**Project report for the course DSCI-6004-03 Natural Language Processing (Fall-23)**

Category Detection using Amazon Queries

# DharmikSai Chilakamari, Jagadeesh Yarra

1 **Master of Science in Data Science, University of New Haven, New Haven, CT**

**dchil3@unh.newhaven.edu, jyarr3@unh.newhaven.edu**

# Abstract

**Background**: The goal of this NLP project is to investigate, from beginning to end, whether we can classify an Amazon customer's inquiry's aim without the assistance of a human by using sophisticated deep learning algorithms. In this case, a chatbot might choose the message to send to the customer after determining the objective or intent of the communication. The goal of employing user queries for category detection is to increase the precision and applicability of search results and user recommendations. When users interface with a system or seek out information, they regularly apply natural language inquiries, which might be imprecise or unclear. We are investigating the business challenge of having to classify a text query from an Amazon customer into one of the application's planned categories. In addition, we must select a model that is both lightweight and effective at performing this task.

**Methods**: To ascertain the most likely category or type, the procedure usually entails studying the query or search word using a wide range of methods, including machine learning, natural language processing, and pattern recognition. People, places, products, and services are a few such categories. Alright, In order to fine-tune this project, a pre-trained language model, such BERT, must be fitted to a particular intent query detection task. By fine-tuning, the model can become more accurate by learning from the data.

**Results**: Improved accuracy by data-driven learning.

Findings: A few metrics, such as accuracy, precision, recall, and F1-score, can be used to examine how well intent detection models work. These metrics show the model's accuracy in identifying the intended purpose of a customer's query.

# Introduction

The process of automatically determining the category, type, or search term of a query or search term based on its text content is known as category detection using queries. This is a crucial job in many applications, including natural language processing, information retrieval, recommendation systems (where users' queries must be matched with relevant results), e-commerce (where products must be classified and recommended to customers), and information retrieval.

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 might be a difficult undertaking because the inquiries themselves can have several topics or categories, be unclear and confusing. All things considered, category.

recognition through querying is a significant field for research and development, with a wide range of possible uses in fields including marketing, information retrieval, and e-commerce. Enhancements in the precision and effectiveness of category identification can result in more appropriate suggestions, pertinent search outcomes, and a deeper comprehension 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 Transform- ers, it has become easier than ever to use and customize BERT- based models for specific NLP tasks.

A diagram of a network

Description automatically generated

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

Because BERT is bidirectional, it can effectively capture contextual information. It can infer a word's meaning from the words around it, which helps with intent detection by allowing the model to consider the entire context of the user query.

BERT can also be trained for specific downstream roles. The BERT weights that have been pre-trained are then trained on the task-specific tagged data. This allows the model to learn and specialize for the following tasks sentiment analysis named entity recognition question answering.

# 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.

<https://www.kaggle.com/code/kagarg/chatbot/input?select=> [multi\_questions.csv](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.

A screenshot of a computer

Description automatically generated

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

After loading the dataset into the Data Frame, we found Unique Categories and mapped it to the dictionary. We have used a data preprocess to clean and standardize the collected data before converting it into the appropriate format for model training.

A pie chart with different colored circles

Description automatically generated

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. Electronics have the highest number of queries (22%) in the dataset taken and appliances have the least number of queries from the taken dataset.

# BERT-base Uncased Model

A BERT-based deep learning model has been implemented for category detection using Amazon datasets. BERT is a pre-trained model developed by Google for NLP tasks. Its transformer architecture captures context and relationships between words in a sentence, making it ideal for text classification.

The Amazon dataset queries are a good benchmark for evaluating BERT-based category detection. Pre-trained BERT model can accurately identify relevant categories of queries. This approach improves accuracy compared to traditional machine learning.

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 etc.

A diagram of a person

Description automatically generated

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.

# Performance Metrics

When it comes to NLP, there are several metrics that can be used to measure the performance of a machine learning model. Some of these metrics are:

* Accuracy
* Measurement of accuracy
* Measurement of precision
* Measurement of recall
* Measurement of F1 (F1-score)

These metrics are very useful when it comes to classifying text data. The purpose of classifying text data is to assign labels or categories to it. The following metrics have been used in this project to measure how well category detection works.

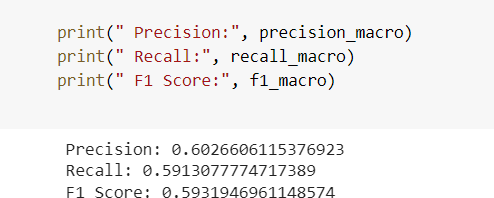
**Accuracy** is the percentage of instances in a dataset that are correctly classified. It’s calculated as the difference between the number of predictions that are true and the number that are false. While accuracy is a good metric for a balanced dataset, it might not be the best metric for an imbalanced dataset where one class dominates the rest.

A close up of numbers

Description automatically generated

**Precision** is the percentage of instances that were correctly classified as positive by the model out of all instances predicted as positive. Precision is calculated as the difference between the total number of true positives and the total number of false positives. A high level of precision means that the model is making only a small number of false positives, which is acceptable for many applications, including spam detection.

**Recall** is a measure of how many instances of a given class of positives belong to that class. It’s calculated as the number of true positives divided by the number of false positives. A high recall indicates that the model is correctly classifying most positive instances in the data set, which is important for applications like disease diagnosis.



**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, Machine learning models' performance in NLP is evaluated using metrics like accuracy, precision, recall, and F1-score, providing a unique perspective for different 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-](https://medium.com/geekculture/banking-query-intent-detector-bbbb20c973bb) [detector-bbbb20c973bb](https://medium.com/geekculture/banking-query-intent-detector-bbbb20c973bb)

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

[https://www.kaggle.com/code/kagarg/chatbot/input?sele](https://www.kaggle.com/code/kagarg/chatbot/input?select=multi_questions.csv) [ct=multi\_questions.csv](https://www.kaggle.com/code/kagarg/chatbot/input?select=multi_questions.csv)

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 .

**GitHub Repository Link**:

https://github.com/codemasterds/NLP