INDIVIDUAL TASK-3

Features extraction thought Experiment : Select a dataset (e.g., photos shopping lists) and describes which features would be important to a machine learning model

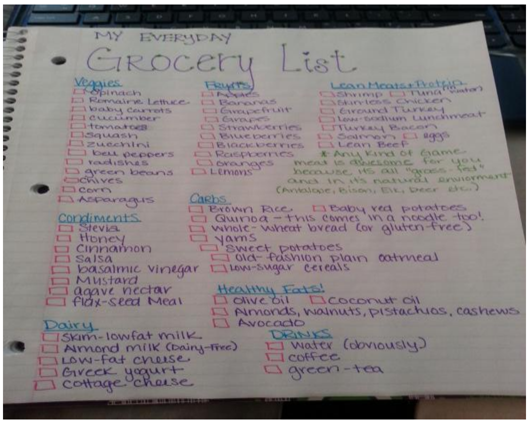
1. Introduction & Problem Definition:-

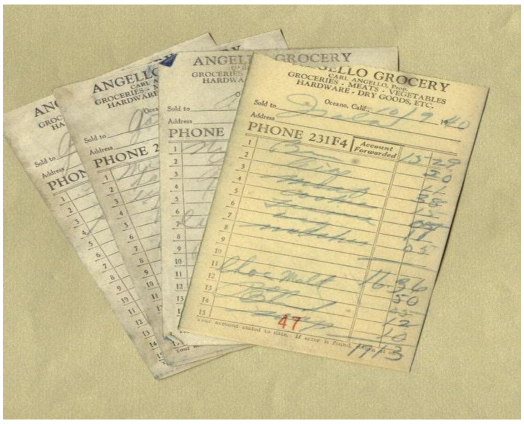
1.1 Dataset Description

Assume we collect **10,000 smartphone photos of handwritten grocery shopping lists**. The images vary in:

- Handwriting styles
- Lighting conditions
- Paper types
- Camera angles
- Spelling mistakes
- Abbreviations

Example images:





1.2 Learning Tasks

We define two machine learning tasks:

Task A – Item Extraction (Structured Prediction)

- Input: Image
- Output: List of items + quantities

Task B – Cost Prediction (Regression)

- Input: Extracted items
- Output: Total estimated cost

This makes the problem multimodal:

- Computer Vision
- Natural Language Processing
- Structured Data Modeling

2. Visual Feature Extraction:-

To extract text from images, we first process visual features.

2.1 Low-Level Features

These are raw or basic image features:

- Pixel intensity values
- RGB color channels
- Grayscale intensity
- Gradients
- Edges (Canny edge detection)
- Corners (Harris detector)
- Stroke thickness

Why important?

- Letters are defined by edges and curves.
- Text regions differ from background in texture and contrast.

2.2 Mid-Level Features

These capture patterns:

- Connected components (grouping strokes into letters)
- Bounding boxes for text lines
- Aspect ratio of characters
- Line spacing
- Word spacing

These features help distinguish:

- Separate words
- Quantities vs item names
- Columns vs lists

2.3 High-Level Features (Deep Learning)

Using CNNs or Vision Transformers, the model learns:

- Letter shapes
- Word shapes
- Common handwriting patterns

3. Semantic Features:-

Using embeddings:

- Word2Vec
- GloVe
- Transformer embeddings

These capture semantic similarity:

- “soda” close to “coke”
- “tuna” close to “fish”

This allows generalization beyond exact vocabulary matches.

4. Structural Features:-

Patterns in the text:

- Number + noun → likely quantity
- Bullet symbols
- Line breaks
- Indentation

Example patterns:

- “2 milk”
- “milk x2”
- “milk (2)”

The model learns these relationships.

5. Features for Cost Prediction:-

After extracting structured data:

Item	Quantity
Milk	2
Bread	1
Eggs	6

6. Item-Level Features:-

- Item category (dairy, produce, meat)
- Average market price
- Brand (if detected)
- Organic vs regular
- Package size

7. Aggregate Features:-

- Total number of items
- Total quantity
- Number of premium items
- Number of produce items
- Price variance

8. Contextual Features:-

If metadata is available:

- Store location
- Day of week
- Season
- Inflation index
- Regional price differences

9. Feature Evaluation & Model Considerations:-

9.1 What Makes a Good Feature?

A good feature:

- Reduces uncertainty about output
- Is robust to noise (messy handwriting)
- Generalizes to new stores/items
- Is computationally efficient

9.2 Feature Engineering vs Feature Learning

➢ **Traditional ML**

- Manual features (edge counts, color histograms)
- TF-IDF vectors
- Domain-specific rules

➢ **Deep Learning**

- Learns hierarchical features automatically
- Less manual engineering
- Requires more data

9.3 Potential Challenges

- Poor lighting
- Extreme handwriting styles
- Abbreviations ("tmlk")

10.Conclusion:-

In this thought experiment, we demonstrated how feature extraction depends on:

1. Input modality (image + text)
2. Learning task (extraction + regression)
3. Model type (traditional vs deep learning)

The same dataset requires:

- Visual features to read text
- Linguistic features to interpret meaning
- Structured features to predict cost