A black background with grey leaves

Description automatically generated

Accuracy Comparison

Week 3

A grey logo on a black background

Description automatically generated

February 12, 2024

Divyanshu

Using Levenshtein, Tokenization, and Character Frequency Methods

# Introduction

This report compares four OCR tools—AWS Textract, Azure OCR, Google OCR, and Tesseract OCR—using three accuracy measurement methods: Levenshtein, Tokenization, and Character Frequency.

## Evaluation Methodologies

## Levenshtein

* The Levenshtein distance, also known as the edit distance, measures the minimum number of single-character edits (insertions, deletions, or substitutions) required to transform one string of text into another.
* In the context of OCR accuracy assessment, the Levenshtein method helps quantify the dissimilarity between the OCR-generated output and the ground truth text. A lower Levenshtein distance indicates higher accuracy.

## Tokenization

* Tokenization involves breaking down a text into smaller units, known as tokens, which can be words, phrases, or even individual characters.
* In OCR accuracy assessment using tokenization, precision, recall, and F1 score are utilized. This methodology assesses accuracy at a more granular token level.
  + **Precision** measures the accuracy of positive predictions.
  + **Recall** evaluates the ability to capture all relevant instances.
  + **F1** **score** provides a balance between precision and recall.

## Character Frequency

* Character frequency analysis involves examining the distribution of individual characters within a text.
* In OCR accuracy evaluation using character frequency methodology, the focus is on the frequency of individual characters in the OCR output compared to the ground truth. This method helps identify common errors, such as misinterpretation of specific characters, and provides a quantitative measure of accuracy based on character-level discrepancies.

# Image 1

A blue label with white text

Description automatically generated

## AWS Textract

A screenshot of a computer program

Description automatically generated

## Azure OCR

A screenshot of a computer

Description automatically generated

## Google OCR

A screenshot of a computer

Description automatically generated

## Tesseract OCR

A screenshot of a computer

Description automatically generated

# Image 2

A green can with a label

Description automatically generated

## AWS Textract A screenshot of a computer Description automatically generated

## Azure OCR

A screenshot of a computer program

Description automatically generated

## Google OCR

A screenshot of a computer

Description automatically generated

## Tesseract OCR

A screenshot of a computer

Description automatically generated

# Image 3

No border


## AWS Textract

A screenshot of a computer program

Description automatically generated

## Azure OCR

A screenshot of a computer program

Description automatically generated

## Google OCR

A screenshot of a computer program

Description automatically generated

## Tesseract OCR

A screenshot of a computer

Description automatically generated

Eg: ‘Vaiue’ should be ‘Value’

‘Og’ should be ‘0g’

‘Omg’ should be ‘0mg’

‘VitaminA0’ should be ‘Vitamin’, ‘A’, ‘0’

‘VitaminC’ should be ‘Vitamin’, ‘C’

‘Calcium0’ should be ‘Calcium’, ‘0’