ESTIMATING STOCK KEEPING UNITS USING MACHINE LEARNING

Introduction:

Stock Keeping Units (SKUs) are unique product codes assigned to individual items within a company’s inventory. Estimating the number of SKUs required for a given business is an important task in inventory management as it directly impacts a company’s ability to effectively manage their inventory levels and make informed decisions about purchasing and stock management. Machine learning is a powerful tool that can be used to estimate SKUs in a more accurate and efficient manner compared to traditional methods. By using algorithms that can learn patterns and relationships in data, machine learning models can make predictions about the number of SKUs a company will need in the future, taking into account factors such as sales trends, seasonal fluctuations, and demand patterns. One common approach to using machine learning for SKU estimation is to use regression models, which can predict the number of SKUs required based on historical sales data. Another approach is to use time series forecasting methods, which can predict future trends and patterns in sales data. The success of machine learning models in estimating SKUs depends on the quality and quantity of the data available, as well as the choice of algorithms used. It is important to carefully select and pre-process the data to ensure that the models are trained on relevant and accurate information. Overall, the use of machine learning in SKU estimation can lead to more accurate and efficient inventory management, resulting in reduced costs and increased profits for businesses.

1.2 Purpose

The purpose of estimating stock keeping units (SKUs) using machine learning is to help businesses improve their inventory management and decision-making processes. By using machine learning algorithms to analyze data and make predictions about the number of SKUs required, businesses can:

1. Increase accuracy

2. Improve efficiency

Overall, the purpose of estimating SKUs using machine learning is to help businesses improve their inventory management processes, reduce costs, and make data-driven decisions to increase profitability.

2. LITERATURE SURVEY

Estimating stock keeping units (SKUs) using machine learning is a complex task that can come with a number of challenges. Some of the existing problems include:

1. Lack of data

2. Complexity of the problem

2.2 Proposed Solution

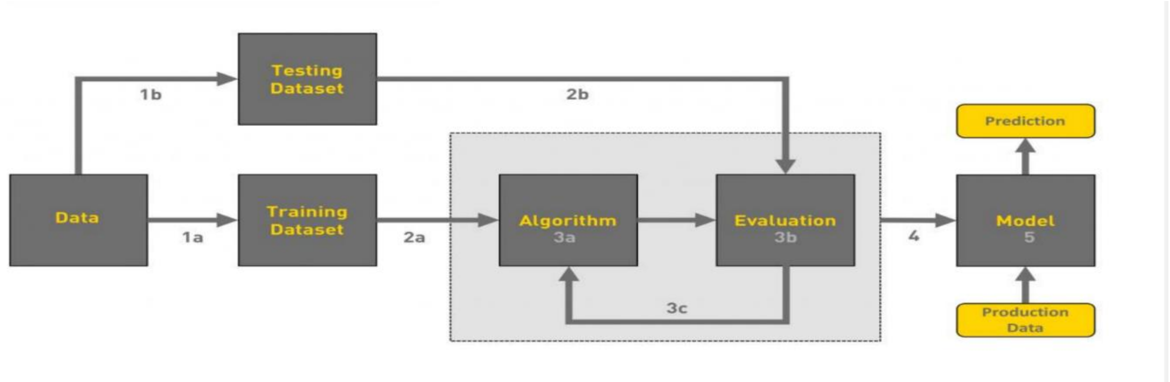
To address the challenges associated with estimating stock keeping units (SKUs) using machine learning, the following solutions can be proposed:

1. Data quality and quantity: Ensure that the data used to train the machine learning model is of high quality and sufficient quantity. This can be achieved by collecting a large and diverse dataset, and carefully pre-processing the data to remove any irrelevant or inaccurate information.

2. Feature engineering: Feature engineering is the process of creating new features or transforming existing features to improve the performance of the machine learning model. This can help to capture complex relationships between variables, and improve the accuracy of the SKU estimation.

3. THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware/Software Designing

Software Requirements:

Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It was created by Guido van Rossum , and first released on February 20, 1991. Its high-level built in Data structures, combined with dynamic typing and dynamic binding , make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing Components together. Python’s simple, easy to learn syntax emphasizes readability and therefore Reduces the cost of program maintenance. Python supports modules and packages, which encourages Program modularity and code reuse. The Python interpreter and the extensive standard library are Available in source or binary form without charge for all major platforms, and can be freely Distributed.

Anaconda Navigator

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda Distribution.Navigator allows you to launch common Python programs and easily manage conda Packages, environments, and channels without using command-line commands. Navigator can search For packages on Anaconda Cloud or in a local Anaconda Repository.Conda is an open-source, Crossplatform, package management system. For this project, we will be using Jupyter notebook and Spyder.

Jupyter Notebook

The Jupyter Notebook App is a server-client application that allows editing and running notebook Documents via a web browser. The Jupyter Notebook App can be executed on a local desktop Requiring no internet access or can be installed on a remote server and accessed through the internet.

Spyder

Spyder is an open-source cross-platform integrated development environment (IDE) for scientific Programming in the Python language. Spyder integrates with a number of prominent packages in the Scientific Python stack, including NumPy, SciPy, Matplotlib, pandas, IPython, SymPy and Cython, as Well as other open-source software. It is released under the MIT license.

Flask

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself.

Hardware Requirements:

Operating System: Windows 7 or above

Processor: Intel Core i5 and above

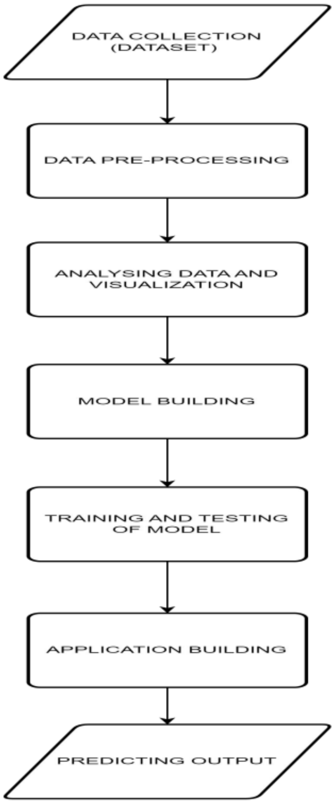
RAM: 4 Gb and above

Storage Space Required: 10gb and above

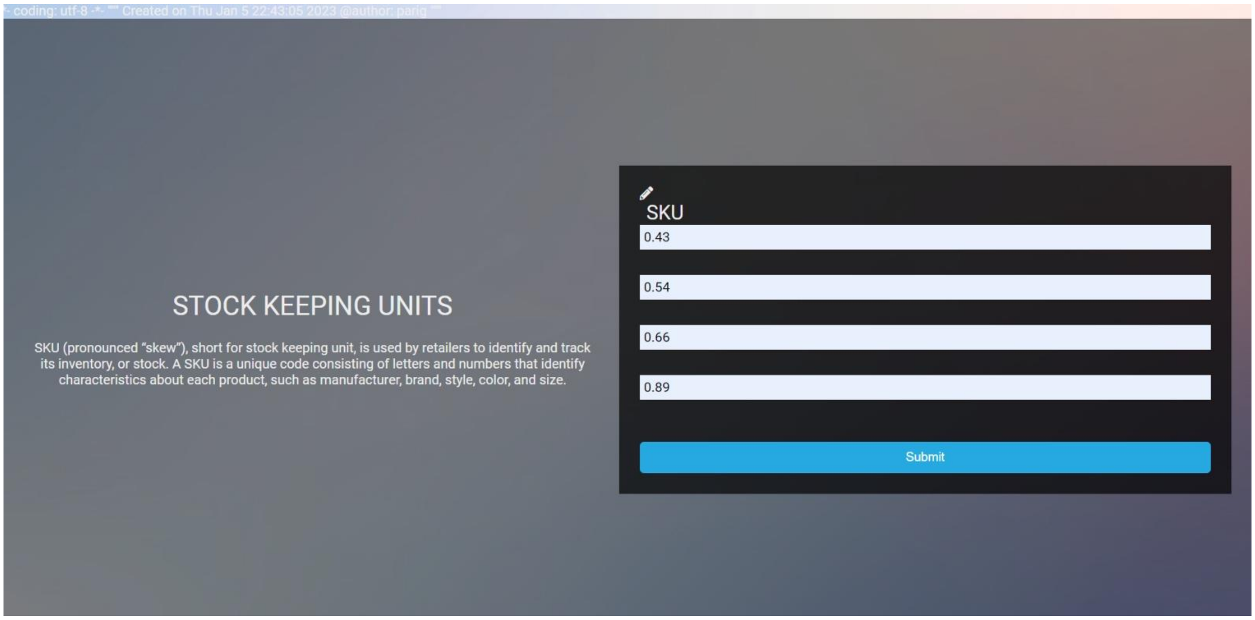
4. EXPERIMENTAL INVESTIGATIONS

The text data need to be organized before proceeding with the project. We will be using PS_20174392719_1491204439457_logs.csv dataset file to fetch the text data of training data. The datas are to be preprocessed in a way such that there is no empty field or outliers. We will create a Function that uses the pre-trained model for predicting custom outputs. Then we have to test and train the model. After the model is build, we will be integrating it to a web application build in flask.

5. FLOWCHART


DATA COLLECTION (DATASET)
DATA PRE-PROCESSING
ANALYSING DATA AND VISUALIZATION
MODEL BUILDING
TRAINING AND TESTING OF MODEL
APPLICATION BUILDING
PREDICTING OUTPUT

6. RESULT



- coding: utf-8 -- """ Created on Thu Jan 5 22:43:05 2023 @author: parig """

STOCK KEEPING UNITS

SKU (pronounced “skew”), short for stock keeping unit, is used by retailers to identify and track its inventory, or stock. A SKU is a unique code consisting of letters and numbers that identify characteristics about each product, such as manufacturer, brand, style, color, and size.

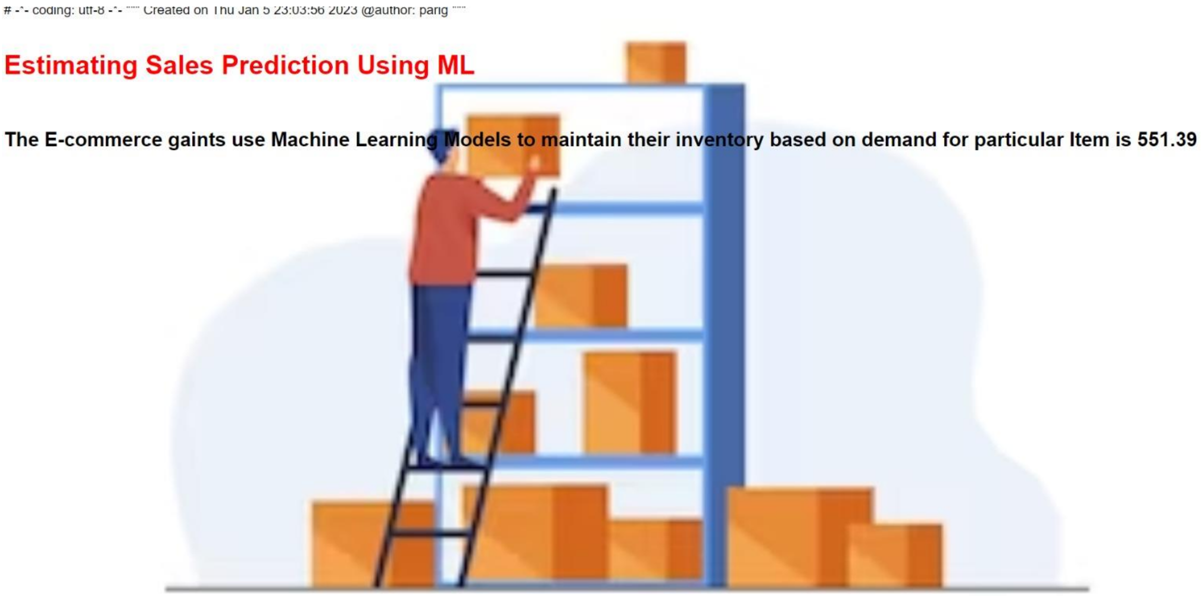
SKU

0.43

0.54

0.66

0.89



-*- coding: utf-8 -*- """ Created on Thu Jan 5 23:03:56 2023 @author: parig """

Estimating Sales Prediction Using ML

The E-commerce gaints use Machine Learning Models to maintain their inventory based on demand for particular Item is 551.39

7. ADVANTAGES

Improved accuracy: Machine learning algorithms can analyze large amounts of data and identify patterns that humans might miss, leading to more accurate predictions of SKU demand. Real-time analysis: Machine learning models can process data in real-time, allowing for more up-to-date predictions and enabling businesses to respond more quickly to changes in demand.

1. Automated forecasting: Machine learning algorithms can automate the forecasting process, reducing the time and resources needed for manual forecasting and minimizing the risk of human error.

2. Scalability: Machine learning models can be easily scaled to handle large amounts of data, making them well-suited for large organizations with a large number of SKUs.

3. Improved efficiency: By automating the forecasting process, machine learning can reduce the time and resources needed for forecasting, allowing businesses to allocate those resources to other areas.

DISADVANTAGES

1. Complexity: Machine learning algorithms can be complex and difficult to understand for those without a strong technical background. This can make it difficult for businesses to interpret the results and make informed decisions based on the forecasts.

2. Data requirements: Machine learning models require large amounts of data to be trained, and the quality of that data is critical to the accuracy of the predictions. If the data is biased or incomplete, the predictions may be inaccurate.

3. Initial investment: Implementing machine learning algorithms can be costly, requiring investments in technology, hardware, and personnel.

4. Maintenance: Machine learning models require ongoing maintenance and updates to ensure that they remain accurate and relevant. This can add to the ongoing costs associated with using machine learning for SKU demand forecasting.

5. Model limitations: Machine learning models are only as good as the data and algorithms they are based on. If the data is limited or the algorithm is not well-suited to the problem, the predictions may be inaccurate.

8. APPLICATIONS

Inventory management: Machine learning can be used to predict SKU demand, allowing businesses to manage their inventory more effectively by avoiding stock shortages or overstocks.

1. Price optimization: Machine learning algorithms can be used to optimize pricing for specific SKUs, helping businesses to maximize their profits while still remaining competitive.

2. Promotional planning: Machine learning algorithms can help businesses plan and optimize promotions, helping to drive sales and increase revenue.

3. Fraud detection: Machine learning algorithms can be used to detect fraudulent activities, such as fake or altered SKUs, protecting businesses from losses and maintaining the integrity of their products.

9. CONCLUSION

In conclusion, machine learning has the potential to be a valuable tool for estimating stock keeping units (SKUs) in the retail and supply chain industries. With its ability to analyse large amounts of data, automate forecasting, and provide real-time predictions, machine learning can help businesses make more informed decisions, reduce waste, and improve efficiency. the end, the best approach will depend on the specific needs and constraints of each business, but for those that choose to implement machine learning for SKU demand forecasting, the potential benefits can be substantial.

10. FUTURE SCOPE

The future of estimating stock keeping units (SKUs) using machine learning looks very promising. As technology continues to advance, machine learning algorithms will become even more powerful and sophisticated, allowing businesses to make more accurate predictions and more informed decisions.

One of the key areas of growth for machine learning in SKU demand forecasting will be the integration of artificial intelligence and the Internet of Things (IoT). The increasing number of connected devices will provide businesses with a wealth of data on customer behavior and preferences, allowing them to make more informed decisions about what SKUs to stock and how much to order. Overall, the future of estimating SKUs using machine learning is very exciting, and businesses that embrace this technology will be well-positioned to compete and succeed in the coming years.

11. BIBLOGRAPHY

https://www.researchgate.net/publication/354937786_Online_Tra
Estimating_stock_keeping_System_Based_on_Machine_Learning

12. APPENDIX

A. SOURCE CODE

APP.PY

```
from flask import Flask, jsonify, make_response, request, abort, render_template

import pandas as pd

import pickle

model = pickle.load(open( "sales_demand_ forecasting.pkl", "rb"))
app = Flask(__name__)

@app.route('/')
def home():
    return render_template('index.html')

@app.route("/y_predict", methods=['POST', 'GET'])
def y_predict():
    x = [[float(x) for x in request.form.values()]]

    print(x)

    cols=["day_1","day_2","day_3","day_4"]

    print(x)
    pred = model.predict(x)

    print(pred[0])
    return render_template('result.html', prediction_text=pred[0])


if __name__ == "__main__":
    app.run()
```

INDEX.HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Demand Forecast form</title>
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.4.1/css/all.css" integrity="sha384-5sAR7xN1Nv6T6+dT2mhtzEpVJvfS3NScPQTrOxhwjIuvcA67KV2R5Jz6kr4abQsz" crossorigin="anonymous">
    <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700" rel="stylesheet">
    <style>
      html, body {
      min-height: 100%;
      }
      body, div, form, input, select, p {
      padding: 0;
      margin: 0;
      outline: none;
      font-family: Roboto, Arial, sans-serif;
      font-size: 16px;
      color: #eee;
      }
      body {
      background: url("https://img.rawpixel.com/s3fs-private/rawpixel_images/website_content/v960-ning-30.jpg?w=800&dpr=1&fit=default&crop=default&q=65&vib=3&con=3&usm=15&bg=F4F4F3&ixlib=js-2.2.1&s=63dd5f402645ef52fb7dfb592aec765a") no-repeat center;
      background-size: cover;
      }
      h1, h2 {
```

```
text-transform: uppercase;
font-weight: 400;
}
h2 {
margin: 0 0 0 8px;
}
.main-block {
display: flex;
flex-direction: column;
justify-content: center;
align-items: center;
height: 100%;
padding: 25px;
background: rgba(0, 0, 0, 0.5);
}
.left-part, form {
padding: 25px;
}
.left-part {
text-align: center;
}
form {
background: rgba(0, 0, 0, 0.7);
}
.title {
display: flex;
align-items: center;
```

```css
margin-bottom: 20px;
}
.info {
display: flex;
flex-direction: column;
}
input, select {
padding: 5px;
margin-bottom: 30px;
background: transparent;
border: none;
border-bottom: 1px solid #eee;
}
input::placeholder {
color: #eee;
}
option:focus {
border: none;
}
option {
background: black;
border: none;
}
.checkbox input {
margin: 0 10px 0 0;
vertical-align: middle;
}
```

```
.checkbox a {
color: #26a9e0;
}
.checkbox a:hover {
color: #85d6de;
}
.btn-item, button {
padding: 10px 5px;
margin-top: 20px;
border-radius: 5px;
border: none;
background: #26a9e0;
text-decoration: none;
font-size: 15px;
font-weight: 400;
color: #fff;
}
.btn-item {
display: inline-block;
margin: 20px 5px 0;
}
button {
width: 100%;
}
button:hover, .btn-item:hover {
background: #85d6de;
}
```

```
      @media (min-width: 568px) {
      html, body {
      height: 100%;
      }
      .main-block {
      flex-direction: row;
      height: calc(100% - 50px);
      }
      .left-part, form {
      flex: 1;
      height: auto;
      }
      }
  </style>
  </head>
  <body>

    <div class="main-block">
     <div class="left-part">

        <h1>Stock Keeping Units</h1>
        <p>SKU (pronounced “skew”), short for stock keeping unit, is used by retailers to identify and track its inventory, or stock. A SKU is a unique code consisting of letters and numbers that identify characteristics about each product, such as manufacturer, brand, style, color, and size.</p>
 </div>
      <form action="{{ url_for('y_predict')}}" method="post">
      <form action="/">
```

```
          <i class="fas fa-pencil-alt"></i>

          <h2>sku</h

        <div class="info">

          <input class="day_1" type="text" name="day_1" placeholder="Enter any Value">

          <input type="text" name="day_2" placeholder="Enter any Value">

          <input type="text" name="day_3" placeholder="Enter any Value">

          <input type="text" name="day_4" placeholder="Enter any Value">

        </di

   <button type="submit" href="/">Submit</button>

      </form>

      {{prediction_text}}

    </div>

  </body>

</html>
```