



UNLEASHING POTENTIAL: HARNESSING FOUNDATION MODELS (LARGE LANGUAGE MODELS) IN BUSINESS AND RESEARCH

FATEMEH H. FARD
ASSISTANT PROFESSOR, UNIVERSITY OF BRITISH COLUMBIA





Announcing a new wave of innovation in Bing Chat and SwiftKey.

[Learn more](#)

What will you do with Bing?

[Try Bing now](#)

Prepare for a pet



Make a meal



Plan a vacation



Create a sports
image



Decorate an
apartment



⚡ GPT-3.5

⚡ GPT-4 🔒

ChatGPT



Tell me a fun fact

about the Roman Empire

Design a database schema

for an online merch store

Write a thank-you note

to my interviewer

Give me ideas

for what to do with my kids' art

Send a message



Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts. [ChatGPT September 25 Version](#)



As of my last update in September 2021, the Okanagan region in British Columbia, Canada, is known for its diverse economy, with several industries contributing significantly to the local economy. While the economic landscape can change over time, tourism, agriculture, and technology were some of the key industries in the Okanagan region:

1. **Tourism:** The Okanagan Valley is a popular tourist destination known for its wineries, vineyards, orchards, and scenic landscapes. Tourism-related activities, including wine tours, outdoor recreation, and cultural events, contribute significantly to the local economy.
2. **Agriculture:** Agriculture is a major industry in the Okanagan, with a focus on vineyards, wineries, fruit orchards, and vegetable farms. The region is renowned for its wine production, especially in areas like Kelowna and Penticton.
3. **Technology:** The technology sector has been growing in the Okanagan, with an emphasis on software development, digital media, and technology startups. Kelowna, in particular, has seen the emergence of a vibrant technology community.
4. **Healthcare and Education:** The presence of healthcare institutions and educational facilities, including the University of British Columbia Okanagan (UBCO), contributes significantly to the local economy.
5. **Manufacturing and Aerospace:** The Okanagan region has a diverse manufacturing sector, including aerospace and defense manufacturing. Several aerospace companies operate in the area.



Regenerate



F

Write 200 words about productivity



Productivity, often regarded as the cornerstone of progress and efficiency, encompasses the ability to achieve maximum output with optimal resource utilization. It's not merely about being busy but rather about focusing on tasks that matter, efficiently managing time, and maintaining a balance between quality and quantity. Effective productivity involves setting clear goals, prioritizing tasks, and organizing one's workflow. It thrives on discipline, concentration, and the ability to minimize distractions, allowing individuals to channel their energy into meaningful endeavors. Time management techniques, such as the Pomodoro method or time blocking, aid in enhancing productivity by breaking work into manageable intervals, fostering intense focus followed by short breaks.

Moreover, productivity is deeply intertwined with motivation and well-being. When individuals find purpose and satisfaction in their work, they tend to be more productive. Employers play a crucial role in this aspect, fostering a positive work environment, recognizing achievements, and encouraging innovation. Embracing technology and automation also enhances productivity, streamlining repetitive tasks and enabling humans to focus on creative problem-solving and strategic thinking.

On a broader scale, societal productivity drives economic growth and development. Nations with productive workforces are more competitive globally, leading to higher living standards and overall prosperity. However, it's essential to emphasize that productivity should not come at the cost of well-being. Balancing productivity with mental and physical health, leisure, and quality time with loved ones ensures a holistic and sustainable approach, fostering a fulfilling life both personally and professionally.

Regenerate

F

Write 200 words about productivity

F

Write 200 words about productivity for graduate students in university



For graduate students in university, productivity is not just a buzzword but a vital skill that can significantly impact their academic and professional journey. Effectively managing time and tasks is crucial as students juggle rigorous coursework, research projects, and personal commitments. Setting clear, achievable goals is the first step. Breaking down large tasks into smaller, manageable steps not only makes the workload seem less daunting but also provides a sense of accomplishment with each milestone achieved.

Prioritization is key; identifying tasks based on deadlines and importance helps allocate time and energy efficiently. Creating a well-organized schedule, incorporating study sessions, research work, and self-care, can prevent burnout and enhance overall productivity. Additionally, utilizing productivity tools and techniques, such as digital planners, time management apps, and the Pomodoro technique, can optimize study sessions by enhancing focus and concentration.

Moreover, fostering a healthy work-life balance is essential. Adequate rest, exercise, and social interactions rejuvenate the mind and enhance cognitive functions, ultimately boosting productivity. Seeking support from professors, peers, or university resources when faced with challenges can also save valuable time and prevent unnecessary stress.

Furthermore, engaging in regular self-assessment and adapting study strategies based on what works best for individual learning styles is fundamental. Ultimately, productivity for graduate students is not just about academic success but also about honing skills that will be invaluable in their future careers, emphasizing the importance of time management, resilience, and adaptability.

Regenerate



F

Write 200 words about productivity for graduate students in university

Prompt Engineering

1. Offer context
2. Include helpful information upfront
3. Give examples
4. Tell it the length of the response you want
5. Define the expected formats
6. Ask it to help you come up with a prompt
7. Use some of these handy expressions
 - "Let's think step by step"
 - "Thinking backwards"
 - "In the style of [famous person]"



Prompt Engineering

1. Give examples

F

Example:

Input: 2020-08-01T15:30:00Z

Add 3 days and convert the following time stamp into

Output: Aug/04/2020 15:30:00

Input: 2020-07-11T12:18:03.934Z

Output:

Source: <https://zapier.com/blog/gpt-prompt/>



To achieve this, you can use a programming language like Python. Here's an example of how you can add 3 days to a given timestamp in the format "YYYY-MM-DDTHH:MM:SSZ" and convert it into the specified format "MMM/DD/YYYY HH:MM:SS":

python

Copy code

```
from datetime import datetime, timedelta

def convert_timestamp(input_timestamp):
    # Parse the input timestamp string to a datetime object
    input_datetime = datetime.strptime(input_timestamp, '%Y-%m-%dT%H:%M:%SZ')

    # Add 3 days to the input datetime
    new_datetime = input_datetime + timedelta(days=3)

    # Format the output timestamp as required
    output_timestamp = new_datetime.strftime('%b/%d/%Y %H:%M:%S')

    return output_timestamp

# Example usage
input_timestamp1 = '2020-08-01T15:30:00Z'
output_timestamp1 = convert_timestamp(input_timestamp1)
print('Output 1:', output_timestamp1)

input_timestamp2 = '2020-07-11T12:18:03.934Z'
output_timestamp2 = convert_timestamp(input_timestamp2)
print('Output 2:', output_timestamp2)
```

FAILURES?



LLMs Failure

- Bias
- Ethics
- Logics
- Hallucination
- Lack of knowledge
- And many others

LLM failure archive (ChatGPT and beyond)

A repo containing failure cases for ChatGPT and similar models, with the purpose of fur useful for:

- comparison with other models
- creation of synthetic data for testing and/or training



Jeremy Howard
@jeremyphoward

Oh I forgot to mention - one other thing that I discuss in some detail (with examples) is what LLMs currently can do, and what they can't. This, from [@snewmanpv](#), is a great example of a GPT 4 mistake, which I use to explain LLM failure modes:



Here is a logic puzzle. I need to carry a cabbage, a goat, and a wolf across a river. I can only carry one item at a time with me in the boat. I can't leave the goat alone with the cabbage, and I can't leave the cabbage alone with the wolf. How can I get everything the other side of the river?



Certainly. Here's a step-by-step solution to this classic river crossing puzzle:

1. First, take the goat across the river and leave it on the other side.
2. Go back alone to the original side and take the cabbage (or the wolf, the solution would still work if these steps are switched) across the river.
3. Leave the cabbage on the other side and take the goat back with you to the original side.
4. Leave the goat on the original side and take the wolf across the river.
5. Leave the wolf with the cabbage on the other side and return alone to the original side.
6. Take the goat across the river one final time.

This way, the goat is never left alone with the cabbage, and the cabbage is never left alone with the wolf, successfully solving the puzzle.



WHAT ARE LANGUAGE MODELS?



ty|



0.0148257168% type

0.0000606981% tympanic

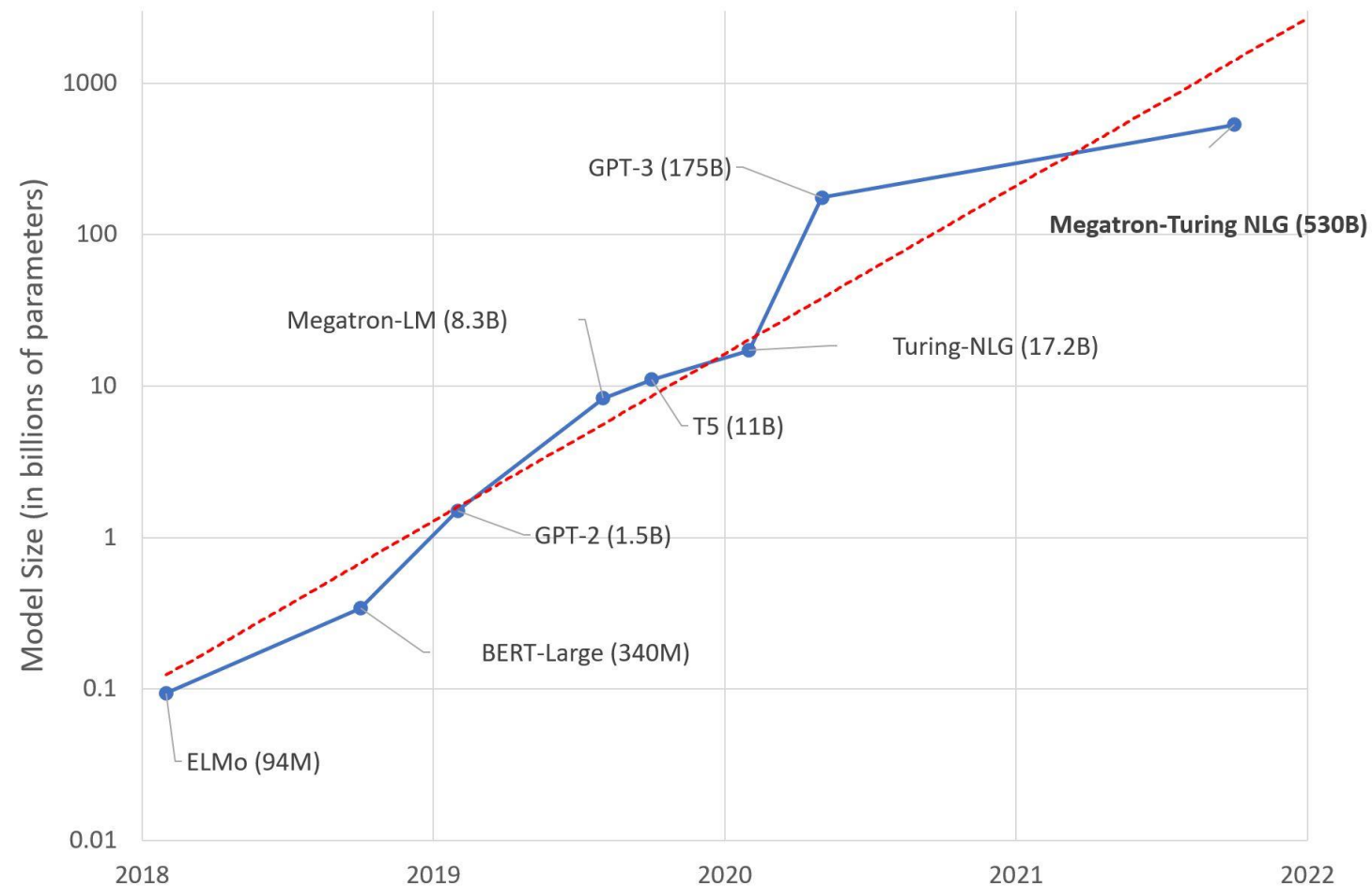
0.0000397285% typography

0.0000396270% typhoon

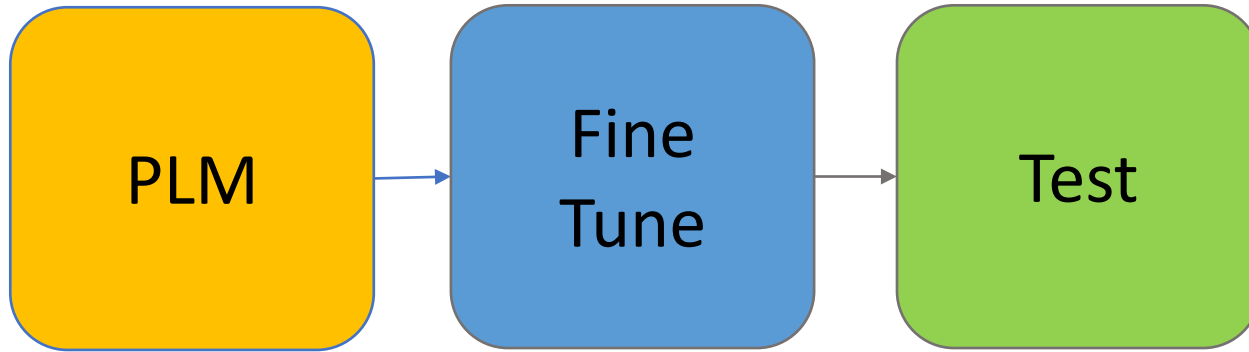
0.0000374782% tycoon

0.0000052224% tye





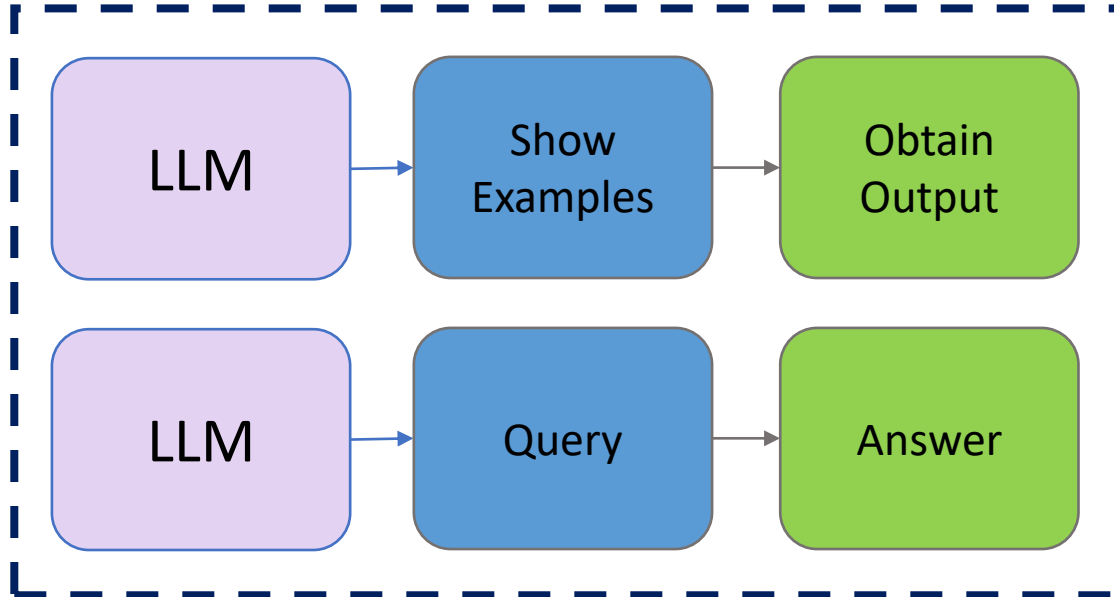
Finetuning Pre-trained Language Models (PLMs)



Finetuning Large Language Models (LLMs)



Expensive and requires GPUs



Examples seen so far





what is llama2?



As of my last update in September 2021, I don't have specific information about "llama2." It's possible that it could be a product, a software, a game, a code name, or something else entirely. If it's a recent development or a niche topic, I recommend checking the latest online sources, websites, or community forums for the most current information. If you provide more context or specify what you're looking for, I'd be happy to help further!





what is llama2 model?



I don't have specific information about a "llama2" model as of my last update in September 2021. It's possible that this model was released or became prominent after that time.



If "llama2" is a specific model, it could refer to a variety of things, such as a machine learning model, a software version, or a specific technology. If you're referring to a machine learning model developed by OpenAI or any other organization, I recommend checking the official OpenAI website or the relevant organization's resources for the most up-to-date and accurate information. If you provide more context or specify the field or application you're interested in, I can offer more targeted assistance.



F

what is llama2 language model?



As of my last update in September 2021, I don't have specific information about a "llama2" language model. It's possible that this model was developed or released after that time, or it could be a specialized or niche model that is not widely known in the general public.



Language models, like OpenAI's GPT-3.5, are designed to understand and generate human-like text based on the input they receive. These models have a wide range of applications, including chatbots, language translation, content generation, and much more. If "llama2" is a specific language model, I recommend checking the official website or resources of the organization or research group that developed it for the most accurate and updated information.



Regenerate



F

llama2 is a language model from Meta. Can you give me some updates about it?



I'm sorry for any confusion, but as of my last update in September 2021, I don't have specific information about a "llama2" language model developed by Meta (formerly known as Facebook). It's possible that this model was developed or released after that time, and I wouldn't have access to information about developments or products that occurred after my last update.

For the latest and most accurate information about the "llama2" language model from Meta, I recommend checking Meta's official website, blog, or press releases. These sources are likely to provide the most recent updates and details about their projects and initiatives.





In software development, a **code clone** refers to a section of source code that is identical or very similar to another section of code within the same codebase or across different codebases. Code clones occur when developers copy and paste code, resulting in duplicated or highly similar code fragments.

Code clones can be categorized into different types:

1. **Type 1 Clones (Exact Clones):** These are identical code fragments found in multiple places within the codebase. They have the same sequence of statements, expressions, and whitespace.
2. **Type 2 Clones (Near Miss Clones):** These are similar code fragments with minor differences, such as variations in variable names, whitespace, or comments. Type 2 clones are often the result of copy-pasting and then making slight modifications.
3. **Type 3 Clones (Semantic Clones):** These clones perform similar tasks but might be implemented differently. They have the same or similar behavior but different syntax or structure. Detecting semantic clones requires a deeper analysis of the code's functionality.

Code cloning can lead to several issues in software development:

- **Maintenance Challenges:** Changes or bug fixes need to be applied to multiple places, leading to higher maintenance efforts and increased chances of inconsistencies.
- **Code Bloat:** Duplicated code increases the size of the codebase, making it harder to understand, manage, and maintain.
- **Bug Propagation:** If a bug exists in a cloned section of code, it can propagate to multiple places, making it challenging to identify and fix all instances.

Python program for implementation of Bubble Sort

```
def bubbleSort(arr):
    n = len(arr)
    # optimize code, so if the array is already sorted, it doesn't need
    # to go through the entire process
    swapped = False
    # Traverse through all array elements
    for i in range(n-1):
        # range(n) also work but outer loop will
        # repeat one time more than n times
        # Last i elements are already in place
        for j in range(0, n-i-1):

            # traverse the array from 0 to n-i-1
            # Swap if the element found
            # than the next element
            if arr[j] > arr[j + 1]:
                swapped = True
                arr[j], arr[j + 1] = arr[j + 1], arr[j]

        if not swapped:
            # if we haven't needed to make a single swap, we
            # can just exit the main loop.
            return
```



The provided Python code implements the Bubble Sort algorithm with an optimization. The optimization involves introducing a boolean variable `swapped` to keep track of whether any swaps were made during a pass through the array. If no swaps are made in a pass, it means the array is already sorted, and the algorithm can terminate early.




```
def insertionSort(arr):  
    n = len(arr) # Get the length of the array  
  
    if n <= 1:  
        return # If the array has 0 or 1 element, it is already sorted, so return  
  
    for i in range(1,  
        key = arr[i] #  
        j = i-1  
        while j >= 0 &  
            arr[j+1] = a  
            j -= 1  
            arr[j+1] = key
```



The provided code is an implementation of the Insertion Sort algorithm in Python. Insertion Sort is a simple sorting algorithm that builds the final sorted array one item at a time. It is much less efficient on large lists than more advanced algorithms such as quicksort, heapsort, or merge sort. However, it has several advantages such as its simplicity and adaptability, especially for small datasets or nearly sorted datasets.

Here's how the algorithm works:

↺ Regenerate



Are these two code clones?

Python program for implementation of Bubble Sort

```
def bubbleSort(arr):
    n = len(arr)
    # optimize code, so if the array is already sorted
    # to go through the entire process
    swapped = False
    # Traverse through all array elements
    for i in range(n-1):
        # range(n) also work but outer loop will
        # repeat one time more than needed.
```

```
def insertionSort(arr):
```

```
    n = len(arr) # Get the length of the array
```

```
    if n <= 1:
```

```
        return # If the array has 0 or 1 element, it is already sorted
```

```
    for i in range(1, n): # Iterate over the array starting from index 1
```

```
        key = arr[i] # Store the current element as the key
```

```
        j = i-1
```

```
        while j >= 0 and key < arr[j]: # Move elements greater than key one position ahead
```

```
            arr[j+1] = arr[j] # Shift elements to the right
```

```
            j -= 1
```

```
        arr[j+1] = key # Insert the key in the correct position
```



No, the provided code snippets are not code clones. They are two different sorting algorithms implemented in Python: Bubble Sort and Insertion Sort. Both algorithms are different in terms of their implementation and sorting logic.

- **Bubble Sort:** The first snippet implements the Bubble Sort algorithm with an optimization to check if the array is already sorted. It uses nested loops to compare adjacent elements and swap them if they are in the wrong order. If no swaps are made in a pass, the algorithm terminates early as the array is already sorted.
- **Insertion Sort:** The second snippet implements the Insertion Sort algorithm. It iterates through the array, considering one element at a time and inserting it into its correct position among the previously sorted elements. It shifts elements greater than the current key to the right to make space for inserting the key in the correct position.

Both sorting algorithms have different approaches and mechanisms, making them distinct and not code clones of each other.

Can the following two codes have same inputs and outputs?

Python program for implementation of Bubble Sort

```
def bubbleSort(arr):
    n = len(arr)
    # optimize code, so if the array is already sorted,
    # to go through the entire process
    swapped = False
    # Traverse through all array elements
    for i in range(n-1):
        # range(n) also work but outer loop will
        # repeat one time more than needed.
```

```
def insertionSort(arr):
```

```
    n = len(arr) # Get the length of the array
```

```
    if n <= 1:
```

```
        return # If the array has 0 or 1 element, it is already sorted
```

```
    for i in range(1, n): # Iterate over the array starting from index 1
```

```
        key = arr[i] # Store the current element as the key to be inserted
```

```
        j = i-1
```

```
        while j >= 0 and key < arr[j]: # Move elements greater than key one position ahead
```

```
            arr[j+1] = arr[j] # Shift elements to the right
```

```
            j -= 1
```

```
        arr[j+1] = key # Insert the key in the correct position
```



Yes, both Bubble Sort and Insertion Sort can have the same inputs and outputs, but they sort the array in different ways.

In the provided code, the input array is `[12, 11, 13, 5, 6]`.

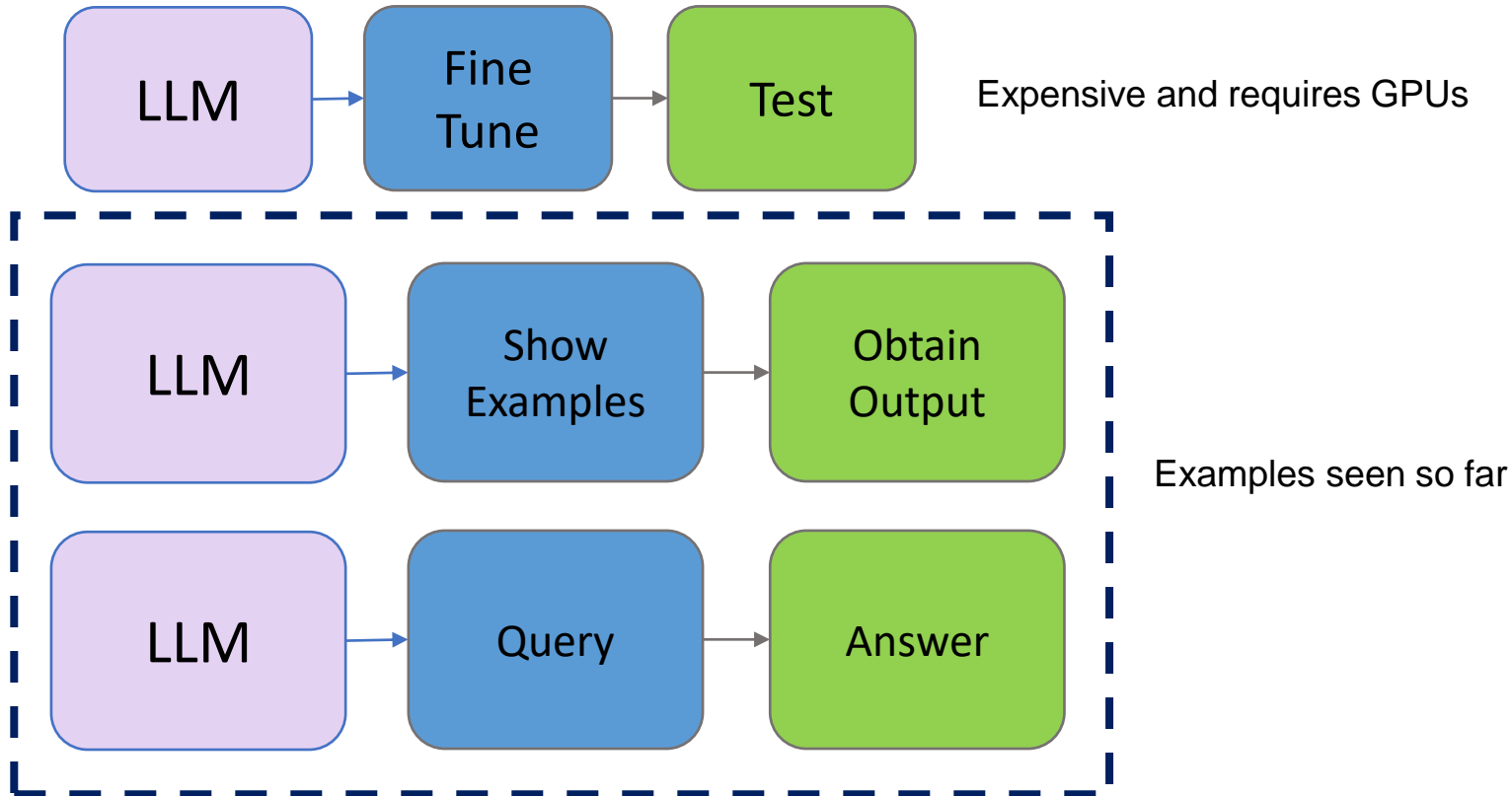
After Bubble Sort, the output array will be `[5, 6, 11, 12, 13]`. Bubble Sort works by repeatedly swapping adjacent elements if they are in the wrong order.

After Insertion Sort, the output array will also be `[5, 6, 11, 12, 13]`. Insertion Sort works by considering one element at a time and inserting it into its correct position among the previously sorted elements.

Both sorting algorithms produce the same sorted output for the given input array. However, the algorithms use different methods to achieve the sorting, making them distinct in their approaches.



Finetuning Large Language Models (LLMs)



Challenges



Many new domains

- Domain specific data
- Domain specific data structure and format

Prompt Engineering

Non-generative tasks (e.g., classification)

Data availability: Zero or few examples

Low resource languages

Inaccurate results are not acceptable

Natural Language Processing 4 Software Engineering



User Feedback Analysis

- User reviews, online discussions

Empirical Studies, MSR

- Data mining
- Finding the reasons, relations, extracting new knowledge

Source Code Rep. Learning

- Comment/Code generation
- Code clone detection

Transfer Learning

- Transferability of the programming languages
- Few shot learning
- Knowledge transfer among tasks/languages using less computational resources



Challenges



Many new domains

- Domain specific data
- Domain specific data structure and format

App Review Classification

Ratings and reviews →

Ratings and reviews are verified ⓘ



Rachel Bronson

★☆☆☆☆ August 23, 2023

I love the app. So much fun. Kept my girls busy for hours while they were able to use it. I am willing to update my review to 5 stars after they fix the issue I'm having. I cancelled the subscription so I wouldn't forget about it and be charged with auto renewal a year from now. The subscription still says it goes through July of 2024-- but it won't let me use the app unless I "resubscribe" and I have to pay another \$30 (on top of the \$42 I already paid for the whole year)! WHAT?!

654 people found this review helpful

Did you find this helpful?

Yes

No



T Graham



★★★★★ September 2, 2023

I love this app! works much better than the app on my partner's i phone. It is great at identifying plants and is good at diagnosing problems too I was really surprised! I also like the fact that I can save the plants that are in my garden specifically so I can learn more about them and research them. Suggestion to Devs-I just wish I could also save a wish list of plants for the future. but I love this! I have tested It against other apps as well as Google reverse image search.

Phone

Tablet

Chromebook

4.6



492K reviews



Challenges With App Review Classification



New Domains (e.g., Security)

New Distributions (e.g., Twitter)

Availability of Dataset

- Time consuming
- Costly
- Imbalanced classes

Multiple Tasks (e.g., Sentiment classification)

Questions



RQ1: Comparing PLMs with MLs

RQ2: Comparing domain specific PLMs with general PLMs

RQ3: Comparing PLMs with MLs for different settings

- Binary vs multi-class setting,
- Zero-shot classification,
- Multi-task setting (i.e. different app-review analysis tasks),
- Classification of user-reviews collected from different resources (i.e., Twitter, App Store).

Challenges with this data

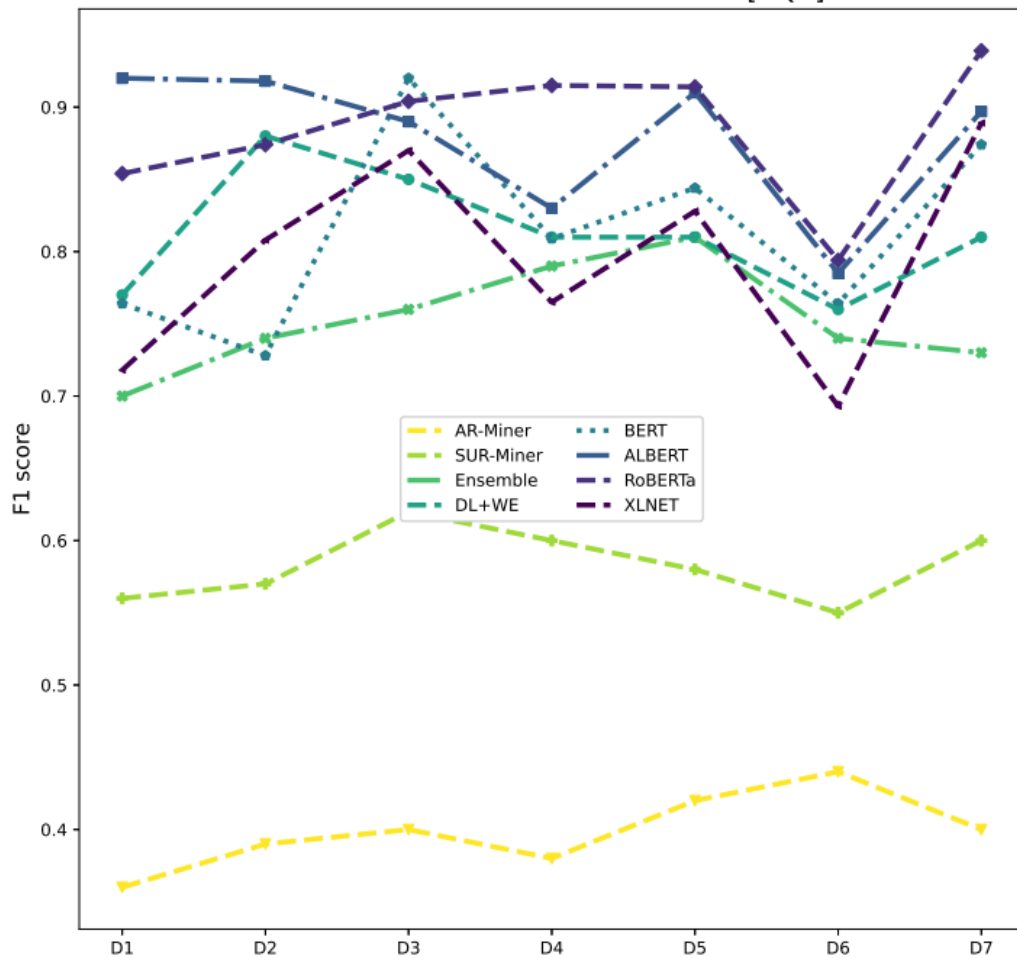
- 16 different labels
- Highly imbalanced classes

Labels in Merged Dataset	Grouped Labels [Label Name: Dataset Initial (# of Reviews)]	# App Reviews in Total
Performance	Performance: D3 (121)	121
Portability	Portability: D3 (119)	119
Usability	Usability: D3 (432)	432
Reliability	Reliability: D3 (587)	587
Usage Scenario	Usage Scenario: D6 (593)	593
Feature Strength	Feature Strength: D6 (644)	644
User Experience	User Experience: D4 (737)	737
Feature Shortcoming	Feature Shortcoming: D6 (1,281)	1,281
Inquiry	Inquiry: D2 (1,100), User Action: D5 (428)	1,528
Problem	Problem Report: D2 (1,437), App Problem: D5 (399), Complaint: D6 (277)	2,113
Rating	Rating: D4 (2,721)	2,721
Bug Report	Bug Report: D1 (2,338), Bug Report: D4 (378), Bug Report: D6 (990)	3,706
Feature Request	Feature Request: D1 (2,323), Feature Request: D3 (558), Feature Request: D4 (299), User request: D6 (404)	3,584
Aspect Evaluation	Aspect Evaluation: D1 (5,937)	5,937
Praise	Praise: D1 (8,112), Praise: D6 (1703)	9,815
Irrelevant	Others: D1 (15,290), Irrelevant: D2 (3,869), Others: D3 (2,183), Neither: D5 (673)	22,015
Total		55,933



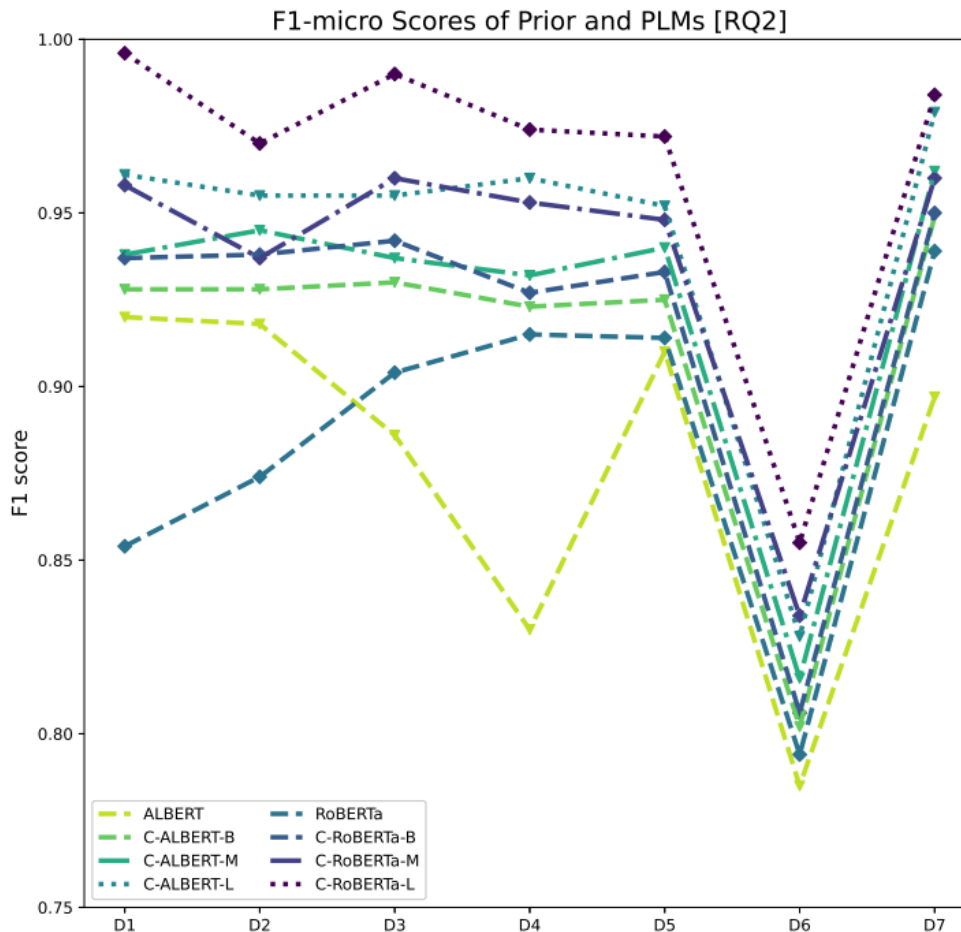
RQ1

F1-micro Scores of Prior and PLMs [RQ1]

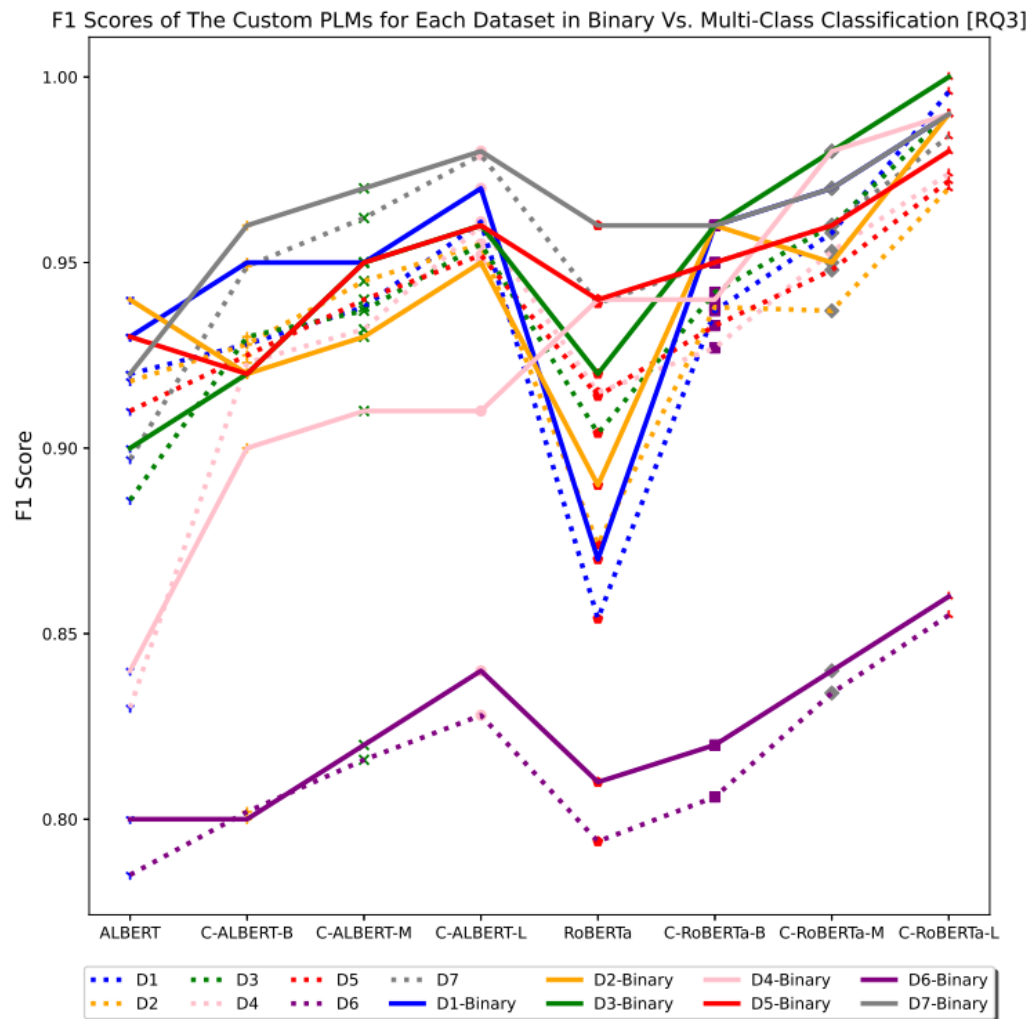


RQ2

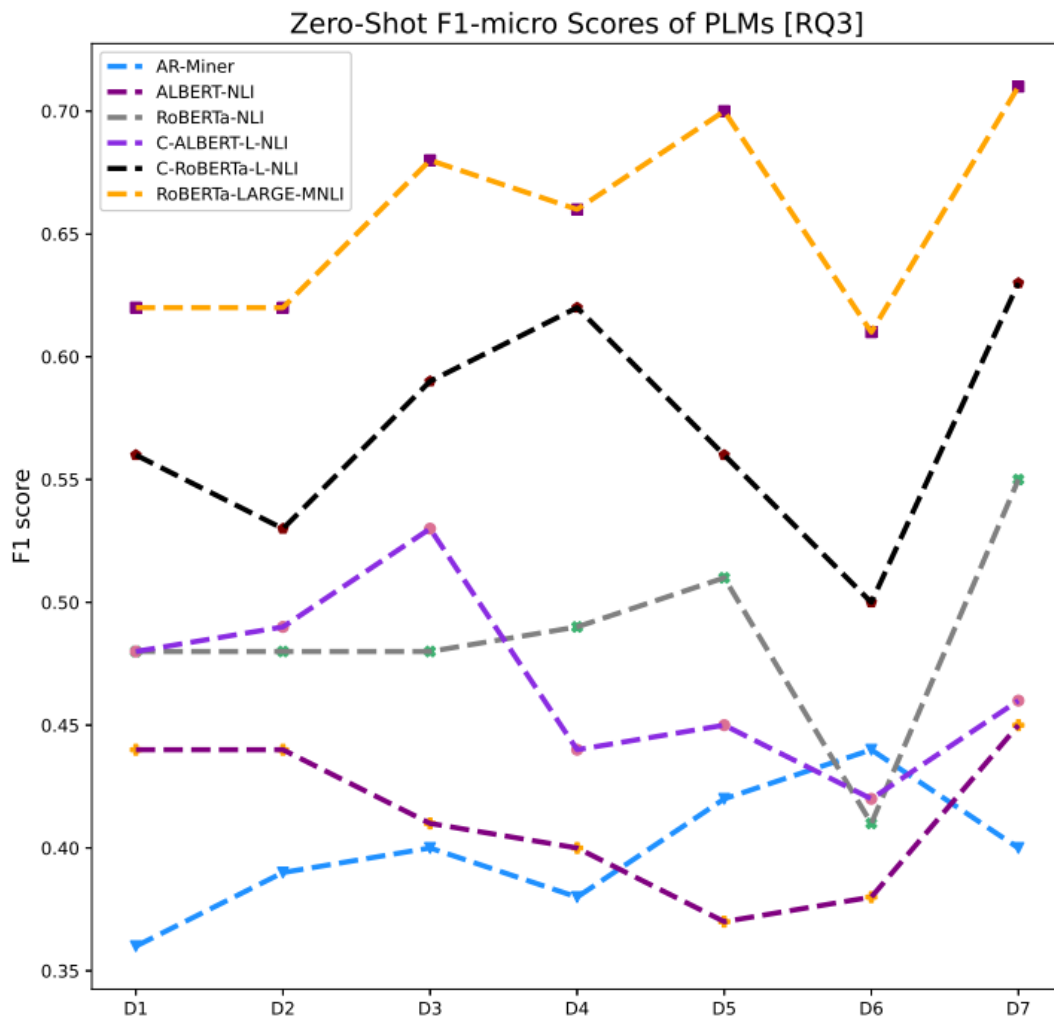
- ❑ Domain specific models slightly improve the prediction time.
- ❑ More custom data improves the performance of the models more.



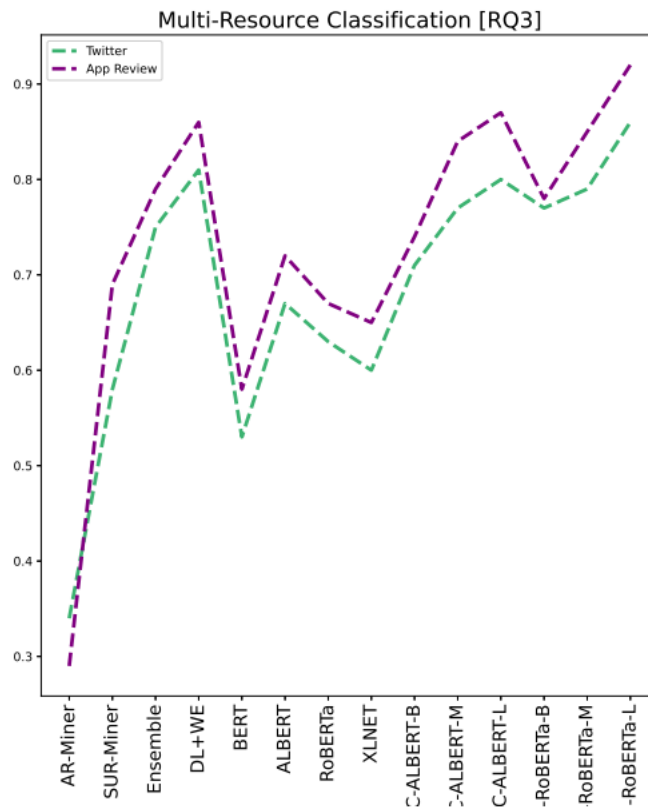
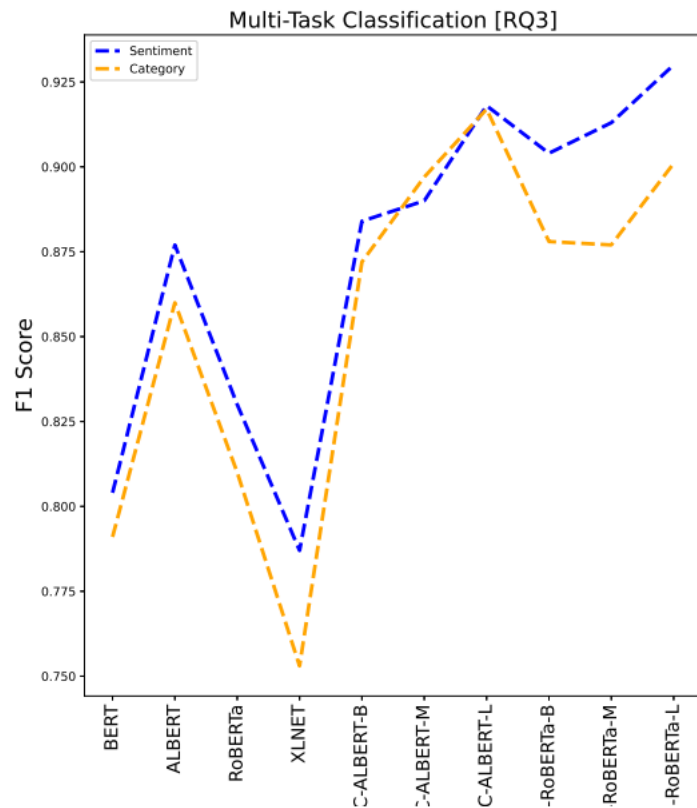
Binary Classification



Zero-Shot



Multi-Task and Multi-Resource



Take Aways



- ☐ PLMs can be used for all settings when higher performance is required
- ☐ Use Custom-PLMs over PLMs
- ☐ Use binary classification

Challenges



Inaccurate results are not acceptable

Retrieval Augmented LLMs



New Data (e.g., API versions, URLs)

Domain Specific Data

Private Information (User Info, Business)

Lack of computational resources

Retrieval Augmented Tool with Generative Power of LLMs



Regular GPT-3.5-Tur



what is llama2



Llama2 is a term that does not have a specific meaning, such as a specific llama named Llama2, a code name for a project. Without more information, I cannot provide an accurate answer. Can you please provide more context?

GPT enhanced with Documentation data



what is llama2



Llama2 is an improved version of the Llama model. It is a collection of pretrained and fine-tuned large language models optimized for dialogue use cases. Llama2 models range in scale from 7 billion to 70 billion parameters and have been trained on trillions of tokens. These models outperform open-source chat models on various benchmarks. Llama2 incorporates architectural tweaks, such as Grouped Query Attention, to enhance its performance. It is designed to be efficient and effective for chat applications.

Please enter your question



Retrieval Augmented Tool with Generative Power of LLMs



Regular G



What is langchain



I'm sorry, but I don't
context or clarify yo

GPT enhanced with Documentation data



what is langchain



LangChain is a library or framework that provides various tools and components for natural language processing (NLP) tasks. It includes modules for document loading, chat models, memory management, query construction, retrieval, and more. LangChain aims to simplify the development and deployment of NLP applications by providing a unified interface and a set of pre-built components.

Retrieval Augmented Tool with Generative Power of LLMs

GPT enhanced with Documentation data



how can I use OpenAI in Langchain, please provide code



Sure! Here's an example code snippet that demonstrates how to use OpenAI in LangChain:

```
from langchain.chat_models import ChatOpenAI
from langchain.chains import LLMChain
from langchain.smith import RunEvalConfig, run_on_dataset

# Chains may have memory. Passing in a constructor function lets the
# evaluation framework avoid cross-contamination between runs.
def construct_chain():
    llm = ChatOpenAI(temperature=0)
    chain = LLMChain.from_string(
        llm,
        "What's the answer to {your_input_key}"
    )
    return chain

# Load off-the-shelf evaluators via config or the EvaluatorType (string or dict)
evaluation_config = RunEvalConfig(
    evaluators=[
        "qa", # "Correctness" against a reference answer
    ]
)
```

Please enter your question



Challenges



Data availability: Zero or few examples

Low resource languages

Lack of computational power

submissions

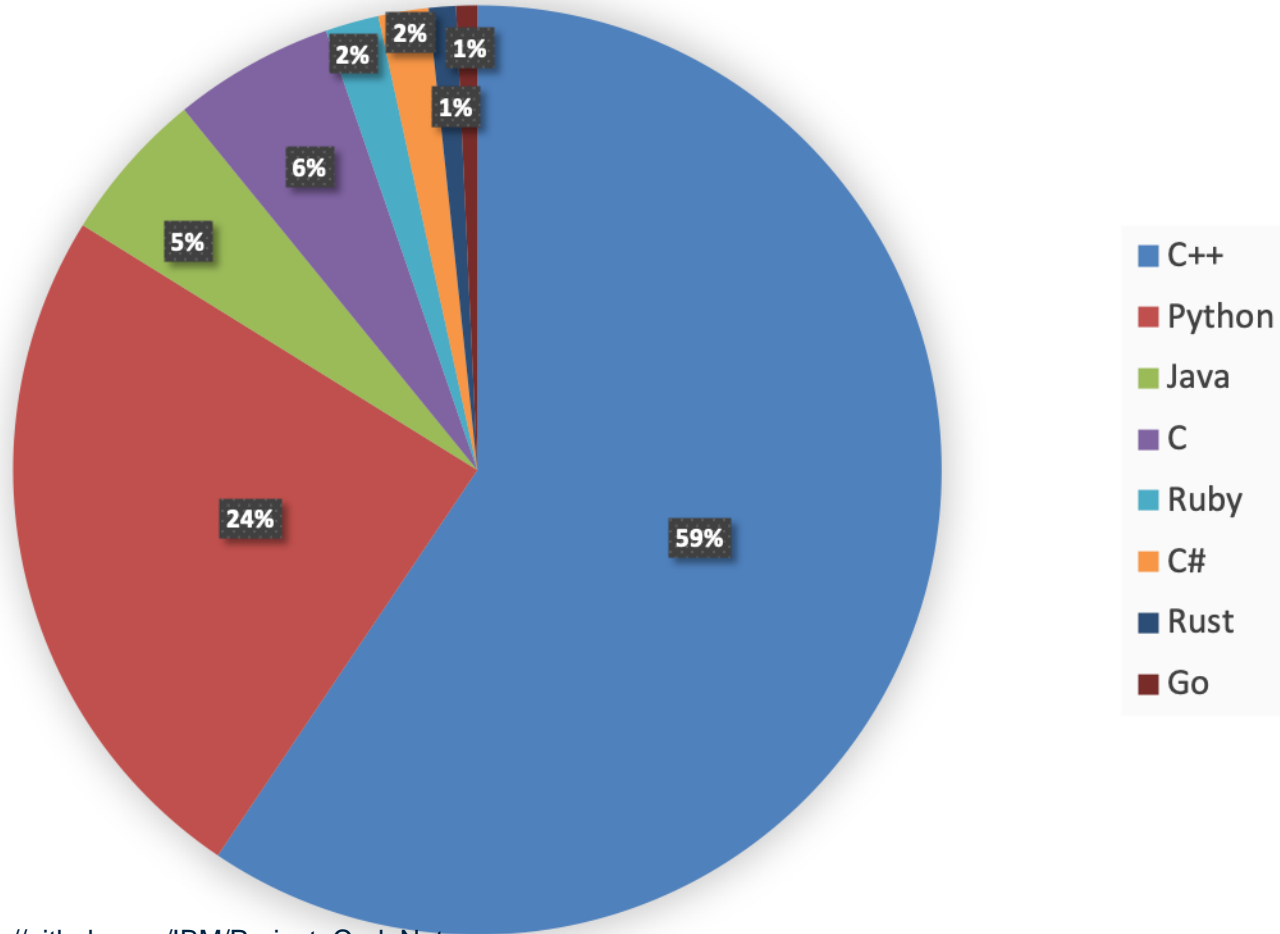


Image from: https://github.com/IBM/Project_CodeNet





Low Resource Languages
and Scientific Programming
Languages



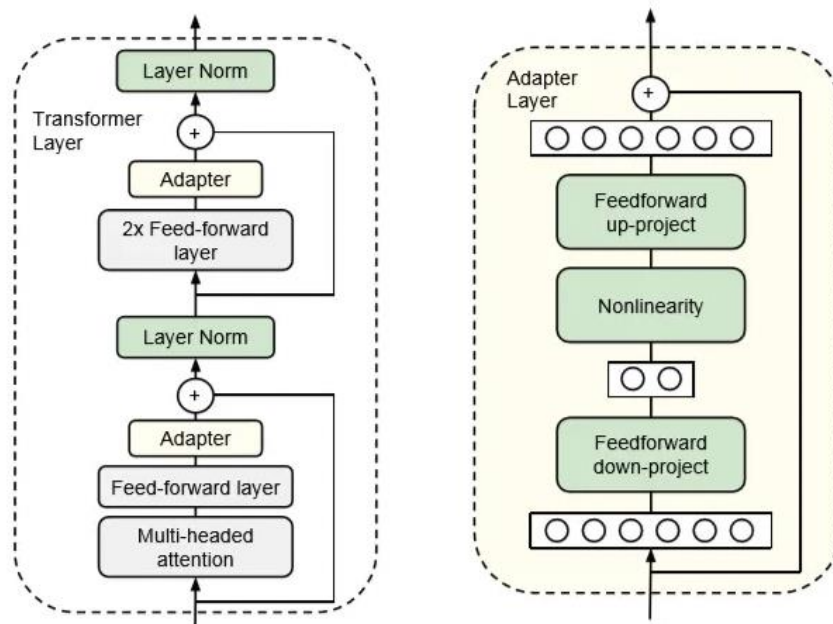
Computational Efficiency



Adapters



1. Fine-tuning
2. Domain adaptation



Internal architecture of Transformer blocks using adapters (Left Figure) and internal architecture of language adapters (Right Figure)

Adapters for SE?



How to?

- Do they work for bimodal transfer?
- How do they perform for code-related tasks?

SE-specific adapters?

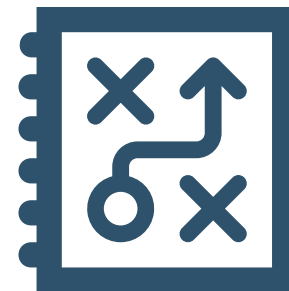
- Other purposes of adapters?
- New adapters for source code?

MODE-X

Goel, Divyam, Grover, Ramansh, and Fatemeh H. Fard. On the cross-modal transfer from natural language to code through adapter modules.

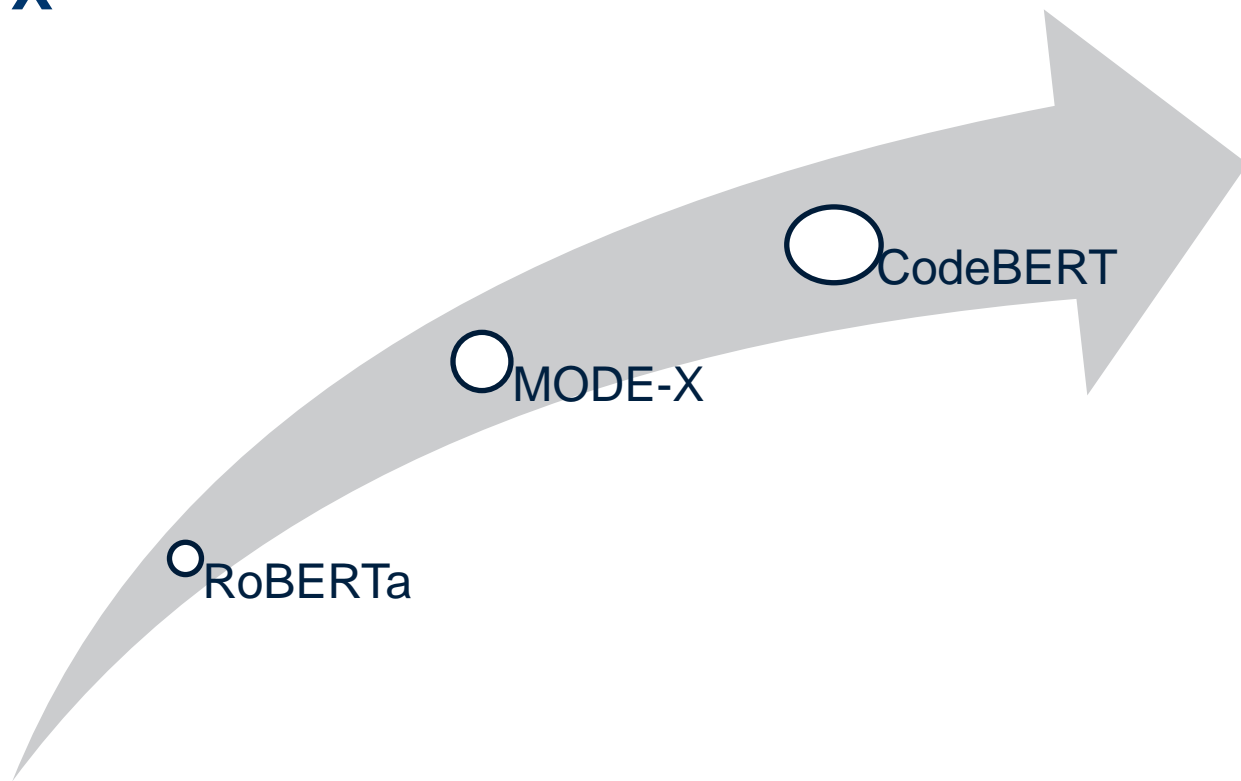


- Knowledge transfer from pre-trained models (PLMs) trained on Natural Language to Source Code
- Performance of adapters for code-PLMs



Goel, Divyam, Grover, Ramansh, and Fatemeh H. Fard. On the cross-modal transfer from natural language to code through adapter modules. In Proceedings of the 30th IEEE/ACM International Conference on Program Comprehension, ICPC '22, page 71–81, New York, NY, USA, 2022.

MODE-X



Accuracy Scores on Cloze Test



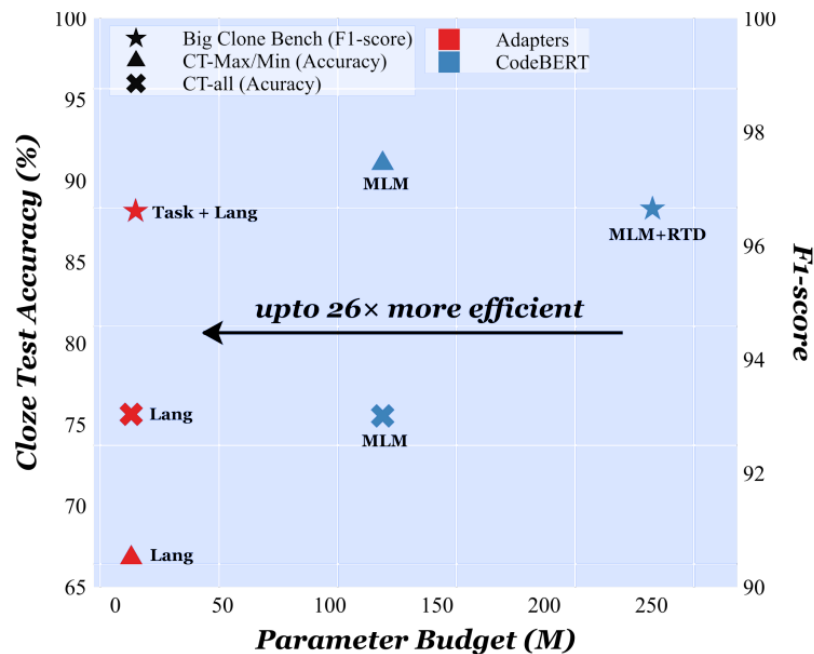
Model	Python	Java
CT max/min		
RoBERTa	59.18	59.75
RoBERTa+LA	66.30	66.81
CodeBERT	79.27	91.08
CT-all		
RoBERTa	54.49	50.75
RoBERTa+LA	74.35	75.63
CodeBERT	83.33	75.53

Code Clone Detection Results

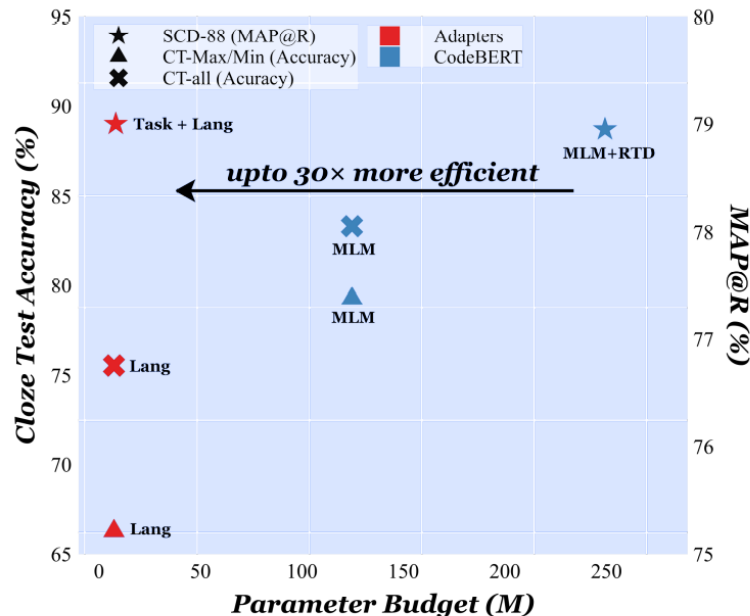


Model	Dataset	Score
RoBERTa	POJ-104 (MAP@R)	81.52
MODE-X (C/C++)		82.40
CodeBERT		86.48
RoBERTa	BCB (F1)	95.61
MODE-X (Java)		96.61
CodeBERT		96.65
RoBERTa	SCD-88 (MAP@R)	73.90
MODE-X (Python)		75.65
CodeBERT		78.95

Computational Efficiency of Adapters

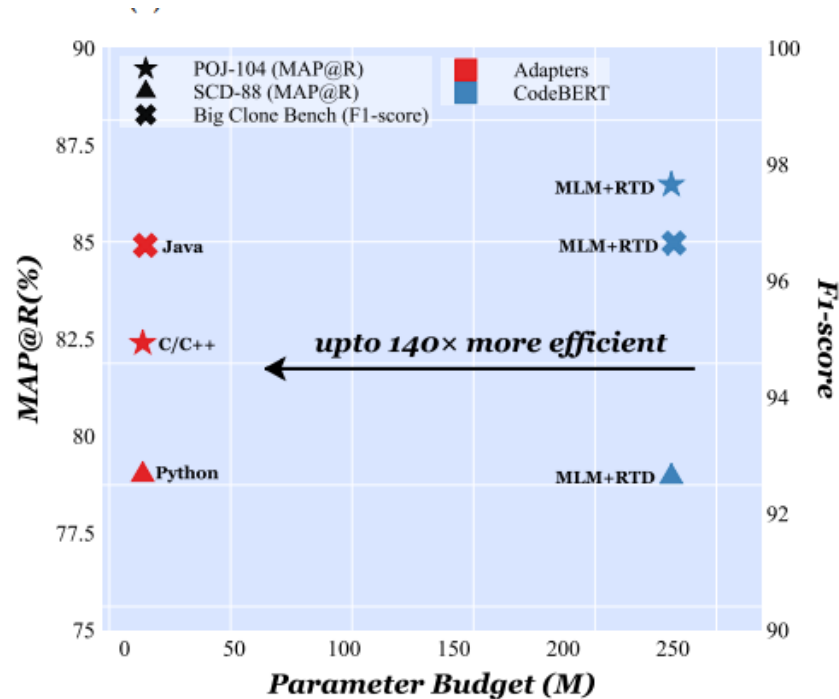


Parameter budget of Java-adapters and CodeBERT in millions



Parameter budget of Python-adapters and CodeBERT in millions

Computational Efficiency of Adapters



Parameter budget of adapters and CodeBERT for code clone detection



- Utilize adapters for knowledge transfer from N-PLM to source code (SE-tasks)
- Adapters are more efficient in terms of the number of parameters, memory usage, and inference time.

Code Summarization Results

Smoot BLEU-4



Models/Languages	Ruby	JS	Go	Python	Java	PHP
GraphCodeBERT + TA	14.53	16.54	23.74	<u>18.73</u>	<u>19.08</u>	25.05
CodeBERT+TA	<u>14.12</u>	<u>15.67</u>	<u>23.21</u>	18.47	18.99	25.55
MODE-X	12.79	14.20	23.05	17.72	18.43	24.27
GraphCodeBERT	12.62	14.79	18.40	18.02	19.22	<u>25.45</u>
CodeBERT	12.16	14.90	18.07	19.06	17.65	25.16
RoBERTa	11.17	11.90	17.72	18.14	16.47	24.02

MODE-X has better or on par results with C-PLMs

Code Summarization Results

Smoot BLEU-4



Models/Languages	Ruby	JS	Go	Python	Java	PHP
GraphCodeBERT + TA	14.53	16.54	23.74	18.73	19.08	25.05
GraphCodeBERT	12.62	14.79	18.40	18.02	19.22	25.45
CodeBERT+TA	14.12	15.67	23.21	18.47	18.99	25.55
CodeBERT	12.16	14.90	18.07	19.06	17.65	25.16

C-PLMs plus adapters have better results than fine-tuning them normally.

Code Summarization Results

Smoot BLEU-4



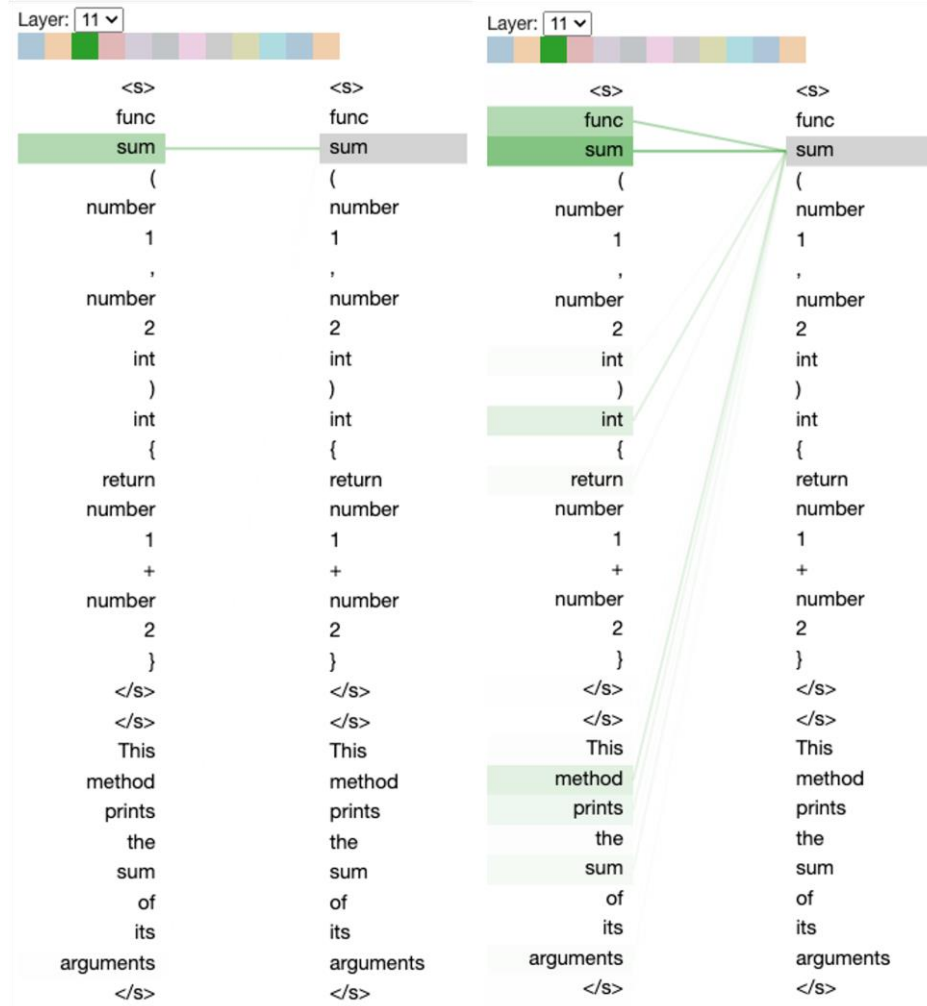
Models/Languages	Ruby	JS	Go	Python	Java	PHP
GraphCodeBERT + TA	14.53	16.54	23.74	<u>18.73</u>	<u>19.08</u>	25.05
CodeBERT+TA	<u>14.12</u>	<u>15.67</u>	<u>23.21</u>	18.47	18.99	25.55
MODE-X	12.79	14.20	23.05	17.72	18.43	24.27
GraphCodeBERT	12.62	14.79	18.40	18.02	19.22	<u>25.45</u>
CodeBERT	12.16	14.90	18.07	19.06	17.65	25.16
RoBERTa	11.17	11.90	17.72	18.14	16.47	24.02
Bimodal Data	52.9K	143.2K	317.8K	458.2K	500.7K	662.9K



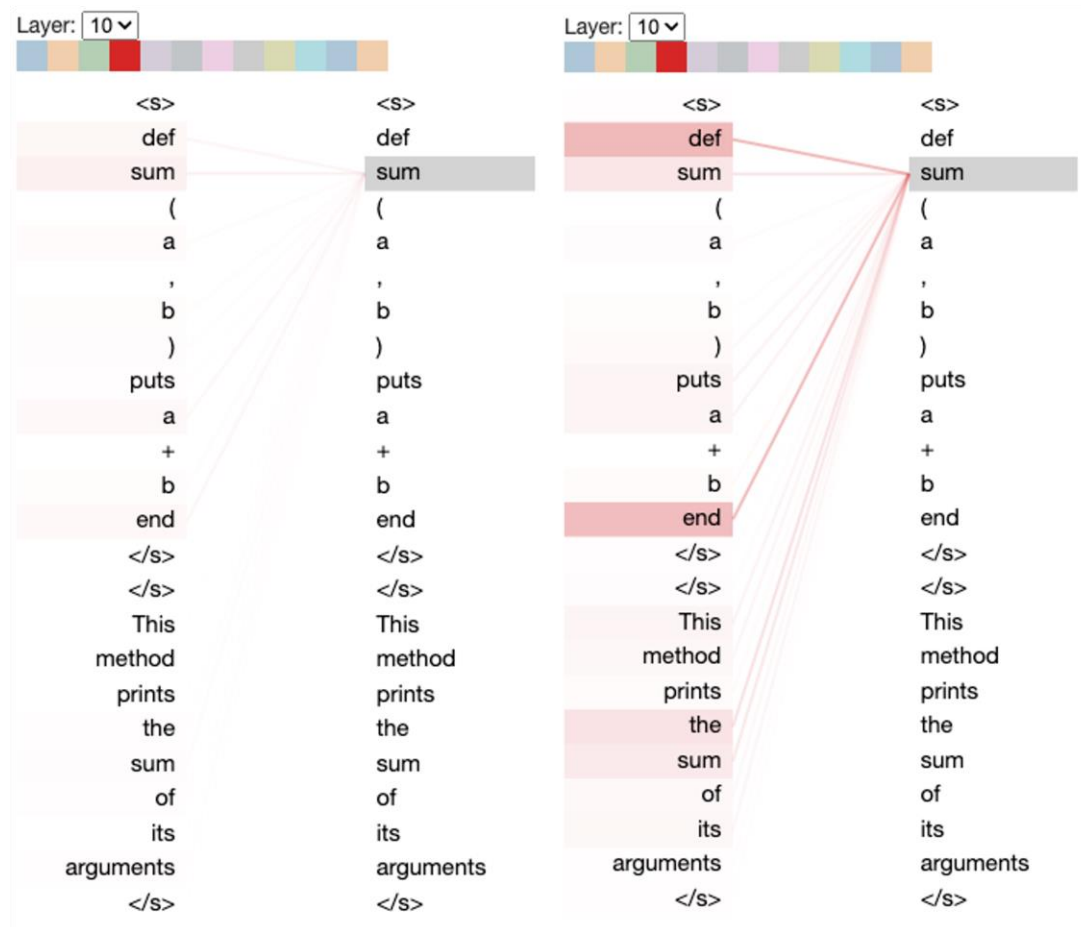
If we encourage the model weights to be closer to the pre-trained model, we could improve the fine-tuning results without using additional data/parameters.



Go Attention



Ruby Attention





How to?

SE-specific
adapters?

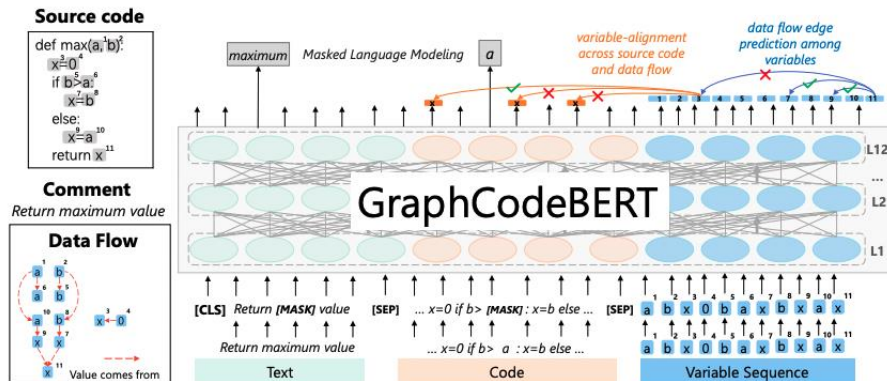
- Other purposes of adapters?
- New adapters for source Code?

CODEBERTER

Iman Saberi, Fatemeh H. Fard, Model-Agnostic
Syntactical Information for Pre-Trained
Programming Language Models.



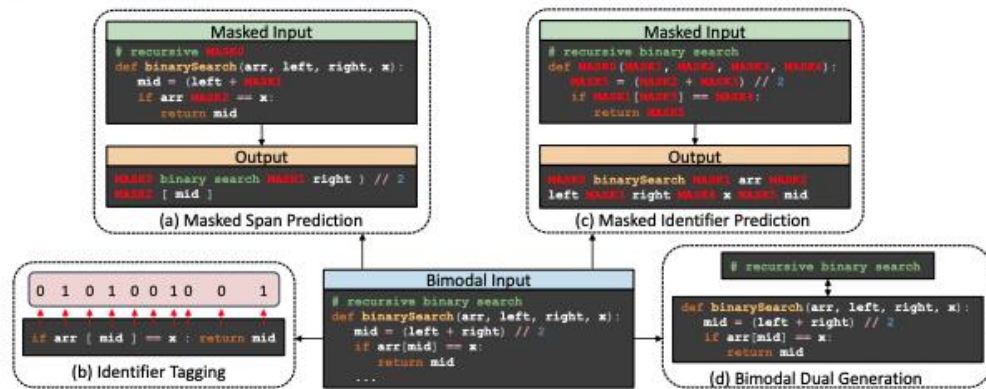
GraphCodeBERT



Picture from: D. Guo, S. Ren, S. Lu, Z. Feng, D. Tang, L. Shujie, L. Zhou, N. Duan, A. Svyatkovskiy, S. Fu et al., "Graphcodebert: Pre-training code representations with data flow," in International Conference on Learning Representations.

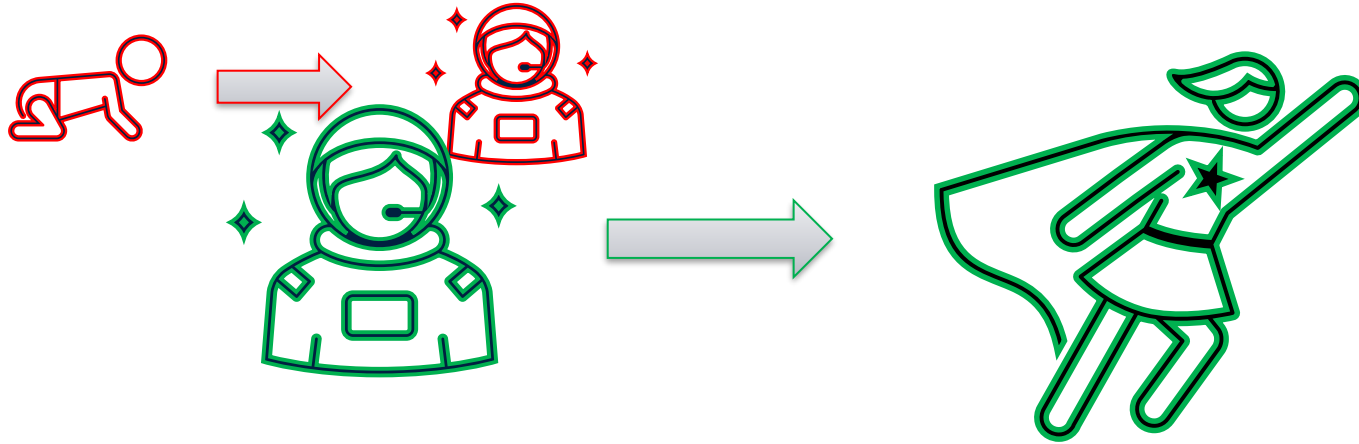
Require Pre-Training

CodeT5

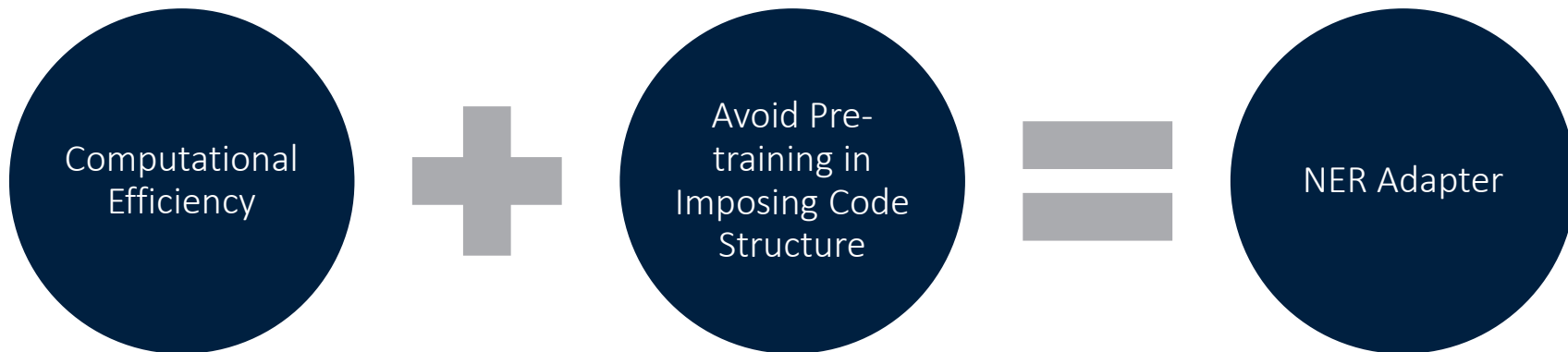


Picture from: Y. Wang, W. Wang, S. Joty, and S. C. Hoi, "Codet5: Identifier-aware unified pre-trained encoder-decoder models for code understanding and generation," in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 8696–8708.

Challenge: How to impose syntactical information of source code to **existing pretrained models**?

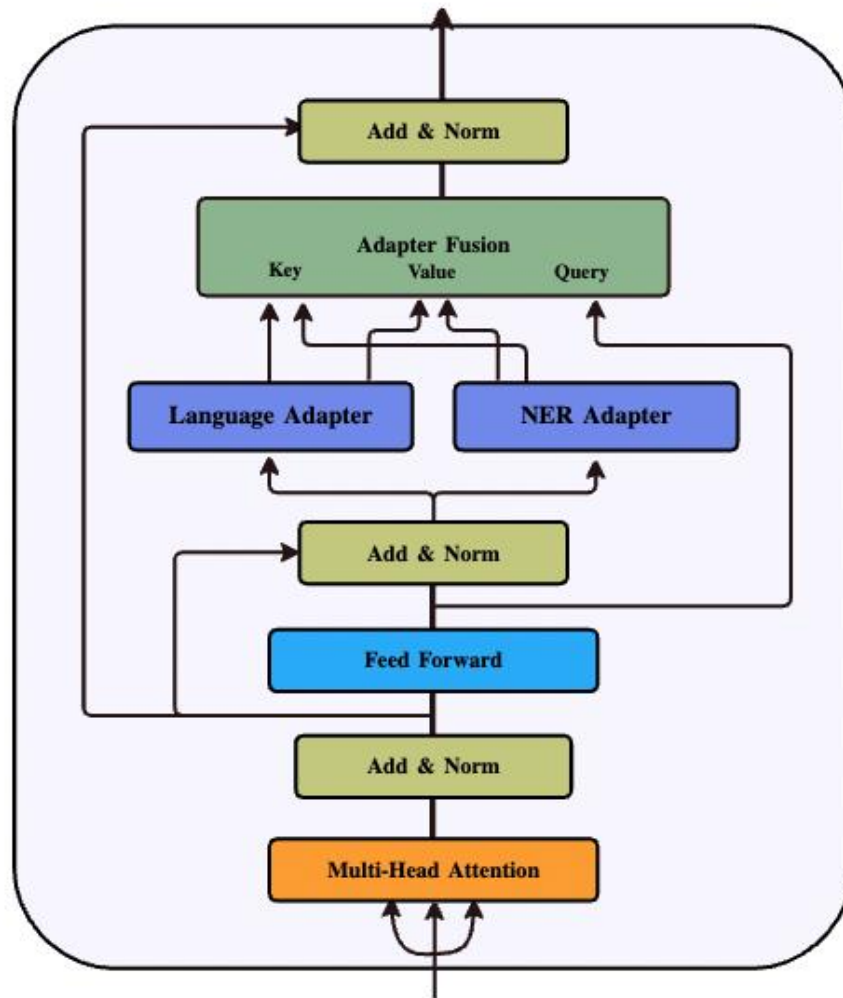


Avoid pre-training while adding new information

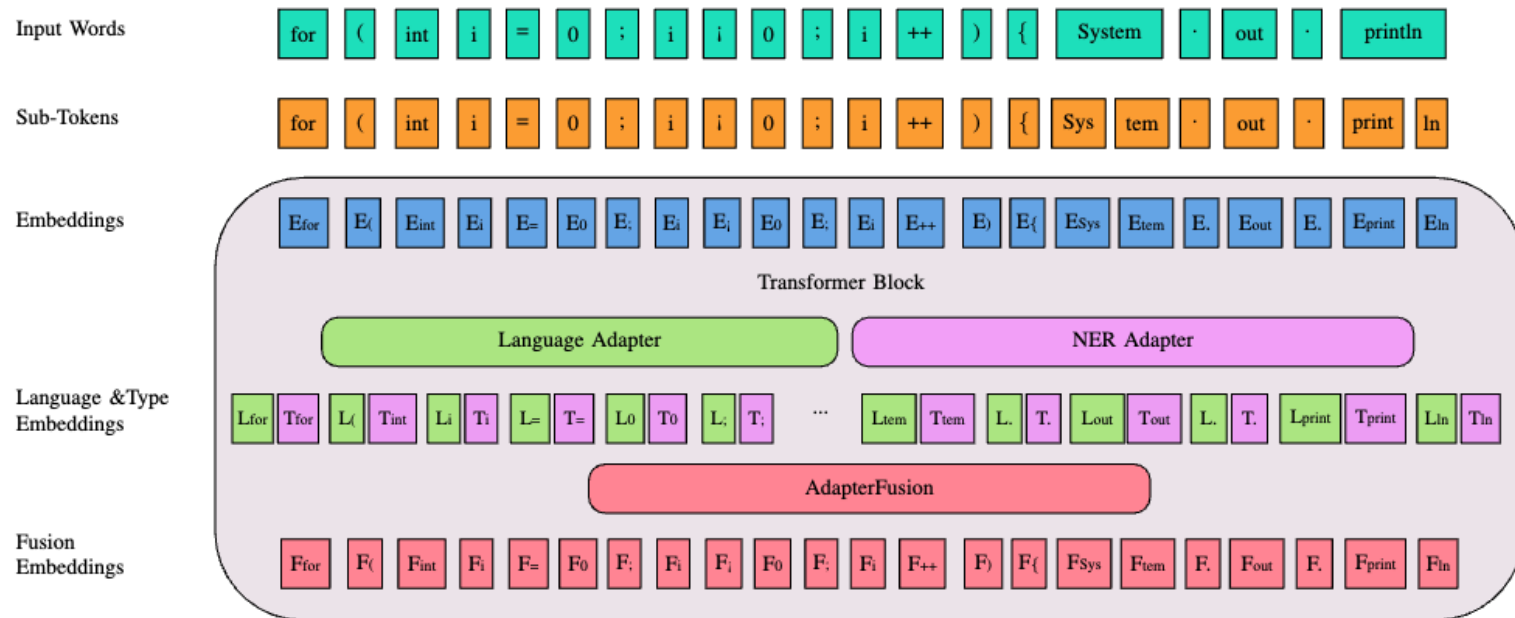


NER Adapter

Token Type
Classification Loss
(TTC)



Overall Architecture



The input data flow for the sample when fed into a transformer block equipped with NER, language and Fusion adapters.

Code Summarization

Automatically generating descriptions of the functionality of a given code



Models	Ruby	JavaScript	Go	Python	Java	Average
CodeBERTER	15.90	16.12	23.34	18.38	19.95	18.738



Works well for languages with
less training data

Code Refinement



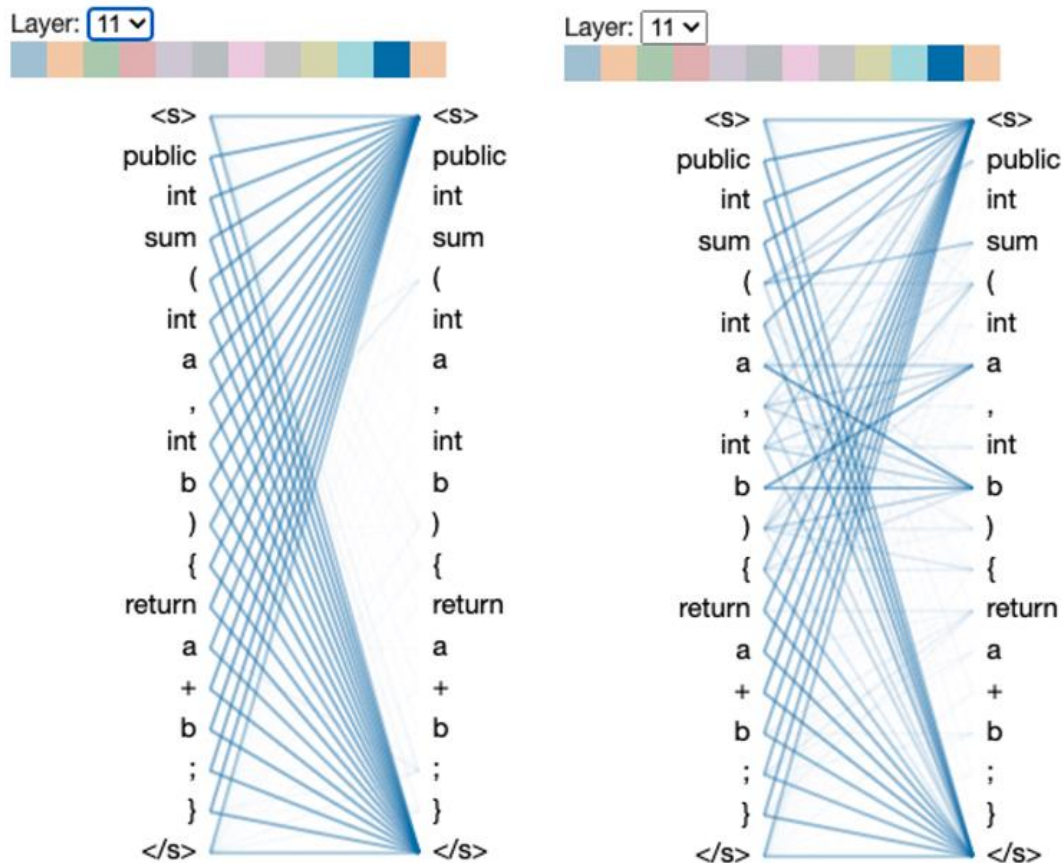
Identify and fix bugs automatically

Method/Model	BLEUNSEdit	Accuracy
Naïve copy	78.06	0.0
LSTM	76.76	10.0
Transformer	77.21	14.7
RoBERTa (code)	77.30	15.9
CodeBERT	77.42	16.4
CodeBERTER	78.20	17.8
CoText	77.91	22.64
NSEdit	71.06	24.04

Attention Change with NER Adapter

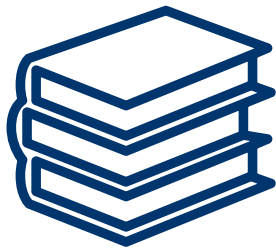


- **CodeBERTER** (right)
- CodeBERT (figure)



SE-Specific adapters can help:

- Avoid pre-training
- Impose new information
- Improve the results



Challenges



Many new domains

- Domain specific data
- Domain specific data structure and format

Prompt Engineering



Non-generative tasks (e.g., classification)



Data availability: Zero or few examples



Low resource languages



Inaccurate results are not acceptable

Summary



Review of LLMs, usages and failure modes

Review of overall process of using pre-trained LMs (and LLMs)

Overview of some challenges

Some techniques to address the challenges



THE UNIVERSITY OF BRITISH COLUMBIA