I

VIDEO PRESENTATION | FINAL YEAR PROJECT MAY 2022 | GRACE Y. LIN

Automatic Product Extraction, Classification, and Analysis of Receipt Data

URN: GL00233 | GL00233@SURREY.AC.UK | SUPERVISOR: PROF. HELEN TREHARNE

II

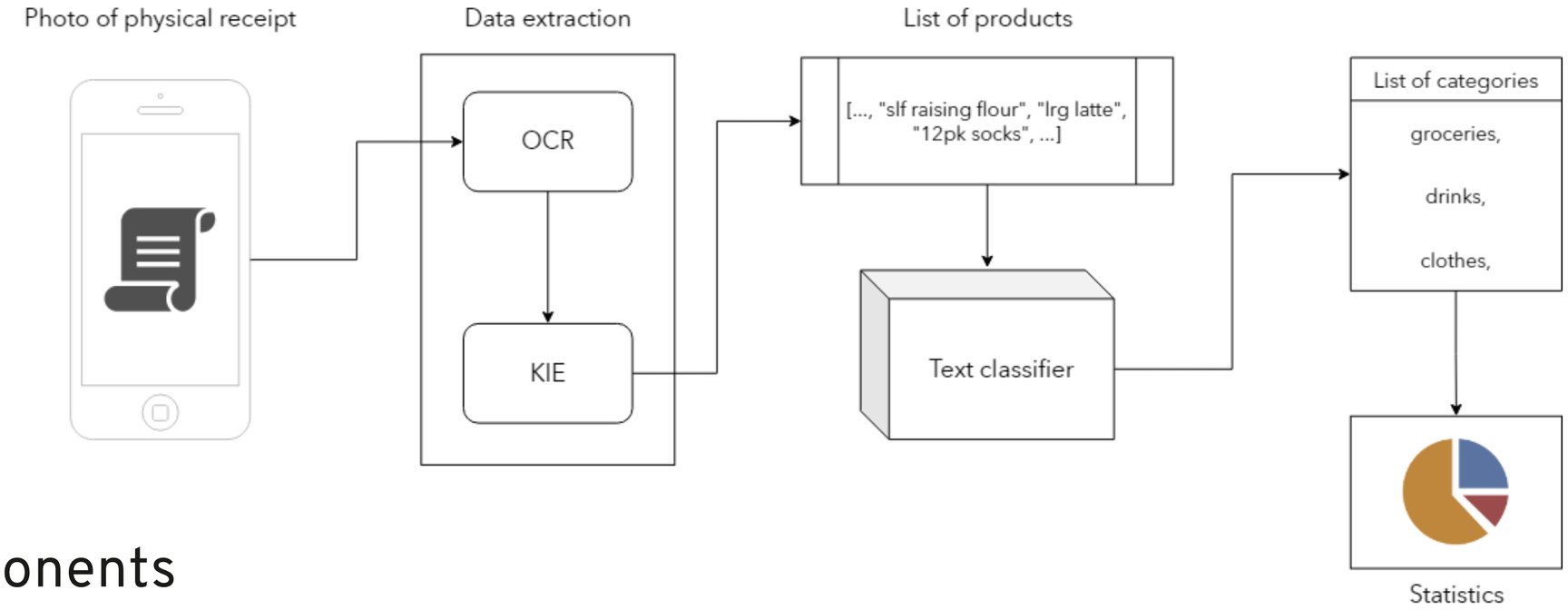
Problem Background

Receipt OCR, Data Extraction, & Classification Research Project

Project overview

GL

An end-to-end pipeline for automatic receipt understanding

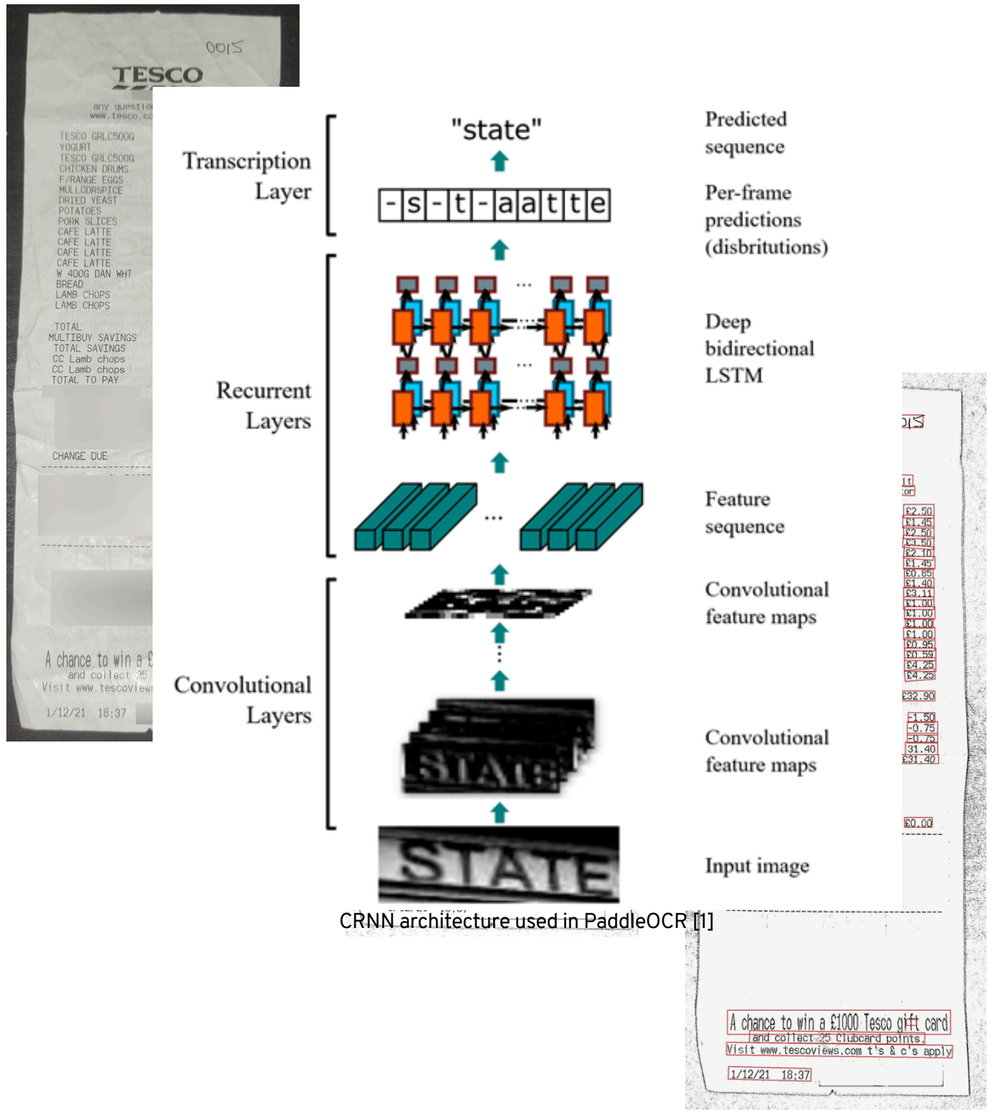


Components

- OCR model
- Key Information Extraction (KIE) model
- Product classification model

IV

OCR Component



CRNN architecture used in PaddleOCR [1]

◆ Pre-processing

Original image | Pre-processed | OCR result

◆ PaddleOCR

- Open source pre-trained OCR model
- PArallel Distributed Deep LEarning
- Published in 2020 by Chinese company Baidu

OCR engine		Results					
		Task 1 - Localisation			Task 2 - Recognition		
		Precision	Recall	Hmean	Precision	Recall	Hmean
TesseractOCR	No processing	0.115	0.198	0.146	0.678	0.697	0.687
	Processed	0.182	0.215	0.197	0.503	0.500	0.540
EasyOCR	No processing	0.597	0.607	0.602	0.597	0.607	0.602
	Processed	0.701	0.698	0.700	0.652	0.534	0.540
PaddleOCR	No processing	0.938	0.920	0.929	0.804	0.689	0.742
	Processed	0.940	0.931	0.936	0.817	0.710	0.759

Table of results from OCR experimentation

```python
from PIL import Image
from paddleocr import draw_ocr

image = 'X510056849111'
img_path = os.path.join(img_dir, image + '.jpg')

result = reader.ocr(img_path, cls=False)

image = Image.open(img_path).convert('RGB')

boxes = [line[0] for line in result]
txts = [line[1][0] for line in result]
scores = [line[1][1] for line in result]
im_show = draw_ocr(image, boxes, txts, scores)

im_show = Image.fromarray(im_show)
im_show.save('result.jpg')
```

PaddleOCR code

V Key Information Extraction component

Functionality

- Label and sort unorganised OCR output based on analysis of the semantic meaning of words and location of text
- Extract relevant fields

LayoutLMv2

- Graph-based model using word embeddings, image embeddings, and layout embeddings
- Transformer layers (BERT inspired)

Fine-tune pre-trained LayoutLMv2 model

```python
from transformers import LayoutLMv2Processor

processor = LayoutLMv2Processor.from_pretrained("microsoft/layoutlmv2-base-uncased", revision="no_ocr")
```

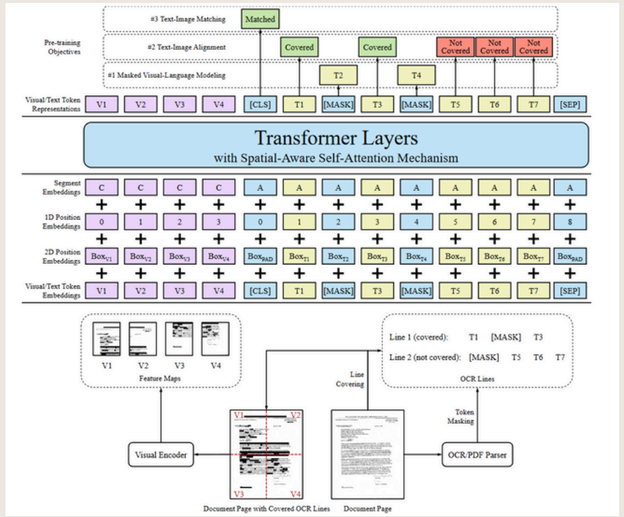
Load pre-trained LayoutLMv2 model

```
5,1,1,1,1,1,1139,444,159,63,0.0,OQ(S
5,1,2,1,1,1,654,719,65,28,93.203644,any
5,1,2,1,1,2,745,714,200,33,92.696281,questions
5,1,2,1,1,3,971,715,132,35,95.961922,please
5,1,2,1,1,4,1131,719,106,33,89.302917,visit
5,1,2,1,2,1,640,758,80,23,86.969009,www.
5,1,2,1,2,2,741,754,116,29,83.386101,tesco.
5,1,2,1,2,3,868,755,382,38,3.396606,com/store-|ocator
5,1,3,1,1,1,511,838,111,36,66.579605,TESCO
5,1,3,1,1,2,648,838,179,38,41.1185,GRLC500G
5,1,3,1,2,1,510,886,132,37,96.89109,YOGURT
5,1,4,1,1,1,0,0,1800,945,95.0,
5,1,5,1,9,1,0,1619,1800,90,95.0,
5,1,9,1,2,1,0,1709,336,94,95.0,
5,1,9,1,2,2,1472,1709,328,144,95.0,
5,1,9,1,6,1,0,2266,1800,739,95.0,
5,1,10,1,1,1,313,932,45,43,37.905384,al
5,1,10,1,1,2,511,933,110,34,91.786697,TESCO
5,1,10,1,1,3,648,933,178,35,55.07774,GRLCSO08
```

Unstructured raw PaddleOCR output (bounding boxes, text)


```
{
  "id": "0015",
  "date": "2021-12-01",
  "storeName": "TESCO",
  "itemList": [
    {
      "itemName": "TESCO GRLC500G",
      "quantity": 1,
      "category": "",
      "price": 2.50
    },
    {
      "itemName": "YOGURT",
      "quantity": 1,
      "category": "",
      "price": 1.45
    },
    {
      "itemName": "TESCO GRLC500G",
      "quantity": 1,
      "category": "",
      "price": 2.50
    },
    {
      "itemName": "CHICKEN DRUMS",
      "quantity": 1,
      "category": "",
      "price": 3.50
    },
```


Labelled and sorted JSON representation of receipt

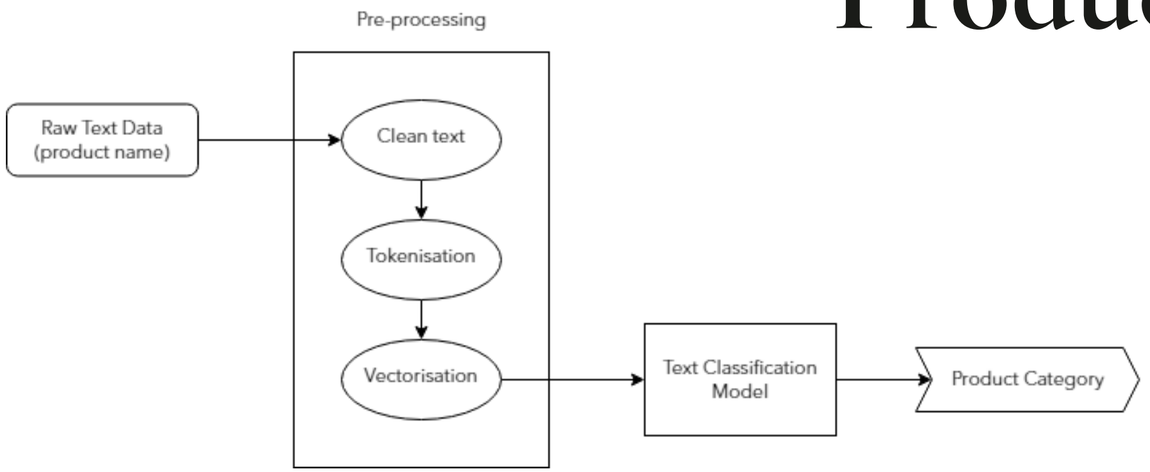


LayoutLMv2 architecture [2]

Product Classification

VI

NLP Text Classification task



Typical text classification pipeline

1. Text cleaning experiments

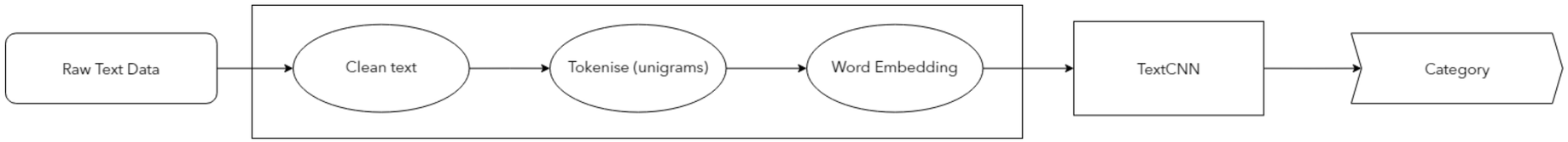
- Case normalisation
- Removing punctuation
- Removing numbers
- Removing stop words
- Stemming
- Lemmatizing

2. Feature mapping experiments

- Bag of words
- Bag of bigrams
- TF-IDF
- Word embeddings

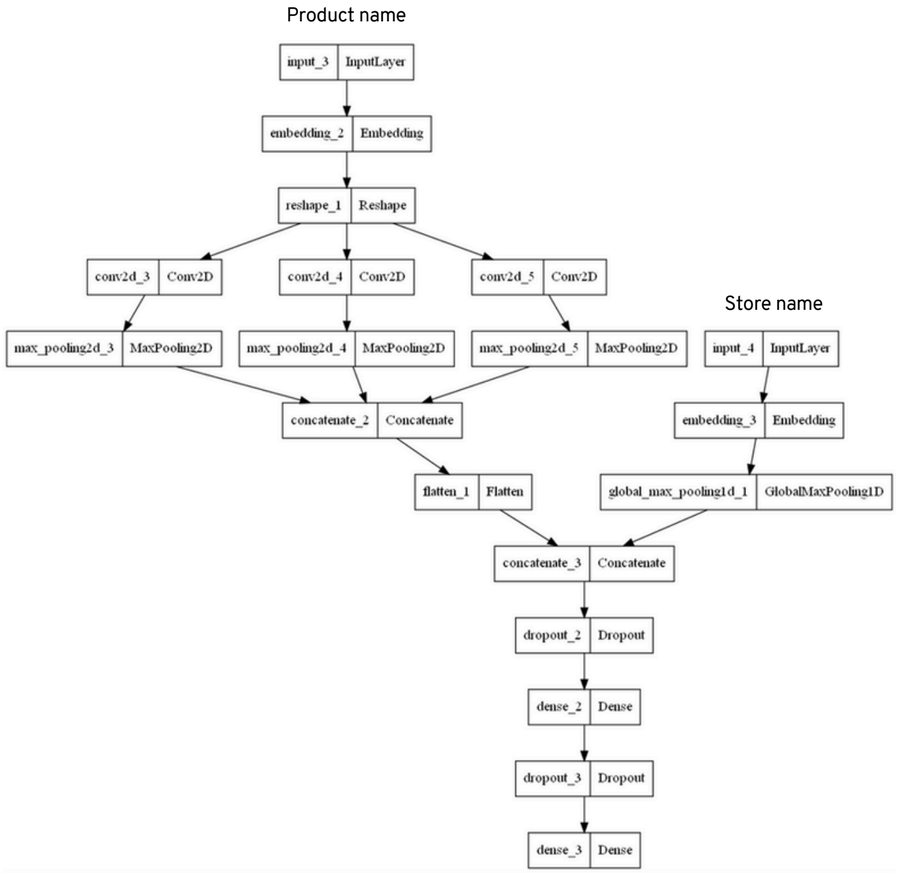
3. Classification model experiments

- Logistic Regression
- Naive-Bayes
- Support Vector Machine
- Shallow Neural Network
- Convoluted Neural Network
- Recurrent Neural Network (and LSTM)
- Transformer based model

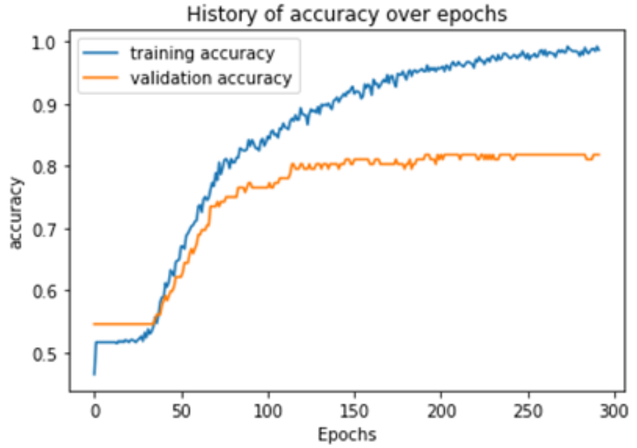


Final text classification pipeline

Text Classification model



Final model structure



Graphs showing improvement from training the model

```
fried chicken, t4
(eating out: 0.724 conf)

BELOW 0.5 CONF
poems, waterstones
(eating out: 0.257)

biscuits, tesco
(groceries: 0.591 conf)
ice cream, tesco
(snacks: 1.0 conf)

BELOW 0.5 CONF
conditioner, sainsburys
(personal hygiene: 0.401)

lemon tea, aldi
(drinks: 0.889 conf)
lemon iced tea, lidl
(drinks: 0.744 conf)
lemon ice tea, t4
(drinks: 0.556 conf)
fried chicken, tesco
(groceries: 0.843 conf)
```

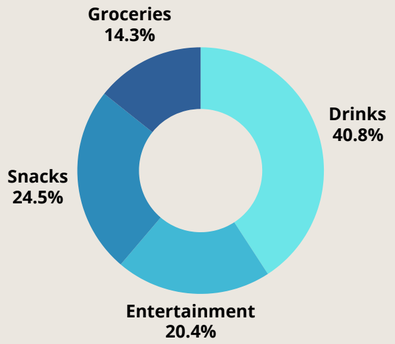
Testing model predictions on unseen data

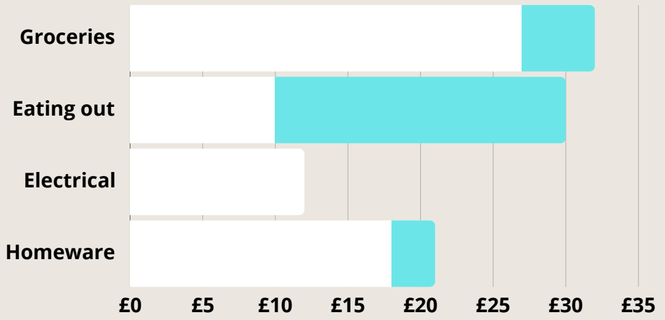
VIII

Generating insights

Mock-up of camera app

Mock-up of mobile application



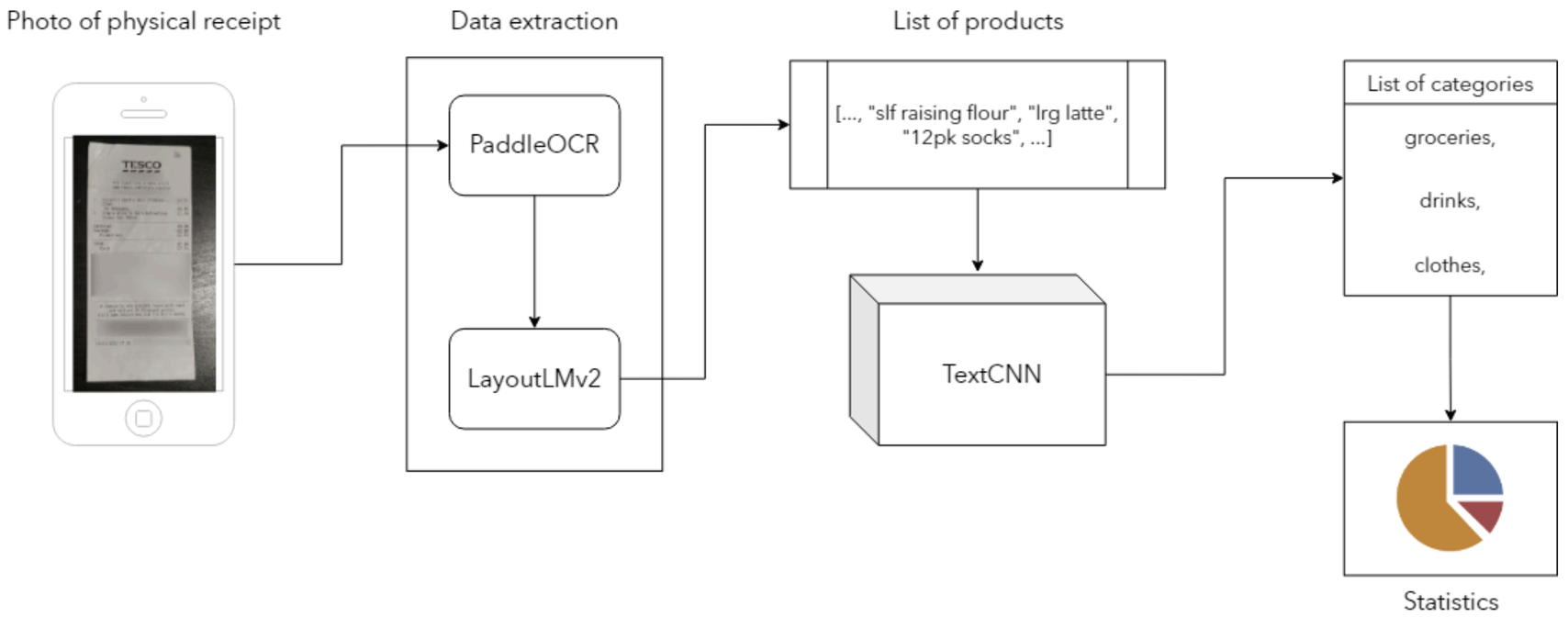


Mock-up of graphs showing spending against different categories

IX

Conclusion

Final proposed pipeline



X

Thank you

E-mail: gl00233@surrey.ac.uk

[1] Y. DU ET AL., ‘PP-OCR: A PRACTICAL ULTRA LIGHTWEIGHT OCR SYSTEM’, OCT. 2020
[2] Y. XU ET AL., ‘LAYOUTLMV2: MULTI-MODAL PRE-TRAINING FOR VISUALLY-RICH DOCUMENT UNDERSTANDING’. JAN. 09, 2022.