Reconhecimento de caracteres por Inteligência Artificial

Mateus Siqueira Carneiro
Universidade Federal de São Paulo
Instituto de Ciência e Tecnologia
São José dos Campos, São Paulo
mateussiqueiracarneiro@outlook.com

Matheus Gomes de Paula
Universidade Federal de São Paulo
Instituto de Ciência e Tecnologia
São José dos Campos, São Paulo
matheusgomes062@gmail.com

***Abstract* — Nesse paper iremos iremos propor o uso de um método analítico para o reconhecimento de palavras manuscritas. Esse método é proposto para o problema do reconhecimento de escrita cursiva offline. Primeiro a obtenção de parâmetros globais, como ângulo de inclinação, linhas de base, largura da linha e altura. Segundo o tratamento da imagem pelo método de segmentação combinando gray scale e informação binária. Terceiro, o uso do Tesseract-OCR da Google para o treinamento e reconhecimento da imagem. Quarto, aplicação do Modelo Oculto de Markov (Hidden Markov Model - HMM) para o reconhecimento das formas e rankeamento dos possíveis caracteres. Finalmente ocorre a decisão do melhor caractere a partir da junção das informações do Tesseract e do HMM.**

Palavras-Chave—Inteligência Artificial, Reconhecimento de padrões, Escrita, OCR, HMM, Hidden Markov Models, Tesseract.

I. INTRODUÇÃO E MOTIVAÇÃO

Apesar de estarmos na era digital, a escrita ainda é uma das atividades mais importantes. Muito do nosso conhecimento ainda se encontra descrito cursivamente em papel. Tais métodos de escrita utilizam diferentes caligrafias variando de pessoa para pessoa, o que pode se mostrar um problema para muita das transcrições para o meio digital, com riscos de não se entender o que está escrito. Propomos o uso de uma Inteligência Artificial junto da aplicação do Modelo Oculto de Markov para resolver esse problema.

II. OBJETIVOS

Neste trabalho focaremos em reconhecer escritas cursivas utilizando Inteligência Artificial. Esperamos que a longo prazo possamos conseguir transcrever para um arquivo digital qualquer texto manuscrito com uma taxa de erro segura. De forma que mesmo o texto possuindo rasuras ele possa ser identificado corretamente dentro da taxa de acerto. Assim confiamos que ao final do curso teremos os conhecimentos necessários para desenvolver um software de Inteligência Artificial para digitalizar escritas manuais que seja relativamente seguro e eventualmente rentável.

III. METODOLOGIA EXPERIMENTAL

Utilizaremos a linguagem de programação Python, sendo utilizado em conjunto bibliotecas como scikit-learn, pytesseract, pillow, numpy, openCV.

Faremos o uso da ferramenta de Inteligência Artificial de reconhecimento de caracteres (Optical Character Recognition - OCR) da Google, o Tesseract-OCR para reconhecer os caracteres. Iremos utilizar como base o paper proposto por Nafiz Arica e T. Yarman-Vural “Optical Character Recognition for Cursive Handwriting”[1] pois apresenta uma abordagem factível e explicativa sobre o processo. Assim, utilizaremos uma estratégia analítica, que utiliza uma abordagem de “baixo para cima”, começando da letra até a palavra. Uma segmentação da palavra se faz necessária para essa estratégia. Com isso reduzimos o reconhecimento para caracteres individuais. Após, aplica-se o ranqueamento dos caracteres segmentados das palavras e aplica-se um algoritmo de busca para achar a palavra com o maior ranking.

O Tesseract-OCR apesar de ser uma boa ferramenta possui limitações, como visto tanto no paper mencionado anteriormente quanto no paper de Ray Smith, “An Overview of the Tesseract OCR Engine”[2], assim teremos uma etapa de pré processamento onde trataremos a imagem com o uso de níveis de cinza, normalização do ângulo e da linha de base, e por fim aplicaremos então o uso de normalização binária da imagem.

A etapa de reconhecimento será feita pelo Modelo Oculto de Markov (HMM) e será empregado para o reconhecimento de forma. As características extraídas das linhas serão alimentadas para um HMM esquerda-direita[3].

IV. BASES

Utilizaremos como base um banco público de imagens fornecido pelo Instituto Nacional de Padrões e Tecnologia (National Institute of Standards and Technology - NIST). O Special Database 19 (nome do banco) contém todo o corpo de materiais de treinamento do NIST para documentos impressos e reconhecimento de caracteres. Ele possui Formulários de Amostra manuscrita de 3600 escritores, 810.000 imagens com classificações verificadas à mão. As imagens possuem dígitos separados, letras maiúsculas e minúsculas e campos de texto livre.

Em conjunto, utilizaremos dos nossos próprios manuscritos em inglês para o treinamento e teste do algoritmo.

V. TÉCNICAS E MEDIDAS DE AVALIAÇÃO

Utilizaremos o modelo de Classificação e um algoritmo de HMM.

Para obter uma medida que é independente do tamanho do texto o número de erros é em geral normalizado para o tamanho do conteúdo esperado (Texto Original). O quociente entre o número de erros e o tamanho do texto é conhecido como taxa de erro.

A taxa de erro é geralmente calculada em dois níveis diferentes:

- Taxa de erro de letra (Character Error Rate - CER).
- Taxa de erro de palavra (Word Error Rate - WER).

Vale lembrar que a taxa de erros em palavras é geralmente maior do que a taxa de erros em letras. Por exemplo, uma taxa de erro de caracteres como 10% significa que aproximadamente metade das palavras de 6 letras conterão pelo menos um caractere errado. Experimentos sugerem que humanos são relativamente intolerantes a erros seguindo a seguinte lógica: uma acurácia por palavra abaixo de 85% leva a um baixo rendimento de produtividade se correção manual é aplicada, isto comparado a escrever tudo do zero. De acordo com Rose Holley no seu artigo “How Good Can It Get? - Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs”[4] uma boa medida de OCR significa acertar em torno de 98 a 99% das palavras, enquanto que abaixo de 90% é ruim. Portanto, adotaremos essa medida de avaliação.

VI. O QUE SERÁ ENTREGUE?

Uma IA capaz de reconhecer padrões em letras e números sendo assim capaz de colocar os caracteres de forma digital em um eventual pdf.

VII. REFERÊNCIAS

1. N. Arica, Fatos T. Yarman-Vural, “Optical character recognition for cursive handwriting”, IEEE Transactions on Pattern Analysis and Machine Intelligence. (https://ieeexplore.ieee.org/abstract/document/1008386)

2. Ray Smith, “An Overview of the Tesseract OCR Engine”, Google Inc. (https://static.googleusercontent.com/media/research.google.com/pt-BR//pubs/archive/33418.pdf)

3. Herv´e Bourlard, “Introduction to Hidden Markov Models”, Ecole Polytechnique F´ed´erale de Lausanne (https://www.cs.ubc.ca/~murphyk/Software/HMM/labman2.pdf)

4. Rosey Holley, “How Good Can It Get? - Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs”, D-Lib Magazine. (http://www.dlib.org/dlib/march09/holley/03holley.html)

5. J. M. White, G. D. Rohrer, “Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction”, IBM Journal of Research and Development. (https://ieeexplore.ieee.org/abstract/document/5390437)

6. Ravina Mithe, Supriya Indalkar, Nilam Divekar, “Optical Character Recognition”, International Journal of Recent Technology and Engineering. (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.673.80614&4rep=rep14&4type=pdf)

7. Sandip Rakshit, Subhadip Basu, Hisashi Ikeda, “Recognition of Handwritten Textual Annotations using Tesseract Open Source OCR Engine for information Just In Time (iJIT)”, Proc. Int. Conf. on Information Technology and Business Intelligence.

(https://arxiv.org/ftp/arxiv/papers/1003/1003.5893.pdf)