

# Introduction

---

Our goal was to create a model capable of accurately extracting and formatting information into predefined fields: **title**, **description**, **category**, **service**, **customerPriority**, **priority**, and **requestType**. After evaluating **T5**, **BART**, and **GPT-2** models, we chose to fine-tune the T5 model due to its versatility and superior performance in various tasks.

## Preliminary Training

---

To estimate the resources and time necessary for training, we conducted an initial training spike. Training Information:

- Model: T5-Small
- GPU: NVIDIA GeForce GTX 960
- Parameters: 60 million
- Dataset Size: 100 samples. This preliminary phase provided us with essential insights for our project's resource and time planning.

## First Training Strategy

---

After completing the spike, we decided to train the **T5-Large** model (which has 220 million parameters) using a dataset of **100,000** entries to achieve optimal training results. We decided to utilize the Azure ML platform for the training. The compute instance (VM) we decided to use on the ML platform is the Standard\_NC6s\_v3, equipped with 6 cores, 112 GB RAM, and a 336-GB disk, and it includes **1 NVIDIA Tesla V100 GPU**.

Unfortunately, we did not get access to a dataset from our industry partner. Consequently, we opted to train our model using a dataset of **10,000** entries generated with ChatGPT.

## First Training

---

Each of our team members manually created data using ChatGPT. We successfully generated a dataset of 1,000 entries. Consequently, we trained the T5-Small model. The model can be tested at: Hugging Face: <https://huggingface.co/TalkTix/t5-ticket-creator>

## Second Training

---

Unfortunately, the accuracy of the fine-tuned model was poor due to the small dataset. We then found a dataset of 8,000 entries that fit our task. We used the T5-Large model for this training. The model can be tested on Hugging Face at: <https://huggingface.co/TalkTix/t5-ticket-creator-8k>

## Second Training Strategy

---

Also, the accuracy of after second training did not meet our requirements. As a result, we decided to change our strategy. We chose to use/train a separate model for each ticket field. For this purpose, we switched from **text-generation** training to **text-classification**. We decided to use **RoBERTa-base**, as it is better suited for classification tasks. RoBERTa is specifically optimized for text classification, its architecture is fine-tuned to excel in understanding context and nuances in language. In contrast, **T5** is a powerful and flexible model for text generation, its strengths lie more in tasks like translation and summarization rather than the specific nuances of text classification. We also switched from manually generating test data to generating it via the ChatGPT API. We managed to generate approximately **53,000 entries** for our dataset.

For the title, affected person, and keywords, we found pre-trained models that we could use to generate them directly.

- The **title** was generated using the "czearing/article-title-generator" model from Hugging Face. [czearing/article-title-generator](#)
- The **affected person** was identified using the "dslim/bert-base-NER" model from Hugging Face. [dslim/bert-base-NER](#)
- **Keywords** were extracted using the "ml6team/keyphrase-extraction-kbir-inspec" model from Hugging Face. [ml6team/keyphrase-extraction-kbir-inspec](#)

The input text served as the **description**. For the fields of **Service**, **Request Type**, **Category**, **Priority**, and **Customer Priority**, we fine-tuned a text classification model for each of these fields.

## Final training

---

With the second strategy, we achieved far better accuracy compared to the first strategy.

- We trained the RoBERTa-base model to classify the input into the following **service** values: **SAP ERP**, **Atlassian**, **Adobe**, **Salesforce**, **Reporting**, **Microsoft Power Platform**, **Microsoft SharePoint**, **Snowflake**, and **Microsoft Office**. Our fine tuned model as well as a training evaluation for Service can be found and tested on Hugging Face at: <https://huggingface.co/TalkTix/roberta-base-service-type-generator-28k>
- We trained the RoBERTa-base model to classify the input into the following **customer priority** values: **Disruption but can work**, **Disruption cannot work**, **Disruption several cannot work** and **Disruption department cannot work**. Our fine tuned model as well a training evaluation for customer priority can be found and tested on Hugging Face at: <https://huggingface.co/TalkTix/roberta-base-customer-priority-type-generator-28k>
- We trained the RoBERTa-base model to classify the input into the following **request type** values: **Incident** or **Service Request**. Our fine tuned model as well a training evaluation for **request type** can be found and tested on Hugging Face at: <https://huggingface.co/TalkTix/roberta-base-request-type>
- We trained the RoBERTa-base model on 28000 entries to classify the input into the following **priority** values: **Low**, **Medium**, **High** and **Very High**. The accuracy wasn't good enough, so we trained the model again on a dataset of 55,000 entries. Unfortunately, the second model also demonstrated poor performance. This is likely due to a lack of versatility in our training data. The fine tuned models as well a training evaluation for **priority** can be found and tested on Hugging Face at:

<https://huggingface.co/TalkTix/roberta-base-customer-priority-type-generator-28k> and

<https://huggingface.co/TalkTix/roberta-base-priority-type-generator-55k>

- We trained the RoBERTa-base model to classify the input into 164 **category** values. Due to the high number of classes and the small dataset size, we faced challenges in achieving high accuracy. So we retrain the RoBERTa-base model to classify the input into **Technical Issues, Billing & Payment, Product Inquiries, Account Management, Polixy Questions, Complaints & Feedback**. With view classes the model achieves a really good accuracy. The fine tuned models as well a training evaluation for **category** can be found and tested on Hugging Face at:

<https://huggingface.co/TalkTix/roberta-base-category-type-generator-43k> and

<https://huggingface.co/TalkTix/roberta-base-category-type-generator-53k>