Ticketing system Task

1. Introduction

This report documents the implementation of topic modeling, label classification, and translation of ticket data using natural language processing (NLP) techniques. The project is designed to preprocess, clean, classify, and extract meaningful insights from the provided data, focusing on issues related to customer support tickets.

The steps include:

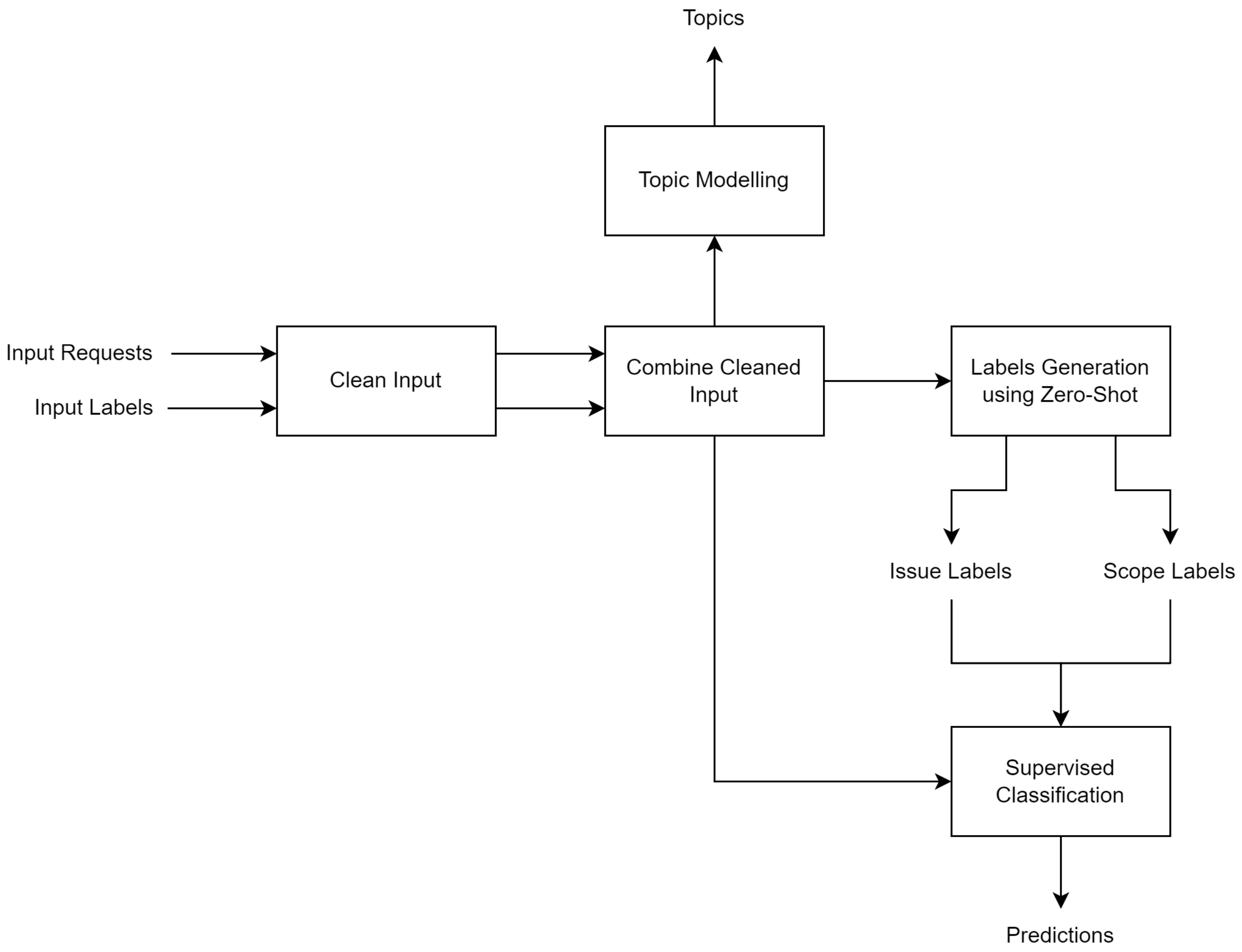
- Data cleaning and preprocessing
- Machine learning-based label generation using zero-shot classification
- Supervised classification using classical and transformer-based models
- Topic modeling using BERTopic with KMeans clustering

---

2. System Overview

The system performs end-to-end processing on a dataset of support tickets, consisting of requests and notes, to classify issue types and identify if requests are within scope. The system includes the following components:

1. **Data Preprocessing:**
    - Clean email text, removing unwanted artifacts like signatures, confidentiality notices, and irrelevant tokens.
    - Translate Arabic to English for better label prediction.
2. **Label Generation:**
    - Utilize zero-shot classification using the `facebook/bart-large-mnli` model to classify requests and notes into four types: Hardware Issue, Network Issue, Software Issue, and Parts Replacement.
3. **Supervised Classification:**
    - Classical models (e.g., Logistic Regression, XGBoost) for supervised classification of issues.
    - Transformers (BERT) for fine-tuned classification of issue types.
4. **Topic Modeling:**
    - Use **BERTopic** for topic modeling to identify key themes in the dataset.
    - Apply **KMeans** clustering within BERTopic to extract stable clusters of topics.
5. **Evaluation:**
    - Model performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion matrices.
    - Topic modeling visualization is used to provide insights into the underlying structure of the text data.



3. Data Preprocessing

3.1 Cleaning Text

The data is first cleaned to remove artifacts such as HTML tags, phone numbers, emails, and special characters. The cleaned text will serve as input for translation and model training. Key steps involved:

- Removal of email signatures, confidentiality notices, and artifacts like `_x000D_`.
- Removal of greetings (e.g., "Dear", "السلام عليكم").

Example of before and after cleaning:

Before: `PC have an issue:_x000D__x000D_Model: Dell_x000D_Ser No: C96PJ5J_x000D_issue: Hard
Disk_x000D__x000D_Best regards_x000D__x000D__x000D__x000D__x000D_تنبيه بإخلاء المسئولية: هذه
الرسالة ومرفقاتها معدة لاستخدام المُرسَل إليه المقصود بالرسالة فقط و قد تحتوي على معلومات سرية أو
محمية قانونيا. إن لم تكن الشخص المقصود، فإنه يُمنع منعا باتا أي عرض أو نشر أو استخدام غير مصرح
به للمحتوى. نرجو إخطار المُرسِل عن طريق الرد على هذا البريد الإلكتروني وإتلاف جميع النسخ الموجودة
لديك. تعد التصريحات و الآراء المذكورة في الرسالة خاصة بالمُرسِل و لا تمثل وزارة الصحة. كما لا تتحمل
الوزارة مسؤولية الأضرار الناتجة عن أي فيروسات قد تحملها هذه
الرسالة._x000D__x000D_CONFIDENTIALITY NOTICE: This e-mail message, including any attachments,
is for the sole use of the intended recipient(s) and may contain confidential and privileged
information or otherwise protected by law. If you are not the intended recipient, you are
notified that any unauthorized review, use, disclosure or distribution is strictly prohibited.
please notify the sender by replying to this email and destroy all copies of the original
message. Statements and opinions expressed in this Email are those of the sender, and do not
necessarily reflect those of Ministry of Health (MOH). Ministry of Health (MOH) accepts no`

`liability for damage caused by any virus transmitted by this Email._x000D__x000D__x000D_MOH Site. <http://www.moh.gov.sa>`

After: `PC have an issue: Model: Dell Ser No: C96PJ5J issue: Hard Disk Best regards`

3.2 Text Translation

Text is translated from Arabic to English to standardize and improve the classification results. The **Google Translator** API is used for this purpose, but other methods such as MyMemory can also be integrated for better control.

- **Translation Importance:** Translating the dataset ensures that the classification and topic modeling processes work effectively across both English and Arabic data.

Example:

Before translation: `السادة شركة المعمر المحترمين توجد لدينا مشكلة في الجهاز التالي: 1 أسم الجهاز :مكيف 3إسم القسم  غرفة السيرفر 4إسم المستشفى : مستشفى البكيرية العام 6وصف المشكلة: التكيف لايعمل`

After translation: `Dear Al-Moammar Company, we have a problem with the following device: 1 Device name: Air conditioner 3 Department name: Server room 4 Hospital name: Al-Bukayriyah General Hospital 6 Problem description: The air conditioner is not working`

4. Label Generation

The label generation process uses zero-shot classification based on the `facebook/bart-large-mnli` model. This model predicts the issue type based on the combined request and note content.

- **Issue Labels:** Hardware Issue, Network Issue, Software Issue, Parts Replacement.
- **Scope Labels:** Whether the request falls within or outside the support scope.

```
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli",
device='cuda')
```

Steps:

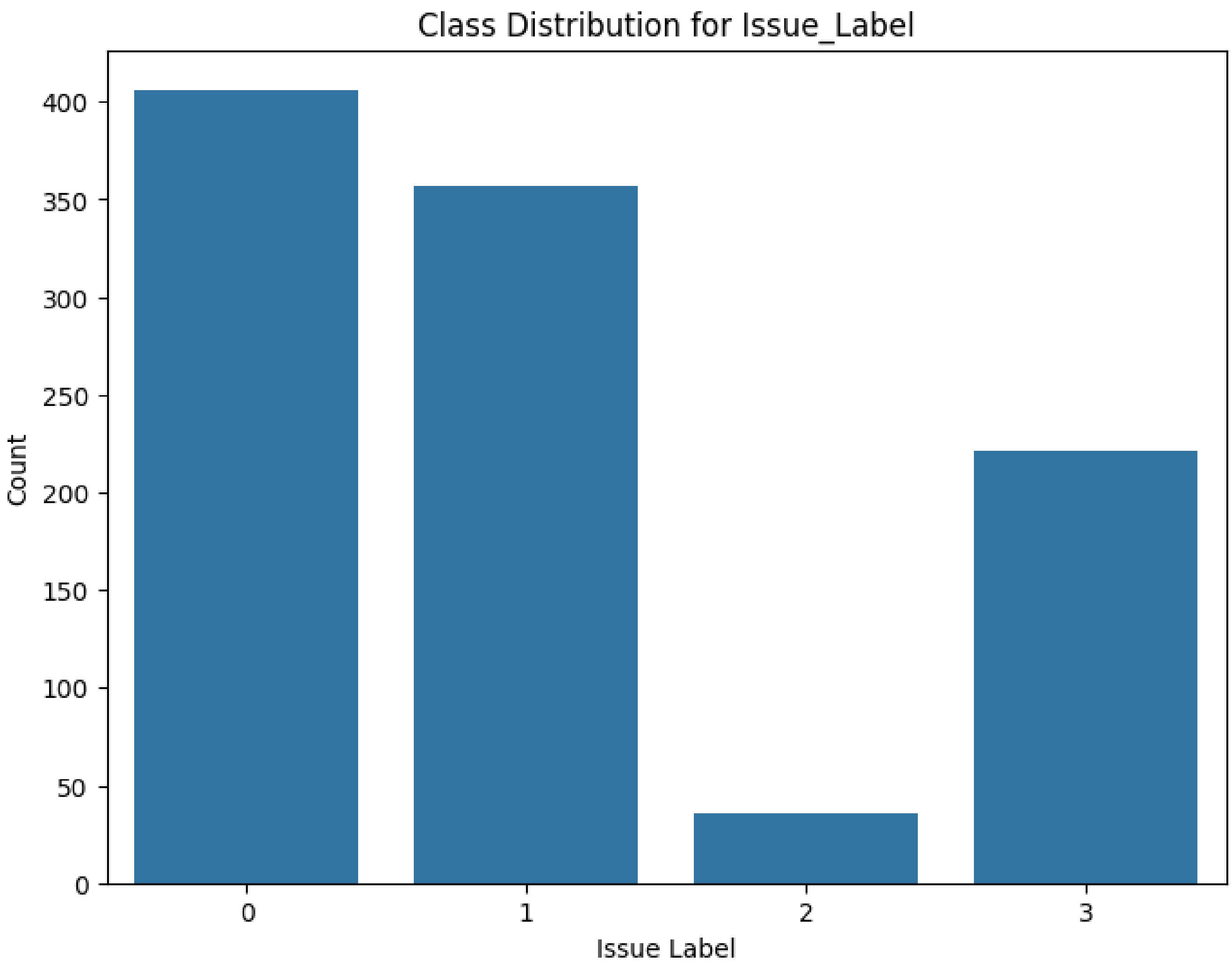
- Combine "Cleaned Request" and "Notes" into a single text field.
- Use zero-shot classification to assign labels based on the issue type and scope.

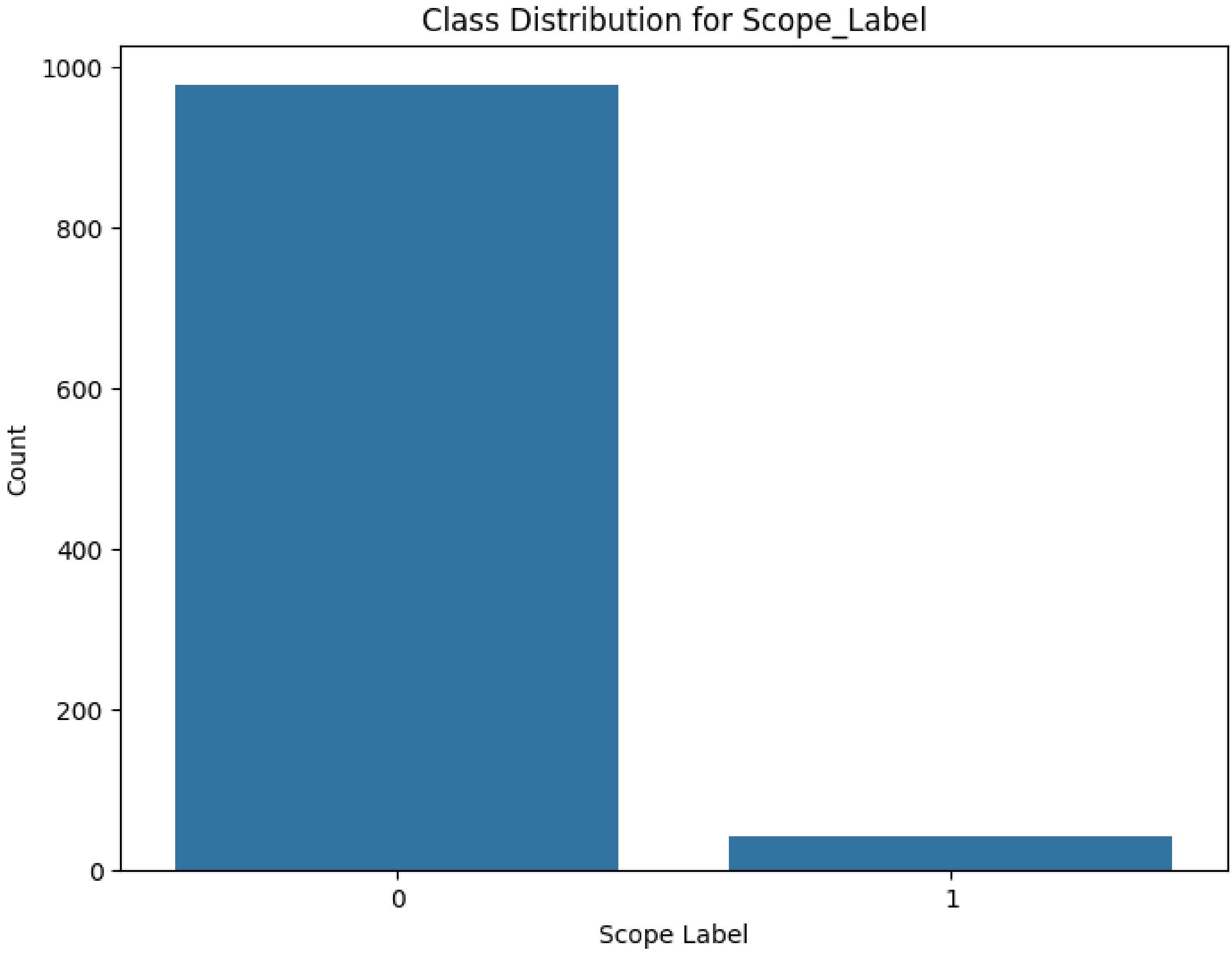
Label Example:

- `Issue_Label`: Hardware Issue
- `Scope_Label`: This request is within the support scope.

5. Supervised Classification

5.1 Checking Class imbalance





As class imbalance is present in the data in a severe way. So, SMOTE was applied on training data.

5.2 Classical Models

We use basic classical models such as **Logistic Regression** and **XGBoost** to classify ticket data into predefined categories. The text data is vectorized using **Count Vectorizer** and **TF-IDF**, and the labels are trained in a supervised fashion.

Steps:

1. Vectorize the cleaned text using `CountVectorizer`.
2. Convert the word vectors to TF-IDF.
3. Train the models using `train_test_split` data.

5.3 Transformer-based Models

We employ transformer-based models like **BERT** for sequence classification. The tokenization process is performed using `BertTokenizer` and the BERT model is fine-tuned on the support ticket data.

```
pythonCopy codetokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertForSequenceClassification.from_pretrained('bert-base-uncased', num_labels=4)
```

Training and evaluation are performed using the `Trainer` class from `transformers`, and metrics such as accuracy, precision, recall, and F1-score are computed to assess model performance.

[Proposed Figure 4: Confusion Matrix for BERT Model on Validation Set]

5.4 Results Analysis

For the Issue Label:

Model	**Accuracy**	**Precision**	**Recall**	**F1**
Logistic Regression	76.9	**77.3**	76.9	76.8
XGBoost	**77.4**	**77.3**	77.4	**76.8**
Transformer (BERT)	75	76.6	**75**	75.4

For the Scope Labe:

Model	**Accuracy**	**Precision**	**Recall**	**F1**
Logistic Regression	**97.5**	**97.2**	**97.5**	**97.2**
XGBoost	96	96	96	96
Transformer (BERT)	96	92	96	94

6. Topic Modeling

6.1 Topic Modeling using BERTopic

We apply **BERTopic** to extract key topics from the dataset. **KMeans** clustering is used within BERTopic to ensure the stability and accuracy of the topics generated. BERTopic allows us to visualize and understand the hidden topics within the combined text of requests and notes.

Steps:

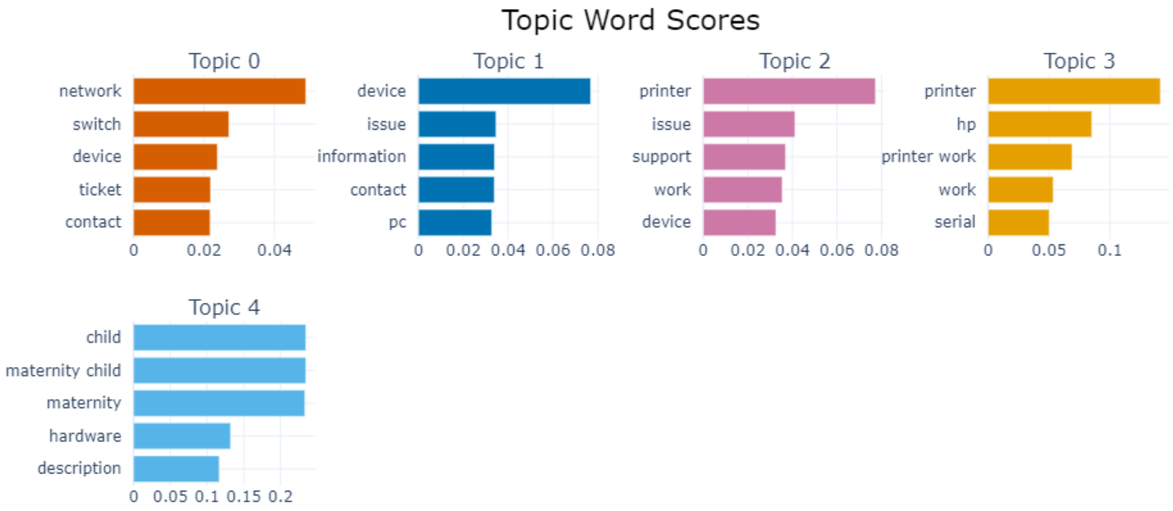
- Clean and lemmatize the text data.
- Fit BERTopic to the cleaned dataset.
- Visualize the topics using bar charts and topic heatmaps.

```
pythonCopy codetopic_model = BERTopic(umap_model=umap_model, hdbscan_model=kmeans_model)
topics, probabilities = topic_model.fit_transform(df_clean['Lemmatized Text'])
```

Hyperparameters:

- **UMAP parameters:** n_neighbors=15, n_components=5, min_dist=0.0.

- **KMeans Clustering:** n_clusters=5.



Topic Word Scores Summary:

1. **Topic 0**: Focus on **network issues** involving switches, devices, and support tickets.
2. **Topic 1**: **Device-related issues**, including PCs, with requests for information or contact.
3. **Topic 2**: **Printer issues**, emphasizing support and functionality.
4. **Topic 3**: **HP printer issues**, focusing on work status and serial numbers.
5. **Topic 4**: **Maternity and child care** support, possibly involving hardware in hospital settings.

Each topic reflects common areas of user support, including network, devices, and printers.

The topics generated from the BERTopic model provide insights into the types of issues raised in the requests and notes. However, We already covered that earlier so, no need to use them for classification.