Assignment Solution

Name : Rishabh Kumar Pandey
College : Vellore Institute of Technology, Vellore
Email : rishabhpandey7681@gmail.com
Contact no : +91 9555217343
Like for the GitHub Repository :

Output

1. Device a taxonomy of intent. (Sample is above) : Have a strong reasoning for why you choose the intent modeling the way you did.

Explanation :

For deriving the taxonomy, first we perform a topic modeling on all the customer reviews. Topic modeling is a type of statistical modeling that uses unsupervised Machine Learning to identify clusters or groups of similar words within a body of text. For example, a topic modeling algorithm could identify whether incoming documents are contracts, invoices, complaints, or more based on their contents.

1. Preprocess process the text to be fit for NLP tasks.
2. Perform sentiment analysis on the reviews. So that we only need to process the reviews which are not positive. Positive reviews don't need to be put in customer support categories.
3. I use the LDA model for the topic model from HggingFace to come up with the most common topics to our customer reviews.
   They are as follows

```
Topic 0:
business novo easy small great use love banking account bank
Topic 1:
account bank novo day money business card support email time
Topic 2:
novo good bank banking business experience service online far customer
Topic 3:
service customer great card issue app novo work quick debit
Topic 4:
novo bank business check deposit need account small work fee
```

4. From the above Five topics, we can design a hierarchy for our model.
   a. Card
   b. Customer Support -> Email
   c. Money -> Debit
   d. Money -> Credit
   e. Money -> Fee
   f. Services -> Online

The idea is to automate most of the things. Given that generally the number of reviews are large, doing manual work is not feasible. The above process follows a logical process of extracting reviews that need customer support(Using sentiment analysis) and then extracting the most important category for it using topic modeling (LDA).

Trade Offs: Using a model for labeling the review text, may not be 100% correct but for most cases it can be assumed to be working fine.

2. **Optional/Stretch : Find a way to auto extract intents from a model. You can prompt ChatGPT/ similar to actual Python/similar code. We are only interested in seeing your knowledge of applicability and not ingenuity in code.**
Explanation:
Intent auto extract can be done using the below two steps:

1. Process the review and get the not positive reviews for all. (Using sentiment Analysis)
2. Using NLP, extract the most common words or central ideas of the non-positive reviews.

The above steps give us an approximate idea of the customer support areas. Then we can manually process the exact hierarchy using our needs.

Using the above techniques, I can find the topic where each review belongs to :

```
zslda_output = lda.transform(X)
topic_assignments = np.argmax(lda_output, axis=1)

for review, topic in zip(negative_reviews, topic_assignments):
    print(f"Review: {review}\nAssigned Topic: {topic}\n")
```

```
Assigned Topic: 0

Review: novo lifesaver business banking convenience awesome customer service make stop financial shop
Assigned Topic: 2

Review: recommend small business owner
Assigned Topic: 0

Review: good experience business account company hassle problem set business account transfer money account smooth sailing good bank deal far
Assigned Topic: 2

Review: far novo month super easy onboard deposit
Assigned Topic: 0

Review: effectively affectivez work great eveeytomdu
Assigned Topic: 3

Review: good bank debit card not month say debit card arrive great potential ill grow pain ok
Assigned Topic: 3
```

3. Teach your model to classify the tickets according to your defined intent.
Explanation.

After the intents are defined, We reduce our problem to a multiclass label classification problem.

1. Labeling out reviews using defined Intents.

```python
def label_function(feedback):
    if 'password' in feedback.lower():
        return taxonomy['Account -> Lost password']
    elif 'void check' in feedback.lower():
        return taxonomy['Checks -> Mobile deposits -> Void checks']
    elif 'unauthorized transaction' in feedback.lower() or 'fraud' in feedback.lower():
        return taxonomy['Debit card -> Declined -> Unauthorized transactions -> fraud']
    elif 'unpaid' in feedback.lower() and 'conflict' in feedback.lower():
        return taxonomy['Invoices -> sent -> unpaid -> conflict']
    elif 'paid' in feedback.lower():
        return taxonomy['Invoices -> sent -> paid']
    elif 'unpaid' in feedback.lower() and 'pending' in feedback.lower():
        return taxonomy['Invoices -> sent -> unpaid -> pending']
    else:
        return -1  # Label for unknown or uncategorized feedback
```

2. After all the reviews we can use, We can use Models for sequence classification like BERT, to classify our reviews.

4. Optional/Stretch : Write notes on how you will "teach" new intents to the model as the new issues will keep coming. An example would be assume you launched a payroll product - and are now starting to see product reviews on issues in that product line.

1. Automating the Review processing pipeline to automatically process all the reviews collected till now.
2. Using NLP, for batch processing/online process to regularly update Our Model using the newly collected training data.
   Using this we obtain new Labels or new issues.

Now this newly found label can be added with other labels. So we came up with new Labels for our model, these new labels can be used to generate the output.

How will you evaluate the model performance ? What does truth mean for this model of intent production ?

Since the problem here is of Multilabel classification, the model can be evaluated using Accuracy or F1 scores. The truth means for this model as the label which is given by human after reading the Review.
Below are some metrics which can be used for evaluation :
Accuracy=Total number of predictions/ Number of correct predictions
Precision=(True Positives+False Positives)/True Positives
Recall=T(True Positives+False Negatives)True Negative
F1 Score=2 · (Precision+Recall)/(Precision · Recall)