# Artificial Intelligence Driven Security Operations

## Security Operations (SecOps)

In recent years, the rapid increase in cyber threats coupled with the growing complexity of IT infrastructures has driven organizations to rethink traditional security practices. One response has been the evolution of Security Operations (SecOps)—a framework that brings together IT security and operations teams to streamline threat detection, incident response, and vulnerability management. By breaking down long‑standing silos, SecOps enables a coordinated, proactive approach to cybersecurity, thereby reducing risk and enhancing operational efficiency. This integrated model is now being further augmented by artificial intelligence, which promises to revolutionize how security teams work and innovate.

The integrated nature of SecOps offers several advantages:

* **Reduced Cyber Risk:** By enabling continuous monitoring and faster incident response, SecOps reduces the risk of significant breaches. The unified approach helps thwart threats before they escalate, thereby reducing downtime and potential damage
* **Improved Operational Efficiency:** Collaboration between IT and security teams streamlines processes and minimizes redundancies. Enhanced workflows and automated processes not only speed up response times but also improve overall productivity
* **Cost Savings:** By reducing the frequency and impact of breaches, organizations can lower the direct and indirect costs associated with incident recovery and system downtime.
* **Enhanced Compliance:** Integrated security operations help organizations meet regulatory requirements and maintain a higher standard of data protection, thereby reducing the risk of fines and reputational damage.

## Artificial Intelligence

At its core, AI refers to the ability of machines or software to simulate aspects of human cognition. According to IBM, AI enables computers to “learn from experience, comprehend complex data, solve problems, make decisions, and even exhibit creativity” . In everyday language, AI can be seen in applications like voice assistants, recommendation systems, and autonomous vehicle.

The journey of AI began in the 1950s with pioneers such as Alan Turing, who introduced the idea that machines could potentially "think" (via what is now known as the Turing Test), and John McCarthy, who coined the term "artificial intelligence" in 1956. Over the decades, research has evolved from symbolic AI and early neural networks to the sophisticated deep learning models that power today’s generative AI systems like ChatGPT and Google Gemini.

### **Key Subfields of AI**

* **Machine Learning (ML):**ML involves creating algorithms that enable systems to learn patterns from data and make predictions or decisions. Techniques include supervised learning (learning from labeled data), unsupervised learning (discovering hidden patterns in unlabeled data), and reinforcement learning (learning through rewards and penalties).
* **Deep Learning:**A subset of ML, deep learning uses artificial neural networks with multiple layers to model complex patterns. It has driven advances in computer vision, speech recognition, and natural language processing .
* **Natural Language Processing (NLP):**NLP enables machines to understand, interpret, and generate human language. It underpins technologies such as chatbots, translation services, and sentiment analysis.
* **Computer Vision:**This subfield focuses on enabling computers to interpret and understand visual information from images and videos, leading to applications in facial recognition, autonomous driving, and medical imaging.
* **Robotics and Automation:**Combining AI with robotics has led to machines that can perform physical tasks, from industrial automation to personal assistance.

### **Modern Advances: LLMs and BERT**

**Large Language Models (LLMs):**LLMs, such as OpenAI’s GPT series, are deep neural networks designed to understand and generate human-like text. They are characterized by:

* **Scale:** They are trained on vast amounts of data (often billions of parameters), enabling them to capture complex language patterns.
* **Generative Capabilities:** LLMs can generate coherent, context-aware text, making them useful for tasks ranging from content creation to dialogue systems.
* **Transfer Learning:** Once pre-trained on large corpora, LLMs can be fine-tuned on specific tasks with relatively smaller datasets, demonstrating versatility across diverse applications.

**BERT (Bidirectional Encoder Representations from Transformers):**BERT, developed by Google, introduced a major shift in NLP by leveraging the Transformer architecture:

* **Bidirectionality:** Unlike traditional language models that process text sequentially (e.g., left-to-right), BERT reads text in both directions simultaneously, allowing it to capture richer context and meaning.
* **Pre-training and Fine-Tuning:** BERT is pre-trained on a large corpus using tasks like masked language modeling and next sentence prediction, and then fine-tuned for specific applications such as sentiment analysis, question answering, and entity recognition.
* **Contextual Embeddings:** BERT produces embeddings that capture nuanced semantic meanings of words depending on their context, outperforming earlier models that relied on static word embeddings.

### **Comparison and Integration**

* **Evolution of Techniques:**Conventional methods laid the groundwork for understanding patterns and making decisions based on structured rules or statistical relationships. Deep learning models like LLMs and BERT have built on these foundations by automating feature extraction and learning hierarchical representations from raw data. The transition from shallow networks to deep architectures (like Transformers) has dramatically improved performance in complex tasks such as language understanding.
* **Advantages of LLMs and BERT:**These models can generalize across various tasks, require less manual feature engineering, and capture subtleties in language that conventional methods struggle with. Their ability to leverage context and learn from massive datasets has led to breakthroughs in tasks previously considered challenging for AI.
* **Challenges:**While LLMs and BERT offer remarkable capabilities, they also bring challenges such as high computational requirements, potential biases in training data, and issues related to interpretability. In contrast, conventional methods, though less powerful in handling complexity, tend to be more interpretable and computationally efficient.

## AI Driven Security Operations

### **2. Key Applications of AI in SecOps**

#### **2.1 Threat Detection and Prevention**

AI-powered systems are revolutionizing threat detection by rapidly analyzing network traffic, user behavior, and system logs. Machine learning (ML) models are trained on historical data to identify anomalies and potential threats that would otherwise be lost in the noise of normal operations. For example, advanced intrusion detection systems (IDS) leverage deep learning to detect zero-day vulnerabilities and unusual patterns that may indicate a breach. This predictive capability not only flags potential incidents in real time but also reduces false positives by continuously refining detection rules based on new data.

#### **2.2 Automated Incident Response**

Once a threat is identified, the speed of the response is critical. AI-driven automation tools in SecOps can trigger immediate responses—such as isolating affected endpoints, blocking malicious IP addresses, or invoking automated playbooks—to contain and mitigate damage before it spreads. This automation, often integrated into Security Orchestration, Automation, and Response (SOAR) platforms, significantly cuts down the mean time to respond (MTTR) compared to manual interventions.

#### **2.3 Threat Hunting and Alert Triage**

Proactive threat hunting is another vital area where AI contributes to SecOps. By analyzing vast amounts of data using natural language processing (NLP) and behavioral analytics, AI systems can surface subtle indicators of compromise and anomalous behavior that may signal an emerging threat. This capability allows even less experienced analysts to formulate complex search queries in plain language, thereby improving the triage process and helping prioritize alerts more effectively.

#### **2.4 Vulnerability Management**

AI enhances vulnerability management by continuously scanning an organization’s infrastructure for weaknesses. By correlating data from various sources—including patch databases, vulnerability scans, and system logs—AI models can prioritize vulnerabilities based on potential impact. This ensures that security teams focus on the most critical issues first, thereby reducing the risk of exploitation before patches can be applied .

#### **2.5 SIEM and Log Analysis**

Traditional Security Information and Event Management (SIEM) systems often generate an overwhelming number of alerts, many of which are false positives. AI integration in SIEM platforms automates log analysis, leverages pattern recognition to filter noise, and correlates events from disparate data sources. The result is enhanced situational awareness and a more efficient allocation of analyst resources, enabling faster investigation and remediation of real threats .

#### **2.6 Endpoint and Network Security**

AI is increasingly used to bolster endpoint and network security. By establishing behavioral baselines for devices and users, AI-driven endpoint detection and response (EDR) solutions can quickly detect deviations from normal operations—such as the appearance of malware or unauthorized access attempts—and initiate immediate remediation actions. Similarly, network security solutions enhanced with AI can analyze traffic flows and detect anomalies indicative of a breach, even in encrypted or heavily trafficked networks .

#### **2.7 Enhancing SOC Workflows**

Beyond detection and response, AI plays a crucial role in streamlining overall SOC workflows. AI systems can automatically generate incident summaries and comprehensive reports, turning raw alerts into actionable intelligence. By automating routine tasks like data correlation, alert classification, and report generation, AI frees up human analysts to focus on higher-level strategic decision-making, thereby increasing overall productivity and reducing burnout

### **3. Benefits of AI in SecOps**

Implementing AI in SecOps delivers several clear advantages:

* **Speed and Efficiency:** Automated threat detection and incident response reduce the time from alert to remediation.
* **Scalability:** AI systems can analyze vast amounts of data across diverse environments (cloud, on-premise, IoT) without additional human resources.
* **Improved Accuracy:** Machine learning models continuously learn from new data, reducing false positives and ensuring that alerts are meaningful.
* **Enhanced Productivity:** By automating repetitive tasks and summarizing incident data, AI allows security analysts to focus on strategic threat mitigation and complex investigations.
* **Proactive Defense:** Predictive analytics enable organizations to anticipate and address vulnerabilities before they can be exploited.

### **4. Challenges and Considerations**

Despite its transformative potential, the integration of AI in SecOps is not without challenges:

* **Data Quality and Bias:** AI models are only as good as the data they are trained on; biased or incomplete data can lead to false alarms or missed threats.
* **Integration Issues:** Incorporating AI into existing security infrastructures—often based on legacy systems—can be complex and resource-intensive.
* **Skills Shortage:** There is a growing need for cybersecurity professionals skilled in AI, machine learning, and data analytics.
* **Privacy and Ethical Concerns:** Automated systems must balance efficient threat detection with respect for privacy and adherence to legal and ethical standards.

### **5. Future Directions in AI-Driven SecOps**

The trend toward integrating AI into SecOps is set to continue as cyber threats evolve. Future directions include:

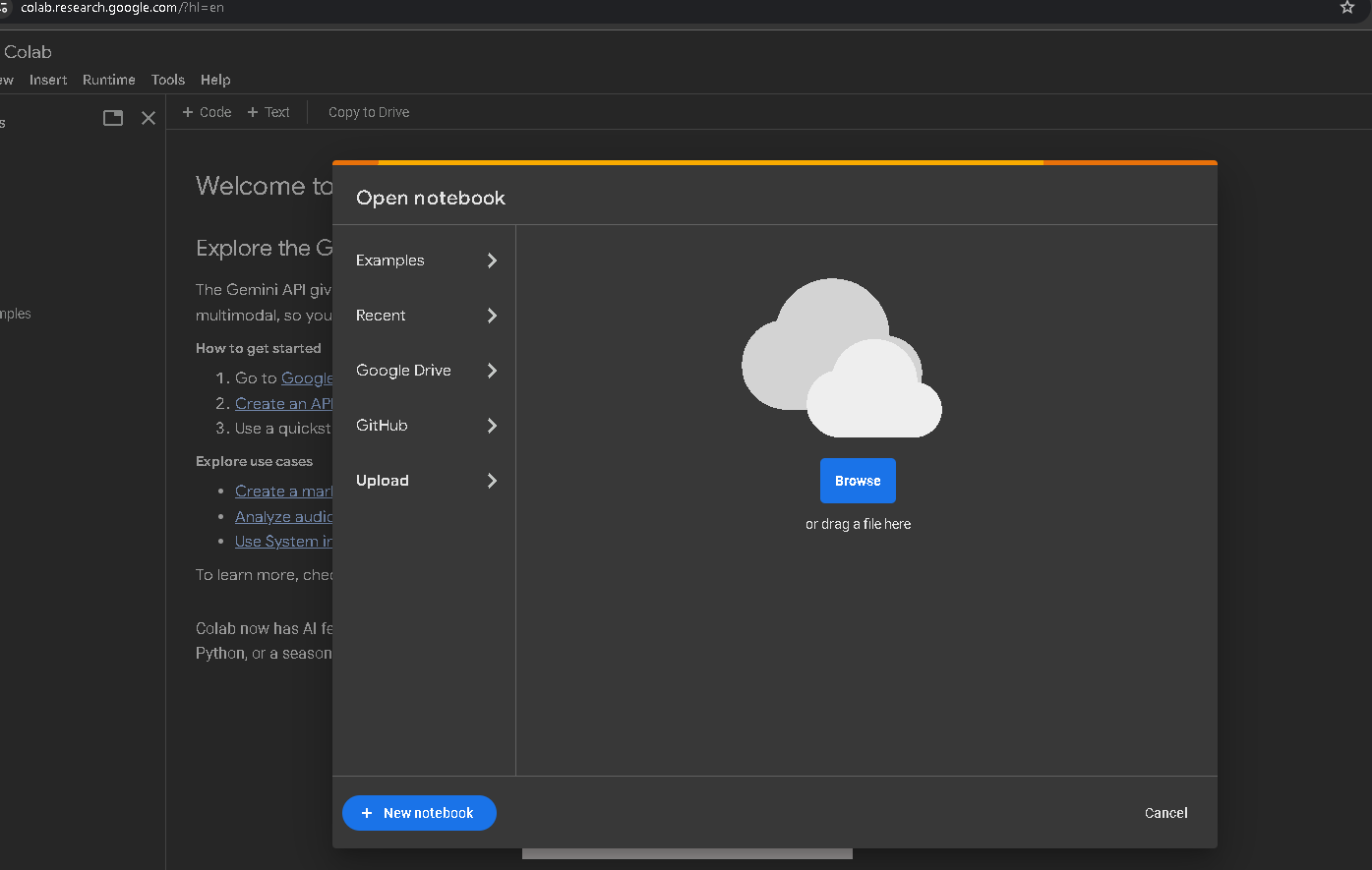
* **Greater Automation:** Continued advancements in AI will further reduce the need for manual intervention in routine security tasks.
* **Improved Predictive Capabilities:** Enhanced machine learning models will offer even more precise threat forecasting and proactive defense mechanisms.
* **Integration with Emerging Technologies:** As IoT, cloud computing, and 5G expand, AI will play an even larger role in securing increasingly complex digital ecosystems.
* **Human-AI Collaboration:** While AI will automate many operational tasks, human expertise will remain critical for strategic decision-making and handling nuanced threat scenarios.

## Laboratory

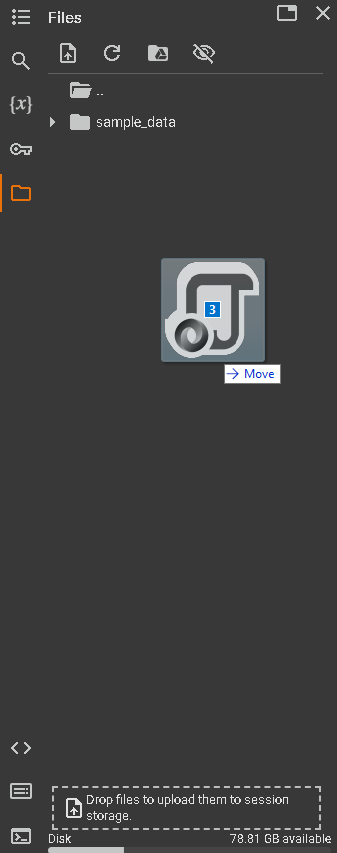
### Key Topics of the laboratory:

1. Working on Google Colab Notebooks

First you have to access the shared files and unzip the archive from the UNITBV drive. After this, go to <https://colab.research.google.com/> and upload the “lab\_soc.ipynb” file.

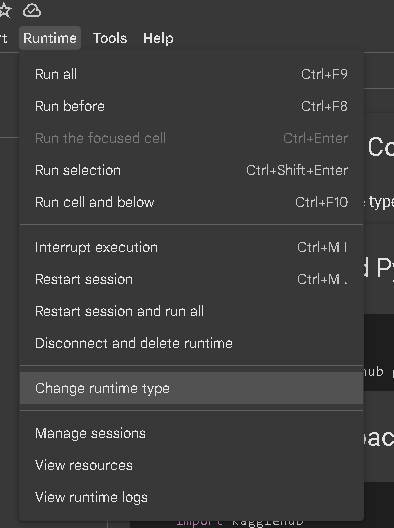


After uploading the file, you will be redirected to a new tab with the Notebook opened. On the left side-menu, upload the .json files provided in the archive downloaded from the GitHub repository, by drag`n`drop.

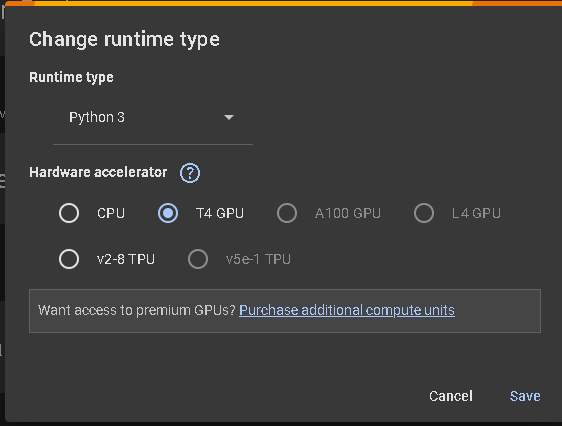


Then, you have to change the runtime to a GPU one, using a free Tesla T4 GPU provided by Google

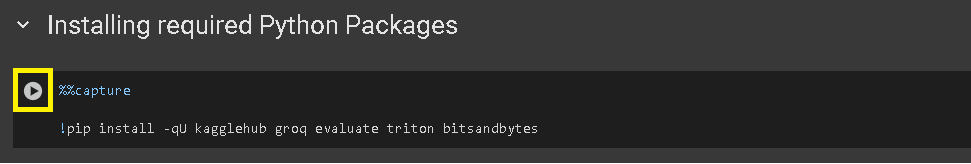
.



Select “T4 GPU” and click “Save”.



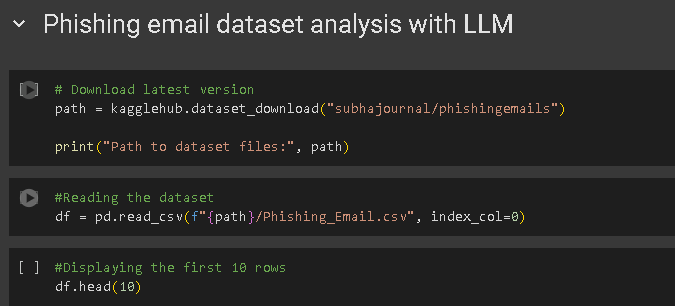
To run each cell, you have to click the top-right button of each cell. In order to have the first half of the laboratory working, you have to run the first cell to install the required Python packages.



The output will be shown below each one.

1. Analyzing a phishing dataset with LLMs.

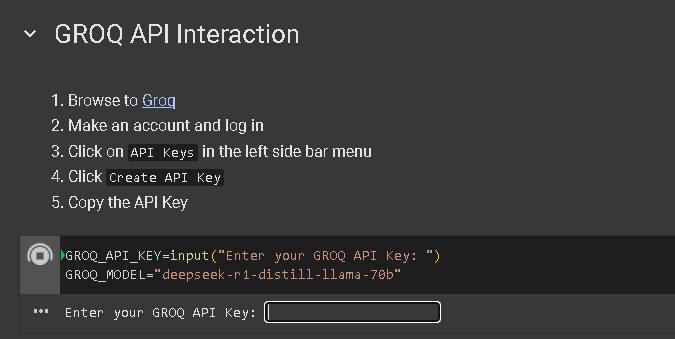
First dataset that we will inspect is the phishing dataset ( <https://www.kaggle.com/datasets/subhajournal/phishingemails> ).



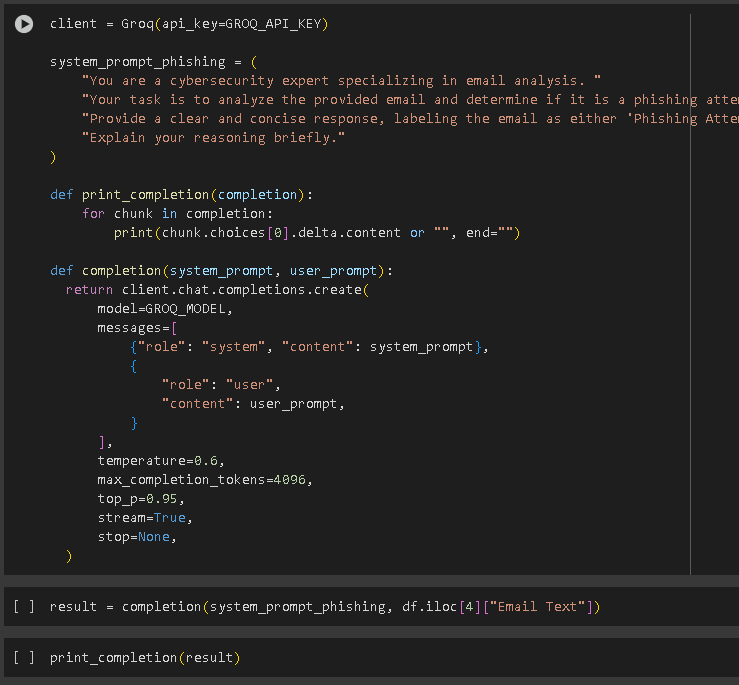
Executing these cells will download the dataset from Kaggle’s database, read the file and display the first 10 rows of the dataset.

To get access to an online LLM API you have to follow the steps provided in the Notebook, to get an API key from Groq.

Running the following cell, will prompt you to enter the GROQ API Key. If you do not insert it, the next steps won’t work for you.



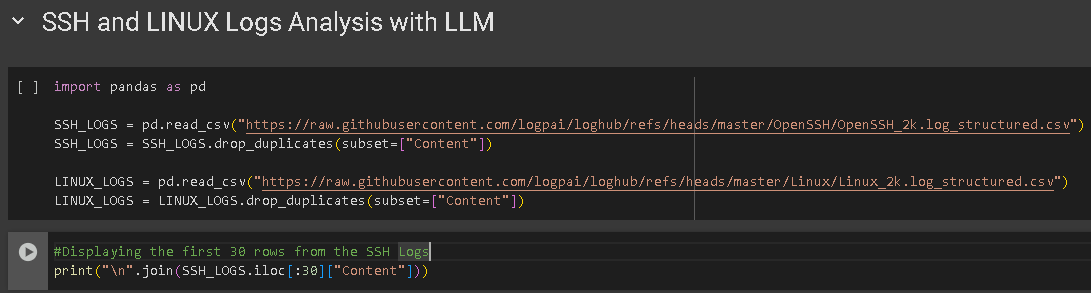
To see how an LLM “thinks” about the provided possible content from a phishing email, you have to run the following three cells.



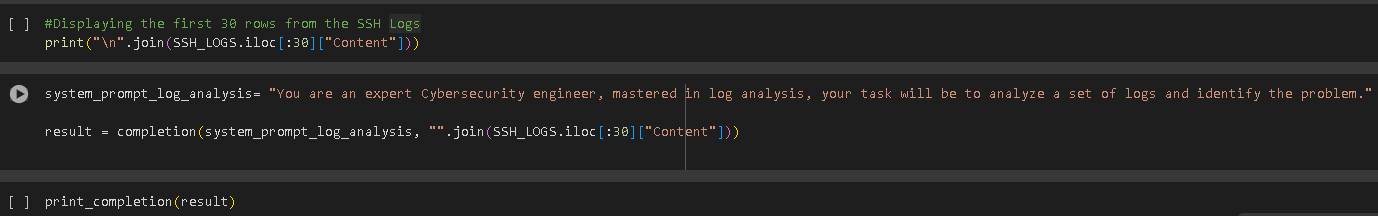
You will see the reasoning of the LLM used to analyze the data and understand why it responded with the conclusion of whether the text is or isn’t a phishing mail.

1. Downloading and analyzing the Linux System and OpenSSH logs using LLMs.

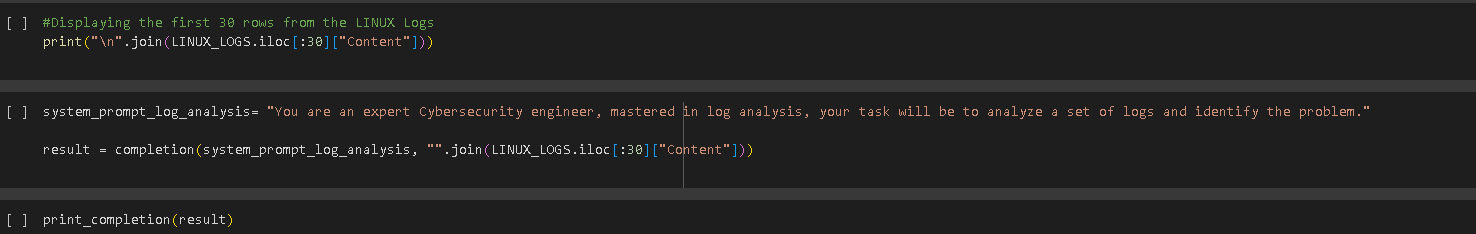
Run this cell to download the datasets.



Running these cells, you will see what the LLM can extract out of the first 30 logs from the SSH Logs.

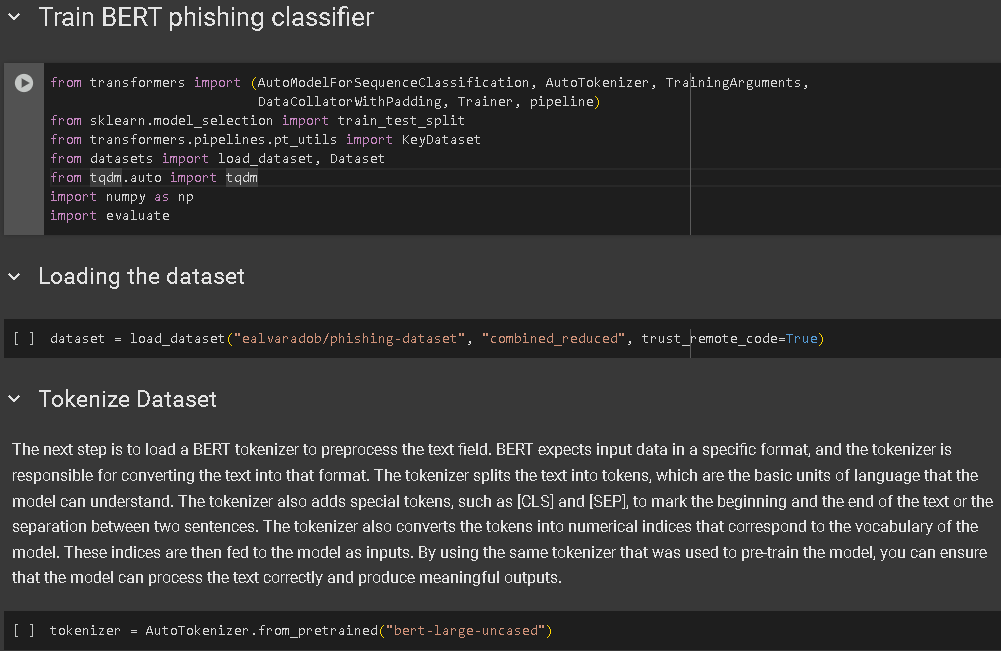


The same goes for the logs of the Linux System.

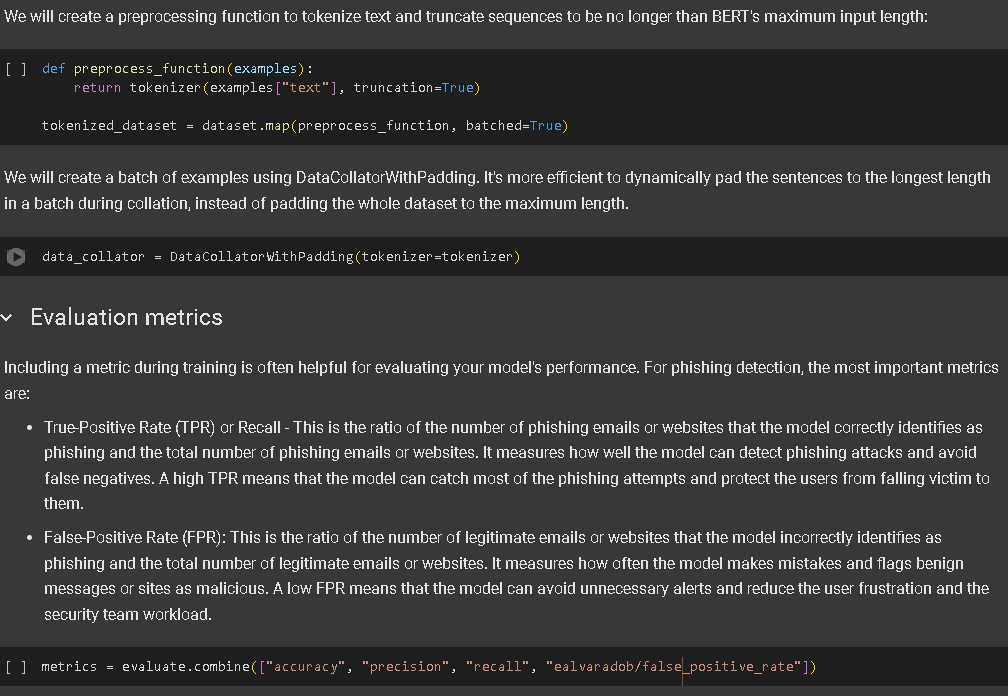


1. Training a BERT model to detect phishing attacks

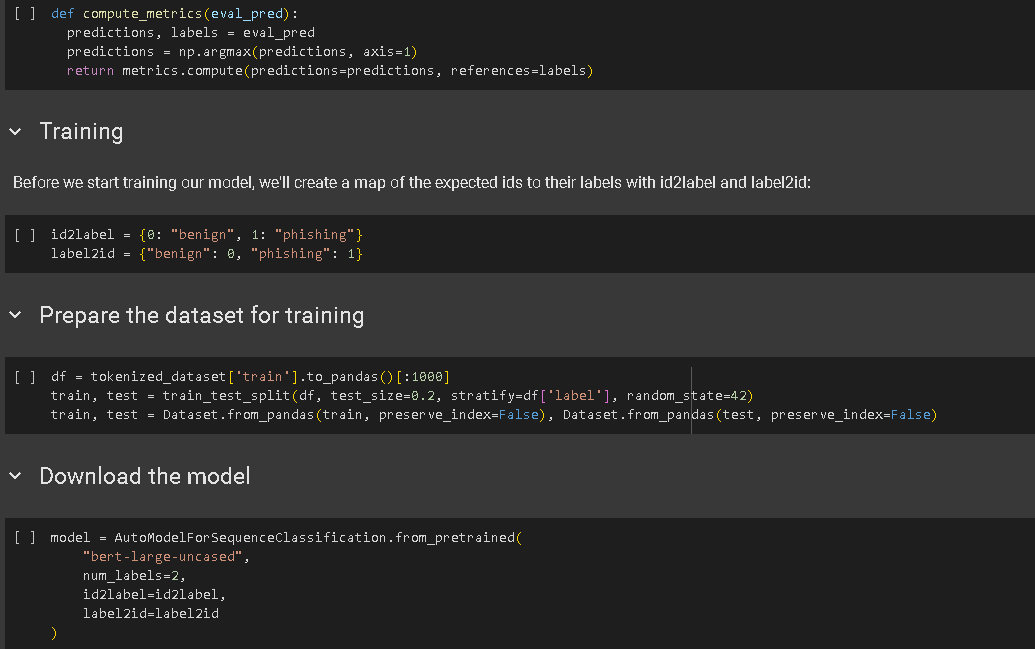
The following cells will import the Python packages needed to load and train a BERT model and will also download the dataset needed for fine-tuning.



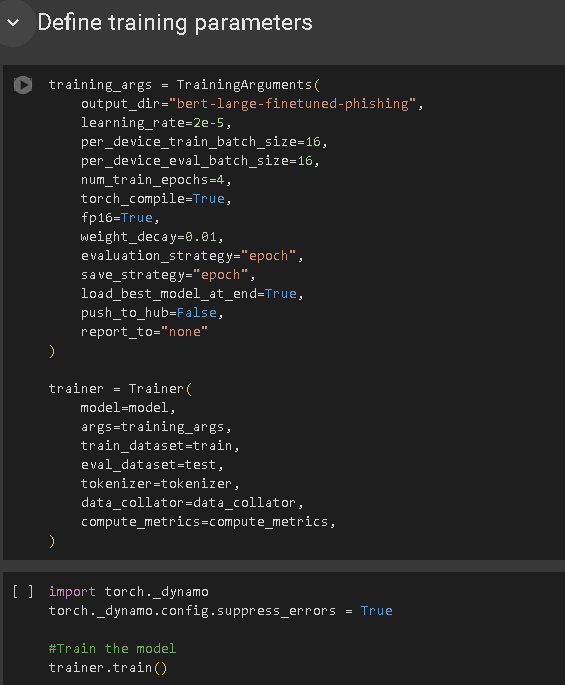
We will create a pre-processing function that will tokenize the dataset and prepare it for training. Also, we will define the metrics needed for evaluating the new model.



Based on these metrics, we will create a function that will be passed as a callback to the training method to evaluate the model. Then, we will create the label to text and text to label mapping and we will split the dataset for training and testing, only using the first 1000 rows, due to performance issues. After splitting the dataset, we will download the model and send the mapping that we have previously created as arguments so the model knows how to handle the dataset.

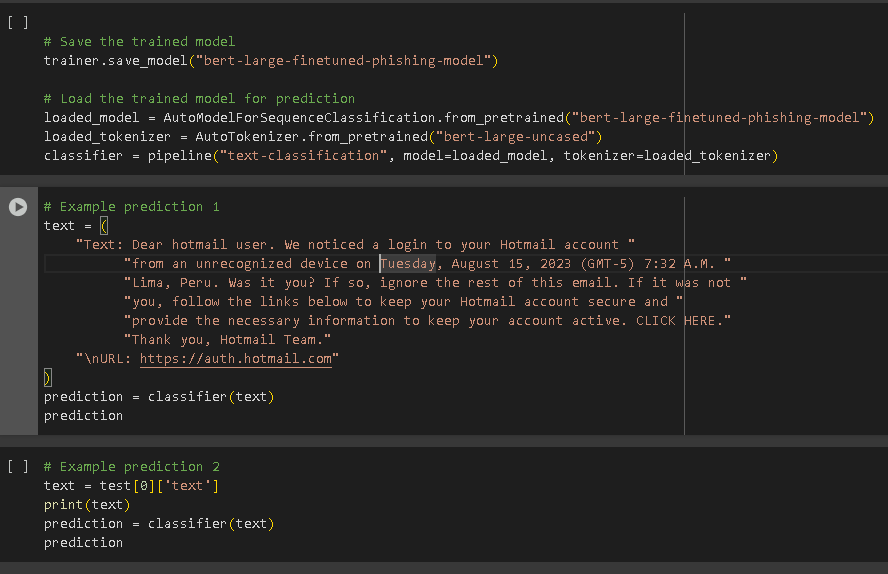


To train the model, we have to set the correct parameters and run the train() function.



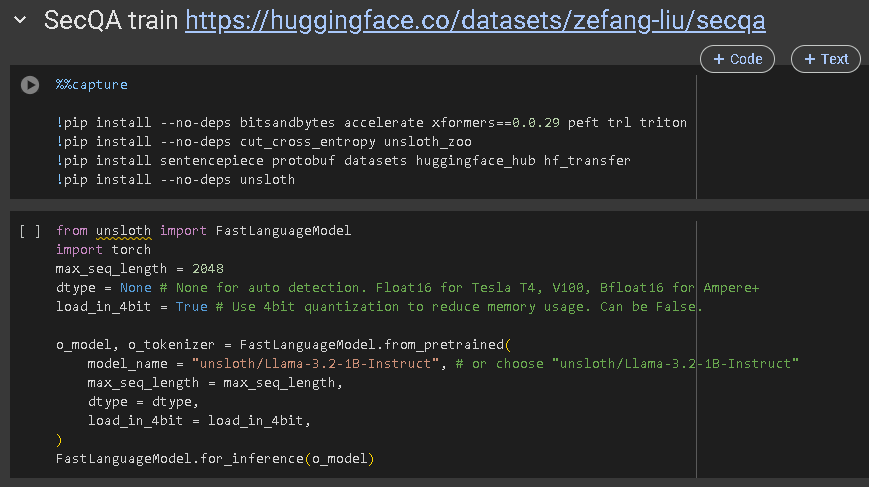
These parameters are just some basic parameters that are used to train simple models, but on our case they do their job pretty well.

After training the model, we have to save it and load it again, into another variable and create a classification pipeline. This pipeline will be used to classify a non-phishing email and a phishing email. From the last 2 cells you will see the probability, ranging from 0 to 1, of the predicted class. (The higher the probability the higher the chance of being the predicted class).



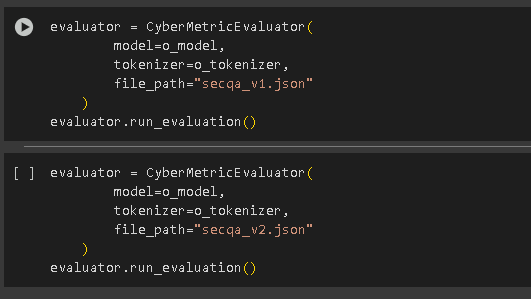
1. Evaluating a local Llama LLM on the Security Q&A (SecQA) dataset

To prepare for the second part of the laboratory, we have to install the required Python packages and download the Llama3.2:1B LLM Model.



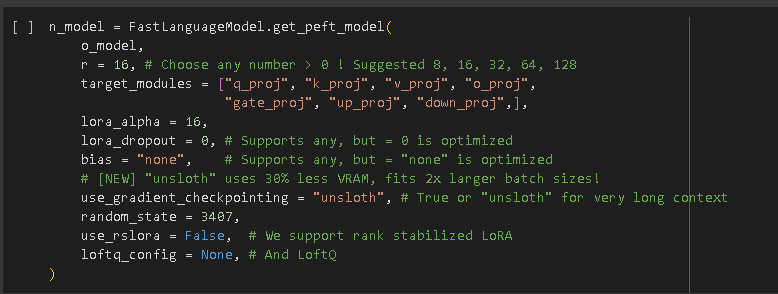
The following big cell is a Python class that has functions to evaluate the SecQA datasets provided from the GitHub repository. You have to run that to continue the lab!

Running the following cells, we will do the benchmark of the accuracy of the model on the dataset. After each cell finishes, you will see the overall accuracy for each dataset.



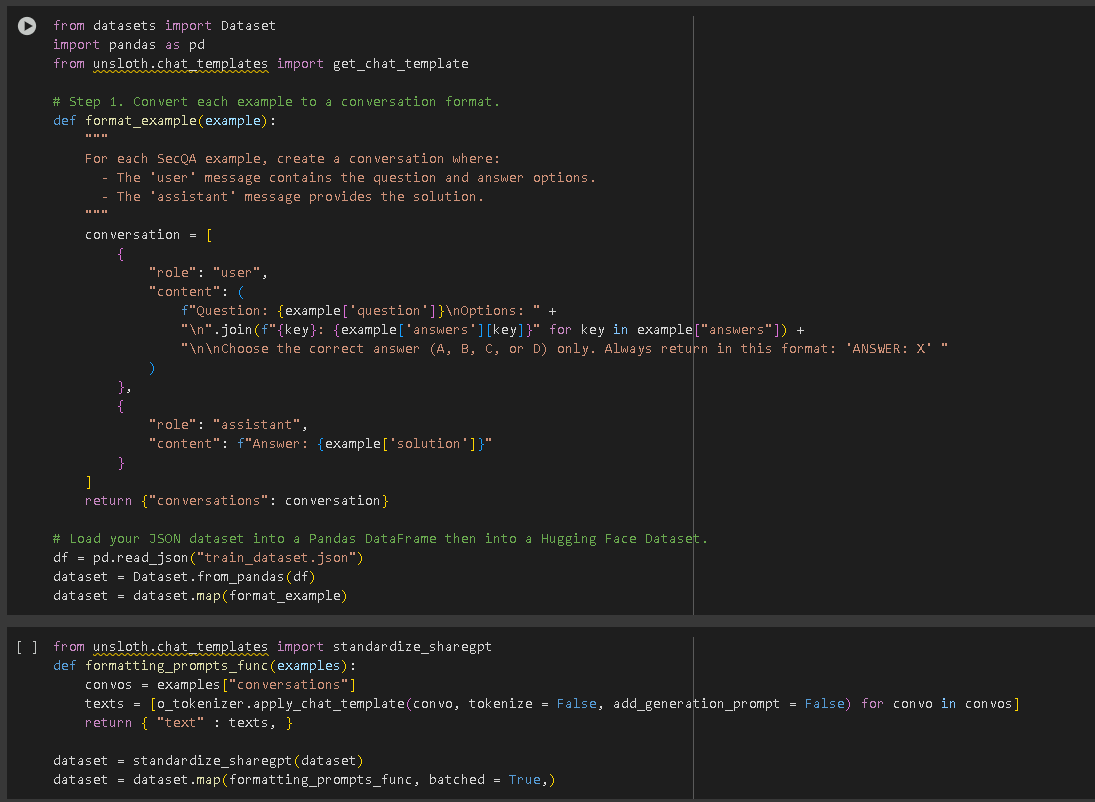
1. Fine-tune the model with a synthetically generated dataset and re-evaluating the LLM accuracy

After evaluating the LLM, we will start preparing the process of fine-tuning it using a simple 30 entries synthetic dataset generated using ChatGPT o3-mini-high. Run the following cell to set the layers and parameters that we want to train.

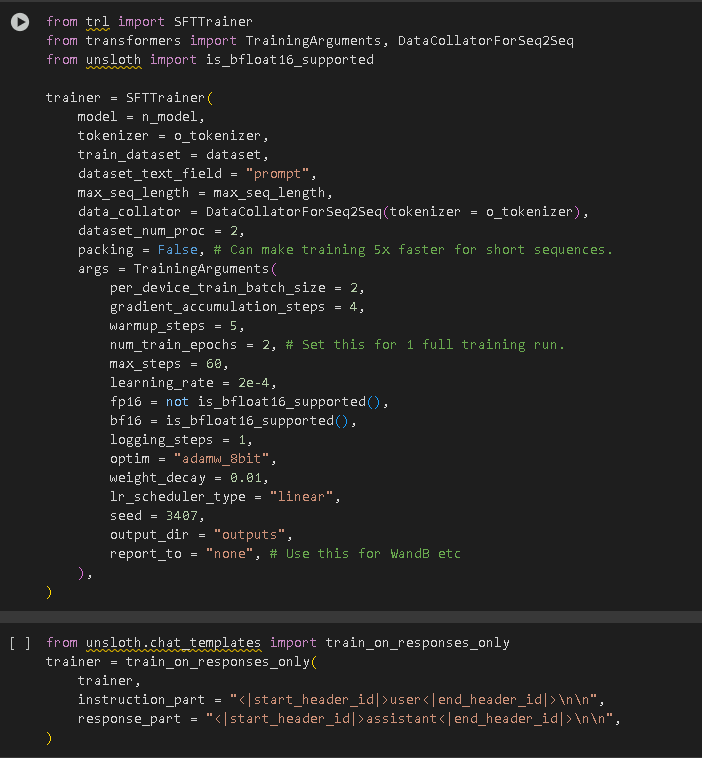


Using PEFT we can train only a part of the model, not the entire model, so we can cut a lot of required hardware resources to achieve comparable results to training an entire model.

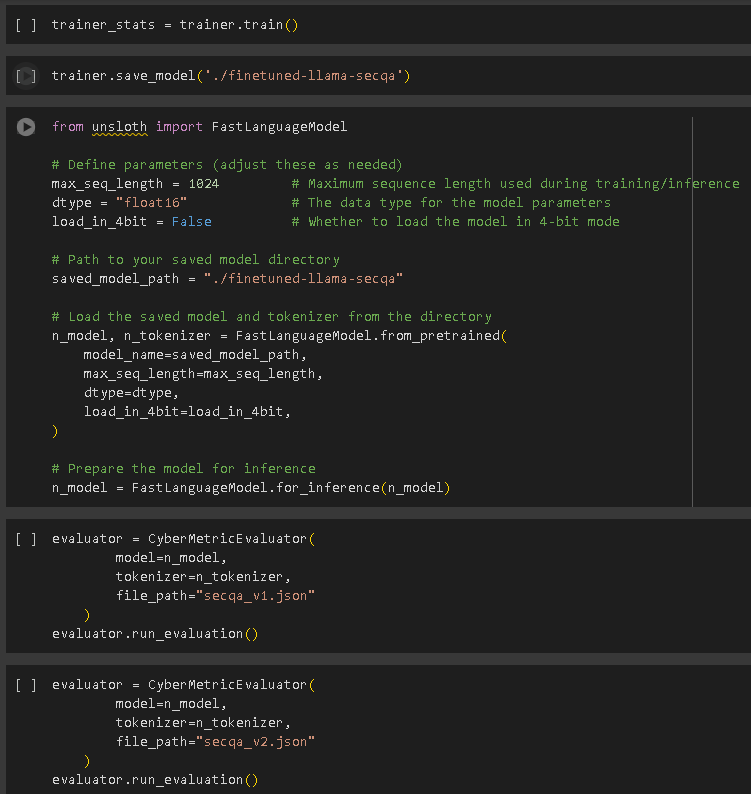
The following cells prepare the dataset in a structure of a conversation, similar to the structure any chatbot is used to function.



Then, similarly to the BERT Model, we have to set the parameters used to train the model. Additionally, we are marking the instruction part and the response part so the LLM knows what was the instruction provided and what was the result it should output for each row used to train.



Then, we will train the model, save it, load it and run the benchmarks on the same datasets we used on the original models.



## Key takeaways

1. Working with Python and GPU Powered Google Colab Notebooks
2. Loading and using open-source datasets to evaluate and fine-tune BERT and LLM Models
3. Interacting and integrating an online LLM Provider
4. Using LLM to evaluate logs, understand the reason for each decision the LLM takes
5. These basic implementations can be highly extended into Agentic behavior to take actions in order to protect systems. For example continuously analyzing the system logs and take actions based on the events that occurs.
6. We can create a Phishing Detection plugin for an email provided.