Category Detection using Amazon Queries

Sai Mohan Majoju[1], Saikumar Jakkula[1]
[1] Master of Science in Data Science, University of New Haven, New Haven, CT
{sjakk2, smajo1}@unh.newhaven.edu

Abstract

Background: This NLP project is an end-to-end exploratory investigation of how advanced deep learning algorithms have the potential to help us classify the purpose of an Amazon customer's inquiry without the need for human interaction. After understanding the purpose or intent of the message, this scenario could make a chat-bot select what message should be sent to the consumer. The motivation for category detection using user queries is to improve the accuracy and relevance of search results and recommendations provided to users. When users search for information or interact with a system, they often use natural language queries that can be ambiguous or unclear. The business challenge we are exploring is that we must categorize a text query from an Amazon customer into one of the intended categories in the application. In addition, we need to choose a model that is lightweight and performs well in performing this task. **Methods**: The process typically involves analyzing the query or search term using various techniques such as natural language processing, machine learning, and pattern recognition, to determine its most likely category or type. Some common categories include products, services, locations, and people. Fine Tuning in this project involves taking a pre-trained language model, such as BERT, and fine-tuning it on a specific intent query detection task. Fine-tuning allows the model to learn from the specific data and improve accuracy. **Results**: The performance of intent detection models can be evaluated using several metrics, including precision, recall, F1-score, and accuracy. These metrics indicate how well the model can correctly identify the intended goal of a customer's query.

Introduction

Category detection using queries refers to the process of automatically identifying the category or type of query or search term based on its text content. This is an important task in many applications, such as e-commerce, where products need to be categorized and recommended to customers, information retrieval, recommendation systems where queries need to be matched with relevant results and natural language processing.

The process of category detection using queries involves analyzing the text of the query or search term, and using various techniques such as natural language processing, machine learning, and pattern recognition to determine its most likely category.

This can be a challenging task, as queries can be complex and ambiguous, and may contain multiple topics or categories. Overall, category detection using queries is an important area of research and development, with many potential applications in industries such as e-commerce, marketing, and information retrieval. Improvements in the accuracy and efficiency of category detection can lead to better recommendations, more relevant search results, and a better understanding of user behavior and preferences.

In this Project, we have used the Amazon Question and answer dataset to train the model with queries and understand the query of the customer and respond with the right category recommendations in the application. We used the dataset from Kaggle, which is a popular platform for data science competitions and provides a vast collection of publicly available datasets for use in research and analysis. In this report, we will describe a dataset obtained from Kaggle, including its origin, contents, and potential applications. to train the model and respond with appropriate category recommendations. The goal is to utilize state-of-the-art techniques and methodologies to enhance the accuracy and efficiency of category detection, resulting in better user experiences, higher conversion rates, and improved business outcomes.

BERT based Deep learning model is implemented in category detection using the Amazon dataset queries. BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained deep learning model developed by Google for natural language processing (NLP) tasks. It is based on the transformer architecture and can be fine-tuned for a wide range of NLP tasks, including text classification, sentiment analysis, question answering, and named entity recognition. BERT is unique in that it is pre-trained on a large corpus of unannotated text data using a masked language modeling (MLM) objective. This pre-training allows the model to learn general-purpose representations of text data, which can be fine-tuned for specific NLP tasks using a smaller labeled dataset. BERT's ability to capture the context of words in a sentence and the relationships between them has led to significant improvements in the accuracy of NLP models across a wide range of tasks. To use a pre-trained BERT model for a specific NLP task, the model can be fine-tuned by adding a task-specific output layer on top of the pre-trained model and training the entire model on a labeled dataset for the task. To use a pre-trained BERT model for a specific NLP task, the model can be fine-tuned by adding a task-specific output layer on top of the pre-trained model and training the entire model on a labeled dataset for the task.

Overall, BERT-based models have revolutionized NLP by providing a powerful and flexible framework for a wide range of NLP tasks. With the availability of pre-trained models and open-source libraries such as Hugging Face's Transformers, it has become easier than ever to use and customize BERT-based models for specific NLP tasks.

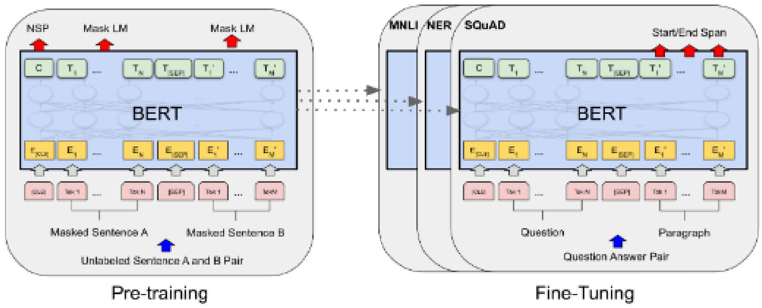


Figure 1: Architecture overview of BERT-base-uncased model.

BERT's bidirectional nature allows it to capture contextual information effectively. It can understand the meaning of a word based on its surrounding words, which is beneficial for intent detection, as it enables the model to consider the full context of the user query.

BERT can be fine-tuned on specific downstream tasks. The pretrained BERT weights are further trained on task-specific labeled data, allowing the model to adapt and specialize for tasks such as sentiment analysis, named entity recognition, question answering, and more.

Source of Data

The data scripts were obtained from Kaggle website which provides a vast collection of publicly available datasets for use in research and analysis. A link for the dataset is provided. https://www.kaggle.com/code/kagarg/chatbot/input?select=multi_questions.csv

The dataset consists of 1.3Million queries in which we have trained the model with 100,000 sample queries from them.

	Category
Appliances	9011
Arts Crafts and Sewing	21262
Automotive	89923
Baby	28933
Beauty	42422
Cell Phones and Accessories	85865
Clothing Shoes and Jewelry	22068
Electronics	314263
Grocery and Gourmet Food	19538
Health and Personal Care	80496
Home and Kitchen	184439
Industrial and Scientific	12136
Musical Instruments	23322
Office Products	43608
Patio Lawn and Garden	59595
Pet Supplies	36607
Software	10636
Sports and Outdoors	146890
Tools and Home Improvement	101088
Toys and Games	51486
Video Games	13307

There were total of eight columns in the dataset obtained from Kaggle platform QuestionType, Asin, AnswerTime, Unix-Time, Question, AnswerType, Answer, Category.

We have used two columns from the dataset to perform category detection. It contains columns named Question and Category the Question belongs to. There are twenty-one unique categories available in the dataset based on the queries.

We detected Unique Categories and mapped them to the dictionary after loading the dataset into Data Frame. Have used a data preprocessing procedure to clean and normalize the gathered data before turning it into a suitable format for model training.

There are different kinds of queries in the dataset in which the customers ask for. Some of the queries in the dataset in the dataset are observed as below.

	Question	labels
0	I have a 9 year old Badger 1 that needs replac...	0
1	model number	0
2	can I replace Badger 1 1/3 with a Badger 5 1/2...	0
3	Does this come with power cord and dishwasher ...	0
4	loud noise inside when turned on. sounds like ...	0
...	...	...
1384584	Flight Sim Seat Possibility? Is it possible to...	20
1384585	Steering Wheel Bar: Can you right foot brake a...	20
1384586	Playseat sale anytime soon? Is there any chanc...	20
1384587	On Sale: Does anyone think they when this play...	20
1384588	What is with the fluctuating price? Why the vo...	20

Fig: Above screenshot shows the sample questions from the dataset used for intent detection.

```python
ax = df['Category'].value_counts().plot.bar(figsize=(14,6),title="Classes vs Number of Records in each class")
```

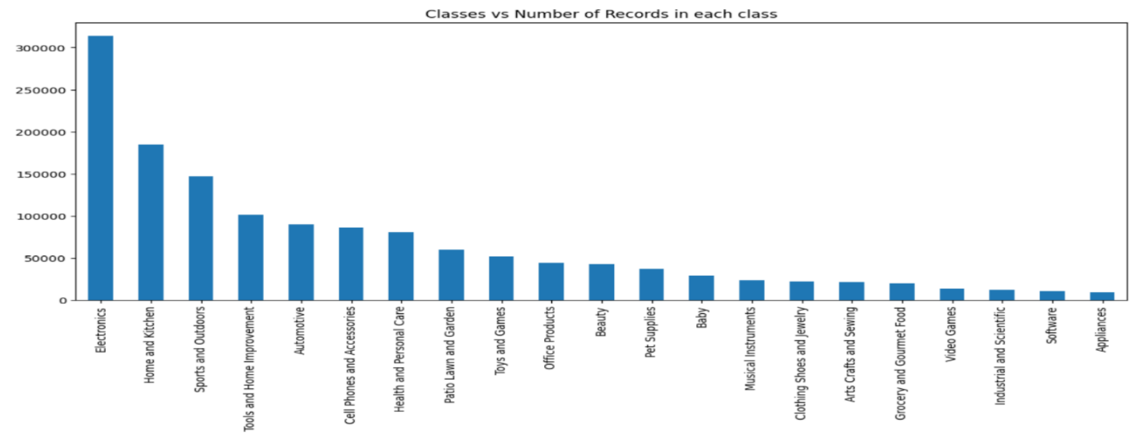


fig: Exploratory data analysis Bar Graph for Classes vs Number of Records in each class

The above exploratory data analysis is based on category and number of queries. X-axis describes all the categories Y-axis describes number of queries for each class or categories. Electronics have the highest number of queries in the dataset taken and appliances have the least number of queries from the taken dataset.

BERT-base Uncased Model

The implementation of a BERT-based deep learning model for category detection using the Amazon dataset queries is a significant contribution to this project. BERT, an acronym for Bidirectional Encoder Representations from Transformers, is a pre-trained deep learning model developed by Google that has been widely adopted in various NLP tasks. The model's transformer architecture allows it to effectively capture the context and relationships between words in a sentence, making it an ideal choice for text classification tasks, including category detection.

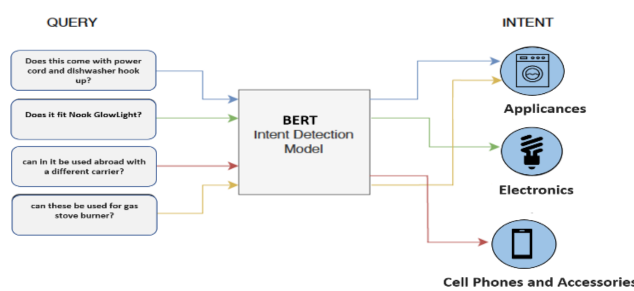
The Amazon dataset queries serve as an excellent benchmark for evaluating the performance of the BERT-based model in category detection. By fine-tuning the pre-trained BERT model on the labeled Amazon dataset queries, the model can learn to identify the relevant category of a given query accurately. This approach has shown significant improvements in the accuracy of category detection, as compared to traditional machine learning approaches.

To fine-tune a BERT-based model for a specific natural language processing (NLP) task, several steps are followed in this project. Firstly, the pre-trained BERT model and tokenizer were loaded. The tokenizer converts input text into tokens that can be processed by the model. Next, the tokenized inputs must be converted to PyTorch tensors, which are the input format required by the BERT model.

After the inputs are in tensor format, a PyTorch dataset must be created to hold the input data and corresponding labels for the task. This dataset is then passed to a PyTorch DataLoader, which helps efficiently load and process the data during training.

Finally, the BERT model can be fine-tuned using the dataset and data loader created in the previous steps. During fine-tuning, the BERT model is trained on the labeled dataset for the specific NLP task, allowing it to adapt to the nuances of the task and optimize its performance.

In summary, the process of fine-tuning a BERT-based model for an NLP task involves loading the pre-trained model and tokenizer, converting inputs to PyTorch tensors, creating a PyTorch dataset, creating a PyTorch DataLoader, and finally fine-tuning the model using the dataset and data loader. This approach has been proven to be highly effective for a wide range of NLP tasks, including text classification, sentiment analysis, and question answering.



Moreover, the BERT-based model can be fine-tuned for a wide range of NLP tasks, such as sentiment analysis, question answering, and named entity recognition. This flexibility has made it a popular choice among researchers and practitioners in the NLP community.

In conclusion, the implementation of a BERT-based deep learning model for category detection using the Amazon dataset queries has significant implications for the field of NLP. The model's ability to accurately classify queries into categories can improve the search experience for users and aid in business decision-making processes.

Performance Metrics

In natural language processing (NLP), several metrics are used to evaluate the performance of machine learning models. Some of the most commonly used metrics are accuracy, precision, recall, and F1-score. These metrics are particularly useful in classification tasks, where the goal is to assign a label or category to a piece of text data. We have used the following metrics to evaluate the performance of category detection in this project.

Accuracy measures the proportion of correctly classified instances out of all instances in the dataset. It is calculated as the ratio of the number of correct predictions to the total number of predictions made. While accuracy is a useful metric for balanced datasets, it may not be the most appropriate metric for imbalanced datasets, where one class is significantly more prevalent than the others.

```
[21] print("Accuracy", count_correct/len(list2))

     Accuracy 0.5912698412698413
```

Precision measures the proportion of correctly classified positive instances out of all instances that the model predicted as positive. It is calculated as the ratio of the number of true positives to the sum of true positives and false positives. High precision indicates that a model is making few false positive errors, which is desirable in many applications, such as spam detection.

Recall measures the proportion of correctly classified positive instances out of all instances that actually belong to the positive class. It is calculated as the ratio of true positives to the sum of true positives and false negatives. High recall indicates that a model is correctly identifying most of the positive instances in the dataset, which is important in applications such as disease diagnosis.

```
print(" Precision:", precision_macro)
print(" Recall:", recall_macro)
print(" F1 Score:", f1_macro)

Precision: 0.6026606115376923
Recall: 0.5913077774717389
F1 Score: 0.5931946961148574
```

F1-score is a harmonic mean of precision and recall and provides a single score that balances both metrics. It is calculated as the harmonic mean of precision and recall, where the weight of each metric is equal. F1-score is a useful metric for imbalanced datasets, where both precision and recall are important.

In summary, accuracy, precision, recall, and F1-score are important metrics for evaluating the performance of machine learning models in NLP. Each metric provides a unique perspective on the model's performance and is particularly useful for different types of classification tasks.

Conclusion

In conclusion, category detection is a critical task in natural language processing that involves identifying the category or topic of a given input. It plays an important role in various applications such as content filtering, information retrieval, and text classification.

There are different techniques and models that can be used for category detection, including rule-based approaches, traditional machine learning algorithms, and deep learning models. Deep learning models such as neural networks have shown to be highly effective in this task, and fine-tuning pre-trained language models has become a popular approach to improve their performance.

To create an effective category detection model, it is important to preprocess and prepare the data, select an appropriate model and hyperparameters, fine-tune the model on the specific task, and evaluate the model's performance using appropriate metrics.

Overall, category detection is a challenging and rewarding task that has numerous practical applications in natural language processing. As the field continues to grow and evolve, we can expect to see even more innovative approaches and models being developed to tackle this important problem.

Reference

https://medium.com/geekculture/banking-query-intent-detector-bbbb20c973bb

https://github.com/sengorajkumar/NLP-Intent-Classification

https://www.kaggle.com/code/kagarg/chatbot/input?select=multi_questions.csv

S. Ramachandran and S. Sankararaman, "Intent Recognition for Banking Domain Using Convolutional Neural Networks," in 2019 3rd International Conference on Intelligent Computing and Control Systems (ICICCS), 2019, pp.703-707.

M. Khodakarami, A. Ahmadvand, and A. Ghodsi, "Intent Classification for Banking Customer Support Using Transfer Learning and Active Learning," in 2020 2nd Conference on Artificial Intelligence and Data Engineering (AIDE),2020, pp. 63-68.

M. L. Prasad, N. M. Padma, and M. N. Giriprasad, "A Comparative Study of Intent Classification Techniques in the Banking Domain," in 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 98-104.