoned	76.92	61.54	53.85	15.38
Abandoned On Street	80.49	63.41	53.66	31.71
Action Center	100	83.33	83.33	0

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Administration	0	0	0	0
Airports	100	79.31	79.31	0
All	50.98	0	0	0
Alley	74.29	42.86	20	65.71
Ambulance	100	100	100	0
Animal	97.44	0	58.97	51.28
Athletics	94.23	100	100	0
Banners	100	100	100	0
Barking Dog	95	95	82.5	92.5
Bite	91.43	71.43	65.71	14.29
Bld / Pkwy	0	21.74	6.52	0
Board Up	0	56.41	92.31	5.13
Boulevards and Parkways	100	76.6	100	0
Broken Asphalt	62.5	17.5	30	0
Brush	100	61.36	81.82	0
Building / Property	72.97	0	0	2.7
Bulky	87.5	35	90	12.5
Bulky Pick Up	92.86	78.57	54.76	57.14
Capital Projects Office	86.84	44.74	0	0
Care	87.8	58.54	60.98	2.44
Catch Basin	0	20.51	0	2.56
City	0	51.28	15.38	0
City Managers Office	95.65	84.78	89.13	26.09
City Property	100	0	85.71	0
Cleaning	0	0	0	0

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Commercial Parking	100	93.75	93.75	0
Commercial Signs / Ads	79.41	58.82	64.71	11.76
Commercial Vehicle	84.21	78.95	73.68	50
Communicable Disease	100	100	0	100
Communications	100	0	100	0
Community Center	0	87.23	80.85	38.3
Community Service	100	100	100	54.17
Construction Issue/Concern	0	53.06	28.57	0
Construction Repair	100	100	100	0
Contractor	85.71	0	65.71	62.86
Contractor Restoration	0	41.67	25	33.33
Crack	74.42	74.42	76.74	18.6
Cruelty or Neglect	79.07	34.88	62.79	0
Culvert	65.91	59.09	77.27	70.45
Curb Box	100	100	100	100
Customer Service	100	0	0	0
Cut / Permit	0	0	3.92	0
Damage	79.41	67.65	0	23.53
Damage / Dis-Repair	0	50	0	0
Dangerous Building	86.36	70.45	2.27	0
Data Not Available	15.62	0	0	0
Dead Animal	0	0	31.43	0
Disabled/Unlicensed on Private Property	66.67	56.41	69.23	7.69
Disease	97.73	68.18	93.18	0

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Disease Control	61.9	0	83.33	0
Disrepair	69.57	0	65.22	2.17
Ditch	65.12	0	0	0
Dumping	25.81	61.29	51.61	0
Dumpsters	73.68	52.63	71.05	68.42
Early Set Out	97.5	75	65	35
Elevator Inspection	100	100	100	0
Emerald Ash	0	64.71	58.82	2.94
Emerald Ash Borer	100	74.19	83.87	6.45
Emergency Management	100	100	100	73.53
Encroachment	48.72	0	0	2.56
Engineering	58.54	34.15	0	0
Facilities / Attractions	85.29	73.53	85.29	17.65
Fire	84.62	0	0	12.82
Flood Barricade	0	95.24	97.62	2.38
Food Establishment	75	84.38	62.5	9.38
Graffiti	100	0	74.36	7.69
Guardrail	0	82.86	71.43	2.86
Hotel / Motel	94.87	89.74	87.18	7.69
Hydrant Repair	100	97.14	77.14	100
IT / Websites	100	100	100	0
Icing	100	100	84.62	66.67
Illegal Dumping	73.17	43.9	53.66	73.17
Illegally Parked	100	100	100	100
Improved Channel	0	60.53	60.53	55.26

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Industrial	100	100	100	100
Injury or Cruelty involving an Animal	59.62	25	0	11.54
Insect Problem	100	100	100	0
Intersection	100	83.78	91.89	0
Investigation	40.48	0	2.38	0
Investigations	62.16	0	16.22	0
KC Police	100	100	23.64	0
Laboratory Service	0	96.67	63.33	73.33
Land Bank	63.16	28.95	39.47	0
Land Development	100	64.71	0	0
Land Trust	81.25	0	0	0
Land Use / Zoning Issue	72.5	55	0	0
Landlord Set Out	100	90.62	90.62	56.25
Landscaping	83.33	46.67	53.33	13.33
Law	100	0	100	0
Lead Poisoning Prevention	100	100	100	0
Leaf and Brush	100	53.85	76.92	84.62
Leaf / Brush	0	78.95	81.58	5.26
Leak	67.44	83.72	65.12	65.12
MFS Referral (Meter Field Services)	40	0	5.71	0
Maintenance	0	0	0	0
Malfunction	72.09	88.37	46.51	2.33
Manhole	88.64	86.36	88.64	0



<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Markings / Paint	77.5	72.5	85	0
Meter	94.74	100	97.37	0
Missing	0	70	63.33	6.67
Municipal Court	100	100	100	0
NHS (Neighborhood Housing Services)	70.73	58.54	0	31.71
Neighborhood Abatement Program	100	100	100	34.15
New	100	100	100	0
New Request	80	71.43	71.43	14.29
No Dumping Sign	90.32	87.1	90.32	0
No Water / Pressure	83.87	83.87	96.77	0
Noise	78.72	89.36	0	0
Not Provided By KCMO	28.57	0	0	2.38
Nuisance	38.24	26.47	14.71	0
Obstructed / Closure	65	47.5	50	0
Other-Maintenance	0	0	0	0
Outage	0	43.18	75	40.91
Outdoor Air Quality	62.5	0	0	0
Owned or Stray at Large	62.5	0	32.5	25
PIAC	0	18.6	18.6	0
Paper Street	100	87.1	67.74	16.13
Park Maintenance	35.71	0	0	11.9
Park Property	100	93.02	93.02	0

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Parked on Unapproved Surface	70	62.5	52.5	45
Parking	100	0	100	0
Parking Lot	95.45	84.09	0	0
Parking Meter	100	47.73	0	0
Parks	53.12	6.25	25	0
Permit	95	70	60	2.5
Permit / License	74.29	65.71	68.57	62.86
Pipeline Referral	94.74	92.11	73.68	60.53
Pipeline Repair	82.05	2.56	0	23.08
Pipeline Restoration Concerns	93.18	68.18	31.82	20.45
Planning	100	0	100	0
Planting	100	84.44	95.56	0
Plate	84.62	87.18	74.36	7.69
Police	0	0	47.5	2.5
Pollutants	100	94.59	94.59	0
Pool / Hot Tub	100	96.67	93.33	16.67
Pothole	86.54	92.31	90.38	21.15
Priority	0	0	30.77	0
Private / Commercial	34.29	65.71	60	14.29
Private Property	20.93	4.65	0	0
Property Maintenance	0	0	12.2	0
Public Facilities	100	100	100	48.48

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Public Improvement Advisory Committee	0	51.35	21.62	0
Public Property	44.74	10.53	2.63	5.26
Public Restroom	100	97.56	100	0
Public Works	0	0	0	0
Quality	0	77.36	67.92	0
Questionable Activity	39.39	42.42	18.18	36.36
Rat Control and Treatment	100	85.37	0	0
Rat Treatment	0	76.6	82.98	0
Recycle	95.83	87.5	50	0
Recycling	89.74	66.67	87.18	0
Regulated Industries	70.45	0	50	0
Removal	0	0	4.88	0
Repair	0	75.76	78.79	81.82
Replacement	95.35	72.09	72.09	27.91
Restoration	100	100	100	85.71
Resurfacing	78.95	63.16	73.68	13.16
Return Call to Citizen	87.5	78.12	81.25	0
Right of Way	57.14	0	28.57	0
Right of Way (ROW)	0	0	0	0
Scooter	100	100	100	35.56
Services	0	25	0	0
Services (CPD)	0	52.78	50	0
Services (NPD)	0	47.62	30.95	0
Sewer	43.24	0	0	21.62

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Sewer Backup / Leak	88.89	70.37	81.48	0
Sewer Odor	85	85	67.5	2.5
Sewer Referral	56.52	43.48	0	0
Sinkhole	54.76	33.33	19.05	59.52
Sinkhole Referral	47.62	0	0	0
Smoking / Tobacco	100	91.18	91.18	17.65
Snow / Ice	87.88	78.79	72.73	0
Solid Waste Operations	100	89.19	100	56.76
Speed Bump	94.74	92.11	94.74	5.26
Storm Damage	76.47	82.35	58.82	0
Storm Drain / Catch Basin	75	72.5	30	60
Storm Water	51.22	0	7.32	21.95
Storm Water Referral	0	0	64.58	70.83
Stray	76.47	47.06	50	38.24
Stray Confined	77.78	81.48	66.67	7.41
Street	0	0	0	0
Street Clean Up	63.16	0	0	21.05
Street Light	97.5	92.5	90	0
Street Name Signs	83.72	32.56	16.28	72.09
Street Preservation	100	97.62	61.9	0
Street Services	0	0	0	0
Street Sweeping	91.84	75.51	85.71	10.2
Street and Traffic	0	41.18	0	14.71
Stump Removal	0	87.18	89.74	0
Ticket	100	100	100	61.54

<b>Level 2 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Tobacco	100	100	100	0
Tow Services	92.31	97.44	97.44	0
Traffic Permit	92.59	0	0	7.41
Traffic Sign	65.62	12.5	40.62	50
Traffic Signal	75.61	19.51	78.05	34.15
Traffic Signs	100	70.27	83.78	0
Traffic Study	0	0	2.38	4.76
Trails	86.67	84.44	62.22	44.44
Trash	100	100	100	0
Trash Cart	100	97.3	100	10.81
Trash Collection	76.47	70.59	76.47	35.29
Treatment	100	100	100	100
Trimming	73.17	31.71	43.9	0
Unapproved Objects	50	38.24	44.12	14.71
Valve Repair	0	86.36	86.36	0
Visibility	76.09	34.78	45.65	0
Water	39.58	0	0	0
Water Main Repair	100	100	100	100
Wildlife	86.21	65.52	82.76	3.45

Table 15: Class Level 1 Accuracy - 311 Kansas City Dataset

<b>Level 1 Classes</b>	<b>Flat Multiple BERT</b>	<b>Multi-Level BERT</b>	<b>Multi-level Tuned</b>	<b>Multi-Level Feature BERT</b>
Animals	98.43	97.06	93.96	38.83
City Facilities	0	71.76	43.85	0.33
Data Not Available	0	0	0	0
Legal	0	74.86	74.31	9.12
Lights/Signals/Signs	96.85	80.69	79.75	32.4
Maintenance	0	0	0	0
Neighborhood	0	21.43	21.43	0
Noise	0	0	0	0
Other	57.07	59.57	55.74	16.31
Parking	0	27.85	50.63	15.19
Parks and Recreation	0	83.07	81.25	26.3
Property Violations	53.68	73.06	77.22	4.88
Public Health	75.59	89.85	88.21	12.54
Public Safety	89.28	88.95	89.73	5.33
Street/Sidewalks	85.79	83.27	82.45	19.2
Traffic	0	0	0	4.65
Trash	93.58	90.06	92.74	71.34
Vehicles	93.16	97.42	97.16	35.14
Water	94.7	90.27	94.13	68.96

## CHAPTER 5

### CONCLUSION AND FUTURE WORK

#### 5.1 Conclusion

The model as believed presents an alternative methodology to the application of multiple BERT models which presents huge limitations with regards to physical infrastructure. The computing resources have been optimized for the usage of the Multi-Level classification. Understandably, Accuracy has been a trade-off factor from the experiments ran the model has seen a decrease in size of about 110 million parameters per classification level. The accuracy however has implied a competitive value in comparison with individual BERT models. The experiments have also proved the point of having one attention layer over attention layers at each classification has shown faster convergence of initial levels of classification.

#### 5.2 Future Work

The methodology has been applied to the real-world dataset with a high possibility of noise and irregularity on the features. The methodology has also been proven with help of the DBpedia dataset. To further the research the model will be evaluated against other standard datasets. The technique will also be exposed to datasets from a different domain to a general domain like medical and observe the performance. Despite having the individual training strategy at different layers for the classification, the sequence of

operations follows a current classifier context to the next level, However, we are planning to pursue if the context can be shared from the next classification layer to correct the prediction at current classification level. The current demonstration of the paper is about the classification, however, adopting this technique to other types of tasks like question answering should also be attempted to further the research.



## REFERENCE LIST

- [1] Adam Conner-Simons, M. C. The computational limits of deep learning. <https://www.csail.mit.edu/news/computational-limits-deep-learning#:~:text=A%20new%20project%20led%20by,towards%20techniques%20that%20are%20more,2020>. [Online].
- [2] Aksoy, C., Ahmetoglu, A., and Gungor, T. Hierarchical Multitask Learning Approach for BERT, 2020.
- [3] Alammam, J. The Illustrated Transformer. <http://jalammar.github.io/illustrated-transformer/>, 2018. [Online].
- [4] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (Minneapolis, Minnesota, June 2019), Association for Computational Linguistics, pp. 4171–4186.
- [5] Hochreiter, S., and Schmidhuber, J. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

- [6] Lu, W., Jiao, J., and Zhang, R. TwinBERT: Distilling Knowledge to Twin-Structured BERT Models for Efficient Retrieval, 2020.
- [7] Maxime. What is a Transformer. <https://medium.com/inside-machine-learning/what-is-a-transformer-d07ddlfbec04>, 2019. [Online].
- [8] Pudipeddi, B., Mesmakhosroshahi, M., Xi, J., and Bharadwaj, S. Training Large Neural Networks with Constant Memory using a New Execution Algorithm. *arXiv* (June 2020).
- [9] Seth, Y. BERT Explained. <https://yashueth.blog/2019/06/12/bert-explained-faqs-understand-bert-working/>, 2019. [Online].
- [10] Si, Y., and Roberts, K. Hierarchical Transformer Networks for Longitudinal Clinical Document Classification, 2021.
- [11] Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., and Zhou, D. MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices, 2020.
- [12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems* (2017), I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, Curran Associates, Inc.

## VITA

Charan Tej Thota is an experienced professional with about 11 years in the Information technology industry, solving complex business problems. His expertise is spread over different domains like Software Engineering, Data Analytics, and Data Engineering in enterprise and research projects with other firms. He is currently working as Lead Data Scientist in Rx Savings Solutions in Kansas City. He has been associated with multiple firms like Mckinsey and Company, Deloitte, and TCS as a Subject Matter Expert, Technology Leader, and Strategist. Charan joined the Master's program at UMKC in Fall 2019 in Computer Science to pursue research in Machine learning and Deep learning. During his time at UMKC, he has conducted multiple research projects under Dr. Yugyung Lee and has been actively contributing to several projects. As a research Intern at T-Mobile, he contributed to patented work (IDF 12784US01 - 4300-89800, currently under external review) based on deep learning models to review the word documents