



UNIVERSIDADE FEDERAL DO RIO GRANDE - FURG
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
CURSO DE MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

Dissertação de Mestrado

**Uma abordagem para Form Understanding baseada em
multitarefas aplicada a documentação técnica de
engenharia**

Gabriel Lavoura dos Santos

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande - FURG, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação

Orientador: Prof. Dr. Nelson Lopes Duarte Filho
Co-orientador: Prof. Dr. Ricardo Nagel Rodrigues

Rio Grande, 2022

Ficha Catalográfica

S237a Santos, Gabriel Lavoura dos.

Uma abordagem para *Form Understanding* baseada em multitarefas aplicada a documentação técnica de engenharia / Gabriel Lavoura dos Santos. – 2022.

106 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2022.

Orientador: Dr. Nelson Lopes Duarte Filho.

Coorientador: Dr. Ricardo Nagel Rodrigues.

1. Extração de Informação 2. *Form Understanding* 3. *Document Understanding* 4. Documentos Impressos I. Duarte Filho, Nelson Lopes II. Rodrigues, Ricardo Nagel III. Título.

CDU 004.8:002

Catálogo na Fonte: Bibliotecário José Paulo dos Santos CRB 10/2344

DISSERTAÇÃO DE MESTRADO

**Uma abordagem para Form Understanding baseada em multitarefas
aplicada a documentação técnica de engenharia**

Gabriel Lavoura dos Santos

Banca examinadora:

CN=Eduardo Nunes
Borges:00057035067, OU=
FURG - Universidade Federal do
Rio Grande, O=ICPEdu, C=BR
[https://pessoal.icpedu.rnp.br/pub
lic/verificar-assinatura](https://pessoal.icpedu.rnp.br/pub
lic/verificar-assinatura)
Rio Grande
2022.08.26 21:43:24-03'00'

Prof. Dr. Eduardo Nunes BorgesDocumento assinado digitalmente
RENATA DE MATOS GALANTE
Data: 26/08/2022 13:57:40-0300
Verifique em <https://verificador.iti.br>

Profa. Dra. Renata de Matos GalanteDocumento assinado digitalmente
RICARDO NAGEL RODRIGUES
Data: 26/08/2022 16:16:04-0300
Verifique em <https://verificador.iti.br>

Prof. Dr. Ricardo Nagel Rodrigues
CoorientadorDocumento assinado digitalmente
NELSON LOPES DUARTE FILHO
Data: 27/08/2022 12:14:14-0300
Verifique em <https://verificador.iti.br>

Prof. Dr. Nelson Lopes Duarte Filho
Orientador

Aos meus pais, Sônia e Sandro, meu irmão Rafael e a Vanessa.

AGRADECIMENTOS

Agradeço aos meus pais, Sônia e Sandro pelo incentivo, dedicação e apoio durante toda minha trajetória e diante aos desafios encontrados neste percurso que caracteriza um ciclo que se encerra e constrói oportunidades para outros darem início. Agradeço também ao meu irmão Rafael por servir como motivação para sustentar a resiliência necessária até o processo de conclusão.

A minha namorada Vanessa que me apoiou de forma inigualável, durante os mais diversos desafios encontrados durante esta jornada.

Ao meu coorientador, Ricardo, cujo o qual colaborou com o suporte técnico necessário e com diversos valores acadêmicos de forma significativa. Ao meu orientador Nelson, que tive o privilégio de trabalhar durante toda a minha jornada acadêmica, contribuindo de forma concisa na construção de diversas inspirações na minha carreira.

E a todos os demais envolvidos, que de forma direta ou indireta contribuíram para que este trabalho se concluísse com êxito. Com pequenas contribuições de todos os envolvidos somam-se diversos ensinamentos que não poderiam ser listados neste espaço, pois seria injusto aos detalhes que a vivência dos mesmos tenha me propiciado.

Se queres vencer o mundo inteiro, vence-te a ti mesmo. — FIÓDOR
DOSTOIÉVSKI

RESUMO

SANTOS, Gabriel Lavoura dos. **Uma abordagem para Form Understanding baseada em multitarefas aplicada a documentação técnica de engenharia**. 2022. 106 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

No âmbito da pesquisa e inovação, nos dias de hoje, muito ainda se discute a importância e necessidade de digitalização, compartilhamento, recuperação e armazenamento de grandes volumes de dados. Entretanto, mesmo com o intenso processo de digitalização vivenciado nos últimos vinte anos, grandes segmentos da indústria ainda dependem de documentos em papel para entrada e ingestão de dados. Como são os casos de documentos históricos, com cunho médico ou de engenharia. Um ser humano compreende um documento ao interpretar o conteúdo semântico de caracteres, palavras, parágrafos e componentes de layout. Para que um sistema computacional consiga entregar um resultado próximo a esse nível de abstração são necessários uma série de passos além de apenas reconhecer os caracteres. Por mais que apenas digitalizar represente um avanço importante, ainda entrega recursos em suma triviais quanto ao contexto e entendimento do documento. Logo, para obter resultados efetivos e um nível semântico adequado torna-se necessário cruzar diferentes técnicas de análise de geometria, espacialidade e utilizar-se de ambos atributos visuais e textuais. Nessa perspectiva, este trabalho tem como objetivo utilizar técnicas de extração de informação baseadas em aspectos espaciais e geométricos do documento digitalizado, assim como técnicas de aprendizado de máquina, para otimizar o processo de entendimento do documento (document understanding) e dos formulários (form understanding). Foi feito um dataset próprio a partir de documentos de engenharia interligados, chamados de databooks. Este dataset foi utilizado em experimentos quanto a classificação de entidade (semantic entity labeling) através do modelo LayoutLM. Dessa forma extraindo informações com semântica atrelada, em documentos de estruturas diversas, através de uma abordagem baseada em multitarefas.

Palavras-chave: Extração de Informação, Form Understanding, Document Understanding, Documentos Impressos.

ABSTRACT

SANTOS, Gabriel Lavoura dos. **A multitasking-based approach to Form Understanding applied to technical engineering documentation.** 2022. 106 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande - FURG, Rio Grande.

Nowadays, digitization, sharing, retrieval, and storage of large volumes of data are hot topics widely discussed in the scope of academic research. Despite the growth of digitalization over the past twenty years, in some industries the data entry relies almost entirely on paper-based record management. E. g: historical, medical, or engineering documents. A human being understands a file by interpreting the semantic content of characters, words, paragraphs, and layout components. A computer system takes a series of extra steps apart from recognizing characters to deliver a result close to a human level of abstraction. Although document digitalization is portrayed as a major task, it nevertheless delivers trivial resources in terms of document context and document understanding. To obtain a semantic understanding close to a human level, both visual and textual attributes must be crossed with techniques such as geometry analysis, spatiality, logic, and many others. This work proposes an approach to extracting information from scanned documents through spatial, geometric, and machine learning-based techniques to improve document understanding and form understanding process. We created our own dataset based on linked engineering documents, called databoks. We also applied the dataset in the LayoutLM model in order to perform Semantic entity labeling. Thus, extracting semantic-linked information from documents with diverse structures through a multitasking approach.

Keywords: Information Extraction, Form Understanding, Document Understanding, Printed Documents.

LISTA DE FIGURAS

1	Exemplos de páginas de um <i>databook</i> . Cada página representa um documento diferente que compõe o conjunto da documentação necessária para um <i>databook</i> estar completo.	19
2	Gabarito de concurso: Documento estruturado.	25
3	Artigos científicos: Documentos semiestruturados	26
4	Documentos não estruturados.	27
5	Exemplo de ligação da estrutura de <i>breakdown</i> em um documento de relatório diário (Cha & Lee 2018).	32
6	Resultado obtido por Aiello et al. (2002)	33
7	Tipos de documentos utilizados no trabalho de (Zhang et al. 2020).	34
8	Exemplo de descrição de modelo proposto por Jaume et al. (2019)	35
9	Arquitetura do modelo pré-treinado <i>LayoutLM</i>	37
10	Diagrama do sistema de extração automática de informação proposto separado em blocos distintos para cada tipo de documento.	41
11	Fluxograma de funcionamento da função <i>PDF-processor-fn</i>	43
12	Fluxograma de funcionamento da função <i>formund-fn</i>	45
13	Trecho do modelo descritivo de um documento para extração do número do relatório.	48
14	Modelo desenhado com os campos encontrados pelo algoritmo sobre um documento de exemplo.	50
15	Exemplo de entidades perguntas: (vermelho), entidades respostas (azul) e em verde a conexão entre elas.	50
16	Arquitetura do módulo de processamento automatizado de documentos.	51
17	Visão geral na qual cada página é dividida em quatro desafios principais.	53
18	Visão geral do <i>pipeline</i> de processamento do <i>input</i>	54
19	Visão geral do <i>pipeline</i> para carimbos e assinaturas proposto.	54
20	Visão geral do <i>pipeline</i> para imagens proposto.	55
21	Visão geral do <i>pipeline</i> para tabelas proposto.	55
22	Visão geral do <i>pipeline</i> para formulários proposto.	56
23	Visão geral das etapas de <i>form understanding</i> , divididas em agrupamento de palavras, classificação de entidade e relacionamento das mesmas.	57
24	Exemplo de separação de entidade textual em palavras individuais.	59
25	Versão final do arquivo de descrição das páginas do <i>dataset</i>	60

26	Exemplo de JSON de saída do algoritmo de <i>form understanding</i>	62
27	Exemplo desenhado de um resultado do processo de extração. Em vermelho: a label das entidade. Em azul: a entidade encontrada. . . .	63
28	Resultado de detecção obtido nos documentos do conjunto nas cores laranja e azul.	66
29	Exemplo de erro de identificação de tabela. Foi identificado a metade de um formulário como se fosse uma tabela.	66
30	Exemplo 1: Comparação de resultados de detecção de tabela antes e depois do <i>data augmentation</i>	67
31	Exemplo 2: Comparação de resultados de detecção de tabela antes e depois do <i>data augmentation</i>	68
32	Exemplo 3: Comparação de resultados de detecção de tabela antes e depois do <i>data augmentation</i>	68
33	Exemplo de documento da Classe 1 utilizado como teste no modelo <i>LayoutLM</i>	70
34	Exemplo de documento da Classe 2 utilizado como teste no modelo <i>LayoutLM</i>	71
35	Exemplo de documento da Classe 3 utilizado como teste no modelo <i>LayoutLM</i>	72
36	Resultado de SEL do modelo <i>LayoutLM</i> em um documento no idi- oma inglês e com o modelo ajustado.	72
37	Exemplo de documento da Classe 1 utilizado como teste no modelo <i>LayoutLMv2</i>	73
38	Exemplo de documento da Classe 2 utilizado como teste no modelo <i>LayoutLMv2</i>	74
39	Exemplo de documento da Classe 3 utilizado como teste no modelo <i>LayoutLMv2</i>	75
40	Resultado de SEL do modelo <i>LayoutLMv2</i> em um documento no idi- oma inglês e com o modelo ajustado.	75
41	Comparação entre resultados de SEL no documento de Classe 1. . . .	76
42	Comparação entre resultados de SEL no documento de Classe 2. . . .	77
43	Comparação entre resultados de SEL no documento de Classe 3. . . .	77
44	Comparação entre resultados de SEL nos documentos em inglês e com os modelos ajustados.	78
45	Comparação entre as classificações originais e as obtidas pelo expe- rimento com 50 páginas do <i>dataset</i> próprio.	80
46	Comparação entre as classificações originais e as obtidas pelo expe- rimento com 300 páginas do <i>dataset</i> próprio.	82
47	Comparação entre as classificações originais e as obtidas pelo expe- rimento com 350 páginas do <i>dataset</i> próprio.	83
48	Comparação entre as classificações originais e as obtidas pelo expe- rimento com 500 páginas do <i>dataset</i> próprio.	84
49	Resultado encontrados ao utilizar os pesos do modelo <i>LayoutLM</i> com <i>fine-tuning</i> do <i>dataset</i> próprio quando aplicado a documentos não es- truturados do tipo 3.	86
50	Comparação entre as classificações originais e as obtidas pelo expe- rimento com 100 páginas do <i>dataset</i> próprio.	94

51	Comparação entre as classificações originais e as obtidas pelo experimento com 150 páginas do <i>dataset</i> próprio.	95
52	Comparação entre as classificações originais e as obtidas pelo experimento com 200 páginas do <i>dataset</i> próprio.	96
53	Comparação entre as classificações originais e as obtidas pelo experimento com 250 páginas do <i>dataset</i> próprio.	97
54	Comparação entre as classificações originais e as obtidas pelo experimento com 400 páginas do <i>dataset</i> próprio.	97
55	Comparação entre as classificações originais e as obtidas pelo experimento com 450 páginas do <i>dataset</i> próprio.	98

LISTA DE TABELAS

1	Parâmetros da regra probabilística aplicada.	36
2	Resultados da métrica de acurácia para a extração para documentos de classe 1.	64
3	Resultado da métrica de acurácia para a extração para documentos de classe 2.	65
4	Resultados dos experimentos do modelo <i>LayoutLM</i> com <i>fine-tuning</i> do <i>dataset</i> próprio.	85

LISTA DE ABREVIATURAS E SIGLAS

AI	Artificial Intelligence
API	Application Programming Interface
ATT	Análise de Textos Técnicos
BBOX	Bounding Box
BERT	Bidirectional Encoder Representation from Transformer
BIM	Building Information Modelling
CBS	Classification Breakdown Structure
C3	Centro de Ciências Computacionais
DocUnd	Document Understanding
DLA	Document Layout Analysis
DPI	Dots Per Inch
EI	Extração da Informação
FaaS	Function as a service
FURG	Universidade Federal do Rio Grande
FUNSD	Form Understanding in Noisy Scanned Documents
GCN	Graph Convolutional Networks
IOU	Intersection over Union
HIS	Hospital Information System
HP	Hewlett-Packard
IBS	Information Breakdown Structure
IDC	International Data Corporation
JSON	JavaScript Object Notation
MDC	Multi-label document classification
MLM	Masked Language Model
MVLM	Masked Visual-Language Model
NER	Named Entity Recognition

NEL	Named Entity Linking
NLI	Natural Language Inference
NLP	Natural Language Processing
NLU	Natural Language Understanding
NSP	Next Sequence Prediction
OCR	Optical Character Recognition
OpenCV	Open Source Computer Vision
PPI	Pixel per Inch
PDF	Portable Document Format
QA	Question Answering
RE	Relation Extraction
Regex	Regular expressions
ROI	Region de interest
SBS	Spatial Breakdown Structure
SEL	Semantic Entity Labeling
SIBGRAPI	Conference on Graphcis Patterns and Images
VRD	Visually-Rich Document
XML	Extensible Markup Language
ZB	Zettabytes

SUMÁRIO

1	Introdução	17
1.1	Contextualização	17
1.2	Motivação	18
1.3	Objetivos	21
1.3.1	Objetivos específicos	21
2	Fundamentação teórica	22
2.1	Extração de informação	22
2.2	<i>Layout</i> de documentos	24
2.3	Form understanding	28
2.4	Document understanding	29
2.5	Tesseract OCR	29
3	Estado da arte	31
3.1	Produção científica	31
3.2	Produtos comerciais	38
3.3	Discussão	39
4	Metodologia	41
4.1	Metodologia para documentos estruturados	42
4.1.1	PDF-processor-fn	42
4.1.2	Formund-fn	44
4.1.3	Implementação no Projeto ATT	50
4.2	Metodologia relativa a documentos semi e não estruturados	52
4.2.1	Pipeline de carimbos e assinaturas	54
4.2.2	Pipeline de imagens	54
4.2.3	Pipeline tabelas	55
4.2.4	Pipeline de formund	55
4.2.5	Implementação no Projeto ATT	57
4.3	Classificação de entidades e criação do <i>dataset</i>	58
5	Resultados	61
5.1	Resultados documentos estruturados	61
5.2	Resultados dos experimentos sobre extração de tabelas	65
5.3	Experimentos com documentos não estruturados	69
5.3.1	Resultados <i>LayoutLM</i> e <i>fine-tuning</i> com <i>FUNSD</i>	69
5.3.2	Resultados obtidos com o modelo <i>LayoutLMv2</i>	73

5.3.3	Comparação entre os modelos	76
5.3.4	Resultados com <i>LayoutLM</i> e <i>fine-tuning</i> com <i>dataset</i> próprio	79
5.3.5	Discussão	86

6 Conclusão 90

6.1 Contribuições 92

6.2 Trabalhos futuros 93

ANEXOS

A Apêndice 94

Referências 99

1 INTRODUÇÃO

Neste capítulo é apresentada uma breve contextualização sobre o tema, seguida pela motivação para a realização deste trabalho e por fim são apresentados os objetivos do mesmo.

1.1 Contextualização

Recentemente, muito se discute a importância e necessidade de digitalização, compartilhamento, recuperação e armazenamento de grandes volumes de dados no âmbito de pesquisa e inovação (Bstieler et al. 2018). Este fenômeno vem ganhando abrangência devido aos atuais avanços tecnológicos alcançados na área de tecnologia da informação, o aumento no número de dispositivos conectados e a crescente disseminação do uso de redes sociais (Trabucchi et al. 2018) (MacGillivray & Reinsel 2019) (Uçkan & Karci 2020).

O *International Data Corporation* (IDC) estima que haverá 41,6 bilhões de dispositivos conectados, gerando 79,4 zettabytes (ZB) de dados em 2025 (MacGillivray & Reinsel 2019). Ainda, segundo o Escritório do Coordenador Nacional de Tecnologia da Informação em Saúde, nos Estados Unidos 80% das receitas médicas já são emitidas de forma digital, o que significa que mais de 1,7 bilhões de prescrições por ano podem ser monitoradas eletronicamente (National Coordinator for Health Information Technology 2017) (Norgeot et al. 2019).

Entretanto, mesmo com o intenso processo de digitalização vivenciado nos últimos 20 anos, grandes segmentos da indústria ainda dependem de documentos em papel para entrada e ingestão de dados, como são os casos de documentos históricos, com cunho médico ou de engenharia (Qi et al. 2020) (Patel & Bhatt 2020) (Promin & Suriyachai 2019). Outro fator preocupante se dá ao fato de que muitas vezes esses documentos não apenas se encontram em forma de mídia física, como também possuem uma grande diversidade em questão de *layout*, conteúdo, características e complexidade de associação (Cha & Lee 2018). Essa diversidade muitas vezes faz com que encontrar informações relevantes usando apenas a pesquisa por palavra-chave se torne uma tarefa custosa e ineficiente

(Das et al. 2018) (Matsubayashi et al. 2018).

Para transcrever automaticamente o texto destes documentos, que ao serem digitalizados são transformados em imagens, é comumente utilizada a ferramenta chamada de *Optical Character Recognition* (OCR) a qual basicamente tem como finalidade transcrever o texto presente nas imagens para texto simples em forma de *strings* (Eikvil 1993).

Enquanto digitalizar os documentos seja uma tarefa importante, ela ainda entrega recursos em suma triviais quanto ao entendimento do contexto do documento. Um ser humano compreende um documento ao interpretar o conteúdo semântico de caracteres, palavras, parágrafos e componentes de layout (Katti et al. 2018) e para que um sistema computacional consiga entregar um resultado próximo a esse nível de abstração é preciso uma série de passos além de apenas reconhecer os caracteres, sendo necessário cruzar diferentes técnicas de análise de geometria, espacialidade e lógica, utilizando-se de atributos visuais e textuais a fim de obter-se um bom nível semântico.

Estes conjuntos de passos podem ser reunidos e resumidos em diversas técnicas as quais conseguem lidar e contornar este tipo de situação onde a máquina deve entregar um entendimento, quanto aos documentos, o mais próximo possível que ao de um ser humano. Dentre as disponíveis na literatura, em especial, tem-se a técnica chamada de *form understanding*, termo que pode ser traduzido literalmente como “entendimento de formulários”.

Esta abordagem está intrinsecamente interligada com o trabalho aqui desenvolvido, servindo como peça chave para a resolução do problema proposto. A técnica de *form understanding* é contextualizada no Capítulo 2, na Seção 2.3.

1.2 Motivação

O projeto denominado de *ATT - Análise de Textos Técnicos* (ATT), em desenvolvimento pelo grupo de docentes e discentes do Centro de Ciências Computacionais (C3), unidade que faz parte da Universidade Federal do Rio Grande (FURG), em conjunto com uma empresa do setor de óleo e gás, trata-se de uma solução de software para suprir a demanda de otimização na análise e processamento de conjuntos de documentos de engenharia densos compostos de milhares de páginas e dezenas de documentos anexados, conhecidos como *databooks*, utilizando-se de conceitos de Inteligência Artificial, *Machine Learning*, *Deep-Learning*, Recuperação de informação, Extração de informação, Teoria dos Grafos, Computação Flexível, Processamento de Imagens, Microserviços e *DevOps*, abordando diversas outras tecnologias necessárias para o desenvolvimento das ferramentas contidas na solução de *software*.

Conceitualmente, *databooks* são documentos impressos que são digitalizados com a finalidade de serem armazenados, assim formando documentos compostos apenas de imagens, fotocópias dos documentos originais. No âmbito do projeto ATT estes documentos

são compostos por normas, documentos técnicos do processo de construção, montagem, compra e venda de insumos, relacionados à indústria de óleo e gás, naval e *offshore*.

Na Figura 1 são apresentados alguns exemplos das possíveis configurações que as páginas de um *databook*¹ possuem.

Figura 1: Exemplos de páginas de um *databook*. Cada página representa um documento diferente que compõe o conjunto da documentação necessária para um *databook* estar completo.

(a) Exemplo 1.

RELATÓRIO DE ENSAIO

Data do Recebimento: 18/02/16 Data da Conclusão: 23/02/16

Cliente: [Redacted] Lota: [Redacted]
 Endereço: [Redacted]
 Solicitante: [Redacted] Auxiliar da Qualidade
 Identificação da amostra declarada pelo cliente: Barra Roscada - 02" SUN - Lota [Redacted]
 Corrente de ensaio: ASTM A193/16 Gr. B16
 Rastreabilidade: [Redacted]
 Norma Aplicável: ASTM A193/16 Gr. B16

1. OBJETIVO
 Realizar ensaio metalográfico na amostra fornecida conforme solicitação do cliente.

2. ENSAIOS E RESULTADOS

2.1. EXAME METALOGRAFICO
 Procedimento utilizado: Preparação de corpo de prova conforme [Redacted] baseada na norma ASTM E 911
 Método de Ensaio utilizado: [Redacted] baseada na norma ASTM E 1023 (Preparação 2200)
 Seção Observada: Longitudinal
 Número de 100X: 400 V / Análise: 3%
 Condições de Preparação: [Redacted] Temperatura de 22,5 °C - Umidade: 60%
 Equipamento e Instrumento utilizado: [Redacted]
 Microscópio: OM-200 - Model: MC-04000
 Termogravimétrico: Q7-022 - Model: K7-008 - Certificado de calibração: CAL-1227-03/15 - Validade: 03/01/16

Amostras: [Redacted] Microestrutura observada: [Redacted]

1 Microestrutura constituída de martensita revenida com presença dispersa de ferrita.

3. LAUDO: O resultado do ensaio atende ao especificado.

De resultados apresentados se referem somente aos dados apresentados.
 Este relatório de ensaio só deve ser reproduzido completo. Reprodução parcial requer autorização escrita da [Redacted]

Aprovado em 04/03/16

(c) Exemplo 3.

Relatório de Databook Digital

Projeto: [Redacted] Cliente: [Redacted]
 Monitoria: [Redacted] Controlador: [Redacted]

Plano de Inspeção: [Redacted]
 Norma de Referência: [Redacted]
 Condição da Superfície: [Redacted]
 Remoção: [Redacted]

Visual de Ajuste
 Processo de Inspeção: [Redacted]
 Método de Análise: [Redacted]
 Observação: [Redacted]

Instrumentos Utilizados
☒ Hita: CO-023
☐ Paquímetro: [Redacted]
☒ Trena: TR-602

Identificação
 Data: [Redacted] Ass: [Redacted] Ass: [Redacted] Ass: [Redacted]
 Data: [Redacted] Ass: [Redacted] Ass: [Redacted] Ass: [Redacted]

Legenda
 Estado: L: Liberado, E: Em andamento, P: Pendente, R: Rejeitado, S: Suspendido, T: Transferido, U: Utilizado, V: Validado, W: Retirado, X: Excluído, Y: Y: Y, Z: Z

Inspector: [Redacted] Coordenador: [Redacted] Cliente: [Redacted]

Data: 04/12/2017

(b) Exemplo 2.

INSPECTION TEST CERTIFICATE
 (ACCORDING TO EN)

ORDER NO. [Redacted] TEST DATE: 07TH SEP 2015
 L/C NO. [Redacted] CERTIFICATE NO. [Redacted]
 INVOICE NO. [Redacted]

DESCRIPTION OF PARTS

ITEM NO.	POSITION	QUANTITY	DIMENSION	STANDARD	MARKING
1		0.020	1.25" - BUN x 22PT	THREADED STUDS ASTM A193 B16 (BLACK OXIDE)	B16
2		0.030	1.25" - BUN x 22PT	THREADED STUDS ASTM A193 B16 (BLACK OXIDE)	B16
3		0.030	1.25" - BUN x 22PT	THREADED STUDS ASTM A193 B16 (BLACK OXIDE)	B16
4		0.030	1.25" - BUN x 22PT	THREADED STUDS ASTM A193 B16 (BLACK OXIDE)	B16

RESULTS OF TESTS

ITEM NO.	POSITION	TENSILE STRENGTH	YIELD STRENGTH	ELONGATION	REDUCED	TEMPERATURE	SERIES	HARDNESS
1		Min. 125 KSI	Min. 105 KSI	Min. 18%	Min. 18%	Min. 600°C	925 - 955°C	Min. 50 HRC
1		143	119	18	18	720	925	28.2 / 30.7 / 30.4
2		142	118	20	20	720	925	28.7 / 30.4 / 30.5
3		143	119	20	20	720	925	28.2 / 30.7 / 30.5
4		132	116	21.0	21.0	720	925	28.2 / 30.7 / 30.5

FINISHED MACRO-STRUCTURE (ASTM E981/96) THE MACRO ETCH RESULTS FOR SURFACE, CENTER AND RANDOM (S1 / S1 / CL)

DECARBURATION - ASTM A923

CHEMICAL ANALYSIS - MATERIALS B16

ANALYSIS POSITION	C (%)	SI (%)	Mn (%)	P (%)	AL (%)	S (%)	V (%)	Cr (%)	Ni (%)	HEAT NO.
REQUIRED VALUE	0.35 - 0.45	0.02 - 0.03	0.40 - 0.70	0.025	0.010	0.005	0.00 - 0.05	0.02 - 0.05	0.00 - 0.05	
TEST VALUE										
1	0.42	0.02	0.56	0.025	0.012	0.005	0.00	0.00	0.00	403101V
2	0.42	0.02	0.56	0.025	0.012	0.005	0.00	0.00	0.00	403101V
3	0.42	0.02	0.56	0.025	0.012	0.005	0.00	0.00	0.00	403101V
4	0.40	0.02	0.57	0.024	0.010	0.005	0.00	0.00	0.00	403101V

CHEMICAL ANALYSIS AND METALLURGICAL TEST RESULTS IN COMPLIANCE WITH ASTM REQUIREMENTS

THE RESULTS OF TESTS ARE SATISFACTORY

WORK INSPECTOR: [Redacted]

Lot: [Redacted]
 Corrente: [Redacted]

Page 1/1

(d) Exemplo 4.

LISTA DE DOCUMENTOS

CLIENTE: [Redacted] FOLHA: 1 of 8

PROGRAMA: [Redacted]

SEAL: [Redacted]

TRILHA: [Redacted]

LISTA DE PLANOS E PROCEDIMENTOS DO

CONTEÚDO: [Redacted]

RESULTADO ELETRÔNICO: [Redacted]

ÍNDICE DE REVISÕES

REV.	DESCRIÇÃO E/OU FOLHAS ATINGIDAS
0	[Redacted]
A	[Redacted]
B	[Redacted]
C	[Redacted]
D	[Redacted]
E	[Redacted]
F	[Redacted]
G	[Redacted]

DATA: [Redacted]

PROJETO: [Redacted]

EXECUÇÃO: [Redacted]

VERIFICAÇÃO: [Redacted]

APROVAÇÃO: [Redacted]

Fonte: Elaborado pelo autor

¹ As informações sensíveis foram ocultadas a pedido da empresa fornecedora dos documentos.

Um dos objetivos principais do Projeto ATT corresponde a auxiliar o processo de inspeção desses *databooks* pelos profissionais especializados. A inspeção desses documentos deve verificar aspectos referentes a completude do mesmo, ou seja, que todos os arquivos que deveriam fazer parte do documento realmente estão presentes. Deve-se também verificar a conformidade, a qual condiz com a validação das normas as quais os documentos devem seguir, sendo elas normativas textuais como formato e *layout* do documento, como também normativas referentes a intervalos de valores aceitos para testes de materiais. Por fim, ainda se deve procurar por possíveis anomalias nesses documentos, que correspondam a situações de má-fé dos fornecedores ou das empresas terceirizadas.

Atualmente, o processo de inspeção é feito por profissionais especializados na área, os quais conferem completude, conformidade e má-fé de forma manual. Como um *databook* pode ter centenas ou até mesmo milhares de documentos, os quais por si só já possuem uma natureza complexa, a inspeção acaba sendo uma tarefa demorada, custosa e suscetível a falhas humanas. Desta forma justificando a necessidade de se implementar um sistema que de forma automática seja capaz de auxiliar os profissionais responsáveis pela inspeção.

Para a resolução da problemática apresentada, pode-se descrever o sistema, abstraindo-se de conceitos técnicos, como um conjunto de passos sequenciais, de forma que um documento submetido a ele seja segmentado, processado através de uma ferramenta de OCR, classificado quanto ao seu conteúdo, que suas informações textuais e visuais sejam extraídas, armazenadas de forma que mantenham as características de conectividade existentes e que então as informações relevantes sejam disponibilizadas para o usuário de forma amigável e eficiente, sem que haja uma grande curva de aprendizagem no que se refere a experiência do usuário.

Dentre as principais características do sistema desenvolvido no Projeto ATT pode-se destacar o módulo de processamento automatizado de documentos, o qual será abordado ao longo deste trabalho, a fim de elucidar a motivação para a elaboração do mesmo.

Para possibilitar o processo de recuperação de informações, no módulo de processamento, foi empregado um sistema de OCR, facilitando a tomada de decisões durante a análise destes documentos como indicado por Tafti et al. (2016) e Mitra & Chaudhuri (2000), seguido pela etapa de *Document Understanding* (Doc. Und.) e por fim a disponibilização das informações em uma interface de usuário intuitiva.

Contudo, como citado anteriormente, *databooks* são constituídos de documentos com conteúdos diversos, consequentemente apresentando estruturas e complexidades diferentes. Além da estrutura esses documentos em sua natureza diferem de, por exemplo, livros ou artigos científicos os quais através do contexto das palavras contidas neles ou da relação semântica é possível realizar suposições quanto ao conteúdo. Os *databooks* são documentos de leitura complexa para não profissionais, possuem dezenas de códigos e jargões específicos os quais dificultam sua interpretação e somam-se ao desafio de apli-

car técnicas as quais consigam lidar com eles. Para que o sistema consiga performar de forma eficiente é necessário que haja mecanismos capazes de lidar com essas diversidades de formatos e natureza entre os documentos.

Além disso, ao contrário das abordagens descritas na literatura, como por exemplo em (Majumder et al. 2020), onde é sabido a priori quais campos deve-se extrair, no Projeto ATT espera-se ser capaz de identificar e recuperar todas as informações contidas no documento sem que haja conhecimento prévio sobre elas.

1.3 Objetivos

Pensando nisso, este trabalho tem como objetivo utilizar técnicas de extração de informação baseadas em informações espaciais e geométricas do documento digitalizado (*form understanding*), assim como métodos de extração que utilizem do conceito de aprendizado de máquina para otimizar o processo de *document understanding*. Assim extraindo informações com semântica atrelada em documentos com estruturas diversas, bem como propor uma metodologia para *document understanding* em documentos estruturados, semiestruturados e não estruturados.

1.3.1 Objetivos específicos

- Levantar o atual estado da arte quanto a *form understanding* aplicável a documentos estruturados, semiestruturados e não estruturados. A fim de contextualizar o termo, *form understanding*, e o seu uso na área de extração de informação.
- Desenvolver um módulo para *form understanding* de documentos estruturados e semiestruturados para o Projeto ATT, utilizando técnicas distintas, com o objetivo de abordar cada documento de maneira otimizada.
- Apresentar resultados de pesquisa sobre abordagens para lidar com documentos não estruturados. Uma vez que são os documentos que apresentam o maior nível de complexidade quanto a sua interpretação.
- Propor um *pipeline* para *document understanding* de documentos não estruturados, tendo como objetivo indicar técnicas e estratégias para lidar com tais tipos de documentos.
- Contribuir com definições específicas sobre *layout* de documentos, estruturados, semiestruturados e não estruturados. Assim entregando para o leitor um melhor entendimento quanto aos formatos de documentos existentes e logo auxiliando no direcionamento de estratégias específicas para os mesmos.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos básicos necessários ao pleno entendimento deste trabalho. Começando por um breve resumo quanto ao processo de extração de informação, seguido por *layout* de documentos, as definições de *form understanding*, *document understanding* e, por fim, *Optical Character Recognition*.

2.1 Extração de informação

De acordo com Hobbs & Riloff (2010) extração de informação, do inglês *Information Extraction* (IE), consiste no processo de digitalização do texto em busca de informações relevantes, incluindo tarefas como extração das palavras de interesse, como elas se relacionam, bem como os eventos existentes, sendo eles por exemplo, o sujeito de uma ação, o local em que a mesma ocorreu assim como quando ela ocorreu. Em Grishman (2015), nada mais é do que o processo de analisar textos e identificar a existência de conjuntos de palavras com semântica atrelada entre elas.

Já em Liu et al. (2019), o autor define formalmente extração de informação como o processo de extrair informação estruturada de documentos não estruturados. Ainda segundo o autor, essa abordagem é um termo clássico utilizado em atividades de processamento de linguagem natural, nas quais pesquisadores visam extrair entidades e relacionamentos de sentenças de texto simples. Uma desvantagem conhecida desta abordagem se deve a possível perda de uma grande quantidade de informação, quando se leva em consideração apenas as sentenças do texto.

Portanto, se pode dizer que, no geral, entende-se que extração de informação refere-se a extração automática de entidades, os relacionamento entre entidades e atributos que as descrevem no documento em que este processo está sendo empregado, podendo estes documentos serem impressos ou digitais (Sarawagi 2008).

Uma vez que a importância dos dados torna-se cada vez mais relevante para as empresas, naturalmente IE vem sendo implementada nos mais diversos tipos de aplicações e áreas, atingindo desde o setores comerciais através de extração de perfil de usuários (Ahmed Awan et al. 2019) (Pudasaini et al. 2022) e análise social (GIURA n.d.), como

até mesmo na área da saúde (Ruan et al. 2018) (Wang et al. 2018).

Considerando o crescimento de dados na era atual, espera-se que o processo de recuperação desses dados seja otimizado e eficiente, automatizando a maior quantidade de passos as quais forem possíveis. Por conseguinte, os atuais métodos de IE utilizados na literatura possuem raízes em técnicas pertencentes as áreas de Processamento de Linguagem Natural (Singh 2018), Aprendizado de Máquina (Liu et al. 2019), Visão Computacional (Manoharan 2019) e Recuperação de Informação (Martinez-Rodriguez et al. 2020).

Segundo o trabalho de pesquisa desenvolvido em Martinez-Rodriguez et al. (2020), o processo de extração pode ser dividido em uma série de subtarefas as quais visam suprir necessidades específicas existentes de acordo com a estrutura (*layout*) e contexto dos dados que estão sendo manipulados. A qualidade dessas subtarefas impacta o processo como um todo, uma vez que o erro em uma delas acaba sendo propagado para as outras de forma a comprometer o resultado final esperado (Singh 2018).

Dentre as diversas possibilidades de configurações de dados existentes, as subtarefas consideradas padronizadas para a maioria dos casos, ou seja, as usualmente implementadas são denominadas *Named Entity Recognition* (NER), *Named Entity Linking* (NEL) e *Relation Extraction* (RE).

NER refere-se a identificar/reconhecer uma definição para a entidade, como por exemplo a palavra Paris pode ser identificada como uma cidade enquanto a palavra queijo pode ser identificada como um alimento. Logo acaba-se por permitir o agrupamento dessas entidades de acordo com uma classificação em comum. Enquanto NEL busca encontrar semelhanças e possíveis conexões entre entidades diferentes, como por exemplo: Apesar da entidade Paris ser uma cidade, a mesma possui uma conexão com a entidade França, que por sua vez é uma palavra identificada como um país. Já o conceito de RE é facilmente confundido com o de NEL. Neste trabalho segue-se a definição de que RE refere-se a encontrar relações semânticas entre as entidades. Levando em conta o contexto em que esta dissertação está envolvida, essas relações semânticas são basicamente relações entre entidades do tipo pergunta e entidades do tipo respostas como por exemplo: Data (pergunta), 13 de junho (resposta). As definições dessas três subtarefas são essenciais e a sua implementação afeta diretamente o processo de *form understanding*.

No contexto de *form understanding*, tem-se especificamente três principais atividades, *Word Grouping* ((Wang et al. 2021) (Lee et al. 2021) (Li et al. 2021)), *Semantic Entity Labeling* ((Xu et al. 2020), (Xu, Xu, Lv, Cui, Wei, Wang, Lu, Florencio, Zhang, Che, Zhang & Zhou 2021) e (Xu, Lv, Cui, Wang, Lu, Florencio, Zhang & Wei 2021)) e *Entity Linking* ((Tang et al. 2021) (Hwang et al. 2020)), as mesmas fazem parte deste trabalho e serão definidas em mais detalhes na Seção 2.3.

2.2 *Layout de documentos*

Documentos podem ser classificados a partir do seu *layout* (Zhong et al. 2019), ou seja, a disposição geométrica e espacial das entidades existentes ao longo do seu conteúdo. Essas entidades e regiões consistem de blocos de texto, linhas, palavras, títulos, parágrafos, figuras, tabelas, formulários, etc.

Usualmente, *Document Layout Analysis* (DLA), processo de detectar e rotular essas diversas regiões de um documento, é a primeira etapa no pipeline de um sistema denominado *document understanding* (Binmakhashen & Mahmoud 2019) (Appalaraju et al. 2021).

Em Kise (2014) documentos escaneados, aqueles cujos originais estão em formato físico impresso, podem ser divididos em seis categorias, de acordo com a disposição dos elementos na página (*layout*). Como esta classificação remete a documentos manuscritos, históricos ou no formato usualmente encontrado em artigos científicos ou revistas, e os documentos alvo para a realização deste trabalho, apesar de também serem escaneados, possuem natureza técnica (industriais e de engenharia) optou-se por utilizar uma classificação mais generalizada. Desta forma abrangendo uma maior diversidade de *layouts* de documentos.

Portanto, de forma genérica, pode-se classificar os tipos de documentos em três categorias principais no que diz respeito a estrutura lógica, sendo elas documentos estruturados, documentos semiestruturados e documentos não estruturados.

Documentos estruturados representam a classe tradicional de documentos, sendo aqueles que, de acordo com Alam et al. (2003), possuem uma hierarquia bem definida, com títulos, parágrafos e níveis de leitura múltiplos ou únicos, muitas vezes utilizando recursos visuais como aumento da fonte ou palavras em negrito para reforçar a hierarquia existente.

De maneira geral, o termo documentos semiestruturados é relacionado tanto com documentos convencionais, tais como contratos e artigos científicos, como também abrange conteúdo web, tais como páginas em HTML. Durante este trabalho refere-se apenas ao seu significado convencional e clássico, uma vez que os documentos alvo são impressos ou em formato PDF. Segundo Chuang et al. (2009) existem quatro critérios que definem um documento semiestruturado e basta apenas um desses para que um documento seja contemplado nessa categoria. O mais explícito e de fácil identificação se refere a presença de itens que denotam o término de uma sentença diferentes das usuais, como são os elementos de pontuação afirmação/negação, exclamação, interrogação, dois pontos etc. Contudo, esta definição está mais relacionada com as entidades textuais de maneira individual do que com o *layout* do documento em si.

Por sua vez, documentos não estruturados carecem das características exibidas pelas classes anteriores. Como aponta Embley et al. (1998), estes documentos não possuem

relações semânticas e hierárquicas bem definidas ou com predicados associados, não possuindo elementos que formam pares aparentes de perguntas e resposta. Portanto há uma dificuldade de interpretação quanto a relação entre o conteúdo disposto no mesmo.

Ao longo das atividades de pesquisa realizadas durante a realização deste trabalho, além das contribuições referentes aos resultados dos experimentos, percebeu-se também ser necessário apresentar propostas de definições próprias para classificação dos tipos de documentos, sendo elas:

Documentos estruturados: São documentos cuja estrutura é conhecida e se repete ao longo dos exemplares, logo possuem um *layout* básico, sabendo-se onde cada campo está localizado. Independente da quantidade de documentos, se eles forem da mesma empresa e do mesmo tipo, todos possivelmente terão a mesma disposição do conteúdo. Desta forma facilitando o processo de extração automatizada de informação. Um exemplo deste tipo de documentos são tabelas de preenchimento de respostas de provas e concursos, como mostra a Figura 2, cuja estrutura é a mesma para todos os alunos e candidatos, sendo preenchida de maneira semelhante entre todos os participantes, logo tornando a correção e avaliação através de sistemas automatizados de análise de documentos um processo trivial.

Figura 2: Gabarito de concurso: Documento estruturado.

Nome:			
Assinatura:			
Avaliação: Sim Não Indeterminado			

1	A	B	C	D	E
2	A	B	C	D	E
3	A	B	C	D	E
4	A	B	C	D	E
5	A	B	C	D	E
6	A	B	C	D	E
7	A	B	C	D	E
8	A	B	C	D	E
9	A	B	C	D	E
10	A	B	C	D	E
11	A	B	C	D	E
12	A	B	C	D	E
13	A	B	C	D	E
14	A	B	C	D	E
15	A	B	C	D	E
16	A	B	C	D	E
17	A	B	C	D	E
18	A	B	C	D	E
19	A	B	C	D	E
20	A	B	C	D	E

21	A	B	C	D	E
22	A	B	C	D	E
23	A	B	C	D	E
24	A	B	C	D	E
25	A	B	C	D	E
26	A	B	C	D	E
27	A	B	C	D	E
28	A	B	C	D	E
29	A	B	C	D	E
30	A	B	C	D	E
31	A	B	C	D	E
32	A	B	C	D	E
33	A	B	C	D	E
34	A	B	C	D	E
35	A	B	C	D	E
36	A	B	C	D	E
37	A	B	C	D	E
38	A	B	C	D	E
39	A	B	C	D	E
40	A	B	C	D	E

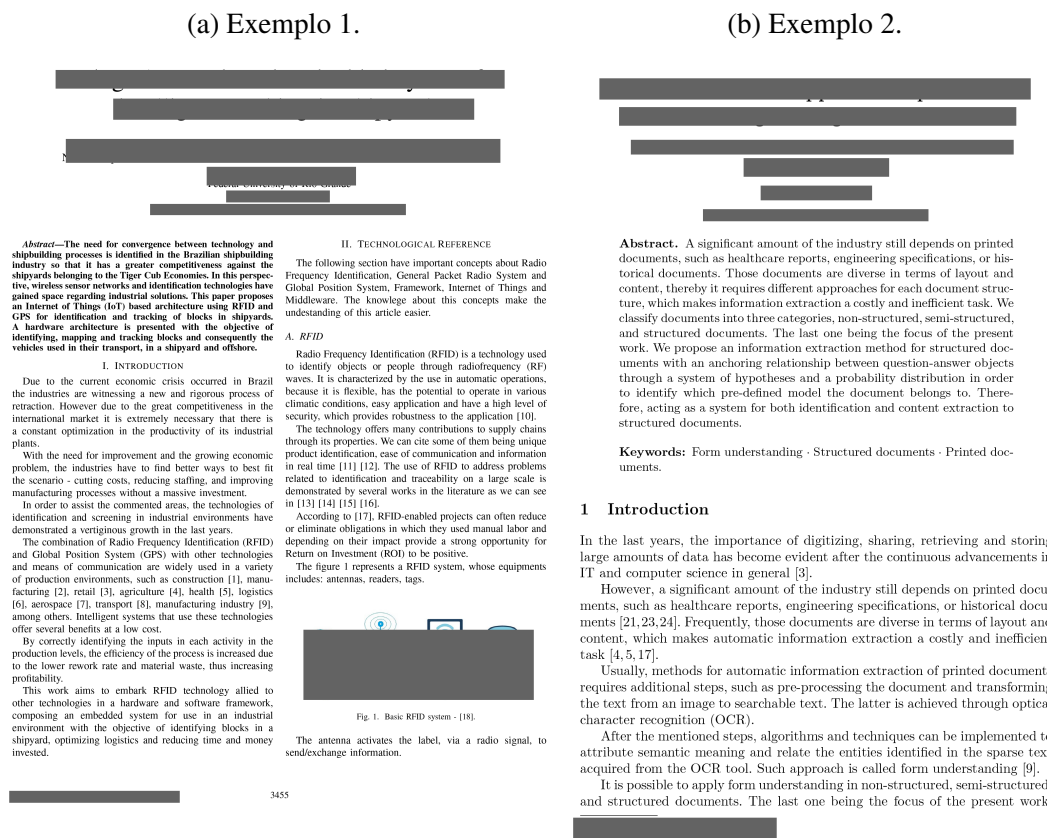
41	DISCURSIVA				
42	DISCURSIVA				
43	A	B	C	D	E
44	A	B	C	D	E
45	A	B	C	D	E
46	A	B	C	D	E
47	A	B	C	D	E
48	A	B	C	D	E
49	A	B	C	D	E
50	A	B	C	D	E
51	A	B	C	D	E
52	A	B	C	D	E
53	A	B	C	D	E
54	A	B	C	D	E
55	A	B	C	D	E
56	A	B	C	D	E
57	A	B	C	D	E
58	A	B	C	D	E
59	A	B	C	D	E
60	A	B	C	D	E

61	A	B	C	D	E
62	A	B	C	D	E
63	A	B	C	D	E
64	A	B	C	D	E
65	A	B	C	D	E
66	A	B	C	D	E
67	A	B	C	D	E
68	A	B	C	D	E
69	A	B	C	D	E
70	A	B	C	D	E
71	A	B	C	D	E
72	A	B	C	D	E
73	A	B	C	D	E
74	A	B	C	D	E
75	A	B	C	D	E
76	A	B	C	D	E
77	A	B	C	D	E
78	A	B	C	D	E
79	A	B	C	D	E
80	A	B	C	D	E

Fonte: Elaborado pelo autor

Documentos semiestruturados: São aqueles documentos que possuem campos estruturados, contudo em sua maioria apresentam uma variação entre um conjunto de documentos similares. Ou seja, não são limitados a um esquema de *layout* fixo, havendo variações. Pode-se citar como exemplo, revistas, artigos científicos (como mostra a Figura 3), *websites* dentre outros.

Figura 3: Artigos científicos: Documentos semiestruturados



Fonte: Elaborado pelo autor.

Documentos não estruturados: São aqueles que possuem um nível de abstração superior, dos quais não se tem ciência prévia do formato, assim como o conteúdo que será analisado pelo sistema, a disposição espacial e geométrica não são conhecidas, desta forma sendo a categoria mais difícil de ser submetida ao processo de extração de informação automatizada de documentos. A Figura 4 demonstra exemplos deste tipo de documento.

2.3 Form understanding

A coleta e organização de dados historicamente se dá através de documentos de formatos específicos, normalmente classificados como formulários. Estes formatos estão presentes em diversas áreas de atuação, tais como relatórios hospitalares, contratos, avaliações escolares e etc (Jaume et al. 2019) (Mao et al. 1996). Assim, confirmando que formulários são uma forma comum e efetiva de coleta de dados.

Para realizar o armazenamento e controle destes formulários, tende-se a digitalizá-los, tornando-os documentos no formato de imagem. Quando se faz necessário, a tarefa de recuperação de informação dos mesmos torna-se complexa do ponto de vista computacional, pois é necessário aplicar um processo de OCR, a fim de tornar as imagens na forma de texto selecionável.

Entretanto, quando se trata de sistemas automatizados para extração de informação de documentos, apenas a etapa de OCR, em muitos casos não é o suficiente. O OCR extrai apenas texto esparsos, isto é, sem informação semântica atrelada, dificultando a extração de conhecimento dos documentos processados. Para auxiliar e automatizar este processo é sugerido a utilização da técnica de *form understanding*.

Segundo Jaume et al. (2019), de maneira geral, entende-se *form understanding* como um processo de extração automatizada de texto e estruturação de formulários baseado na detecção e reconhecimento das palavras contidas no mesmo. Para isso, primeiramente é feita a análise das informações espaciais do *layout* do documento, logo após se faz o cruzamento destes dados com o texto, podendo então realizar a identificação de entes que são classificados como perguntas, respostas e cabeçalhos presentes no próprio formulário. Por fim, é necessário compreender e assimilar como estas entidades estão relacionadas entre si.

Ainda de acordo com Jaume et al. (2019), *form understanding* quando aplicado a documentos não estruturados, pode ser descrito em três principais etapas, sendo elas:

- *Word Grouping* - Etapa que consiste em agregar entidades compostas por múltiplos *tokens* em uma única célula semântica (Lassoued & Deleris 2016). Por exemplo, no caso em que se tenha uma *question* composta pelos *tokens* “Número do documento”, a etapa de *word grouping* caracteriza-se por agregar as palavras “número”, “do”, “documento” em uma única entidade, para posteriormente ser classificada dentre algumas classes pré-definidas. Nesta etapa, diversas técnicas são aplicadas sendo algumas delas, como por exemplo, técnicas baseadas em estatística, distribuição espacial, bem como técnicas mais robustas baseadas em *Graph Convolutional Networks* (GCN) (Kipf & Welling 2016).
- *Semantic Entity Labeling* (SEL) - Atividade responsável por classificar entidades em determinadas categorias pré-definidas, sendo elas: Q - *Question*, A - *Answers*,

H - *Header* e O - *Others*. Esta etapa baseia-se na utilização de diversas *features* disponíveis no documento, como, por exemplo, disposição espacial, características da fonte do texto (tamanho da fonte e peso da fonte) e a relação semântica entre as entidades. Ou seja, utiliza-se do contexto em que o *token* está inserido no documento (Pham et al. 2016). Além disso, para se obter êxito durante a etapa de SEL também faz-se uso de redes neurais artificiais em conjunto com técnicas de *Natural Language Processing* (NLP) (Chowdhury 2003) e *Natural Language Understanding* (NLU) (Allen 1988).

- *Entity Linking* - Última etapa do processo de *form understanding*, cujo objetivo corresponde a realizar a predição da relação entre as entidades semânticas classificadas na etapa de SEL, ou seja, as informações classificadas são então empregadas como entrada para algoritmos de aprendizado de máquina capazes de analisar as possíveis relações entre as entidades de *Question* e *Answer* disponíveis para aquele documento (Han & Sun 2012).

Neste trabalho, tem-se como importante tarefa alvo a atividade de *Semantic Entity Labeling*, realizando uma exploração das técnicas e métodos disponíveis para tal na literatura.

2.4 Document understanding

Document understanding é o termo utilizado para referir um conjunto de técnicas aplicadas em um documento com o objetivo de realizar a extração de conhecimento do mesmo, integrando aspectos como por exemplo *form understanding* (Wang et al. 2020) e *table understanding* (Deng et al. 2020).

Normalmente, os dados de entrada deste tipo de sistema consistem de documentos de estruturas complexas, podendo conter desde um simples texto, como uma página de um livro, até mesmo uma estrutura mais imprevisível como uma página de revista, onde normalmente o *layout* contém imagens, textos em diversas fontes e tamanhos, bem como apresenta uma distribuição esparsa e muitas vezes com uma ordem de leitura não muito explícita. Este tipo de documentos são os mais confusos de lidar pois, com as abordagens usualmente utilizadas na literatura, o processo de extração automatizada de conhecimento torna-se complexo.

2.5 Tesseract OCR

Optical Character Recognition (OCR) é um processo muito utilizado, e de grande utilidade, que consiste em transformar textos-imagens, ou seja, textos impressos que foram digitalizados, em texto selecionável e editável. O sucesso do resultado de uma extração

de texto através do processo de OCR está diretamente relacionado com a qualidade do documento. Pois, quanto maior for a qualidade da digitalização, quanto mais linear for o *background* da página a ser processada no documento e quanto menos ruídos apresentar, melhor será o resultado da extração (Patel & Bhatt 2020). Atualmente existem diversas ferramentas de OCR no mercado, como por exemplo *Google Vision* (que está incluso na ferramenta *Google Docs*), *Abbyy FineReader*, *Microsoft OCR* (*Azure Cognitive Services*), dentre outros.

No desenvolvimento deste trabalho e do Projeto ATT, optou-se pela ferramenta *Tesseract OCR*, devido a disponibilidade de código fonte e pelo fato de ser um *framework* utilizado em diversos trabalhos com resultados positivos como (Hasnat et al. 2009), (Thakare et al. 2018) e (Agbemenu et al. 2018).

Tesseract OCR é um motor de OCR, inicialmente desenvolvido pela HP, posteriormente mantido pelo Google e atualmente disponibilizado na plataforma *Github* no formato de código aberto (Smith 2007) (Koistinen et al. 2017), possuindo suporte a diversos idiomas. É disponibilizado no formato de *framework*, linha de comando e *Application Programming Interface* (API), característica esta que facilita sua integração ao desenvolvimento de aplicações.

O Tesseract, mesmo sendo *open-source*, apresenta resultados de acurácia equiparável à outras opções de motores de OCR comerciais disponíveis no mercado (Tafti et al. 2016), já que o mesmo foi desenvolvido com o foco em reduzir o número de casos de rejeição à obter uma acurácia alta (Patel et al. 2012).

3 ESTADO DA ARTE

Neste capítulo são apresentados brevemente trabalhos de pesquisa e desenvolvimento, os quais possuem como tema principal a problemática de realizar as tarefas de extração, manipulação, classificação ou compreensão de documentos dos mais diversos tipos, sendo suas soluções genéricas ou específicas.

3.1 Produção científica

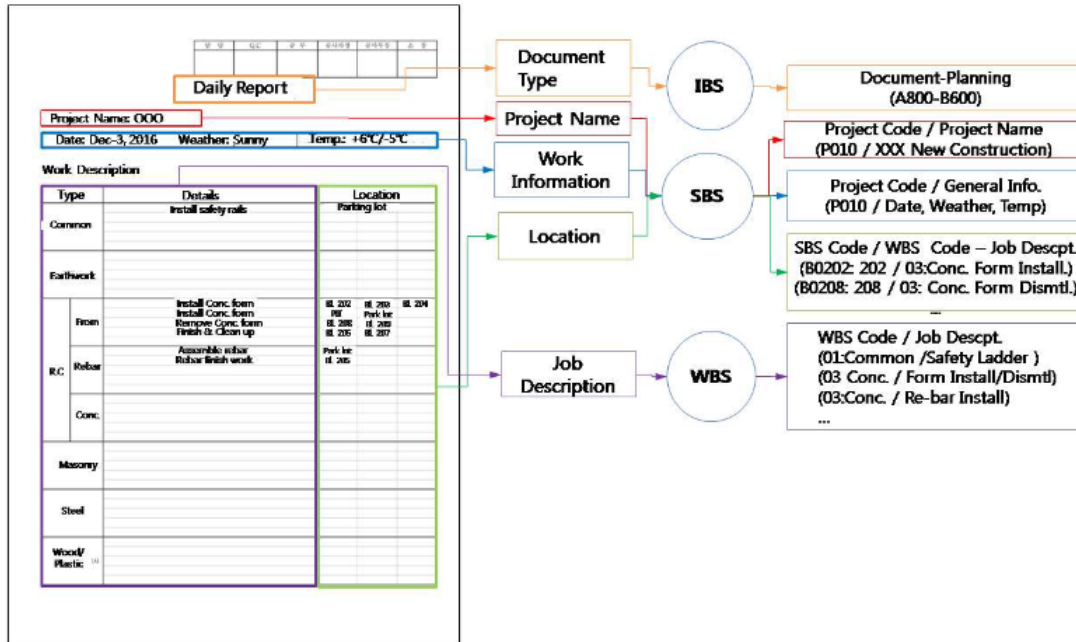
Digitalizar, classificar, manipular, entender e utilizar com eficiência grandes volumes de dados é uma área de pesquisa científica que engloba os mais diversos setores, entregando desde soluções genéricas quanto abrangendo problemáticas mais específicas.

Como se pode observar em Promin & Suriyachai (2019), onde é apresentado o design e implementação de um sistema de gerenciamento de documentos médicos digitalizados chamado SCAN, o qual utiliza-se da tecnologia de OCR para transcrever esses documentos em formatos que sejam suportados pelo banco de dados integrado a um *Hospital Information System* (HIS), o qual irá utilizar esses documentos em conjunto com dados já existentes no mesmo para auxiliar os profissionais de saúde quanto a rápida aquisição e manipulação de dados referentes a seus pacientes, bem como disponibilizar um sistema capaz de armazenar a relação entre esses diversos tipos de documentos.

Já Cha & Lee (2018) estabeleceu uma estrutura de gerenciamento de documentos não estruturados de projetos de construção que utiliza um banco de dados baseado em um *building information modelling* (BIM) para o armazenamento e utilização eficiente de informações, visando um gerenciamento, pesquisa e recuperação local eficaz de documentos relacionados ao processo de construção. Foram utilizados sistemas de *spatial breakdown structure* (SBS), *classification breakdown structure* (CBS) e *information breakdown structure* (IBS) para linkar e classificar, ambos em questão de tipo e contexto, os termos presentes nos documentos.

Uma vez classificados os documentos são armazenados de duas maneiras, na forma estruturada e não estruturada, como mostra a Figura 5.

Figura 5: Exemplo de ligação da estrutura de *breakdown* em um documento de relatório diário (Cha & Lee 2018).



Fonte: Elaborado por Cha & Lee (2018).

Entrando nas soluções com cunho mais genérico, tem-se Aiello et al. (2002) onde foi proposto um sistema de *document understanding* capaz de analisar coleções de documentos heterogêneos de forma a utilizar o *generic knowledge* referente a estrutura de cada um. Ou seja, as *features* as quais os caracterizam (posicionamento do texto, tamanho da fonte, cor e etc). Os mesmos então são classificados em duas classes. A classe de documentos genéricos e a classe de documentos específicos, sendo a última referente aqueles documentos que possuem características conhecidas ou semelhantes, como por exemplo em arquivos de transações ACM em que o texto é usualmente escrito utilizando a fonte *Times Roman* em um tamanho de 10 *pixels*. Neste trabalho cada documento é descrito de duas maneiras, lógica e geometricamente, demonstrada na Equação 1:

$$\mathcal{D} = (\mathcal{G}, \mathcal{L}) \quad (1)$$

O conjunto $\mathcal{G}$ representa uma descrição de *layout* a fim de transcrever a apresentação do documento utilizando grafo, na qual os nodos são os objetos, sendo eles tanto imagens quanto textos, pertencentes aos documentos e as arestas representam a existência e o potencial valor de uma relação entre dois ou mais objetos do documento. O formato dos objetos no texto são considerados retangulares, quando um objeto possui um formato desconhecido é então levada em consideração a *bounding-box* do mesmo.

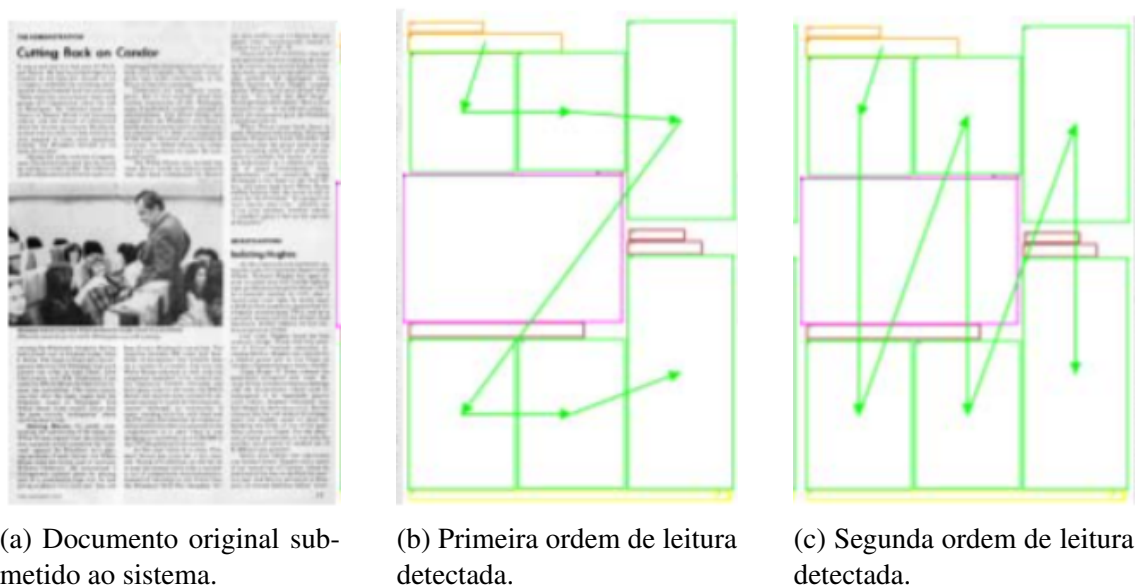
Já o conjunto $\mathcal{L}$ é responsável por organizar o conteúdo lógico relacional entre as entidades do texto, definindo as relações de chamadas de *BeforeinReading*, as quais identificam a possível hierarquia de leitura entre os objetos, a ordem na qual cada um deve ser

lido para manter a compreensão do documento. De forma análoga ao conjunto anterior, é utilizada a representação por forma de grafos, onde os nodos correspondem aos objetos e as arestas a existência da relação lógica entre eles.

Ambos conjuntos de *features* são extraídos de acordo com hipóteses formadas pela experiência adquirida em um processo anterior de observação e pelo módulo de processamento de linguagem natural, o qual usa *n-grams* para computar as probabilidades de transição entre dois blocos de texto. Para este processo foram utilizados um total de 400.000 arquivos.

A Figura 6a representa um exemplo de documento analisado pelo modelo proposto no qual as imagens 6b e 6c representam tanto o *layout* quanto às ordens de leitura identificadas pelo sistema, sendo a Figura 6b correspondendo ao *layout* e ordem corretos.

Figura 6: Resultado obtido por Aiello et al. (2002)



(a) Documento original submetido ao sistema.

(b) Primeira ordem de leitura detectada.

(c) Segunda ordem de leitura detectada.

Fonte: Elaborado por Aiello et al. (2002)

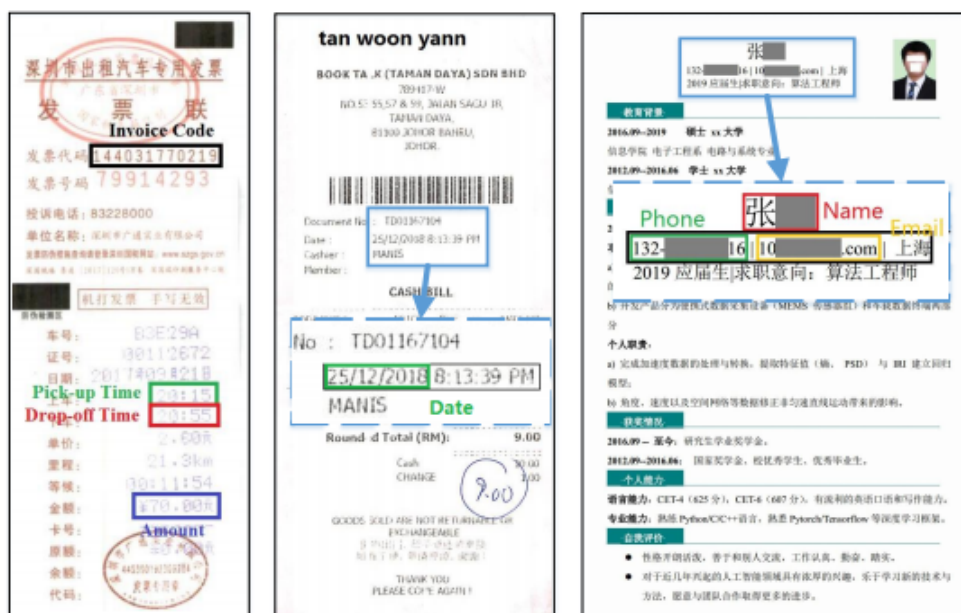
Em relação a trabalhos mais recentes, pode-se citar como exemplo a pesquisa desenvolvida por Devlin et al. (2018), a qual apresenta o modelo BERT, *Bidirectional Encoder Representations from Transformer*, que propõe utilizar objetos pré treinados do tipo MLM (*masked language model*) em conjunto com *Next Sentence Prediction* (NSP), com o objetivo de solucionar tarefas dentro do contexto de *language understanding system* que necessitem de soluções capazes de entender a relação entre duas ou mais sentenças, como são os casos de *Question Answering* (QA) e *Natural Language Inference* (NLI), sem que haja a necessidade de que o sistema tenha arquiteturas específicas para cada tarefa a qual está sendo empregado.

Já Zhang et al. (2020) propõe um *framework* que una os conceitos de *text reading* e *information extraction* para treinar uma rede capaz de realizar a tarefa de *VRD understanding*, ou seja, que o modelo computacional consiga assimilar de forma ordenada o

conhecimento contido nestes tipos de documentos de forma semelhante ao processo de compreensão de informação desempenhado pelos seres humanos. O *framework* é separado em três módulos: *text reading*, *multimodal context block* e *information extraction*.

O módulo de *text reading* é responsável por localizar e reconhecer no documento tudo que seja um possível texto. Inicialmente tem como dados de entrada imagens que representam documentos e entrega como primeiro resultado da rede ResNet as *bounding-boxes*, coordenadas das posições das regiões que contêm os textos detectados. Essas coordenadas são entregues para um algoritmo capaz de reconhecer sequências de caracteres, assim obtendo as características textuais de cada texto. O contexto da informação se dá então pelo módulo multimodal, que por sua vez irá prover tanto o contexto textual (utilizando mecanismos de atenção) quanto o visual (através de redes neurais convolucionais), já que o primeiro modela as relações implícitas entre as imagens do documento enquanto o último resume os padrões locais de cada texto. Com ambas as *features* textuais e contextuais o módulo de *information extraction* aplica uma *Bidirectional-LSTM*, a fim de eliminar a dependência entre caracteres, seguida por uma rede neural que é então treinada e encarregada de entregar como saída as *labels* referentes ao texto. O *framework* foi testado utilizando três *datasets* com documentos da vida real formados por faturas, currículos e notas fiscais, como demonstrado na Figura 7.

Figura 7: Tipos de documentos utilizados no trabalho de (Zhang et al. 2020).



Fonte: Elaborado por Zhang et al. (2020).

Para a realização da metodologia para documentos estruturados aqui em questão, dois artigos foram de extrema relevância, servindo como base para a proposta desenvolvida.

O primeiro foi o trabalho proposto em Jaume et al. (2019), o qual apresenta um *dataset* para *form understanding* chamado FUNSD que tem como objetivo auxiliar na tarefa de

extrair e estruturar conteúdo textual de formulários. Inicialmente foram separados 25.000 documentos, dos quais 3200 passaram por um processo de elegibilidade e 199 foram anotados. Além do *dataset*, este trabalho contribuiu com uma definição formal para “*Form Understanding*” e com um modelo de descrição das entidades semânticas presentes em um documento em uma forma interligada, onde entidades são definidas como perguntas ou respostas, sendo as respostas ancoradas nas perguntas. Esse modelo se dá por uma lista em que cada entidade é descrita por um identificador único, um rótulo que define a entidade como pergunta ou resposta, a *bounding-box* e todas as entidades ligadas a ela. As *bounding-boxes* são descritas em coordenadas, seguindo o esquema descrito na Equação 2 e o modelo em si é representado no formato JSON como mostra a Figura 8.

$$(x_{left}, y_{top}, x_{right}, y_{bottom}) \quad (2)$$

Figura 8: Exemplo de descrição de modelo proposto por Jaume et al. (2019)

```
{
  "form": [
    {
      "id": 0,
      "text": "Registration No.",
      "box": [94,169,191,186],
      "linking": [
        [0,1]
      ],
      "label": "question",
      "words": [
        {
          "text": "Registration",
          "box": [94,169,168,186]
        },
        {
          "text": "No.",
          "box": [170,169,191,183]
        }
      ]
    },
    {
      "id": 1,
      "text": "533",
      "box": [209,169,236,182],
      "label": "answer",
      "words": [
        {
          "box": [209,169,236,182],
          "text": "533"
        }
      ],
      "linking": [
        [0,1]
      ]
    }
  ]
}
```

Fonte: Elaborado por Jaume et al. (2019)

Já o segundo trabalho que serviu de base foi o de Medvet et al. (2011), no qual é proposto um método para realizar a extração de informação como uma das etapas necessárias para a problemática de *document understanding* de documentos impressos de múltiplas páginas, os quais possam ser representados através de um conjunto de classes

disponíveis, chamadas modelos, que por sua vez são formadas a partir da semelhança de *layout* e conteúdo entre os documentos. A descrição das classes foi feita através de uma interface de usuário na qual, a partir dos documentos de exemplos submetidos para a etapa de processamento, entrega como saída os blocos de objetos contidos no texto, extraídos pela ferramenta de OCR.

No trabalho o autor associa o documento com dois elementos que os descrevem, sendo eles:

- Um esquema que contém uma série de objetos referentes ao arquivo e tem como objetivo encontrar valores para estes objetos baseado no conteúdo.
- Um modelo o qual consiste de uma série de regras, uma para cada objeto presente no esquema, o qual irá descrever como encontrar os valores nos blocos de documentos.

Os blocos relevantes são então classificados de acordo com a probabilidade de serem consistentes, de a informação estar presente em uma sequência de blocos identificada pelo sistema, utilizando como base a geometria (tamanho, posição e página) e informação textual (tamanho do texto, alinhamento e conteúdo) de cada documento. Os parâmetros utilizados na abordagem probabilística foram estimados utilizando *maximum likelihood method* (Equação 3) para as amostras de classes identificadas.

$$P = \prod_{i=1}^n \left(\sum_k s_{i,k} I(k) N(\mu_{i,k}^{y'}, \sigma_{i,k}^{y'}) N(\mu_{i,k}^h, \sigma_{i,k}^h) \times N(\mu_{i,k}^{x''}, \sigma_{i,k}^{x''}) N(\mu_{i,k}^{w'}, \sigma_{i,k}^{w'}) M_{i,k} \right) \quad (3)$$

Onde o significados de cada parâmetro é descrito na tabela 1.

Tabela 1: Parâmetros da regra probabilística aplicada.

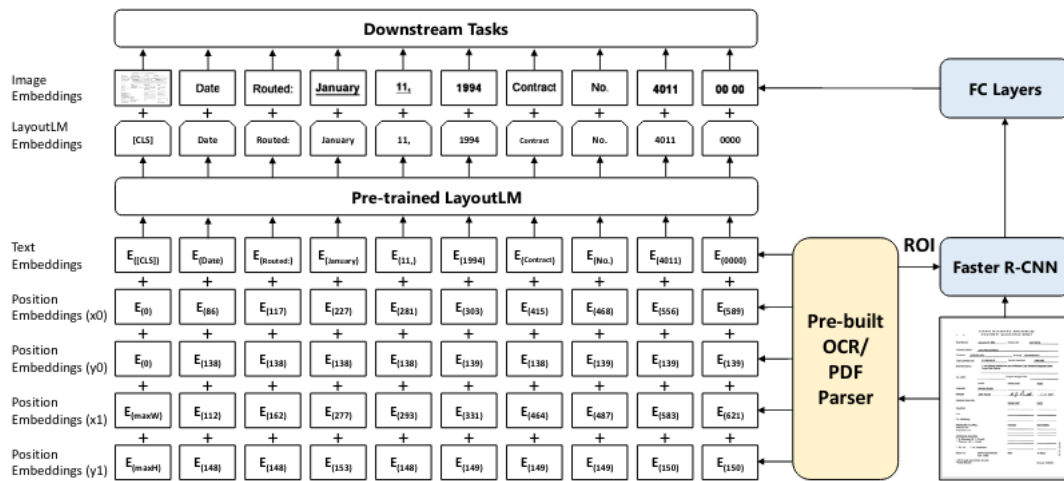
Parâmetro	Significado
$\mu_{i,k}^{x''}$	média da posição relativa ajustada de x''
$\sigma_{i,k}^{x''}$	escala da posição relativa média de x'' ,
$\mu_{i,k}^{y'}$	média da posição relativa y'
$\sigma_{i,k}^{y'}$	escala da posição média relativa de y'
$\mu_{i,k}^{w'}$	média da largura para o carácter w'
$\sigma_{i,k}^{w'}$	média da largura do carácter w'
$\mu_{i,k}^h$	média da altura h
$\sigma_{i,k}^h$	escala da altura h

O sistema foi então avaliado em um *dataset* com 807 documentos impressos com múltiplas páginas, pertencentes a três principais classes, sendo elas: patentes, *invoices*, e *datasheet* de componentes eletrônicos.

Já para os resultados preliminares referentes a documentos semiestruturados e não estruturados, adotou-se como base outros dois artigos, sendo o primeiro proposto por Xu et al. (2020) e denominado *LayoutLM*.

LayoutLM é um modelo pré-treinado baseado em arquitetura BERT, que utiliza a informação textual e do *layout* para o processo *document understanding* de documentos escaneados através da posição relativa do *token* no documento por um *2d position embedding* e um *image embedding* que utiliza um modelo de R-CNN para produzir os *embeddings* das regiões de interesse - *Regions Of Interest (ROI)*. Dessa forma o *embedding 2d* consegue entregar as relações entre os *tokens* enquanto o *image embedding* fica responsável por representar características visuais dos *tokens* como a tipografia, cores e etc, conforme apresentado na Figura 9.

Figura 9: Arquitetura do modelo pré-treinado *LayoutLM*.



Fonte: Elaborado por Xu et al. (2020).

Em sua arquitetura, o *LayoutLM* propõe também, um *Masked Visual-Language Model (MVLM)* e *Multi-label document classification (MDC)*. O MDC tem como objetivo gerar melhores representações em nível de documento, enquanto o MVLM tem como objetivo aprender a representação da linguagem referente ao documento de forma que durante o pré-treinamento alguns *tokens* de entrada tenha seu conteúdo mascarados e a relação de posição mantida, assim fazendo com que o modelo treinado faça a predição dos *tokens* de acordo com o contexto. Assim o modelo consegue entender o contexto de linguagem, como também utiliza as informações de posição 2D correspondentes. O autor destaca que não está no escopo da pesquisa realizar a atividade de *word grouping* e *entity linking*.

Já o segundo artigo, denominado *LayoutLMv2*, nada mais é do que uma melhoria da primeira versão do *LayoutLM*, no qual são aceitas três modalidade de entrada: imagem, texto e *layout*. Outra mudança se dá pelo fato que nessa nova versão o modelo combina os *image embeddings* no processo de *fine-tuning* (ajuste), de forma que essas informações

são integradas no processo de pré-treinamento, fazendo uso das vantagens atreladas a arquitetura de *Transformers* para que o modelo consiga então aprender a relação cruzada entre as informações visuais e textuais. Para os *text embeddings* é utilizado *WordPiece* (Wu et al. 2016), para o *visual embedding* é utilizada a arquitetura ResNeXt-FPN (Xie et al. 2017) e para o *layout embedding* são utilizadas duas *embedding layers* de forma análoga a abordagem utilizada na primeira versão *LayoutLM*.

3.2 Produtos comerciais

Existem no mercado ferramentas para extração de informação. Nenhuma delas genérica o suficiente para tratar diversos documentos sem que seja necessário um *dataset* completo e que sejam feitas diversas anotações de forma manual. A seguir são apresentadas algumas destas soluções disponíveis.

- Microsoft Azure Form Recognizer - Essa ferramenta apresenta diversas funções, dentre elas extração de tabelas, formulários e pares-chave, ou seja perguntas e respostas, disponibilizando os resultados em formato texto ou JSON, utilizando-se da mesma descrição adotada no trabalho aqui proposto e no apresentado em Jaume et al. (2019). A ferramenta visa o processamento de faturas, cartões de visitas e notas fiscais de venda, podendo ser carregados seus próprios modelos. Os documentos devem ser estruturados visto que a solução utiliza conceitos de aprendizagem de máquina para realização da extração de informação. Trata-se de ferramenta que envolve custos para utilização.
- Google Document AI - Também realiza a extração de informação e, assim como o anterior, é voltada para documentos estruturados, tais como faturas. A sua função principal é extrair conhecimento destes documentos para gerar *insights* sobre os mesmos. Também se trata de ferramenta que envolve o pagamento de serviços a serem adquirido em conjunto, como containerização, *web-services* e acesso a plataforma do Google AI.
- Nanonets - Plataforma que utiliza conceitos de *form understanding* e *document understanding* com foco no preenchimento automatizado de tabelas, utilizando-se dos dados extraídos. Também necessita de uma base de dados e envolve custo.

3.3 Discussão

No que diz respeito aos trabalhos apresentados neste capítulo, tanto os de natureza acadêmica como os comerciais possuem particularidades que contribuíram para a decisão da não utilização dos mesmos na solução do problema aqui em questão.

No que se refere as soluções comerciais, as três ferramentas não se mostraram adequadas, conforme a seguir justificado.

O *Microsoft Azure*, apesar de apresentar características que atendem algumas das necessidades relatadas no Projeto ATT (extração de tabelas, extração de pares pergunta-resposta, saída em formato JSON e etc), tem como principal empecilho o fato de ser uma abordagem aplicável apenas a documentos estruturados. Como se tem como objetivo atingir resultados em documentos tanto estruturados como semiestruturados, a sua implementação tornou-se inviável.

Considerando que se espera que as aplicações aqui vislumbradas envolvam o processamento mensal de um elevado volume de páginas, uma vez que na base de dados do projeto existem *databooks* que possuem mais de 10000 páginas, o custo da ferramenta *Nanonets* a inviabilizou.

Com relação ao *Google Document AI*, a característica sigilosa dos *databooks* envolvidos no projeto a inviabilizou, apesar de ser uma das excelentes ferramentas existentes no mercado.

Para as soluções acadêmicas, a solução de Promin & Suriyachai (2019) apresenta um sistema robusto, direcionado para documentos médicos. As suas características assemelham-se as que se deseja obter ao final do Projeto ATT. Contudo, a sua utilização não é viável aqui, considerando que os *templates* dos documentos médicos escaneados são conhecidos e semelhantes entre si, o que não ocorre no Projeto ATT.

A solução de Aiello et al. (2002) utiliza descrição de *layout* da página fazendo uso de artifícios visuais como tamanho, tipo de tipografia, se a entidade está em negrito ou itálico, etc, para facilitar a extração das informações. Nos *databooks* os *layouts* dos documentos não respeitam uma regra de formatação definida. Cada fornecedor tem seus próprios *templates* e maneiras de preencher as informações, podendo utilizar elementos tipográficos e visuais de diferentes formas. Além disso, os fornecedores estrangeiros, principalmente aqueles que além de utilizarem idiomas diferentes também utilizam outros tipos de alfabetos, podem adotar artifícios de maneiras diferentes. Por exemplo, é comum que um documento com caracteres chineses tenha ao lado de cada elemento a sua tradução para o inglês entre parênteses, com uma fonte reduzida e em itálico.

O trabalho de Devlin et al. (2018), por ser baseado em soluções que abordam problemas como *QA* e *NLI*, implicitamente espera-se que as entidades que estão sendo extraídas e processadas no documento de alguma forma tenham relação entre si, trazendo a ideia do contexto por trás das palavras e sentenças. Como os *databooks* são compostos por

muitos códigos constituídos por letras e números, este tipo de solução não é atrativa para o problema.

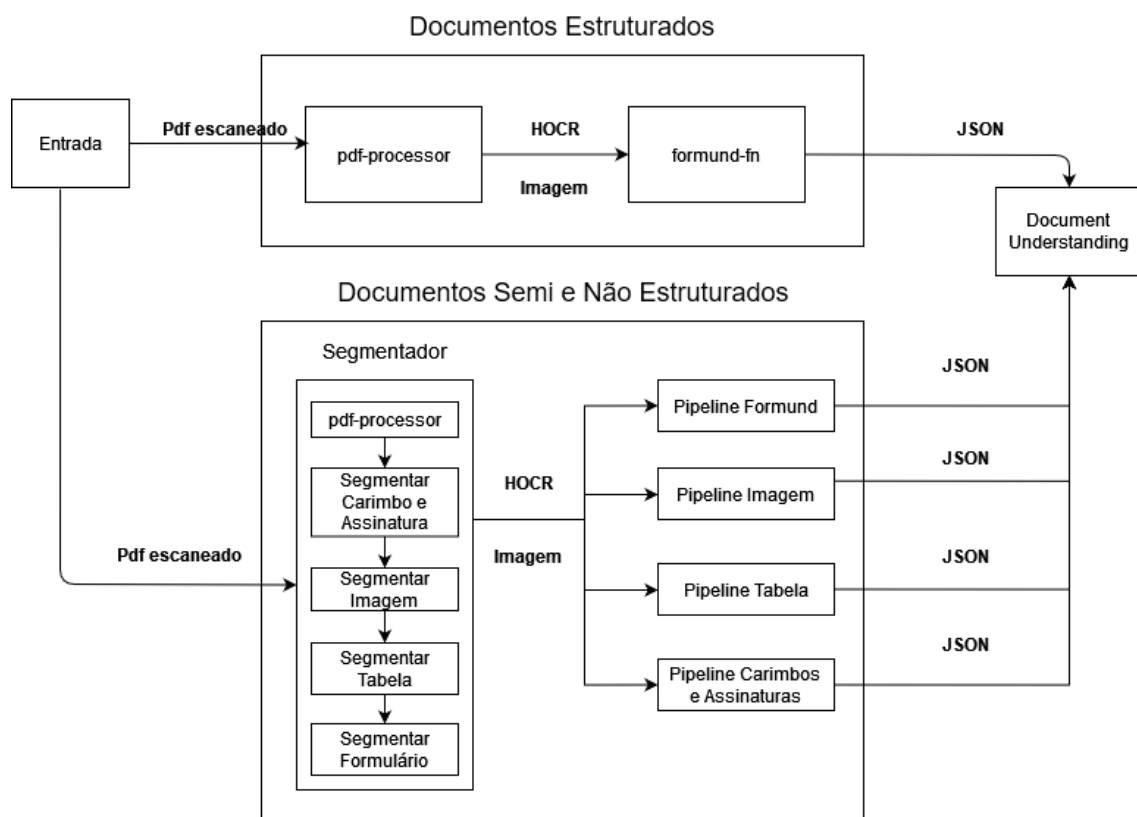
Por sua vez, as soluções de Cha & Lee (2018) e Zhang et al. (2020) entregaram resultados próximos ao que se espera atingir no Projeto ATT. Ambas foram estudadas com o objetivo de entender as estratégias e tecnologias que possivelmente auxiliariam na resolução do problema proposto. Contudo, foram as soluções desenvolvidas por Jaume et al. (2019), (Medvet et al. 2011) e Xu et al. (2020) que de fato serviram de referência neste trabalho. No capítulo 4 é apresentado o que foi implementado ou adaptado a partir dos trabalhos desses autores.

4 METODOLOGIA

Este capítulo descreve a metodologia proposta para o desenvolvimento do sistema de software vislumbrado, capaz de suprir a demanda por otimização na análise e processamento de *databooks*.

Inicia-se por apresentar, na Figura 10, uma *big picture* da estrutura do sistema, através dos blocos que o constituem e suas inter-relações. Após, detalham-se as funcionalidades desses blocos e como ocorrem os fluxos de dados intra e entre eles.

Figura 10: Diagrama do sistema de extração automática de informação proposto separado em blocos distintos para cada tipo de documento.



Fonte: Elaborado pelo autor.

Isto posto, a metodologia aqui apresentada é dividida em duas partes. Primeiramente, aborda-se os aspectos referentes a documentos estruturados (Seção 4.1). Após,

apresentam-se os aspectos referentes a documentos semiestruturados e não estruturados (Seção 4.2).

Quanto aos documentos estruturados também é descrita a arquitetura de como a abordagem foi implementada no Projeto ATT (Seção 4.1.3). Já para os documentos semiestruturados e não estruturados, é detalhada a metodologia utilizada por trás dos experimentos realizados e na construção do *dataset* próprio utilizado nos mesmos.

4.1 Metodologia para documentos estruturados

A metodologia de *document understanding* para documentos estruturados, como mostrado na Figura 10, tem como base duas funções principais, sendo elas: *PDF-processor-fn* e *formund-fn*, que serão apresentadas nas Seções 4.1.1 e 4.1.2, respectivamente.

Já a etapa de extração, implementada na função *formund-fn*, é baseada em duas etapas: extração de entidades e extração de tabelas, que serão apresentadas nas Seções 4.1.2.1 e 4.1.2.2, respectivamente.

4.1.1 PDF-processor-fn

Como função principal do *pipeline* de processamento de documentos automático no Projeto ATT, tem-se a função chamada *PDF-processor-fn*, descrita pelo fluxograma mostrado na Figura 11.

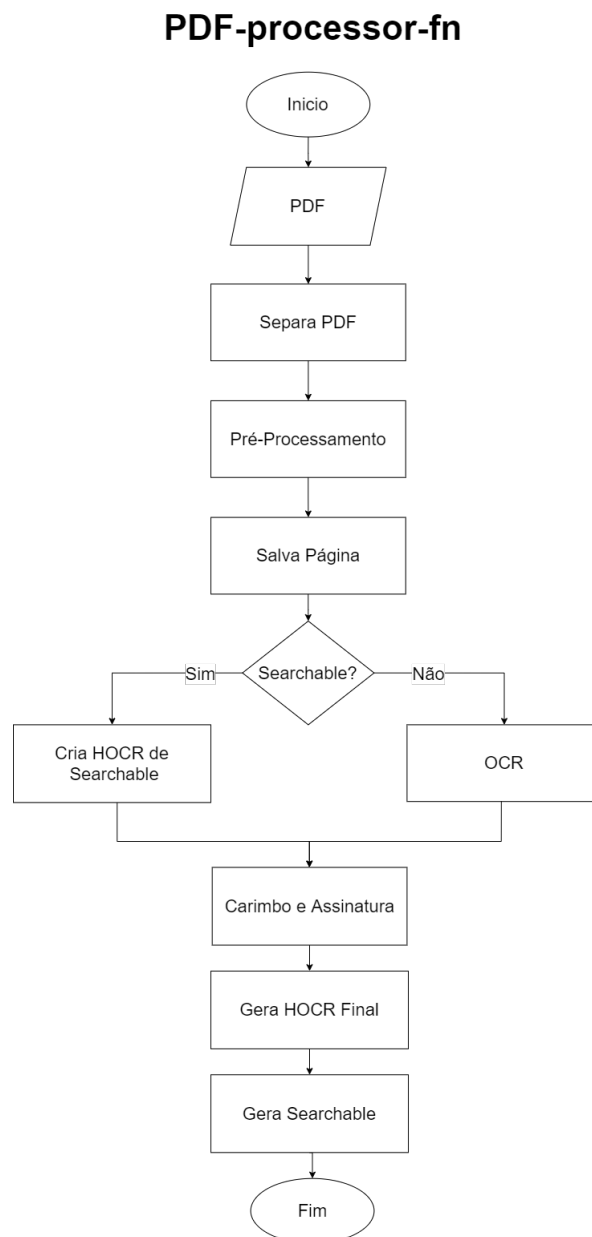
A *PDF-processor-fn* recebe como entrada um PDF escaneado, o qual irá passar pela etapa de separação do documento em páginas individuais. Em seguida, cada página, separadamente, é submetida a várias etapas de pré-processamento, devido a que os documentos podem ser de baixa qualidade gráfica. Ou seja, possuem inclinação derivada do processo de escaneamento e muitas das vezes outros tipos de ruídos diversos (manchas, riscos, defeitos de impressão, etc). Estas técnicas são importantes pois contribuem para um melhor desempenho das próximas etapas às quais o documento será submetido.

As principais técnicas de pré-processamento utilizadas através da ferramenta *OpenCV* são:

- Correção de PPI.
- Correção de inclinação (*Skew detection and correction*);
- Remoção de ruído.
- Binarização.

As páginas são então salvas em um arquivo temporário até o final do processamento da função, de modo a ficarem disponíveis para os demais serviços do sistema.

Figura 11: Fluxograma de funcionamento da função *PDF-processor-fn*.



Fonte: Elaborado pelo autor.

Após, é realizada uma verificação, a fim de identificar se a página já contém o texto embutido ou não. Quando negativo é aplicado o processo de OCR pela ferramenta *Tesseract OCR*, gerando o HOOCR da página, um padrão de representação de informação do OCR que contém informações espaciais e geométricas das secções de texto, métricas de confiança de reconhecimento, assim como informações de metadados como família e tamanho da fonte, descrito através de XML. Caso contrário, é aplicada uma função para extrair a camada de texto do documento *searchable*, sendo formatado também como HOOCR.

A seguir, é realizada uma verificação quanto a presença de carimbos e assinaturas, desenvolvida pelo Projeto ATT. Essa informação então é adicionada ao HOOCR referente a

página que está sendo processada, a qual será então agrupada, formando um único HOCR com todas as páginas do documento submetido como entrada.

É gerado então o documento *searchable* final com o PDF de entrada e o HOCR final, cujas páginas individuais passaram por correções durante o pré-processamento. Para que a camada *searchable* seja adicionada ao PDF original, de modo que a mesma sobreponha a imagem de forma alinhada, é realizada uma correção através de homografia inversa, reposicionando o documento conforme a inclinação e posição geométrica original.

Ao longo do *pipeline* a função entrega como saída o HOCR completo, o PDF individual de cada página e o PDF *searchable* completo do documento, para utilização por outros serviços que compõem o Projeto ATT.

4.1.2 Formund-fn

A função chamada *formund-fn*, apresentada na Figura 12, é a responsável pelo processo de *form understanding*.

Ela admite como entrada o arquivo HOCR gerado pela função *PDF-processor-fn* e o nome da classe alvo referente ao documento a ser processado, podendo ser Classe 1 ou Classe 2 em formato de *string*. Então a função entrega como saída um arquivo JSON relativo às duas classes de documentos disponíveis para processamento.

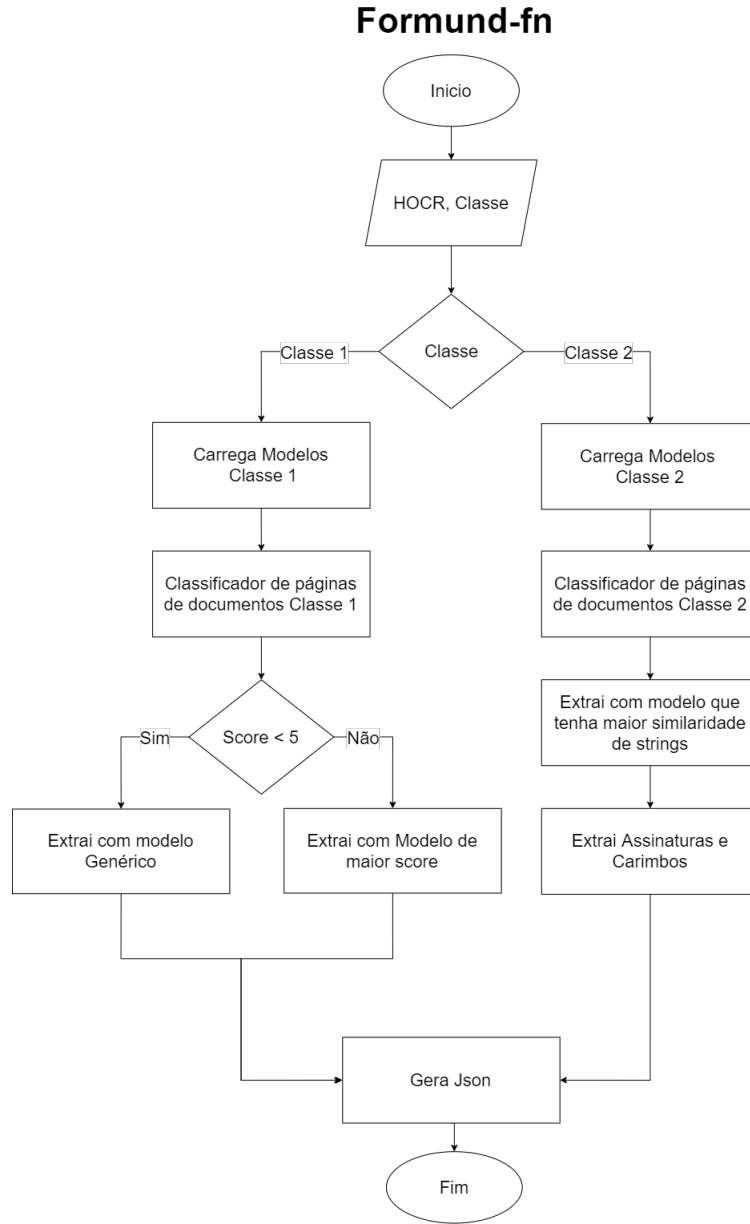
No fluxograma da Figura 12 estão representados os fluxos referentes as Classe 1 e Classe 2, sendo elas associadas a normas e documentos de construção e montagem, respectivamente. Em ambos os métodos, foram mapeados modelos descritivos da geometria, elaborados manualmente no formato JSON, relativos a cada possível tipo de página contida nos documentos de interesse.

Quando o documento é de Classe 1, existem quatro tipos de modelos de conteúdo e um modelo generalista (união de todos os modelos), com a finalidade de extrair o máximo de informações possível. Sendo então necessário aplicar um classificador de documentos e a seleção do modelo é baseada em uma abordagem probabilística, descrita no trabalho proposto por (Medvet et al. 2011), no qual é calculado um *score*.

O *score* é adquirido através do processo de verificação de modelos, nos quais todos eles são testados e aquele que obtiver o maior valor é considerado o modelo correto. Quando o valor encontrado for menor do que cinco, valor esse obtido através de testes empíricos, considera-se que o modelo genérico é o que melhor descreve a página do documento.

Para calcular o *score* são utilizadas hipóteses, que nada mais são do que um conjunto de até cinco palavras pertencentes a mesma linha. A Equação 4, mostra a fórmula de cálculo, onde t_i representa uma palavra pertencente a um documento com um total de T palavras.

Figura 12: Fluxograma de funcionamento da função *formund-fn*.



Fonte: Elaborado pelo autor.

$$hypo = \bigcup_{i=1}^T \{(t_i), (t_i, t_{i-1}), (t_i, t_{i-1}, t_{i-2}), (t_i, t_{i-1}, t_{i-2}, t_{i-3})\} \quad (4)$$

As hipóteses e suas *bounding boxes* (descrição geométrica da posição de uma palavra) são então convertidas em uma lista de *features* h_{feat} , composta por posição x , y , altura (h), largura (w) e texto da palavra (t), como descrito na Equação 5.

Esse conjunto de *features* serão utilizadas para o cálculo da probabilidade que representará o *score*.

$$h_{feat} = \langle x, y, w, h, t \rangle \quad (5)$$

Mais detalhes quanto as probabilidades utilizadas para calcular o *score* são mencionados na Seção 4.1.2.1.

No caso da Classe 2, existem doze modelos de documentos. Enquanto a Classe 1 utiliza uma abordagem probabilística baseada em *score*, neste caso é realizada uma comparação baseada em similaridade de *strings*, utilizando o título do documento ou palavras-chave como guia para a classificação. Para isso, adota-se expressões regulares visando encontrar as palavras de interesse, baseadas na sua localização espacial.

Esta classe de documentos possui assinaturas e carimbos. Portanto, neles é realizada uma verificação, através de redes neurais, quanto a existência destes itens, recuperando esta informação que foi gerada pela *PDF-processor-fn*. Sendo identificado a presença de carimbos e assinaturas é gerado um campo do tipo booleano indicando a presença ou ausência dos mesmos. Posteriormente será implementada a extração de informações dos dados dos carimbos e assinaturas, etapa está ainda em fase de pesquisa e desenvolvimento, em outros trabalhos associados ao Projeto ATT.

Ao final do processo de extração das informações é gerado um JSON que contém os dados textuais relacionados semanticamente. Este processo realiza a extração de entidades e elementos de tabelas, conforme abordado em detalhes nas Seções 4.1.2.1 e 4.1.2.2, de forma a finalizar o processo de *form understanding*

4.1.2.1 Extração de entidades

O método de extração de entidades é baseado no proposto por Jaume et al. (2019), onde segmentos de texto, entidades, são classificados como perguntas (*questions*), respostas (*answers*) e textos (*headers e others*).

O modelo de *layout* apresentado neste trabalho segue a seguinte organização: primeiro tem-se um objeto que é composto por um campo *string* chamado de *name* que indica o título do documento a ser modelado, e um *array* de objetos, denominado de *blocks*, objetos estes que por sua vez descrevem as entidades alvos a serem extraídas do documentos.

Cada entidade possui um *id*, um *type* que representa qual das três classificações o mesmo representa (pergunta, resposta e texto) e um campo chamado *links* o qual é responsável por associar a entidade do tipo resposta a sua pergunta correspondente (quando a entidade é uma pergunta este campo é deixado vazio).

Recentemente, implementou-se um método baseado em múltiplas âncoras, que utiliza várias referências para uma resposta. Desta forma tornando o método de extração mais robusto. Deveu-se isso aos casos em que havia a perda do campo de referência (âncora). Nessa situação perdia-se também a resposta associada. Assim, atualmente pode-se marcar mais de uma referência para campos importantes que necessitem de redundância para garantir uma extração correta. É importante mencionar que neste caso detalhes de implementação não foram aqui discutidos, pois este método está ainda em fase de testes.

Logo após, tem-se a descrição espacial da posição da entidade. Ou seja, suas coorde-

nadas no documento. Essa informação ao ser salva no formato JSON contém um campo chamado *type*, que indica qual o tipo daquela descrição, que pode ser:

- *Region*, indicando uma região a ser buscada.
- *Anchored_region*, que representa uma posição relativa a entidade associada, ou seja utiliza-se o campo associado como referencia para o sistema de coordenadas do bloco atual.
- *Derived* que representa um campo que deriva de outro. Por exemplo, para os campos páginas de documentos, que muitas vezes se encontram compostos de quatro entidades como: “Folha 1 de 3”, pode-se marcar como resposta do tipo *derived* as entidades “1 de 3” e descrever uma expressão regular que consiga extrair este padrão no campo adicional *text model*.

Por fim, tem-se o campo *text model*. Este campo assume valores diferentes de acordo com o tipo de entidade que está atrelado.

Em uma entidade do tipo *form_answer* podem-se utilizar os tipos:

- *Any*, quando não há informação prévia do formato esperado para a resposta.
- *Regex*, onde se pode descrever uma lista de expressões regulares.
- *Derived_regex*, utilizado quando o *bounding_box_model* da entidade seja configurado como *derived*.

Já quando a entidade é do tipo *form_question*, espera-se que seja passado qual o algoritmo que será usado para identificar o texto (referente a essa entidade no documento), como também um conjunto de variações do texto esperado para este campo.

Na Figura 13 é apresentado um trecho exemplo do JSON para descrição de modelos, para extrair o número de um documento. Na Figura se pode observar os campos citados até então, de forma que no primeiro elemento de *blocks* tem-se uma entidade do tipo *form_question*, no qual sua *bounding_box_model* é do tipo *region*. Por ser uma pergunta, possui dois campos para preencher em *text model*, sendo o primeiro referente ao algoritmo distância de edição de Levenshtein (Yujian & Bo 2007) usado para procurar pelo elemento. Já no campo adicional *texts* tem-se um conjunto de variações do texto esperado para este campo, sendo neste caso “Relatório”. No segundo elemento de *blocks* tem-se uma entidade do tipo *form_answer* que tem sua *bounding_box_model* do tipo *anchored_region* e, por ser uma pergunta em *text model*, tem-se apenas um campo, o qual foi preenchido por *any*.

Quanto a esse modelo descritivo, assim como foi aplicado na *formund-fn* e de acordo com (Santos et al. 2021), o que define se uma entidade pertence ou não ao *block* é a probabilidade p_h chamada de probabilidade de hipótese, descrita na Equação 6.

Figura 13: Trecho do modelo descritivo de um documento para extração do número do relatório.



Fonte: Elaborado pelo autor.

$$p_h = p_t * p_{bb} \quad (6)$$

Onde p_{bb} representa a probabilidade de *bounding box*, enquanto p_t representa a probabilidade associada ao texto encontrado na hipótese.

O cálculo da probabilidade de *bounding box* pode ser feito de maneiras diferentes. Quando um bloco é reconhecido como do tipo “*text*” sua p_{bb} é definida como 1. Para um bloco reconhecido como do tipo “*region*” é feita uma verificação quanto as coordenadas da hipótese h_{bb} com as coordenadas do bloco q_{bb} , utilizando uma distribuição uniforme, conforme mostra a Equação 7.

$$p_{bb} = \begin{cases} 1, & \text{if } h_{bb} \subset q_{bb} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

Para um bloco reconhecido como “*anchored-region*”, de forma análoga ao anterior, é feita uma verificação quanto as coordenadas da hipótese com as do bloco, conforme

mostra a Equação 8. Sendo as coordenadas do bloco dadas como $a_{bb} = anc_{bb} + q_{bb}$, onde anc_{bb} representa as coordenadas da *bounding box* da pergunta.

$$p_{bb} = \begin{cases} 1, & \text{if } h_{bb} \subset a_{bb} \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Por fim, o cálculo da probabilidade do texto identificado na hipótese é feito utilizando a distância de *Levenshtein*, de forma que dentre as possíveis palavras encontradas aquela que possui o maior valor de distância é a escolhida, conforme mostra a Equação 9.

$$p_t = 1 - \max(lev_distance) \quad (9)$$

4.1.2.2 Extração de Tabelas

Para realizar a extração de tabelas com semântica foi utilizado o método de descrição de *layouts* apresentado na Seção 4.1.2.1, com a diferença de que agora uma pergunta possui múltiplas respostas, de forma que se a mesma não for encontrada, os dados não são extraídos pois não teriam uma relação semântica atrelada. Como estes segmentos de texto tratam-se de códigos, sequências alfanuméricas, seriam usualmente considerados apenas textos esparsos, podendo assim gerar falsos positivos no sistema. Atualmente está sendo desenvolvido um método, chamado de múltiplas âncoras, com a finalidade de utilizar mais de uma referência associada ao campo de interesse, de forma que torne o processo de extrair tabelas mais robusto, visto que, conforme acima mencionado, quando se perde a entidade do tipo pergunta, no cabeçalho da tabela, perde-se todo o conteúdo da respectiva coluna.

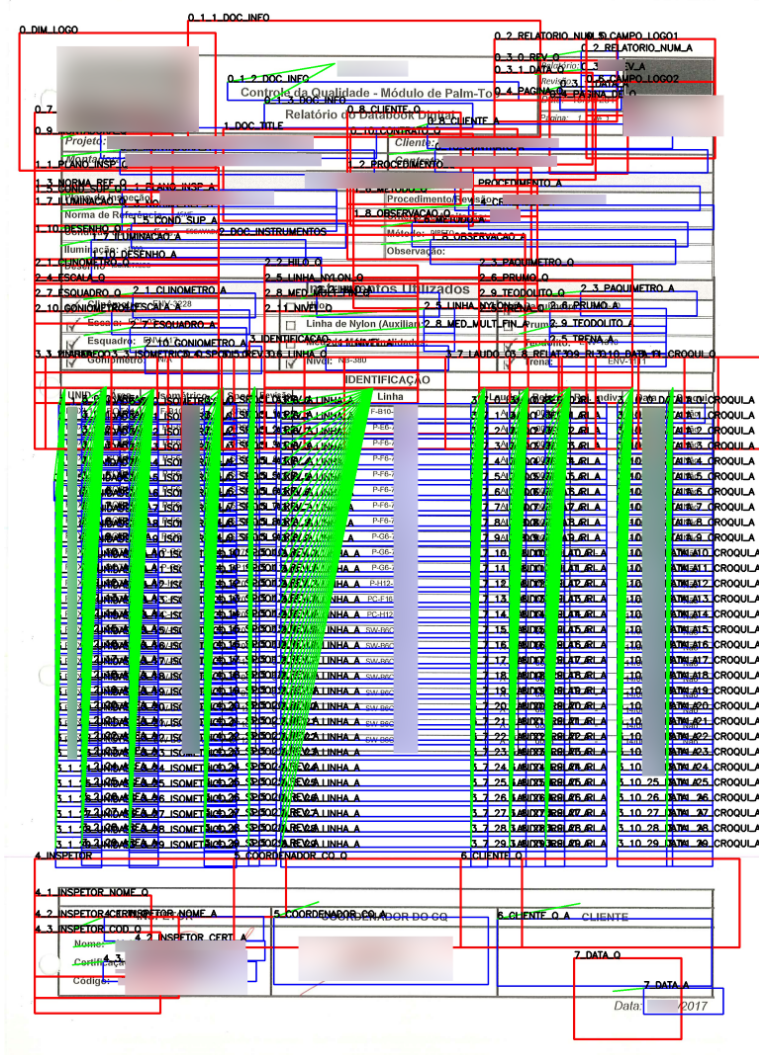
Na Figura 14, há a representação do modelo completo desenhado sobre um exemplar de página de documento de *databook*. Os dados sensíveis foram omitidos.

Já na Figura 15 tem-se a visão aproximada desses elementos, exemplificando a relação entre eles.

Em vermelho tem-se as entidades modeladas de perguntas e de texto, em azul tem-se as respostas e em verde são representadas as ligações de ancoragem (também chamados de links ou referências) entre entidades do tipo pergunta-resposta. No centro do documento encontra-se o método de descrição para extração de tabelas descrito anteriormente, onde há uma maior densidade de campos do tipo resposta (em azul) e ancoragens (em verde).

Cabe destacar que até o presente momento este modelo é descrito manualmente. Um método em desenvolvimento será apresentado na Seção 4.2, cujo objetivo é aprender o modelo baseado na anotação de campos de interesse, reduzindo assim a necessidade de descrever modelos de forma manual.

Figura 14: Modelo desenhado com os campos encontrados pelo algoritmo sobre um documento de exemplo.



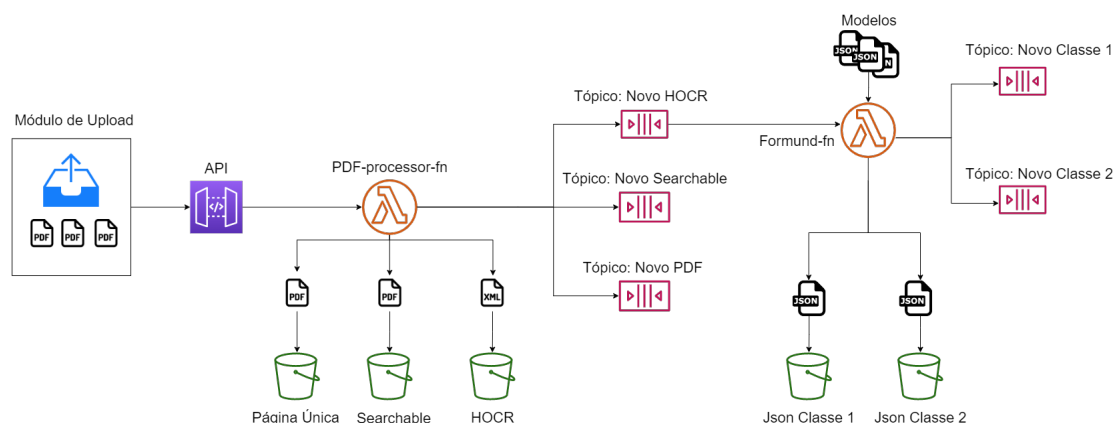
de processamento foi desenvolvido utilizando o conceito de função como serviço, mais conhecido como *function as a service (FaaS)*.

Levando também em conta tratar-se de projeto de pesquisa, desenvolvimento e inovação, a ser utilizado em ambiente de possível alta rotatividade de usuários, foi escolhido utilizar ferramentas com curvas de aprendizado relativamente baixas e amigáveis a novos usuários.

Assim, para o desenvolvimento das funções *PDF-processor-fn* e *formund-fn*, apresentadas nas Seções 4.1.1 e 4.1.2, foi escolhido o *framework OpenFaaS*, devido a sua versatilidade, documentação de fácil entendimento e possibilidade de admitir diversas linguagens de programação, de forma independente do resto do sistema, como *Python*, *Javascript*, *Go* e etc.

A Figura 16 apresenta a arquitetura básica de operação do módulo de processamento de documentos, com algumas abstrações de elementos do sistema devido a confidencialidade dos métodos utilizados no restante da arquitetura.

Figura 16: Arquitetura do módulo de processamento automatizado de documentos.



Fonte: Elaborado pelo autor.

De maneira geral, tem-se o módulo de *upload* no qual o usuário irá, através da interface, carregar os documentos que deseja que sejam processados pelo sistema. Após, tem-se uma API responsável por gerenciar estes arquivos e disponibilizá-los para a função *PDF-processor-fn*. Esta função processa os documentos em formato PDF, entregando três saídas: as páginas individuais pré-processadas, o HOOCR e o PDF original em formato *searchable* (contendo o texto embutido, o que possibilita realizar buscas por *strings*). Essas saídas são armazenadas em *buckets* (*containers* para objetos).

A cada etapa de escrita nos *buckets* são geradas mensagens no serviço de mensageria utilizado pela plataforma. Assim servindo como gatilho para a execução dos outros serviços.

Cada vez que uma mensagem é publicada no tópico referente ao HOOCR, a função *formund-fn* é disparada através de um orquestrador.

A *formund-fn* então utiliza modelos de descrição geométrica do documento para adicionar semântica ao texto esparsa gerado pelo OCR. Entregando então um JSON referente a uma das duas classes disponíveis no sistema e publicando a mensagem de gatilho ao tópico específico da classe.

4.2 Metodologia relativa a documentos semi e não estruturados

Como apresentado na Figura 10, para os documentos semiestruturados ou não estruturados foi modelado e proposto o desenvolvimento de um *pipeline* especial, baseado em métodos de aprendizado de máquina profundo, composto de diversas etapas e técnicas recentemente publicadas na literatura.

A abordagem tem como principal característica separar o processamento em quatro tarefas, aplicar redes neurais específicas no processamento de cada uma e, por fim, utilizar o conjunto de saídas obtidos para construir o *document understanding* completo da página. Ou seja, ao lidar de maneira separada com cada um dos quatro elementos principais que compõem uma página do *databook* torna-se possível aplicar modelos que sejam especializados para cada um deles e que representem os melhores resultados no estado da arte para cada um.

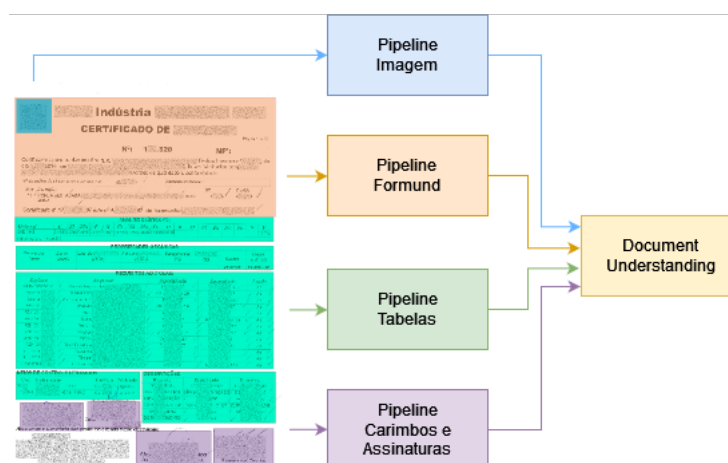
Esta abordagem proposta, de separação do documentos em multitarefas, tem como base os estudos realizados na literatura, condizendo a cada elemento específico. Foram, pois, utilizados os seguintes trabalhos como referência: para imagens (Soto & Yoo 2019) e (Rani et al. 2016), para assinaturas (Zhu et al. 2008), (Ahmed et al. 2012) e (Peng et al. 2009), para carimbos (Zhu et al. 2006), (Micenkov & van Beusekom 2011) e (Forczmański 2010), enquanto para tabelas (Prasad et al. 2020), (Paliwal et al. 2019) e (Shafait & Smith 2010).

Vale ressaltar, que a tarefa referente ao *pipeline* de *formund* é o foco principal deste trabalho, enquanto as outras três (carimbos e assinaturas, imagens e tabelas) são abordados em trabalhos paralelos, também associados ao Projeto ATT.

A Figura 17 mostra como é realizada a divisão dos elementos que compõem a página, evidenciando cada *pipeline* específico, de acordo com o esquema de cores: azul para imagens, laranja para *formund* (formulários), verde para tabelas e roxo para ambos carimbos e assinaturas.

O processo proposto inicia-se através da submissão de um *databook* completo a um processo chamado de segmentador, o qual compõe-se por uma sequência de passos. Primeiramente, é aplicada a *pdf-processor* que, de forma análoga a função implementada para documentos estruturados, realiza a separação do documento página por página, aplica pré-processamento em cada uma delas, realiza o processo de OCR e então obtém como saída tanto a imagem tratada como o HOCR.

Figura 17: Visão geral na qual cada página é dividida em quatro desafios principais.



Fonte: Elaborado pelo autor.

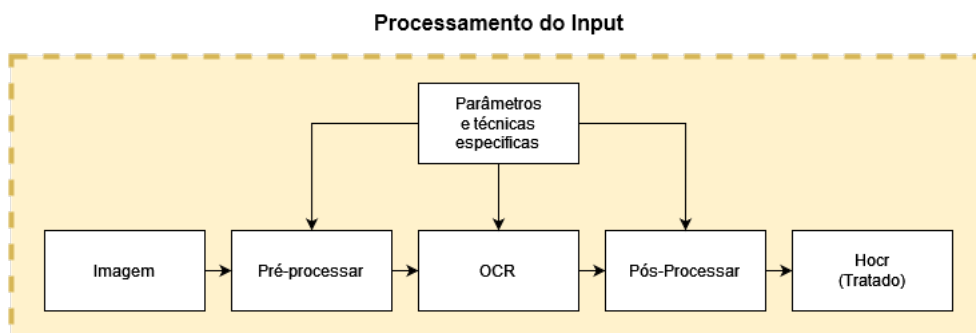
Ambas saídas são submetidas a um segmentador estrutural que nada mais é do que um conjunto de redes neurais para identificar os elementos contidos no documento, em busca de detectar a presença de imagens, carimbos/assinaturas e tabelas. Sendo o que restar da página considerado formulário correspondente ao *pipeline* de *formund*. Deve-se considerar que formulários normalmente não possuem uma representação visual bem definida, como é o caso de tabelas que normalmente possuem linhas que determinam o seu formato, ou carimbos e assinaturas, os quais podem utilizar técnicas de detecção de imagens em documentos para os quais existem diversas soluções na literatura.

Após realizar a detecção destes elementos, as suas *bounding boxes* são identificadas e então criadas *tags* específicas para cada um, as quais serão adicionadas no arquivo HOCR da página. Esse arquivo é usado como entrada para os quatro *pipelines* subsequentes, representados nas Seções 4.2.1, 4.2.2, 4.2.3 e 4.2.4.

Cada *pipeline* apresenta em comum dois blocos: Identificar/Recortar e Processamento de *Input*.

O bloco de identificar e recortar deve percorrer o HOCR, procurar por sua *tag* específica e realizar a subtração de todas as imagens referentes aos elementos que sejam detectados com a mesma. Já o bloco chamado de processamento do *input*, ilustrado na Figura 18, tem como responsabilidade implementar ajustes e configurações específicas para a determinada tarefa a qual está sendo submetido. A entrada desse bloco se dá através das imagens subtraídas no processo anterior, as quais são pré-processadas com as técnicas e parâmetros específicos do *pipeline*, assim como também são novamente submetidas a ferramenta de OCR. Esta ferramenta, por sua vez, também é ajustada para os parâmetros mais adequados, o que resulta em outro documento HOCR o qual, visando corrigir eventuais erros de OCR, é pós-processado utilizando-se de um conjunto de métodos específicos. Ao final, tem-se como saída um HOCR pós-processado que servirá de entrada para as próximas tarefas dos *pipelines* específicos.

Figura 18: Visão geral do *pipeline* de processamento do *input*.

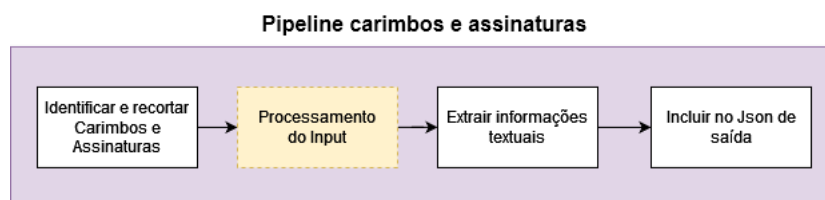


Fonte: Elaborado pelo autor.

4.2.1 Pipeline de carimbos e assinaturas

Na etapa de carimbos e assinaturas, Figura 19, inicia-se pela identificação e extração dessas entidades a partir de redes neurais específicas. Uma vez com essas informações recortadas são aplicadas técnicas de ajustes de orientação, rotação, reconstrução do texto e etc, para que possa ser aplicado um método de *form understanding* próprio para carimbos e assinaturas, o qual irá retornar o texto contido no carimbo e o possível *match* entre a assinatura encontrada e a do técnico responsável por assinar o documento.

Figura 19: Visão geral do *pipeline* para carimbos e assinaturas proposto.

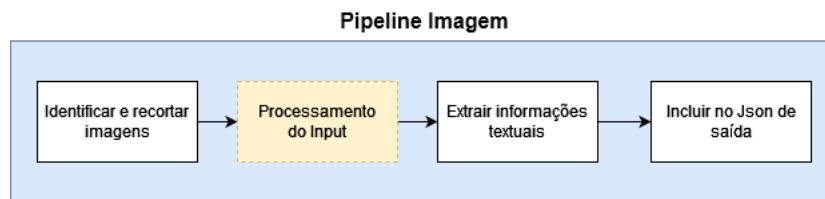


Fonte: Elaborado pelo autor.

4.2.2 Pipeline de imagens

Conforme a Figura 20, é representado em azul o *pipeline* responsável por tratar e extrair informações de imagens presentes em cada página do *databook*. Ali tem-se os dois blocos já citados anteriormente e que são responsáveis por entregar (no formato de imagem e HOCR) as informações que irão servir como entrada para o próximo bloco, que objetiva buscar relações semânticas entre as entidades extraídas (através do processo de extração de informações), buscando relacionar as palavras encontradas caso haja alguma relação, podendo também classificar a entidade encontrada em alguma classe pré-definida. Por fim, o resultado é incluído no arquivo de extração final, no formato JSON, e usado para compor o *document understanding* completo do *databook*.

Figura 20: Visão geral do *pipeline* para imagens proposto.



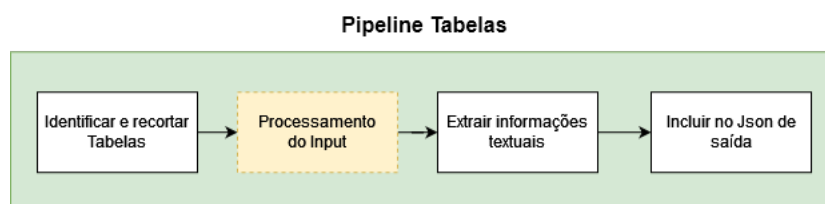
Fonte: Elaborado pelo autor.

4.2.3 Pipeline tabelas

Para o *pipeline* de extração de tabelas, representado na cor verde (Figura 21), tem-se novamente os dois blocos iniciais comuns a todos os *pipelines* específicos, seguido pelo módulo responsável por implementar as técnicas para extração de tabelas, usando como referência o trabalho desenvolvido em Paliwal et al. (2019). Foram realizadas duas séries de experimentos, utilizando os pesos da rede neural convolucional de Mebsout (2021), a qual se utiliza da tecnologia YOLOv3 (análoga a utilizada pelo *pipeline* de carimbos e assinaturas), pré-treinada por um conjunto de tabelas recortadas dos mais diversos tipos de documentos.

A primeira série de experimentos foi realizada com 19 páginas de documentos reais, enquanto a segunda utilizou a estratégia de *data augmentation* para aumentar a quantidade de amostras. Para tal foram aplicados métodos como *noisy*, *occlusion*, *rotation* e *scale*, gerando ao total um adicional de 136 páginas para serem aplicadas no processo de treinamento do modelo. Com as tabelas encontradas corretamente, os resultados, assim como nos outros *pipelines*, são representados no formato JSON. Os resultados referentes aos experimentos relativos as tabelas serão apresentados no Capítulo 5.

Figura 21: Visão geral do *pipeline* para tabelas proposto.



Fonte: Elaborado pelo autor.

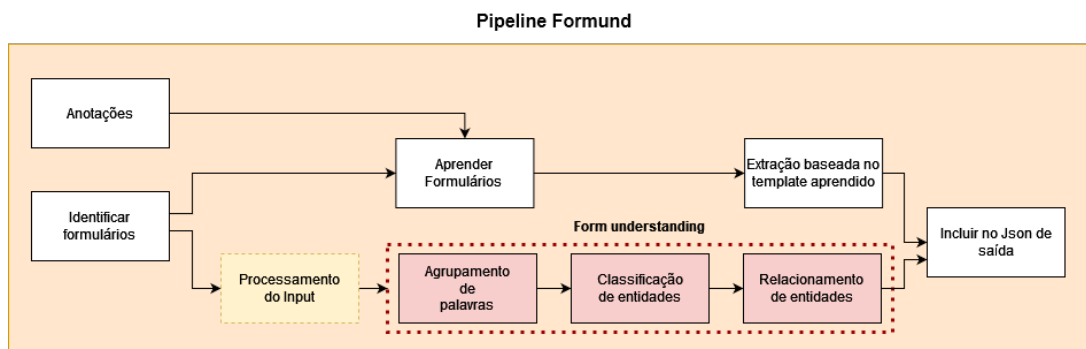
4.2.4 Pipeline de formund

No *pipeline* para *form understanding* proposto tem-se como primeira atividade a identificação do formulário que se dá de acordo com a abordagem descrita anteriormente: remove-se do arquivo HOCR todos os elementos que não sejam imagens, tabelas, carimbos e assinaturas, o conteúdo que restar assume-se que seja do tipo formulário. Essas informações são então resgatas e submetidas ao processamento de *input*, de forma análoga

aos outros *pipelines* específicos descritos anteriormente.

Na Figura 22 pode-se observar dois processos em paralelo propostos para lidar com os formulários de documentos semi e não estruturados.

Figura 22: Visão geral do *pipeline* para formulários proposto.



Fonte: Elaborado pelo autor.

Desses dois *pipelines*, o primeiro utiliza-se do conceito de aprender formulários através de anotações dos campos de interesse no documento levantadas pelo profissional responsável por realizar a análise dos documentos, cujo o qual possui *know-how* para determinar quais campos são necessários, logo trazendo um alto grau de confiabilidade quanto o que espera-se extrair. A partir deste processo tem-se um possível *template*, composto de *bounding boxes* baseadas nas médias e desvio padrão dos valores descritos.

De posse do *template* gerado baseado no aprendizado dos formulários, realiza-se a extração dos campos com informação semântica, através do processo de *form understanding* para documentos estruturados e então é feita a inserção desta informação no arquivo JSON de saída.

Por outro lado, o outro caminho do *pipeline* tem-se a abordagem baseada em aprendizado de máquina, sendo esta uma abordagem mais genérica e robusta. Nesta abordagem inicialmente submete-se a informação correspondente a formulários pela etapa de pré-processamento, seguida pela aplicação da ferramenta de OCR, o pós-processamento e logo a seguir é submetida a um conjunto de técnicas de *form understanding*, representado pela área tracejada na Figura 22, a qual é expandida na Figura 23.

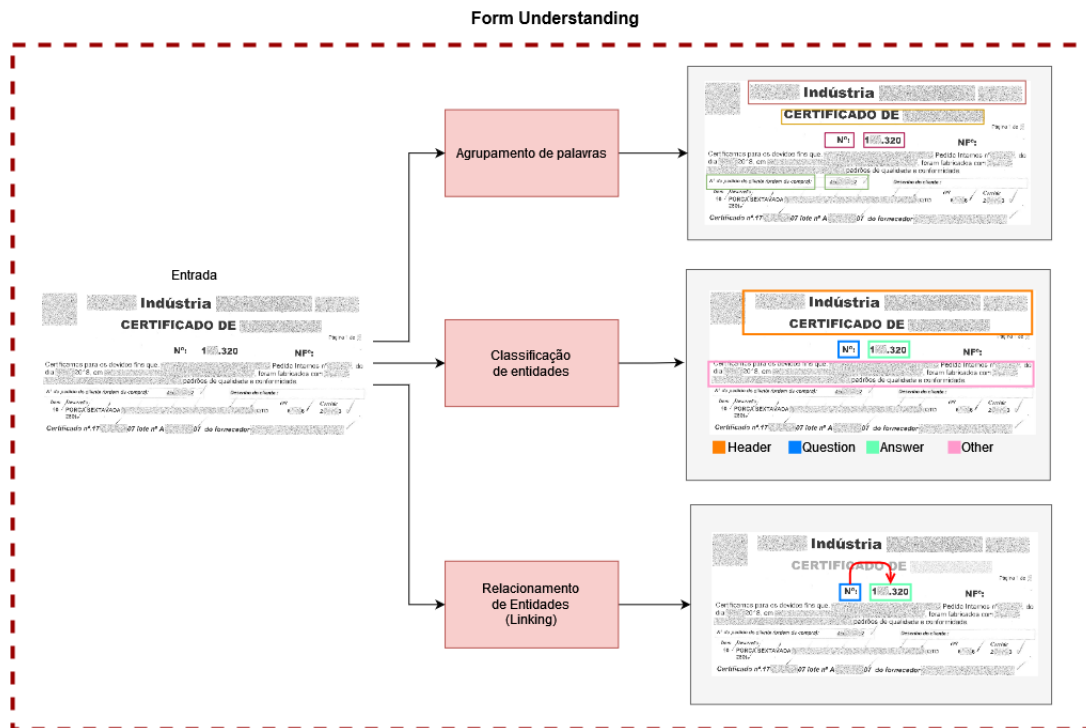
Este módulo, representado na Figura 23, é responsável por realizar três principais atividades para *form understanding* segundo Jaume et al. (2019). Sendo a primeira delas o agrupamento semântico de palavras (*Word Grouping*) onde palavras do mesmo grupo semântico estão contornadas, formando as células semânticas. Seguida por classificação de entidades (*Semantic Entity Labeling*), a qual é responsável por classificar as entidades em classes pré-definidas (conforme as cores dos contornos na Figura 23). E por fim o relacionamento de entidades (*Entity Linking*), onde na Figura 23 a entidade *Question* está contornada em azul, a entidade *Answer* em verde e o *link* entre as duas em vermelho.

De forma análoga ao outro caminho, por fim tem-se a saída do processo sendo arma-

zenada em formato JSON.

Vale ressaltar que *semantic entity labeling* é a principal contribuição do trabalho aqui proposto, portanto foram realizados experimentos quanto a mesma, os quais serão abordados no Capítulo 5. Assim como foi desenvolvido um *dataset* próprio, o qual é abordado na Seção 4.3.

Figura 23: Visão geral das etapas de *form understanding*, divididas em agrupamento de palavras, classificação de entidade e relacionamento das mesmas.



Fonte: Elaborado pelo autor.

4.2.5 Implementação no Projeto ATT

Cada *pipeline* apresentado representa um microserviço que ocorre em paralelo com os outros no sistema do Projeto ATT, o qual possui uma arquitetura própria e dependências específicas para cada técnica a ser empregada. Desta forma há uma separação de responsabilidades e independência por parte dos desenvolvedores responsáveis, característica crucial para o desenvolvimento do Projeto ATT.

A arquitetura em microserviços, aliada com a abordagem proposta de *document understanding* dividida por partes possibilita que o sistema consiga processar uma quantidade maior de informação, sendo altamente escalável e robusto a falhas, tanto pela redundância quanto pelo desacoplamento dos microserviços, uma vez que os mesmos se comunicam por API's ou serviços de mensageria.

Desta forma ao fim do processo tem-se um JSON de saída de forma análoga ao método estruturado apresentado anteriormente neste trabalho. Representando assim o processo de *document understanding* completo, ou seja, obtendo um documento de descrição que

conecta os textos com a semântica atribuída ao mesmo bem como todas as relações necessárias para o devido entendimento do documento, extração de informação e processamento automatizado das mesmas.

4.3 Classificação de entidades e criação do *dataset*.

A metodologia utilizada para o problema de *form understanding* de documentos não estruturados, como dito anteriormente é dado em três grandes subtarefas, dentre as quais *Semantic Entity Labeling* (SEL), também chamada de Classificação de entidades, é o foco do trabalho aqui proposto.

Para tal utilizou-se do modelo pré-treinado *LayoutLM* (Xu et al. 2020) (já mencionado no Capítulo 3), no qual foi feito um *fine-tuning* (ajuste fino) com o objetivo de auxiliar o modelo quanto a tarefa de reconhecer os padrões presentes nos *databooks* utilizados. Esta etapa se faz necessária, pois o modelo *LayoutLM* é pre-treinado com 6 milhões de documentos e esta vasta gama de variedade de *templates* e *layouts* pode acabar prejudicando os resultados do modelo nos documentos do Projeto ATT, tendo em vista que em sua maioria os documentos originalmente utilizados para o treino deste modelo são provenientes de países estrangeiros os quais possuem ordens hierárquicas de leitura diferentes das utilizadas no Brasil. Portanto este *fine-tuning* é feito de forma a dar um viés sutil aos padrões reconhecidos e aprendidos pela rede, logo se adaptando aos tipos de documentos que constituem os *databooks*. Os primeiro experimentos com o modelo, o *fine-tuning* foi feito utilizando dos documentos disponibilizados no *dataset* criado por (Jaume et al. 2019), contudo se viu a necessidade de criar um *dataset* próprio com documentos mais próximos da realidade observada nos *databooks*.

Para que o modelo consiga realizar a classificação das entidades entre “*Answer*”, “*Question*” e “*Text*”, é preciso que as informações do *layout* das páginas sejam disponíveis de maneira descritiva. Uma vez que o método aqui proposto para documentos estruturados (descrito na Seção 4.1) obteve resultados satisfatórios, foi possível então utilizar os JSON’s de saída obtidos por ele e modificá-lo no formato exigido pelo modelo.

Haviam diversas modificações necessárias referentes a ajustes nas *bounding boxes* das entidades, logo foi feito um *script* em *Python* que fosse capaz de realizar as mudanças em larga escala. A primeira modificação necessária relaciona-se com o fato de as entidades do tipo “*answer*” serem originalmente armazenadas com suas coordenadas referenciadas a pergunta que a mesma está atrelada, portanto não utiliza das referenciais globais da página. Essa configuração não é aceita pelo modelo, logo todas as entidades do tipo pergunta tiveram suas coordenadas ajustadas para a referência global de forma que cada *bounding box* segue o formato (x_0, y_0, x_1, y_1) , onde (x_0, y_0) corresponde às posições do canto superior esquerdo e as coordenadas (x_1, y_1) representam a posição do canto inferior direito. Após foi feita uma normalização dos valores para que os mesmos fiquem

entre uma escala de 0 – 1000.

Com a normalização das *bounding boxes* é então criado um *dataframe* no qual possui as colunas referentes ao nome do arquivo (que condiz com o valor individual da página), a *bounding box*, a coluna com a entidade textual encontrada (ex: “relatório de ultrassom”) e qual a *label* dessa entidade (“*answer*”, “*question*” ou “*text*”).

Foi feito então uma separação da entidade textual em palavras unitárias (formato aceito pelo modelo), para isso primeiramente foram filtradas palavras indesejadas, como pro exemplos palavras com apenas 1 caractere. Após foi necessário fazer a aferição das *bounding boxes* dessas palavras separadas, foram utilizadas a quantidade de caracteres encontrada na linha inteira a que a palavra pertence ($line_{len}$), a quantidade de caracteres na própria palavra ($word_{len}$), a suposição que cada caractere ocupada aproximadamente um valor de 5pixel e a largura da bbox da entidade original ($bbox_{width}$).

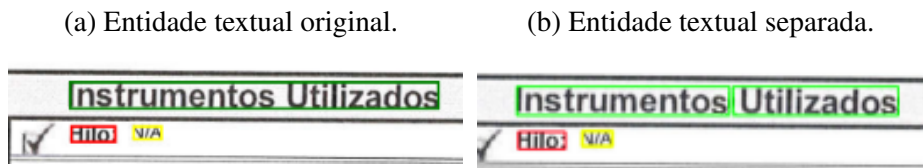
As Equações 10 e 11 demonstram as medidas x_0 e x_1 da nova palavra, é importante ressaltar que as medidas de altura (y_0 e y_1) foram mantidas.

$$x_1 = x_0 + \left(\frac{bbox_{width} * word_{len}}{line_{len}} \right) \quad (10)$$

$$x_0 = x_1 + 5 \quad (11)$$

A Figura 24a demonstra como exemplo a entidade “Instrumentos Utilizados” da maneira original como foi extraída pelo processo de *form understanding* para documentos estruturados enquanto a Figura 24b demonstra a mesma entidade após ter passado pelo processo de separação em palavras unitárias.

Figura 24: Exemplo de separação de entidade textual em palavras individuais.



Fonte: Elaborado pelo autor.

As *labels*, classes, são então passadas para uma extensão do sistema de *tags IOB* chamado de sistema *IOBES* (Cho et al. 2013), onde é acrescentado o prefixo “S” como um *singleton*, ou seja um *token* unitário. As *labels* de “*anwser*” e “*question*” receberam esse prefixo enquanto as *labels* de “*Text*” foram trocadas pelo prefixo “O” apenas.

Para a criação do *dataset*, ao total foram escolhidos 10 tipos de documentos, nos quais 7 possuem formato retrato e 3 são no formato de paisagem. Para cada tipo de documento foram separadas 50 páginas individuais, cada uma no formato *JPEG*, com as mesmas dimensões, DPI e podendo estar tanto em escalas de cinza quanto coloridas.

Ao total o *dataset* é constituído de 500 imagens e um arquivo JSON o qual contém

as informações descritivas do *layout* das páginas, de forma análoga ao arquivo desenvolvido para os documentos estruturados. As imagens foram mantidas, pois existem versões mais atuais do modelo que levam em consideração a informação visual proveniente das imagens puras para auxiliar na classificação. Logo, abrindo possibilidades para aperfeiçoamentos e novas abordagens.

A Figura 25 mostra uma tabela com as informações utilizadas para a descrição do *layout* e das entidades contidas nas páginas que compõem o *dataset*.

Figura 25: Versão final do arquivo de descrição das páginas do *dataset*.

index	filename	x0	y0	x2	y2 ▲	line	label
15	35	197	684	278	710	NORMA	S-QUESTION
16	35	283	684	315	710	DE	S-QUESTION
17	35	320	684	499	710	REFERÊNCIA:	S-QUESTION
93	35	595	743	702	761	ESCAVADA	S-ANSWER
21	35	195	740	321	771	CONDIÇÃO	S-QUESTION
22	35	326	740	357	771	DA	S-QUESTION
23	35	362	740	536	771	SUPERFÍCIE:	S-QUESTION
95	35	389	799	450	816	>1000	S-ANSWER
25	35	194	796	369	826	ILUMINAÇÃO:	S-QUESTION
96	35	349	852	476	871	ISOMETRICO	S-ANSWER

Fonte: Elaborado pelo autor.

Para a implementação no modelo, o *dataset* foi dividido em dois conjuntos, sendo eles treino e teste, os quais obedecem a proporção de 80/20, respectivamente.

Para cada um dos conjuntos foram criados três arquivos. O primeiro contém em cada linha uma entidade textual individual e sua *label*, no segundo arquivo consta a entidade textual individual e sua respectiva *bounding box* normalizada, no terceiro documento se tem novamente a entidade textual, porém dessa vez com a sua *bounding box* original seguida pela altura, largura e nome da página a qual pertence.

Por fim é feito um arquivo chamado *labels* o qual consta com o gabarito de todas as possíveis *labels* encontradas no *dataset*, o que no caso do *dataset* aqui desenvolvido são: *S-ANSWER*, *S-QUESTION* e *O*.

Todos esses documentos são então entregues ao modelo o qual inicia o processo de treino com 2 épocas de treinamento. O mesmo é procedido pelo processo de teste o qual entrega um arquivo com as previsões e as métricas de *f1-score*, *loss*, *precision* e *recall* como métodos de avaliação do modelo.

5 RESULTADOS

Neste capítulo são apresentados os resultados obtidos tanto para documentos estruturados quanto os resultados dos experimentos realizados aos documentos semi e não estruturados.

5.1 Resultados documentos estruturados

Durante o desenvolvimento do Projeto ATT foi realizada, como parte do processo preliminar de *document understanding* em documentos estruturados, a extração de informações utilizando modelos descritivos, focando, mais especificamente, a parte de *form understanding*.

Para isso, foram mapeados um conjunto de documentos considerados padrão, assim caracterizando o *layout* dos possíveis modelos.

Cada novo documento submetido ao sistema teve suas páginas separadas e pré-processadas de forma individual. O texto extraído de cada uma foi então submetido a uma verificação de hipóteses as quais foram utilizadas para definir qual o modelo mais adequado para aquela página. Com o modelo definido, foi então realizado o processo de *form understanding*, entregando como saída um JSON com as informações adquiridas e o mapeamento das mesmas.

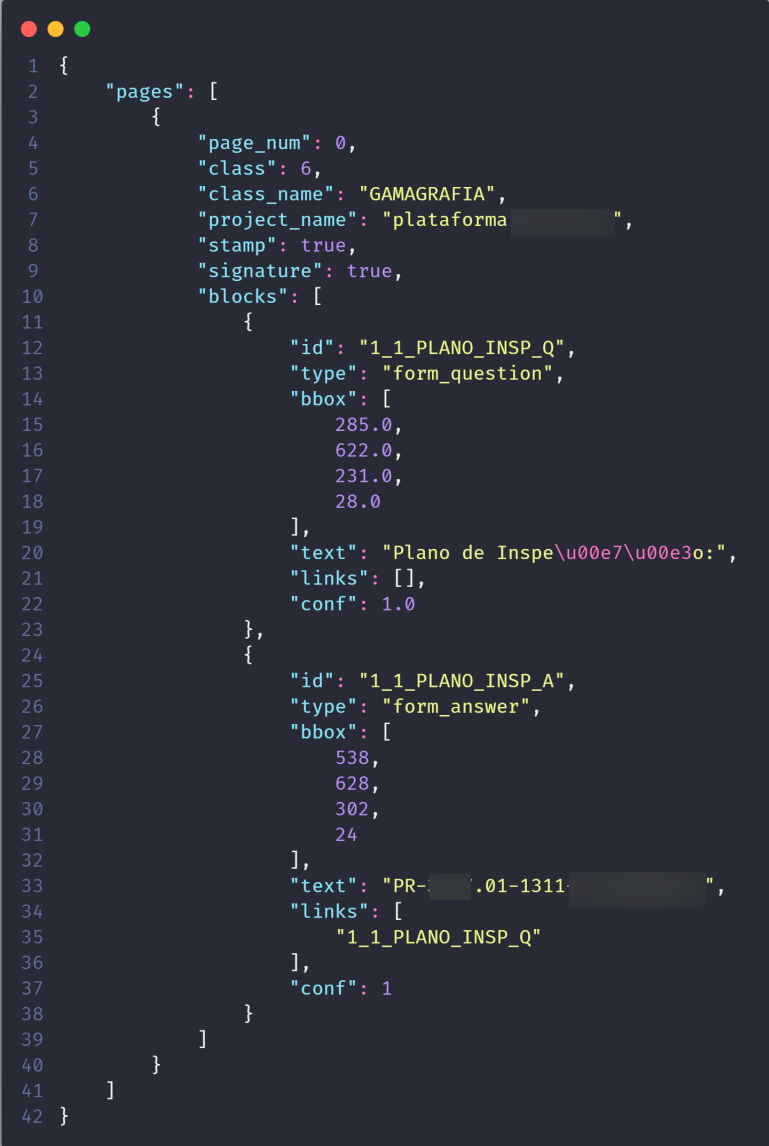
Na Figura 26 é apresentado o JSON de saída, o qual consiste de uma lista de objetos pertencentes a cada uma das páginas individuais. Dentro de cada objeto, há todas as informações necessárias para reconstrução do mapeamento geométrico e semântico do documento, como por exemplo número da página, classe do documento identificada, nome da classe, nome do projeto ao que o documento é pertencente, presença de carimbos e assinaturas, posição geométrica dos objetos e etc.

Ainda na Figura 26, pode-se perceber que, após a descrição dos blocos, há um exemplo do tipo de entidade pergunta com o id único de “1_1_Plano_INSP_Q”. A mesma é associada ao seu respectivo tipo, a *bounding box* descrevendo a posição espacial no documento, o texto extraído e um índice de confiança.

Logo em seguida é apresentado um exemplo de entidade resposta, a qual foi extraída

utilizando-se do conceito de ancoragem. Neste caso, a resposta encontrada foi ancorada a pergunta de id “1_1_Plano_INSP_Q”. Sendo, de forma análoga, associada a um id próprio, um tipo, a informação espacial, o texto extraído, um índice de confiança e a informação adicional de *link*, o qual é responsável por estabelecer a relação semântica das duas entidades.

Figura 26: Exemplo de JSON de saída do algoritmo de *form understanding*.



```

1 {
2   "pages": [
3     {
4       "page_num": 0,
5       "class": 6,
6       "class_name": "GAMAGRAFIA",
7       "project_name": "plataforma ",
8       "stamp": true,
9       "signature": true,
10      "blocks": [
11        {
12          "id": "1_1_PLANO_INSP_Q",
13          "type": "form_question",
14          "bbox": [
15            285.0,
16            622.0,
17            231.0,
18            28.0
19          ],
20          "text": "Plano de Inspe\u00e7\u00e3o:",
21          "links": [],
22          "conf": 1.0
23        },
24        {
25          "id": "1_1_PLANO_INSP_A",
26          "type": "form_answer",
27          "bbox": [
28            538,
29            628,
30            302,
31            24
32          ],
33          "text": "PR- .01-1311 ",
34          "links": [
35            "1_1_PLANO_INSP_Q"
36          ],
37          "conf": 1
38        }
39      ]
40    }
41  ]
42 }

```

Fonte: Elaborado pelo autor.

Na Figura 27 é apresentado o resultado da extração de informação, indicando em azul as entidades encontradas. Deve-se levar em consideração que o pleno sucesso deste método é diretamente associado ao sucesso da ferramenta de OCR, pois quando o mesmo falha em identificar alguma palavra que seja caracterizada como pergunta, pelo método de ancoragem não será possível recuperar a informação referente a resposta, independente

da mesma ter sido identificada corretamente pelo OCR. A Figura também apresenta o resultado da extração de tabelas que, mesmo com a técnica de ancorar respostas a mais de uma pergunta, ainda assim acabou por perder algumas informações por problemas relativos ao OCR.

Figura 27: Exemplo desenhado de um resultado do processo de extração. Em vermelho: a label das entidade. Em azul: a entidade encontrada.

The screenshot displays a complex form titled 'Módulo de Palm-Top' and 'Relatório do Databook Digital'. It contains numerous input fields, checkboxes, and tables. Red text highlights entity labels, while blue text shows the entities found. The bottom section features a table with three columns: 'INSPECTOR', 'COORDENADOR DO Q', and 'CLIENTE Q'. The table is currently empty, with only the headers visible.

Fonte: Elaborado pelo autor.

Para avaliar os resultados obtidos pelo método foi desenvolvido um algoritmo responsável pela validação dos resultados, o qual percorre o JSON de saída do sistema e realiza uma comparação entre a contagem de quantos campos foram identificados pelo sistema de extração de informação e quantos elementos eram esperados de fato naquele documento. Para isto, as entidades de interesse foram separadas em três classes, *text* (texto), *questions* (perguntas), e *answers* (respostas). É importante ressaltar que foram

desconsiderados os erros cometidos pela ferramenta de OCR, de forma que subtraiu-se todas as palavras que não foram encontradas por não existirem ou terem um valor de distância de edição maior ou igual a própria palavra, o que descaracteriza completamente a mesma.

Uma vez que os documentos contidos no *dataset* utilizados possuem tabelas com número de linhas variáveis, foi feita uma média de linhas de tabela por página específica para cada modelo de documento.

O método leva em média 13.5 segundos para identificar e descrever o *layout* de uma única página. Já o processo de descrição manual por profissionais demora em torno de 4 horas para cada modelo novo.

Todo processo de *form understanding*, contando as saídas de JSON, HOCR, PDF searchable e imagens dos modelos desenhados gerou aproximadamente 40Gb ao total e totalizaram mais de 21 horas de experimentos.

A Tabela 2 apresenta os resultados referentes aos documentos de Classe 1. Percebem-se os resultados elevados encontrados para documentos do tipo A3. Isso se deve ao fato de que haviam poucas amostras do mesmo no *dataset*, cerca de menos de 5% do total. Em experimentos futuros espera-se aumentar a quantidade de amostras para então reavaliar a efetividade do método em documentos do tipo A3.

Para os outros dois tipos de documentos de Classe 1 foram obtidos resultados acima de 85%. Sendo que para os do tipo A1 foram obtidos os melhores resultados.

Não há resultados para “*undefined*”, pois o sistema não encontrou nenhum documento no qual extraiu uma quantidade inferior a n blocos, condição essa que foi definida como o gatilho para que um documento fosse atribuído com esta categoria, a qual significa que o sistema não conseguiu encontrar um modelo para a página. O número específico n foi definido de forma empírica, após experimentos realizados no *dataset* durante a fase de construção dos *templates* dos modelos.

Tabela 2: Resultados da métrica de acurácia para a extração para documentos de classe 1.

Tipo de documento	<i>Text</i>	<i>Questions</i>	<i>Answers</i>
A1	0.98	0.97	0.95
A2	0.97	0.85	0.85
A3	1.0	1.0	1.0

Como demonstra a Tabela 3 os resultados preliminares do método de *form understanding* para Classe 2 apresentaram valores acima de 87% para entidades do tipo *text*, acima de 91% para *questions* e 78% para tipo *answers*. O decréscimo no resultado das entidades *text* para essa classe de documento se deve a detecção de logotipos como texto. Espera-se eliminar essa situação uma vez que logomarcas (imagens) serão tratadas em um *pipeline* próprio, conforme proposto no Capítulo 4.

Tabela 3: Resultado da métrica de acurácia para a extração para documentos de classe 2.

Tipo de documento	<i>Text</i>	<i>Questions</i>	<i>Answers</i>
01	0.97	0.96	0.86
02	0.87	0.94	0.83
03	0.87	0.93	0.80
04	0.88	0.91	0.90
05	0.91	0.91	0.79
06	0.88	0.91	0.90
07	0.88	0.93	0.78

De modo geral, se pode perceber que em ambas as classes de documentos todos os modelos obtiveram resultados¹ de extração promissores, demonstrando um desempenho aceitável da abordagem, quanto a tarefa proposta.

5.2 Resultados dos experimentos sobre extração de tabelas

Como etapa fundamental para a abordagem proposta de *form understanding*, como dito anteriormente, tem-se a identificação de tabelas. A seguir são apresentados os resultados obtidos nos experimentos com os dados reais (descritos na Seção 4.2).

Observa-se na Figura 28 documentos caracterizados como não estruturados, visto que cada laboratório tem o seu modo único de representar os dados, seguindo tanto as definições apresentadas na literatura quanto as propostas no Capítulo 2. A identificação correta de dados tabulares está demarcada nas cores laranja e azul nos primeiros experimentos e rosa no segundo.

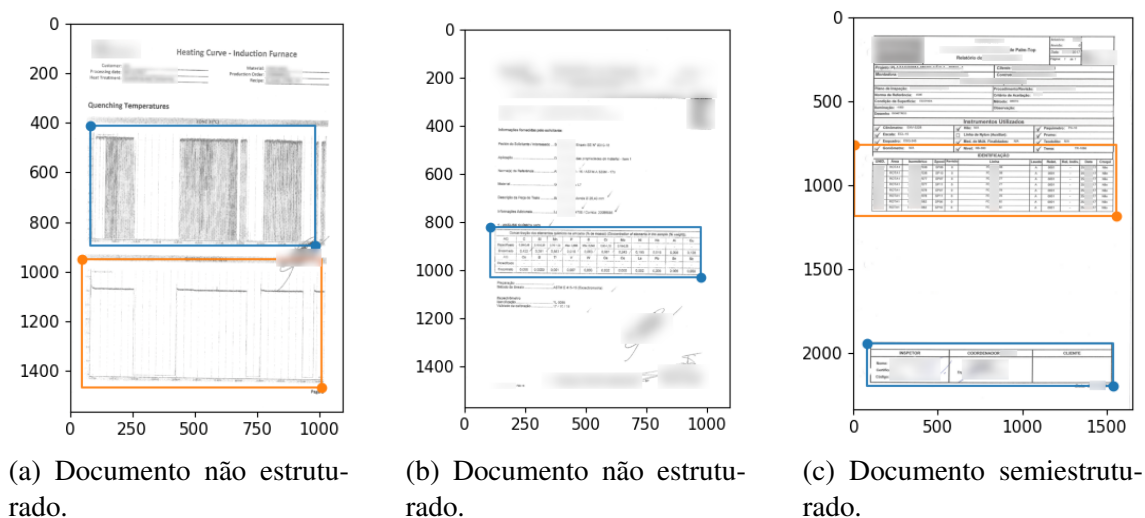
Na Figura 28c observa-se que há o acerto da tabela principal, mesmo ocorrendo um erro quanto a largura da *bounding box* identificada em laranja. Vale ressaltar que ao fim do documento em azul o modelo infere que há uma tabela, porém, nas definições utilizadas pelo Projeto ATT o trecho identificado não é considerado uma tabela e sim um formulário, trazendo assim a necessidade de formalizar com precisão o que, de fato, é considerado uma tabela ou não.

Já na Figura 29, o modelo errou ao inferir a tabela principal, pois o documento apresenta uma formação tabular, tornando difícil até mesmo para humanos uma identificação correta.

A fim de melhorar esses resultados foi utilizada a técnica chamada *Data Augmentation* para aumentar a quantidade de amostras disponíveis para o processo de treino do modelo.

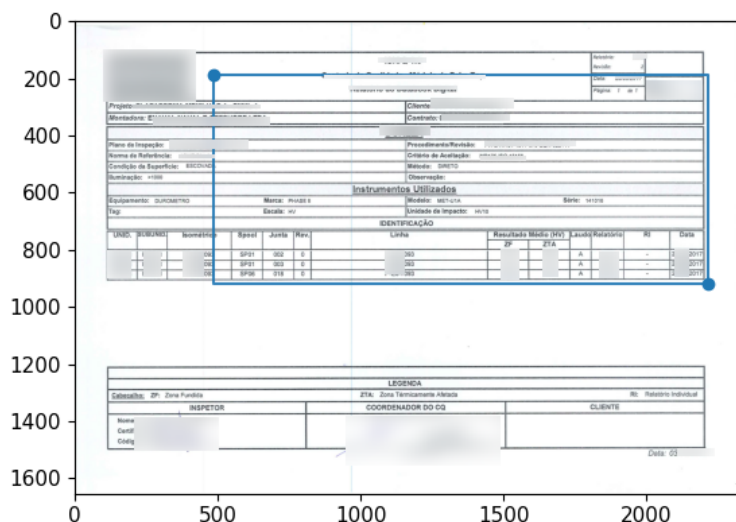
¹Os resultados preliminares referentes aos documentos estruturados aqui apresentados foram submetidos e aceitos na conferência 34^a SIBGRAPI - Conference on Graphics, Patterns and Images. O artigo intitulado “A Form Understanding Approach to Printed and Structured Engineering Documentation” foi apresentado em 22 de Outubro de 2021 na seção referente a Pattern Recognition & Applications.

Figura 28: Resultado de detecção obtido nos documentos do conjunto nas cores laranja e azul.



Fonte: Elaborado pelo autor.

Figura 29: Exemplo de erro de identificação de tabela. Foi identificado a metade de um formulário como se fosse uma tabela.



Fonte: Elaborado pelo autor.

Para isso, foram utilizadas as técnicas de adicionar ruídos como rotação das imagens, mudança de escala e, por fim, a técnica de *occlusion* a qual oculta uma parte da amostra. Todas as quatro técnicas utilizadas adicionaram 34 amostras, totalizando 136 novas amostras as quais foram utilizadas no processo de treinamento do modelo. Após o novo treino o modelo apresentou *recall* de 0.89, *precision* de 1.00 e *f1-score* 0.94. O modelo também entregou um *Intersection over Union (IoU)*² médio de 0.786.

²Métrica que especifica a sobreposição entre a *bounding box* original e a entregue pelo modelo. Quanto mais próximo do valor 1, melhor é o IoU.

As Figuras 30, 31 e 32 apresentam exemplos de comparações de detecções de tabelas realizados pelo modelo antes e depois do treino com *data augmentation*.

É possível perceber que na segunda série de experimentos o modelo conseguiu detectar a tabela em sua totalidade, não mais perdendo parte dela, como também conseguiu detectar tabelas que possuem padrões de células não usuais.

Figura 30: Exemplo 1: Comparação de resultados de detecção de tabela antes e depois do *data augmentation*.

The image shows two versions of a technical certificate form, labeled (a) and (b). Both forms are titled 'CERTIFICADO DE ANÁLISE QUÍMICA' and 'INDÚSTRIA'. They contain various sections for chemical analysis, mechanical properties (DUREZA), and tensile strength (TRAÇÃO). The forms are filled with data, including sample numbers, test results, and laboratory information. The forms are marked with red lines and boxes, indicating specific areas of interest or comparison.

(a) Antes.

(b) Depois.

Fonte: Elaborado pelo autor.

Figura 31: Exemplo 2: Comparação de resultados de detecção de tabela antes e depois do *data augmentation*.

RELATÓRIO DE INSPEÇÃO DIMENSIONAL

Cliente: [Redacted] PCS: 407 PI J2 Data: 25/04/18 Folha: 1/1

Descrição: [Redacted] PARAF: [Redacted] BT - NAT. Procedimento: [Redacted] Revisão: 00

Quantidade: [Redacted]

Item	Características	Mínimo	Máximo	Encontrado	AP	RE
01	[Redacted]	mm	mm			
02	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando		OK	
03	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando		OK	
04	Ø de Rosca (Nominal)	1,1	1,1		OK	
05	Gravação	J + B7	J + B7		OK	
06	Visual	Isento de Rebarbas	Isento de Rebarbas		OK	
07	Chavito	Referência 30°	Referência 30°		OK	

Resultado da Inspeção: ☒ Aprovado ☐ Reprovado ☐ RNC

(a) Antes.

RELATÓRIO DE INSPEÇÃO DIMENSIONAL

Cliente: [Redacted] PCS: 407 PI J2 Data: 25/04/18 Folha: 1/1

Descrição: [Redacted] PARAF: [Redacted] BT - NAT. Procedimento: [Redacted] Revisão: 00

Quantidade: [Redacted]

Item	Características	Mínimo	Máximo	Encontrado	AP	RE
01	[Redacted]	mm	mm	mm	OK	
02	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando	CR - aluando	OK	
03	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando	CR - aluando	OK	
04	Ø de Rosca (Nominal)	1,1	1,1	1,1	OK	
05	Gravação	J + B7	J + B7	J + B7	OK	
06	Visual	Isento de Rebarbas	Isento de Rebarbas	Isento de Rebarbas	OK	
07	Chavito	Referência 30°	Referência 30°	Referência 30°	OK	

Resultado da Inspeção: ☒ Aprovado ☐ Reprovado ☐ RNC

(b) Depois.

Fonte: Elaborado pelo autor.

Figura 32: Exemplo 3: Comparação de resultados de detecção de tabela antes e depois do *data augmentation*.

INDÚSTRIA

CERTIFICADO

Cliente: [Redacted] Nº: [Redacted]

Descrição: [Redacted]

Procedimento: [Redacted]

Revisão: [Redacted]

Quantidade: [Redacted]

Item	Características	Mínimo	Máximo	Encontrado	AP	RE
01	[Redacted]	mm	mm			
02	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando		OK	
03	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando		OK	
04	Ø de Rosca (Nominal)	1,1	1,1		OK	
05	Gravação	J + B7	J + B7		OK	
06	Visual	Isento de Rebarbas	Isento de Rebarbas		OK	
07	Chavito	Referência 30°	Referência 30°		OK	

Resultado da Inspeção: ☒ Aprovado ☐ Reprovado ☐ RNC

(a) Antes.

INDÚSTRIA

CERTIFICADO

Cliente: [Redacted] Nº: [Redacted]

Descrição: [Redacted]

Procedimento: [Redacted]

Revisão: [Redacted]

Quantidade: [Redacted]

Item	Características	Mínimo	Máximo	Encontrado	AP	RE
01	[Redacted]	mm	mm	mm	OK	
02	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando	CR - aluando	OK	
03	Rosca Calibração	Calibrador Anel - Não Passa	CR - aluando	CR - aluando	OK	
04	Ø de Rosca (Nominal)	1,1	1,1	1,1	OK	
05	Gravação	J + B7	J + B7	J + B7	OK	
06	Visual	Isento de Rebarbas	Isento de Rebarbas	Isento de Rebarbas	OK	
07	Chavito	Referência 30°	Referência 30°	Referência 30°	OK	

Resultado da Inspeção: ☒ Aprovado ☐ Reprovado ☐ RNC

(b) Depois.

Fonte: Elaborado pelo autor.

5.3 Experimentos com documentos não estruturados

Conforme informado anteriormente, optou-se por realizar os estudos a partir da atividade de *Semantic Entity Labeling*, por ser uma atividade desafiadora e apresentar tanto estudos quanto trabalhos recentes, com resultados significativos. Foram realizados experimentos utilizando os modelos de código aberto desenvolvidos pela empresa Microsoft nas plataformas *Github* e *Huggingface*, cujo modelo e pesos da rede foram disponibilizados. Por motivos de confidencialidade, o nome dos documentos e modelos serão chamados de Classe 1, 2 e 3. No qual os documentos de Classe 1 e 2 podem ser considerados documentos semiestruturados e estruturados, enquanto os de Classe 3 são não estruturados.

5.3.1 Resultados *LayoutLM* e *fine-tuning* com *FUNSD*

No documento de normativa, Classe 1, representado na Figura 33, pode-se observar que há o acerto em alguns campos do tipo *Answer*, como: número do contrato, registro CREA e número do documento. Porém o modelo erra em sua grande maioria. Este resultado era esperado, considerando-se que o mesmo não foi re-treinado e nem utiliza o pacote de idiomas adequado para o documento.

No modelo de documentos do tipo construção e montagem, Classe 2, conforme Figura 34 tem-se novamente a presença do acerto de algumas respostas, como por exemplo nos campos método, critério de aceitação e instrumentos utilizados. Porém, há a ocorrência do acerto em um conjunto de entidades classificadas corretamente como *Question* e *Answer*, no campo referente a “condição de superfície”.

Deve-se ressaltar que mesmo o modelo tendo classificado corretamente o conjunto de entidades como perguntas e respostas em sequência, o mesmo não apresenta uma relação semântica nem uma associação entre eles, pois tanto o *LayoutLM* (Xu et al. 2020) quanto o *LayoutLMv2* (Xu, Xu, Lv, Cui, Wei, Wang, Lu, Florencio, Zhang, Che, Zhang & Zhou 2021) são especializados apenas em classificar entidades (SEL), ou seja não fazem o *link* entre as entidades.

Como última classe de documento testado no modelo *LayoutLM* tem-se a capa de documentos do tipo fornecedores, a qual configura a Classe 3 e é demonstrado na Figura 35. Pode-se observar que há uma maior incidência de classificação de entidades do tipo *Question* e *Answer*, assim como também parte do *header* foi classificado corretamente. Outro destaque se dá ao campo “Número do pedido do cliente (ordem de compra)”, onde o mesmo classificou de maneira correta todos os *tokens* pertencentes à mesma célula semântica que formam a frase que caracteriza a *Question* relativa a ordem de compra.

Na seção de análise química e propriedades mecânicas houve uma ocorrência de classificações corretas em um grau mais elevado quando comparado aos demais documentos anteriormente apresentados.

Figura 33: Exemplo de documento da Classe 1 utilizado como teste no modelo *LayoutLM*.

[illegible]

Fonte: Elaborado pelo autor.

Por fim, a Figura 36 apresenta o resultado do *LayoutLM* em um documento no idioma inglês e ajustado para documentos do tipo formulário a partir do *dataset FUNSD* (Jaume et al. 2019). No geral pode-se observar que o mesmo apresenta um resultado superior e satisfatório para a atividade de classificação, evidenciando assim que, ao decorrer do Projeto ATT, quando for obtido um *dataset* anotado para realizar o ajuste destes modelos e adicionar-se o pacote de idiomas correspondente ao do documento a ser processado, deverá obter-se melhores resultados.

Figura 34: Exemplo de documento da Classe 2 utilizado como teste no modelo *LayoutLM*.

Relatório de Controle da Qualidade Projeto: [redacted] Cliente: [redacted] Montadora: [redacted] Contrato: [redacted]		Relatório: 0012 Revisão: 0 Data: 03/12/2017 Página: 1 de 1																																																																														
VISUAL DE AJUSTE Piano de Inspeção: [redacted] Norma de Referência: [redacted] Condição da Superfície: [redacted] Iluminação: >1000																																																																																
Instrumentos Utilizados ☒ Hinc: CS-023 ☐ Paquímetro: ☒ Trena: TR-892 ☒ Luxímetro: [redacted] ☒ Medidor de Espessura: ENV-3404 ☐ Medidor de Multifunções:																																																																																
IDENTIFICAÇÃO <table border="1"> <thead> <tr> <th>Unid.</th> <th>Área</th> <th>Identificação</th> <th>Spool</th> <th>Junta</th> <th>Rev.</th> <th>Incha</th> <th>TP</th> <th>DI</th> <th>Espes.</th> <th>L.</th> <th>Rel. Ind.</th> <th>Data</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>R</td> <td>AT</td> <td>7038</td> <td>SP05</td> <td>019</td> <td>0</td> <td>PC</td> <td>038</td> <td>TP</td> <td>60</td> <td>-</td> <td>03/12/17</td> </tr> <tr> <td>1</td> <td>R</td> <td>AT</td> <td>7038</td> <td>SP05</td> <td>019</td> <td>0</td> <td>PC</td> <td>038</td> <td>TP</td> <td>60</td> <td>-</td> <td>03/12/17</td> </tr> <tr> <td>1</td> <td>R</td> <td>AT</td> <td>7037</td> <td>SP05</td> <td>019</td> <td>0</td> <td>PC</td> <td>037</td> <td>TP</td> <td>60</td> <td>-</td> <td>03/12/17</td> </tr> <tr> <td>1</td> <td>R</td> <td>AT</td> <td>7038</td> <td>SP05</td> <td>022</td> <td>0</td> <td>PC</td> <td>038</td> <td>TP</td> <td>60</td> <td>-</td> <td>03/12/17</td> </tr> <tr> <td>1</td> <td>R</td> <td>AT</td> <td>7032</td> <td>SP05</td> <td>017</td> <td>0</td> <td>PC</td> <td>032</td> <td>TP</td> <td>60</td> <td>-</td> <td>03/12/17</td> </tr> </tbody> </table>			Unid.	Área	Identificação	Spool	Junta	Rev.	Incha	TP	DI	Espes.	L.	Rel. Ind.	Data	1	R	AT	7038	SP05	019	0	PC	038	TP	60	-	03/12/17	1	R	AT	7038	SP05	019	0	PC	038	TP	60	-	03/12/17	1	R	AT	7037	SP05	019	0	PC	037	TP	60	-	03/12/17	1	R	AT	7038	SP05	022	0	PC	038	TP	60	-	03/12/17	1	R	AT	7032	SP05	017	0	PC	032	TP	60	-	03/12/17
Unid.	Área	Identificação	Spool	Junta	Rev.	Incha	TP	DI	Espes.	L.	Rel. Ind.	Data																																																																				
1	R	AT	7038	SP05	019	0	PC	038	TP	60	-	03/12/17																																																																				
1	R	AT	7038	SP05	019	0	PC	038	TP	60	-	03/12/17																																																																				
1	R	AT	7037	SP05	019	0	PC	037	TP	60	-	03/12/17																																																																				
1	R	AT	7038	SP05	022	0	PC	038	TP	60	-	03/12/17																																																																				
1	R	AT	7032	SP05	017	0	PC	032	TP	60	-	03/12/17																																																																				
LEGENDA Cabeçalho: L: Laudo Espe: Espessura Rel. Ind: Relatório Individual TP: Tipo DI: Diâmetro Laudo: [redacted] INSPEÇÃO COORDENADOR CLIENTE Nome: [redacted] Permissão: [redacted] Código: [redacted] Sig: Metatopon - 0 Data: 04/12/2017																																																																																

Fonte: Elaborado pelo autor.

Figura 35: Exemplo de documento da Classe 3 utilizado como teste no modelo *LayoutLM*.

Indústria

CERTIFICADO

Nº: 100001 NFº:

Certificamos para os devidos fins que os produtos constantes em nosso(s) Pedido nº 100001 de dia 10/01/2020 em nome da empresa [redacted] foram fabricados com matéria prima qualificada e dentro dos mais rigorosos padrões de qualidade e conformidade.

Nº do pedido do cliente (ordem de compra): [redacted] Descrição do cliente: [redacted]

Item: PARAFUSO ESTUJO 1" BUNC A 14-1/2" (98mm) ZN/Ni ASTM A193-1B Gr.B7 (QTD 37)

Certificado nº 7 do lote nº 7 do fornecedor [redacted]

ANÁLISE QUÍMICA (%)

Material	C	Si	Mn	P	S	Cr	Ni	Mo	Cu	Al	B	Ti	Ba	Zn	Nb	Fe	N	P
ASTM A193 B7	0.210	0.25	0.71	0.018	0.007	0.82	0.18	0.22	0.120	0.01100	0.0030	0.0016		3.0210		0.00710	0.0040	

PROPRIEDADES MECÂNICAS

Dimensões (mm)	Seção (mm)	Lim. de Escapamento (MPa)	Lim. de Resistência (MPa)	Alongamento (%)	Estricção (%)	Dureza	Carga Aplicada
		807.84	885.79	19.00	55.05	30730/32	

Fonte: Elaborado pelo autor.

Figura 36: Resultado de SEL do modelo *LayoutLM* em um documento no idioma inglês e com o modelo ajustado.

R&D QUALITY IMPROVEMENT SUGGESTION/SOLUTION FORM

question answer question answer
Name/Phone Ext.: W. Hamann, P. Harper, P. Martinez Date: 9/3/92

question answer
Supervisor/Manager: J. S. Wigand R&D Group: Licensee

question answer
Suggestion: Discontinue coal retention analyses on licensee submitted product samples. (Note: Coal Retention Testing is not performed by most licensees) Other R&D physical measurements as ends stability and inspection for sort spots in cigarettes are thought to be sufficient measures to assure cigarette physical integrity. The proposed action will increase laboratory productivity.

question answer
Suggested Solution(s): Delete coal retention from the list of standard analyses performed on licensee submitted product samples. Special requests for coal retention testing could still be submitted on an exception basis.

header question question
Have you contacted your Manager/Supervisor? Yes No

question answer
Manager Comments: Manager, please contact suggester and forward comments to the Quality Council.

other
qip.wp

other
597005708

Fonte: Elaborado pelo autor.

Na próxima seção são apresentados os resultados obtidos ao realizar os mesmos procedimentos de testes apresentados nesta seção, porém submetendo os documentos ao modelo *LayoutLMv2*. Sendo este o atual novo estado da arte e que, segundo a literatura, apresenta melhorias consideráveis se comparado com *LayoutLM*.

A principal característica que difere o modelo *LayoutLMv2* do *LayoutLM* dá-se através da incorporação dos *embeddings* visuais durante o pré-treinamento, enquanto o modelo *LayoutLM* adiciona os mesmos apenas durante a etapa de ajuste fino. Pode-se destacar também que o modelo *LayoutLMv2* utiliza a ferramenta *Detectron2* da empresa Facebook como base para o *backbone* visual.

5.3.2 Resultados obtidos com o modelo *LayoutLMv2*

Pode-se observar que na Figura 37 o documento de Classe 1, assim como no *LayoutLM*, também apresenta uma classificação correta nos campos *Answer*, contudo não são observadas melhorias significativas esperadas nos resultados.

Figura 37: Exemplo de documento da Classe 1 utilizado como teste no modelo *LayoutLMv2*.

The image shows a document form with various sections and fields. The form is overlaid with a dense network of colored bounding boxes and labels, indicating the model's predictions for different parts of the document. The labels include 'question', 'answer', 'other', and 'embed'. The form contains various text fields, checkboxes, and a table at the bottom. The text is mostly in Portuguese, with some fields containing technical or administrative information. The bounding boxes are primarily in shades of green, blue, and red, indicating different levels of model confidence or classification. The text is mostly in Portuguese, with some fields containing technical or administrative information.

Fonte: Elaborado pelo autor.

Figura 39: Exemplo de documento da Classe 3 utilizado como teste no modelo *LayoutLMv2*.

[illegible]

Fonte: Elaborado pelo autor.

qual o mesmo foi ajustado.

Figura 40: Resultado de SEL do modelo *LayoutLMv2* em um documento no idioma inglês e com o modelo ajustado.

[illegible]

Fonte: Elaborado pelo autor.

5.3.3 Comparação entre os modelos

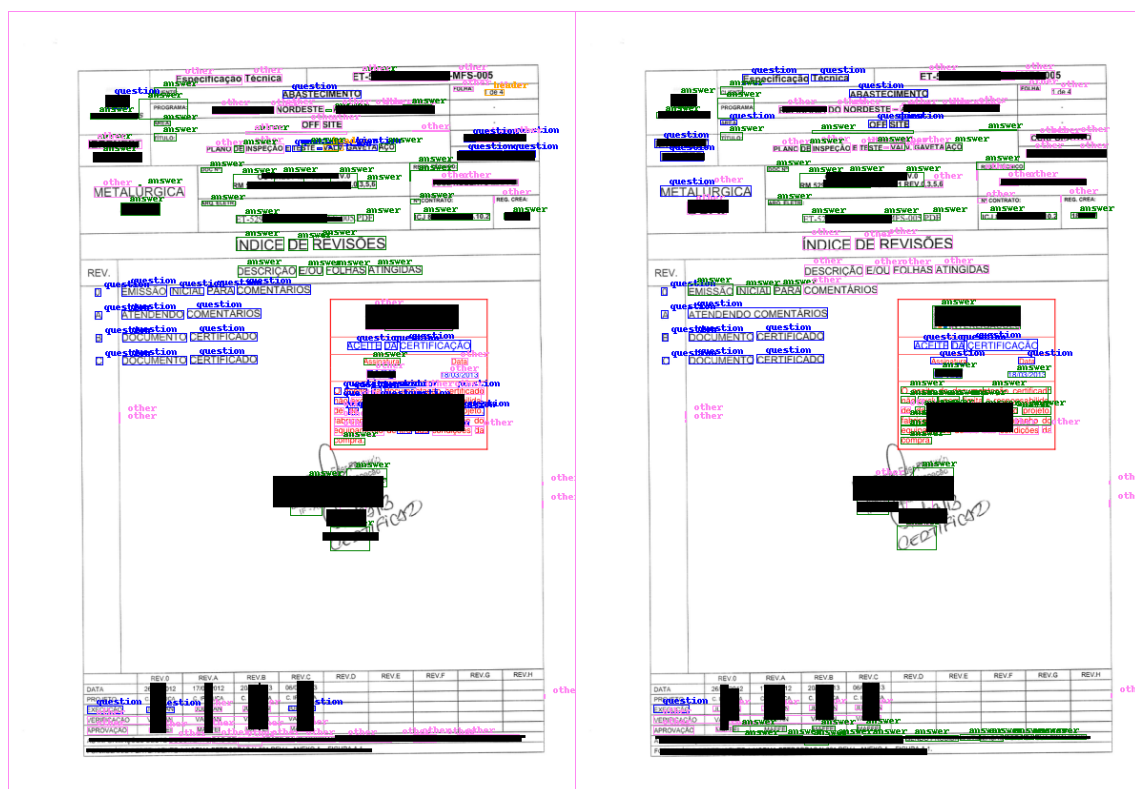
A seguir, são apresentados os resultados de comparação entre os modelos *LayoutLM* e *LayoutLMv2*.

De maneira geral, é possível perceber que nos modelos Classe 1 e Classe 2, nas Figuras 41 e 42, respectivamente, há pouco ganho na quantidade de elementos classificados corretamente.

Figura 41: Comparação entre resultados de SEL no documento de Classe 1.

(a) *LayoutLM*.

(b) *LayoutLMv2*.



Fonte: Elaborado pelo autor.

Já no documento Classe 3, Figura 43, a diferença é significativa, uma vez que o modelo *LayoutLMv2* classificou corretamente boa parte dos campos, assim como o título do documento.

Notou-se também que modelos com diversas linhas apresentam resultados semelhantes entre o modelo *LayoutLM* e *LayoutLMv2*. Possivelmente este fato ocorre devido a presença de linhas (tabelas) que formatam os documentos, gerando ruídos no OCR e na segmentação. Reforçando esta afirmação, o documento Classe 3 apresentou um resultado melhor nas partes às quais não há a presença de tabelas.

Figura 42: Comparação entre resultados de SEL no documento de Classe 2.

(a) LayoutLM.

(b) LayoutLMv2.

Fonte: Elaborado pelo autor.

Figura 43: Comparação entre resultados de SEL no documento de Classe 3.

(a) LayoutLM.

(b) LayoutLMv2.

Fonte: Elaborado pelo autor.

Os resultados positivos encontrados nos documentos do tipo Classe 3, somados aos resultados encontrados em documentos nos quais o idioma condiz com o usado durante o treino do modelo, Figura 44, acabam por demonstrar que há a possibilidade de melhorar os resultados encontrados até então, trazendo assim boas expectativas para os próximos passos dos experimentos.

Figura 44: Comparação entre resultados de SEL nos documentos em inglês e com os modelos ajustados.

(a) LayoutLM.

(b) LayoutLMv2.



header
RAW QUALITY IMPROVEMENT
SUGGESTION/SOLUTION FORM

question **answer**
Name/Phone Etc.: M. Hamann / P. Haver / P. Hollatz **DATE:** 2.9.92

question **answer**
Supervisor/Manager: W. H. Schmid **Position:** BA Group

question
Suggestion: Discontinuing local retention analyses on licensed submitted product samples; (total) local retention testing is not performed by pure licensees; unless 100% physical measurements are made regularly and inspection for new areas in laboratories are thought to be sufficient measures on future (possible) physical contacts the proposed action will increase laboratory productivity.

question **answer**
Suggested Solution(s): Delete local retention from the list of standard analyses performed on licensees submitted product samples; Special requests for local retention testing should still be submitted on an exception basis.

header **question** **question**
Have you contacted your Manager/Supervisor: Yes No

question **answer**
Manager/Supervisor: Manager: Please contact Suggester and forward comments to the Quality Council.

100-20
 592005208



MONTHLY INVOICE

Bill To:
Mr. Joseph Gail
Fountainhead Continental, Inc.
355 Madison Avenue
Suite 4500
New York, NY 10022

Invoice Number: 10010022
Invoice Date: 06/01/2012
Invoice Period: 05/01/2012 - 05/31/2012
Due Date: 06/15/2012

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number	Invoice Date	Invoice Period	Due Date	Balance Due
Gail Residence	10010022	06/01/2012	05/01/2012 - 05/31/2012	06/15/2012	\$11,812.80

Project Description	Invoice Number
---------------------	----------------

Fonte: Elaborado pelo autor.

5.3.4 Resultados com *LayoutLM* e *fine-tuning* com *dataset* próprio

Os dez tipos de documentos utilizados para criar o *dataset* próprio fazem parte dos documentos de Classe 2, utilizados na Seção 5.3.1 e foram renomeados para as siglas A,B,C,D,E,F,G,H,I e J por questões de manter o sigilo exigido no Projeto ATT. Os documentos do tipo A-F e J são no formato retrato e os documentos G-I são no formato paisagem.

Os documentos G e J são coloridos, enquanto os outros são em escala de cinza.

Para a realização dos experimentos o modelo *LayoutLM* foi utilizado com um *fine-tuning* feito com os documentos do *dataset*. Após cada experimento um novo modelo de documento foi adicionado, consequentemente adicionando 50 páginas a mais no processo de treinamento.

Todos os experimentos foram realizados na plataforma *Google Colab*, com CPU Intel(R) Xeon(R) 2.30GHz, 12GB de RAM e placa de vídeo Nvidia K80/T4 com 12GB/16GB.

Como o *dataset* é desbalanceado, optou-se por utilizar as métricas de *precision*, *recall* e *f1-score* para a avaliação do modelo, tendo em vista que as mesmas disponibilizam um melhor cenário de avaliação e com menor tendência do que apenas a métrica de acurácia, normalmente utilizadas para avaliar modelos de classificação balanceados. O modelo também entrega o valor obtido para a *loss*, no qual foi utilizada a função de *cross entropy*.

O primeiro experimento foi realizado com apenas o documento do tipo A e o modelo foi treinado utilizando 40 páginas, sendo que dez foram separadas para o conjunto de teste. Ao total, o processo de treinamento levou em média 37 minutos para ser realizado. De acordo com a Tabela 4 pode-se perceber que logo no primeiro experimento foram obtidos valores acima de 0.82 para as métricas de *precision*, *recall* e *f1-score*. Mostrando assim resultados promissores.

É importante ressaltar que devido a natureza da problemática trazida neste trabalho, se torna de maior interesse ter um valor de *precision* alto, uma vez que se espera minimizar os falsos positivos. Este fato ocorre uma vez que, dentre as possíveis *labels* utilizadas no *dataset*, apenas as referentes a “*S-Answer*” e “*S-Question*” são levada em consideração no cálculo dessas métricas, já que o grande desafio encontrado quanto a extração de informação dos databook se dá pelo entendimento e interpretação das informações de formulário e suas relações pergunta-resposta. As entidades classificadas como “*Text*” são úteis para auxiliar no processo de identificação da classe da página em si. Ou seja, a qual tipo de documento a mesma pertence e, portanto, não contribuem para o processo ligado a etapa de SEL a qual este trabalho se concentra. Retirando essas informações das métricas é possível ter um panorama real quanto a classificação das informações relevantes ao objetivo proposto.

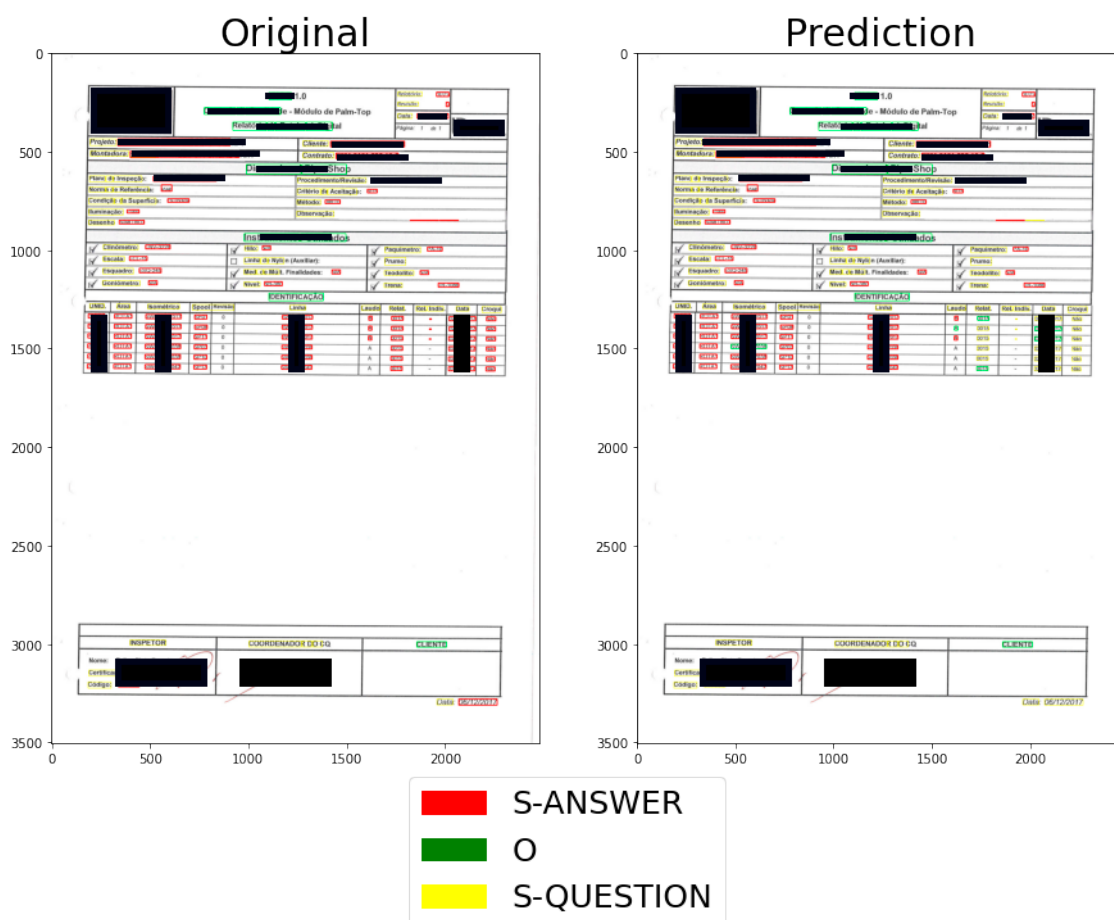
Na Figura 45 é apresentada uma comparação quanto a classificação original (esquerda) e a classificação feita pelo modelo (direita). Percebe-se que há, em sua maioria,

acertos quanto a predição, porém também são aparentes os erros quanto a classificação das entidades no formato de tabela. Percebe-se que várias linhas da tabela que deveriam ser classificadas como “*S-answer*” foram classificadas como “*S-Question*”.

É possível perceber também que o modelo tenta classificar as informações extraídas dos carimbos e assinaturas. Como o JSON de saída do *form understanding* de documentos estruturados que foi utilizado para criar o *dataset* não contava com os métodos específicos, os quais esperam-se ser utilizados para realizar de fato a extração dessas informações, os resultados encontrados apresentam uma qualidade inferior a do resto dos elementos presentes no documento.

No primeiro experimento foi encontrado um valor de *loss* de 0.486, este valor também se mostrou promissor, porém tem-se como objetivo encontrar valores de *loss* o mais próximos possíveis de 0. Assim, adicionou-se os documentos do tipo B ao *dataset* e foi realizado o segundo experimento.

Figura 45: Comparação entre as classificações originais e as obtidas pelo experimento com 50 páginas do *dataset* próprio.



Fonte: Elaborado pelo autor.

No segundo experimento tem-se 80 e 20 páginas para os conjuntos de treinamento e teste, respectivamente. Desta vez os resultados das métricas ficaram acima de 0.85, va-

lor superior aos encontrados no experimento anterior. Assim como também o valor de *loss* encontrado decaiu para um valor abaixo de 0.4. Houveram algumas melhorias na classificação, desta vez com a tabela com menos erros, porém com entidades erroneamente classificadas como “O” ao invés de “*S-Answer*” (Figura 50, no apêndice A). O processo de treino levou em média 63 minutos.

No terceiro experimento foram adicionados os documentos do tipo C, os quais possuem *layout* consideravelmente similares aos anteriores. Os resultados para a métrica desta vez foram superiores a 0.92, e a *loss* decaiu para 0.21. A Figura 51 no A apresenta o resultado de predição de uma página do conjunto de teste.

O processo de treino levou em média 84 minutos.

O quarto experimento adicionou os documento do tipo D e após o treino demonstrou valores de métricas superiores aos anteriores. Todas as três métricas entregaram o valor de 0.965, com um decréscimo de *loss* considerável de 0.21 para 0.1. Na Figura 52 no apêndice A foi possível perceber que a tabela apresentou em sua totalidade apenas dois erros de classificação. Porém, os formulários apresentaram uma quantidade de erros superior. O processo de treino levou em média 110 minutos para ser realizado.

Estes valores demonstraram que, ao aumentar a quantidade de documentos que são apresentados aos modelos, a performance do mesmo é aperfeiçoada. Contudo, como os documentos são similares, há a possibilidade do modelo estar decorando os padrões ao invés de estar aprendendo e se adaptando.

Para o quinto experimento foi adicionado o documento de tipo E, que apresentou resultados diferentes do padrão o qual o modelo estava exibindo até então. Segundo a Tabela 4, até então as métricas apenas incrementaram seus valores a cada experimento realizado. Porém, no quinto experimento houve um decaimento nos resultados, passando de 0.96 para 0.95 em média. Esta situação ocorre pois os documentos do tipo E possuem uma maior diferença quanto ao seus *layout* em relação aos documentos anteriores. Logo ele apresenta mais variabilidade descritiva. Esta ocorrência é demonstrada no valor de *loss* encontrado, o qual apresentou um aumento significativo, passando de 0.10 para 0.134. O processo de treino levou em média 131 minutos para ser realizado.

A predição após o treino com 250 páginas (Figura 53 no apêndice A) demonstrou que o modelo ainda não consegue classificar corretamente as informações mais superiores do formulário. Em contrapartida, o formulário anterior a tabela demonstrou uma classificação quase perfeita, assim como a própria tabela em si.

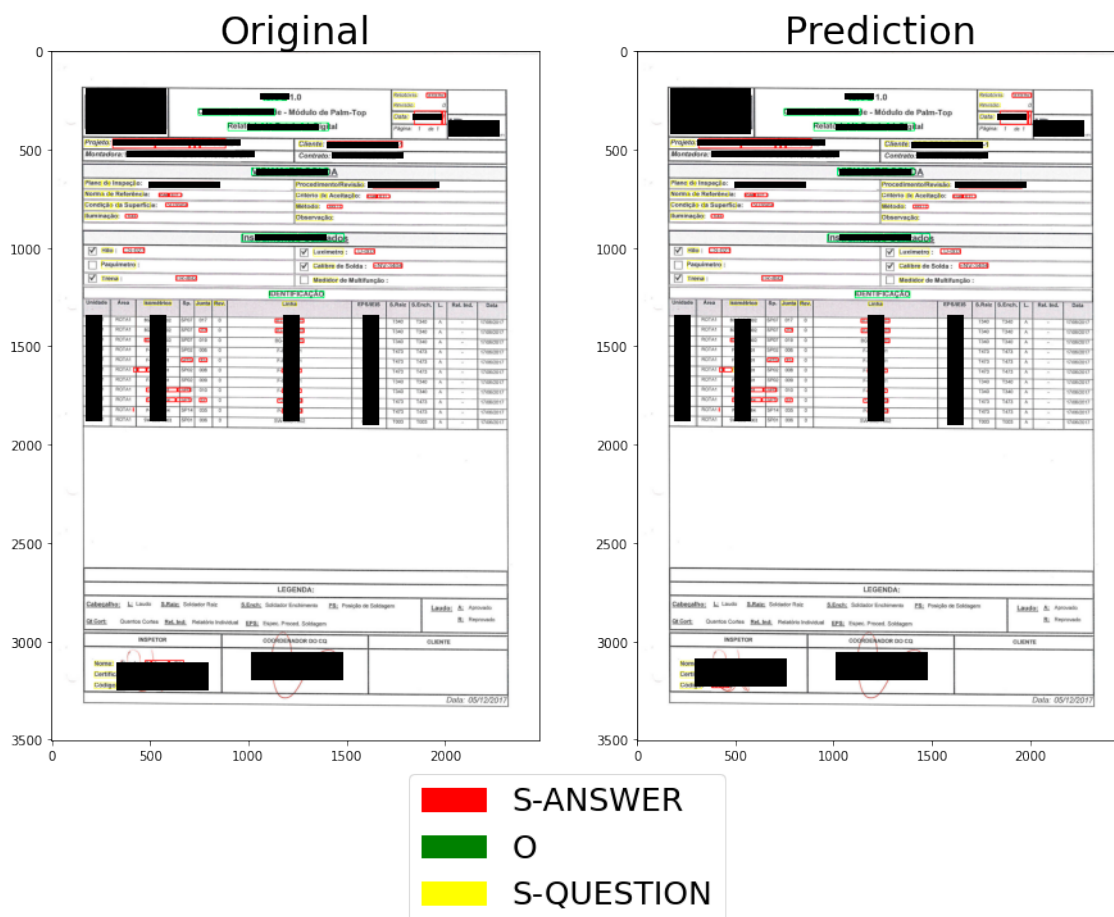
O decaimento dos resultados pode dar indícios de que o modelo estava de fato apenas decorando os padrões descritivos encontrados nos documentos, devido a grande quantidade de páginas similares entregues a ele.

Ao adicionar os documentos do tipo F no sexto experimento, os quais também possuem *layouts* com diferenças significantes aos anteriores, se obteve novamente o padrão de comportamento esperado, no qual os valores das métricas aumentaram novamente para

0.96 enquanto a *loss* decaiu 0.117. O processo de treino levou em média 173 minutos.

Com 300 páginas (Figura 46) tem-se uma melhora na classificação dos formulários superiores e dos bons resultados encontrados nas tabelas até então.

Figura 46: Comparação entre as classificações originais e as obtidas pelo experimento com 300 páginas do *dataset* próprio.



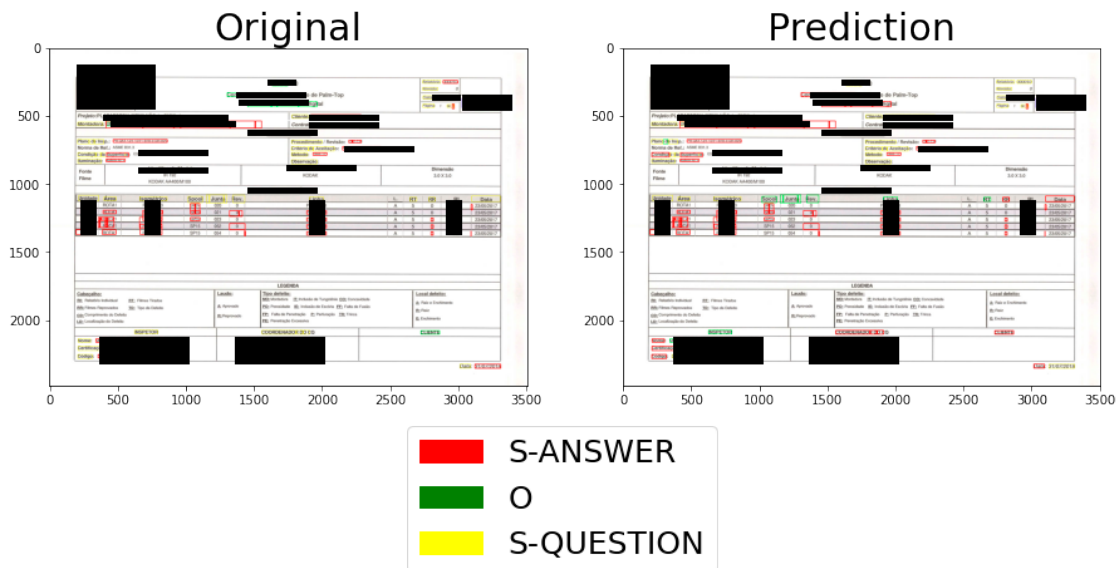
Fonte: Elaborado pelo autor.

Com o sétimo experimento foram adicionados os documentos do tipo G, os quais introduziram ao modelo os documentos no formato paisagem, aumentando ainda mais a variabilidade descritiva de modelos. Obteve-se como resultado o aumento nas métricas de avaliação, com *precision* e *f1-score* com o valor de 0.969 enquanto para *recall* foi obtido um valor de 0.97. A *loss* também caiu para 0.092, valor próximo de 0 como esperava-se atingir. Logo, foi possível concluir que o modelo ainda estava convergindo, uma vez que ao ser apresentado a um tipo de *layout* completamente diferente dos anteriores o modelo conseguiu não apenas continuar entregando valores de métricas semelhantes a experimento anterior, como também conseguiu reduzir o seu valor de *loss*. O processo de treino levou em média 198 minutos.

A Figura 47 mostra os resultados de predição do modelo em uma página do tipo paisagem, demonstrando que ao adicionar os documentos do tipo paisagem o modelo possui

alguns erros de classificação. Porém, em sua maioria, continuou entregando resultados satisfatórios.

Figura 47: Comparação entre as classificações originais e as obtidas pelo experimento com 350 páginas do *dataset* próprio.



Fonte: Elaborado pelo autor.

No oitavo experimento foi adicionado o documento do tipo H, também em formato paisagem e com *layout* significativamente diferente aos anteriores. Novamente tem-se mudanças leves nos resultados. *precision* e *f1-score* continuaram iguais enquanto *recall* teve um leve decaimento para 0.969 e a *loss* aumentou de 0.092 para 0.099. Ainda assim, o modelo continua demonstrando estar se generalizando e adaptando aos novos tipos de documentos inseridos no processo de treino.

Com 400 páginas (Figura 54 do apêndice A) é possível perceber que o documento em si possui poucas entidades para classificar. Isto se deve a que o processo de *form understanding* para documentos estruturados, que foi aplicado anteriormente, não conseguiu captar todas as informações necessárias. Porém, dentre as poucas entidades disponíveis, o modelo conseguiu classificar com sucesso grande parte delas. O treino levou em média 225 minutos.

Com o novo experimento foram adicionados os documentos do tipo I, os últimos no formato paisagem. Os resultados obtidos demonstram aumento nas métricas para valores acima de 0.97 enquanto houve uma queda de 0.99 para 0.76 de valor de *loss*. O treino levou em média 256 minutos.

Com 450 páginas (Figura 55 no apêndice A) o modelo consegue classificar corretamente praticamente todas as entidades da página, até mesmo as entidades do tipo “O”.

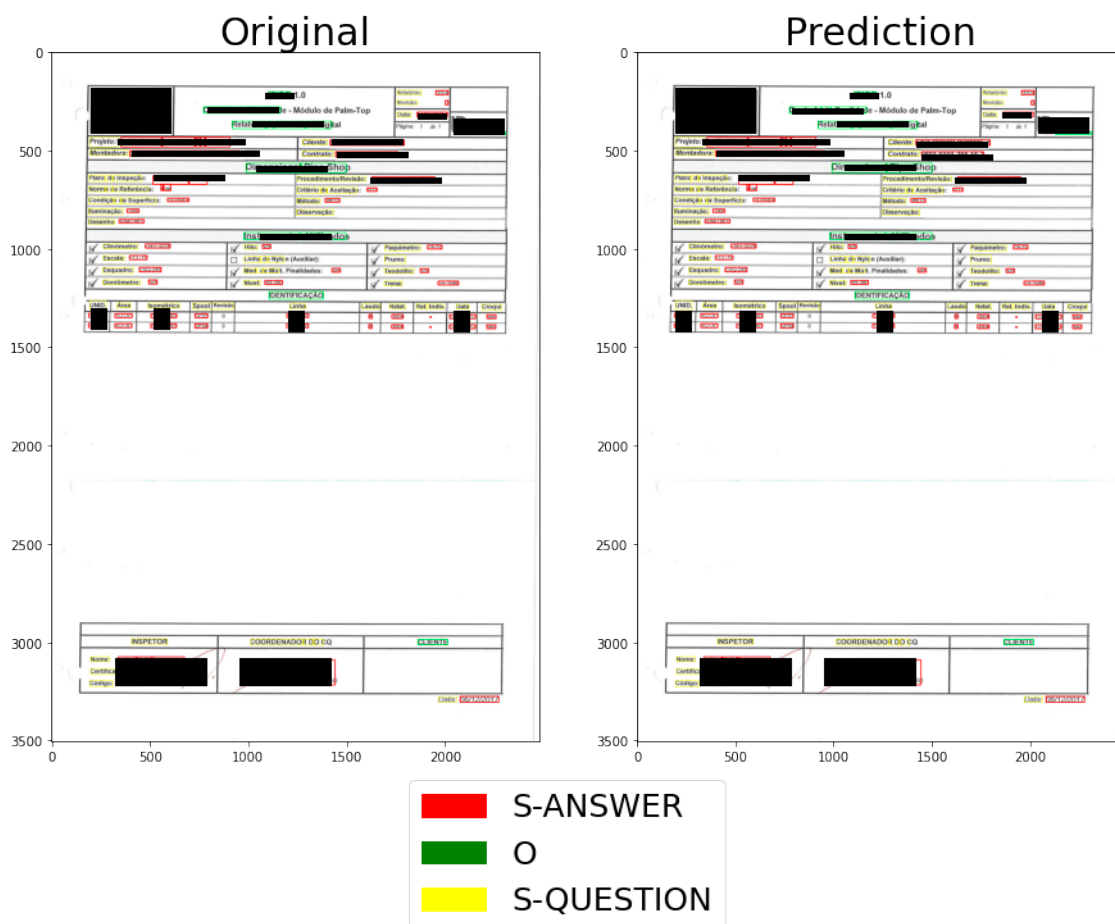
Os resultados obtidos até então são suficientes para demonstrar o sucesso nos experimentos do modelo com o *fine-tuning* no *dataset* próprio, uma vez que as métricas atingiram valores acima do esperado e o valor de *loss* está próximo de 0. Porém, optou-se

por realizar um último teste para assegurar a qualidade dos resultados.

No décimo e último experimento foram adicionados os documentos do tipo J que estão no formato retrato. Mais uma vez os resultados obtidos pelo modelo foram satisfatórios, demonstrando ainda um sutil acréscimo nos valores das métricas, entregando 0.977 para ambas *precision* e *f1-score* e 0.978 para *recall*. O valor obtido de *loss* decaiu ainda mais, atingindo a marca de 0.065.

Por fim, a Figura 48 mostra a comparação realizada no último teste. É possível perceber que o modelo entrega uma classificação praticamente perfeita entre a sua descrição e a descrição original. Observa-se que até mesmo a região de carimbo e assinatura, a qual não contempla o escopo aqui proposto, acabou por também apresentar resultados de classificação aceitáveis. O treino levou em média 292 minutos.

Figura 48: Comparação entre as classificações originais e as obtidas pelo experimento com 500 páginas do *dataset* próprio.



Fonte: Elaborado pelo autor.

Ao total todos os experimentos somaram 1569 minutos de treinamento.

A Tabela 4 mostra um resumo de todos os resultados encontrados durante os experimentos.

Tabela 4: Resultados dos experimentos do modelo *LayoutLM* com *fine-tuning* do *dataset* próprio.

Documentos usados	Nº de Páginas	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>Loss</i>
A	50	0.827	0.851	0.839	0.486
A,B	100	0.851	0.861	0.856	0.399
A-C	150	0.927	0.926	0.926	0.210
A-D	200	0.965	0.965	0.965	0.100
A-E	250	0.952	0.951	0.952	0.134
A-F	300	0.961	0.964	0.962	0.117
A-G	350	0.969	0.970	0.969	0.092
A-H	400	0.969	0.969	0.969	0.099
A-I	450	0.975	0.976	0.976	0.076
A-J	500	0.977	0.978	0.977	0.065

5.3.4.1 Inferência com documentos não estruturados do tipo 3

Após o último experimento, foram entregue ao modelo exemplos de documentos não estruturados do tipo 3, similares aos utilizados na Seção 5.3.1, com o objetivo de utilizar os pesos do modelo após o *fine-tuning* com o *dataset* próprio para analisar a performance do modelo nesses documentos já que os mesmos possuem *layouts* consideravelmente diversificados.

A Figura 49a mostra que o *fine-tuning* resultou em diferenças sutis no resultado. Destacam-se as tabelas as quais obtiveram resultados interessantes. A maior parte dos seus elementos foram classificados como “*S-Answer*”, o que é considerado corretamente, havendo apenas confusão quanto a alguns elementos do cabeçalho da tabela, o que já era esperado.

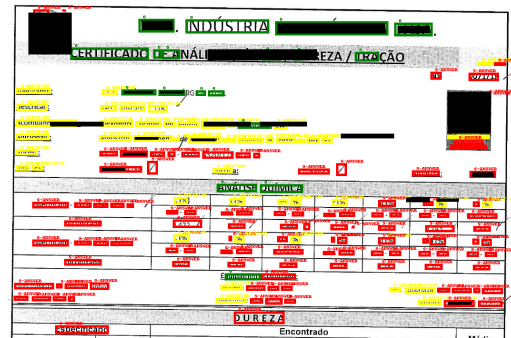
Na Figura 49b também pode-se observar o resultado positivo quanto a classificação na tabela, mas assim como na Figura 49c houveram classificações errôneas quanto aos formulários que precedem a tabela, o que contribui para a ideia de que utilizar o mesmo modelo para classificar ambos elementos simultaneamente não aparenta ser uma escolha de projeto eficiente.

Vale ressaltar na Figura 49d que o modelo conseguiu uma classificação muito próxima da desejada, novamente trazendo a maior parte dos erros na parte superior da página onde se encontra o formulário. Contudo, praticamente todos os elementos de baixo foram classificados corretamente, com exceção dos carimbos e assinaturas, o que também já era esperado, uma vez que na abordagem aqui proposta indica-se a análise separada desses elementos.

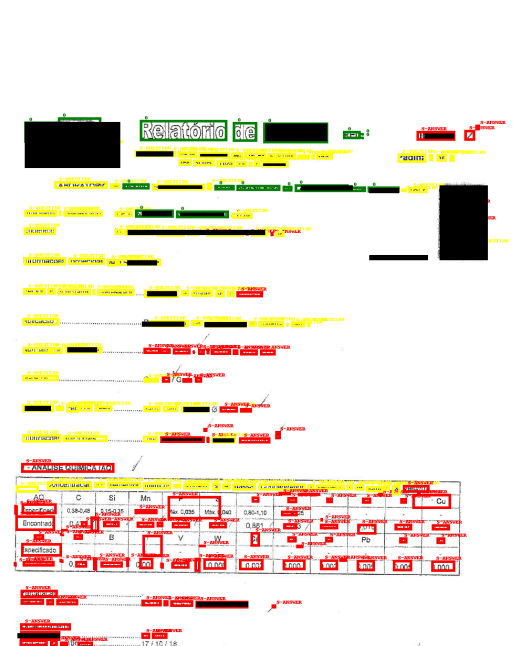
Figura 49: Resultado encontrados ao utilizar os pesos do modelo *LayoutLM* com *fine-tuning* do *dataset* próprio quando aplicado a documentos não estruturados do tipo 3.



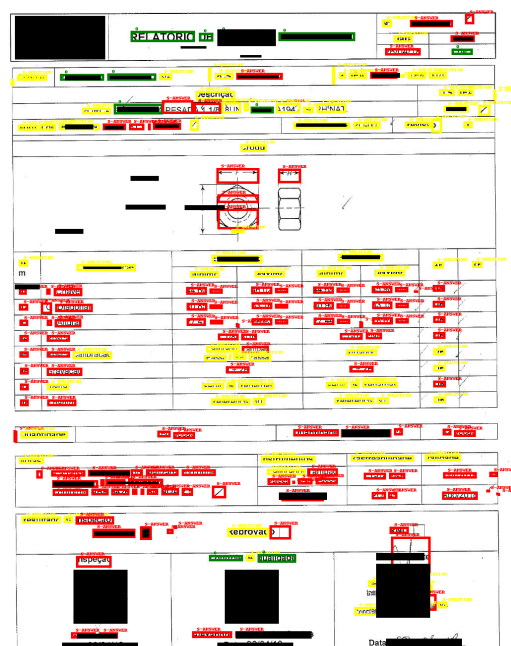
(a) Exemplo 1.



(b) Exemplo 2.



(c) Exemplo 3.



(d) Exemplo 4.

Fonte: Elaborado pelo autor.

5.3.5 Discussão

Ao longo do capítulo foram apresentados experimentos utilizando abordagens e documentos diferentes. Foram abordadas duas grandes frentes. A primeira, de documentos estruturados (Seção 5.1) e, a segunda, de documentos semi e não estruturados (Seção 5.3). Sendo a última dividida em duas partes: experimentos com *dataset FUNSD* e experimentos com um *dataset* próprio. Ainda foram adicionados resultados de experimentos realizados paralelamente quanto a identificação de tabelas, os quais tem como objetivo auxiliar no processo de segmentação do documento em elementos visuais distintos, o que foi proposto neste trabalho.

Com relação aos resultados obtidos para o processo de *form understanding* de documentos estruturados foi possível observar que, a princípio aparenta ser custoso por ser

necessário realizar a descrição do modelo para cada tipo de documento estruturado que for adicionado a abordagem. Contudo, uma vez que a descrição é feita em novos documentos nunca vistos pelo sistema, desde que contenham aquela descrição de *layout* são facilmente processados e suas informações extraídas com um grau de acurácia satisfatório. A abordagem não é perfeita, uma vez que logotipos presentes nas páginas são erroneamente processados e classificados com a *label* “text”, assim como quando um elemento de cabeçalho de tabela não é identificado como “answer”, todos os elementos de “question” atrelados a eles são perdidos.

Ainda assim, as extrações realizadas nos documentos de Classe 1 e 2 possuem acima de 80% de acerto. Demonstrando assim, ser uma contribuição interessante para a problemática em questão.

Com relação as tabelas, foi percebido que é possível utilizar um modelo de detecção simples, como YOLO, desde que haja um número de amostras significativo no processo de treino. Este número de amostras foi atingindo utilizando técnicas de *data augmentation* os quais acabaram por fornecer 10 vezes mais amostras dos que as contidas originalmente.

Para os resultados obtidos quanto aos documentos não estruturados, foram realizados dois tipos de experimentos principais, além dos resultados paralelos referentes a extração de tabelas.

O *fine-tuning* do *LayoutLM* utilizando o *dataset FUNSD* obteve resultados aceitáveis no processo de treino e teste, mas mostrou resultados preliminares não promissores quando aplicados aos documentos que compõem um databook, como os documentos de Classe 1 (normativa), 2 (construção e montagem) e 3 (fornecedores). Esses resultados eram esperados já que o modelo tinha sido ajustado com documentos do *dataset FUNSD* os quais, além de em sua maioria serem documentos que não possuem elementos visuais de extrema importância como tabelas e formulários, ainda possuíam um sistema de classificação de *label* diferente do desejado, o que levou a que o mesmo realizasse classificações desnecessárias, como diferenciar um título entre a *label* “Text” (texto) e “Header” (cabeçalho).

Em dois dos três tipos de documentos que foram testados, o modelo apresentou resultados com baixa qualidade. Apenas no documento de Classe 3 o mesmo demonstrou um certo acerto quanto as entidades textuais. Destacando-se uma inesperada classificação correta de boa parte da tabela referente a análise química. Essa situação foi pontual devido a configuração da tabela, a qual possui uma característica conhecida pelo modelo na qual a pergunta possui sua resposta no elemento logo a esquerda. A primeira linha da tabela foi classificada de maneira correta com a entidade “Material” sendo a pergunta e os elementos químicos como resposta. Além disso, a tabela logo abaixo, de propriedades Mecânicas, também obteve resultados interessantes, uma vez que sua configuração dos pares pergunta-resposta é diferente, com a resposta sendo o elemento logo abaixo da pergunta.

Apesar dos resultados não terem sido inicialmente positivos, os experimentos demonstraram que havia a possibilidade do modelo se adequar a tarefa referente a problemática aqui proposta e que, para isso, seria necessário realizar algumas modificações.

A primeira modificação se deu no sistema de *label* do modelo. Foram retiradas todas as *labels* que não eram de interesse, ficando apenas as três essenciais (*S-answer*, *S-question* e *O*). Após, foi detectada a necessidade de realizar o *fine-tuning* com documentos que fossem mais próximos da realidade dos *databooks*. Para isso, se percebeu a oportunidade de utilizar os resultados obtidos na primeira parte da abordagem aqui proposta, nos próximos experimentos.

Quanto aos experimentos realizados com o *dataset* próprio, logo no primeiro (50 páginas) já foi possível obter resultados acima do esperado com métricas de avaliação maiores que 0.80. Foram realizados experimentos nos quais aumentou-se o número de épocas de 2 para 5. Embora o resultado das métricas tenha sido superior, por volta de 0.98, o acréscimo de tempo para realizar o treino (1 hora e 53 minutos) somado a possibilidade de o modelo estar caindo em sobre-ajuste acabou mostrando-se não ser uma troca vantajosa.

Além disso, foi feito um experimento utilizando duas épocas de treinamento, mas 20 páginas ao invés de 50. Os resultados encontrados para as métricas foram em média 0.57 com o alto valor de *loss* de 1.32. Concluiu-se que o melhor cenário se dava ao continuar realizando todos os experimentos seguintes apenas com duas épocas de treino e 50 páginas de cada documento.

Conforme os experimentos foram sendo realizados, foi possível perceber que a diferença entre documentos no formato de retrato e paisagem não causou impacto significativo no modelo. De início, as previsões para documentos em paisagem continham mais erros devido ao fato de que havia sido entregue ao modelo uma quantidade muito maior de documentos no outro formato, não havendo assim amostras o suficiente para que o mesmo conseguisse interpretar e aprender o formato paisagem. Nos testes posteriores, conforme a quantidade de páginas dos dois formatos foi se igualando, notou-se que o modelo conseguiu se generalizar para ambos, de modo que não perdeu qualidade em nenhum dos dois formatos.

Com relação aos dois documentos digitalizados em cores, o *LayoutLM* não deu indícios de que o fato tenha contribuído ou dificultado o processo de aprendizado dos padrões. Mostrando, neste caso, não ser significativo.

Por outro lado, percebeu-se que o fato do documento apresentar tanto tabelas quanto formulários na mesma página acabou por prejudicar a classificação das entidades. Isso se deve a que formulários e tabelas serem elementos com grandes semelhanças entre si, muitas vezes confundindo até mesmo seres humanos quanto a sua classificação.

Por exemplo, usualmente tabelas possuem como pergunta as informações que estão contidas em seu cabeçalho, enquanto as respostas se dão pelas informações provenientes

das linhas logo abaixo da mesma. Já formulários geralmente seguem o padrão de que o elemento mais a esquerda será a pergunta com o elemento logo a direita sendo a resposta. Essas classificações se tornam subjetivas no momento em que se tem uma tabela no qual os elementos de cada linha se comportam como um formulário, com pergunta seguida de resposta. Quando este tipo de situação acontece trás a ambiguidade, pois afinal pode ser interpretada como uma tabela dentro de outro tabela ou um formulário dentro de uma tabela. Essa ambiguidade acaba por também confundir o modelo no processo de aprendizado e faz com que o mesmo não consiga generalizar os padrões que haviam sido aprendidos. Para resolver essa adversidade, o ideal seria processar esses elementos visuais de maneira separada, reforçando a abordagem proposta posteriormente. Desta forma, se fixa uma definição para tabela e outra para formulário, de forma que as mesmas serão seguidas rigidamente pelo processo de separação desses elementos, logo retirando do *LayoutLM* a responsabilidade de aprender a classificar em ambos os tipos de elementos.

Seguindo a abordagem de separação de elementos, por consequência se resolve a classificação referente aos carimbos e assinaturas presentes nos documentos. O *LayoutLM* mostrou-se capaz de entregar uma classificação acima da esperada nesses elementos. Contudo um modelo especializado nos mesmos possui maior chances de entregar os resultados esperados e desejados pelo Projeto ATT.

Como o modelo do *LayoutLM* utilizado para realizar a classificação tem como idioma primário o inglês, há a possibilidade de melhorar tanto o processo de treino quanto os resultados obtidos. Uma vez que haja uma versão estável do *LayoutLMV2* que tenha como escolher o Português Brasileiro como idioma principal, acredita-se que o modelo consiga se generalizar com uma quantidade menor de amostras. Assim diminuindo a necessidade de se ter um *dataset* grande como também diminuindo o tempo necessário para o treino do mesmo.

6 CONCLUSÃO

O presente trabalho apresentou uma abordagem para *form understanding* baseada em multitarefas, de forma a ser capaz de realizar o processo de extração de informações relevantes em documentos técnicos de engenharia, também chamados de *databooks*.

Inicialmente foi feita uma contextualização da fundamentação teórica, trazendo breves descrições quanto ao processo de extração de informação e o que significa *form understanding* e *document understanding*, apresentando ao leitor os conceitos necessários quanto a teoria que respalda o trabalho desenvolvido. Logo após foi realizado um levantamento do estado da arte, primeiramente de soluções acadêmicas, seguido por soluções comerciais, e apresentada uma breve discussão quanto as mesmas. Entregando um panorama geral quanto as soluções que estão sendo desenvolvidas no entorno deste tema.

Mostrou-se necessário então disponibilizar uma definição formal para os formatos de documentos encontrados, sendo eles documentos estruturados, semiestruturados e não estruturados.

Então foi apresentada a metodologia utilizada, dividida em duas grandes vertentes. A primeira condizente com os documentos estruturados, enquanto a outra focou nos documentos não estruturados. Também foi descrito o processo de criação e desenvolvimento do *dataset* próprio, bem como, por fim, as metodologia foram validadas por meio de trabalho experimental.

Os experimentos quanto a técnica de *form understanding* para documentos estruturados desenvolvida entregou resultados satisfatórios em questão de acurácia e, apesar do tempo inicial que deve ser despendido realizando a descrição manual do *template* quando deseja-se analisar um novo tipo de documento, ainda assim o tempo de execução para avaliar um lote de documentos mostrou-se vantajoso quando em comparação com a análise manual que era realizada anteriormente, tendo-se validado o uso da abordagem no Projeto ATT.

O treino do YOLO para detecção de tabelas em primeiro momento não mostrou resultados relevantes, porém ao se utilizar da estratégia de *data augmentation* foi possível atingir resultados interessantes, o que demonstrou que estes tipos de modelos simples podem ser utilizados para tarefas específicas como essa, desde que haja uma quantidade de

amostras expressivas para o processo de treino dos mesmos.

De forma análoga, os primeiros experimentos do *LayoutLM* resultaram em classificações genéricas e de baixa qualidade nos documentos alvo. Tendo em vista que o modelo havia passado pelo *fine-tuning* com o *dataset FUNSD*, é correto afirmar que o mesmo não se beneficiou deste ajuste, uma vez que o tipo de informação que estava classificando, as *labels*, não condizia com as *labels* de interesse para os *databooks*. Portanto, foi percebida a necessidade de ter um *dataset* próprio, constituído de páginas que pertencem a *databooks* reais do Projeto ATT.

Com a criação do *dataset* próprio notou-se a necessidade de realizar pequenas mudanças no método de *form understanding* para documentos estruturados, de forma que as entidades textuais sejam compostas apenas por palavras individuais, assim se aproximando do formato o qual o modelo *LayoutLM* e vários outros utilizam.

Após adaptar as *labels* para os pares de pergunta-resposta e realizar o *fine-tuning*, observou-se que o modelo foi capaz de se generalizar de maneira eficiente. Com apenas 50 páginas o modelo foi capaz de entregar métricas de avaliação acima de 0.82, o que por sua vez validou a utilização do mesmo na etapa de *Semantic Entity Labeling* a qual era proposta. Além disso, com os experimentos realizados, ao aumentar a quantidade de amostras e a diversidade do *dataset*, o modelo novamente conseguiu se generalizar de modo a entregar métricas de avaliação satisfatórias, tanto para páginas no formato paisagem quanto no formato retrato.

Em suma, o presente trabalho demonstra a complexidade por detrás do processo de extração de informações de maneira completa, em quaisquer tipos de documentos, também chamado como *document understanding*. Problemática a qual tem sido foco não só do ambiente acadêmico como também de gigantes da tecnologia como *Google* e *Microsoft*.

Foi demonstrado que seguir a abordagem proposta de multitarefas, dividindo cada elemento de uma página e avaliando-a de maneira separada, apesar de trabalhoso, entrega resultados animadores e oferece subsídios para o aprimoramento linear e focal de cada um desses elementos. Também foi possível concluir que, dentre os inúmeros desafios atrelados a problemática, uma das dificuldades aqui encontrada ao fazer a extração de informações de documentos não estruturados se dá pela análise de elementos como formulários e tabelas, demonstrando que realizar a análise pontual dos mesmos, o *form understanding*, é essencial e longe de ser uma tarefa considerada trivial. Abrindo assim um amplo campo de pesquisa com grandes oportunidades para contribuições acadêmicas e comerciais.

6.1 Contribuições

As principais contribuições entregues por este trabalho podem ser resumidas em:

- Definição formal quanto aos *layouts* de documento estruturado, semiestruturado e não estruturado, a qual entrega um melhor entendimento quanto aos formatos de documentos existentes e logo auxilia no direcionamento de estratégias específicas para os mesmos.
- Proposta de um *pipeline* completo para *document understanding* baseado em multitarefas que consiste na separação e análise individual dos elementos visuais do documento, visando entregar uma extração de informação otimizada.
- Proposta e desenvolvimento do método de *form understanding* para documentos estruturados baseado na descrição de *templates*.
- Desenvolvimento de um *dataset* próprio constituído de dez tipos diferentes de modelos de *databooks* reais, totalizando 500 páginas e um JSON descritivo quanto ao *layout* das mesmas.
- Necessidade de haver uma descrição formal e fixa quanto ao que é considerado tabela e o que é considerado formulário, quando se opera com esses dois tipos de elementos.
- Resultados promissores quanto a performance do modelo *LayoutLM* aplicado a tarefa de *semantic entity labeling*, demonstrando sua capacidade de classificação das entidades pergunta-resposta quando feito um *fine-tuning*, logo contribuindo para o processo de *form understanding* de documentos não estruturados.
- A validação da proposta quanto a análise individual dos elementos visuais presentes em cada página dos *databooks*, reforçando o conceito de multitarefas da abordagem.

6.2 Trabalhos futuros

Como próximos passos sugere-se modificar o método de *form understanding* para documentos estruturados de forma que cada entidade de texto extraída contenha apenas uma palavra individualmente ao invés de uma sentença como é atualmente. Essa forma viabiliza utilizar seu JSON de saída para construir novos *dataset* já no formato aceito pelo *LayoutLM*, assim como outros modelos, logo possibilitando novos experimentos.

Ainda assim, se pode também aperfeiçoar o método para realizar a separação das entidades textuais utilizado no *script* de criação do *dataset*, pois em alguns casos o valor de 5 *pixels* que foi assumido para cada carácter não condiz com a realidade do documento. Assim também possibilitando que o *dataset* já existente no projeto possa ser aperfeiçoado e adaptado no formato usualmente utilizado pelos modelos propostos na literatura.

Como passos futuros quanto aos documentos não estruturados, sugere-se como principal atividade a implementação de uma rede neural baseada em grafos para predição de *link*, de forma que a mesma seja capaz de predizer a conexão existente entre os pares de pergunta-respostas, assim cobrindo a etapa de “*entity linking*” que condiz com a terceira etapa do processo de *form understanding*.

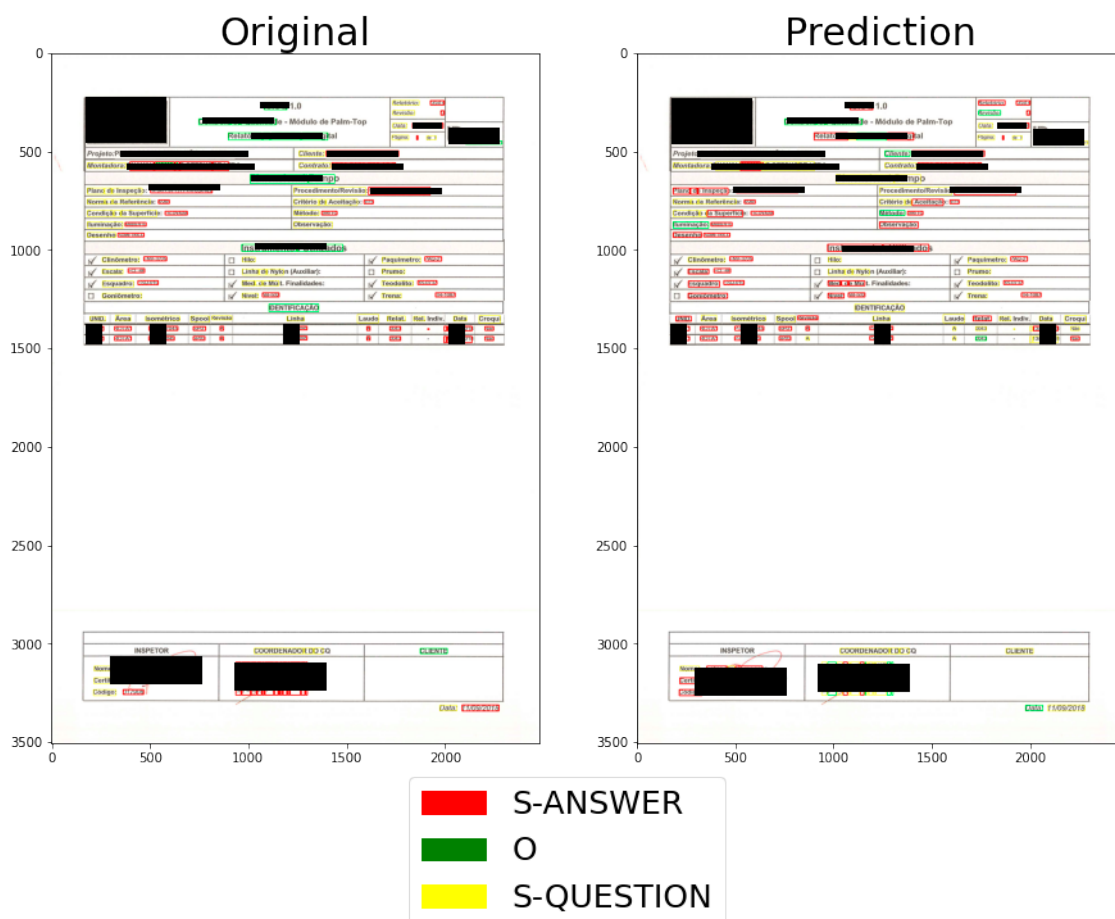
Sugere-se também realizar a classificação das entidades textuais presentes em formulários e tabelas separadamente, como sugerido na abordagem proposta para *document understanding*.

Por fim, sugere-se como passos futuros realizar uma série de experimentos no *dataset* próprio com a versão estável do modelo *LayoutLMV2*, trocando o idioma primário do modelo de inglês para português. Com esta mudança, acredita-se que a quantidade de documentos necessárias para o treino do modelo seria reduzida, assim reduzindo o tamanho exigido para que um *dataset* obtenha resultados satisfatórios, como também diminuindo o tempo necessário de treinamento do modelo.

A APÊNDICE

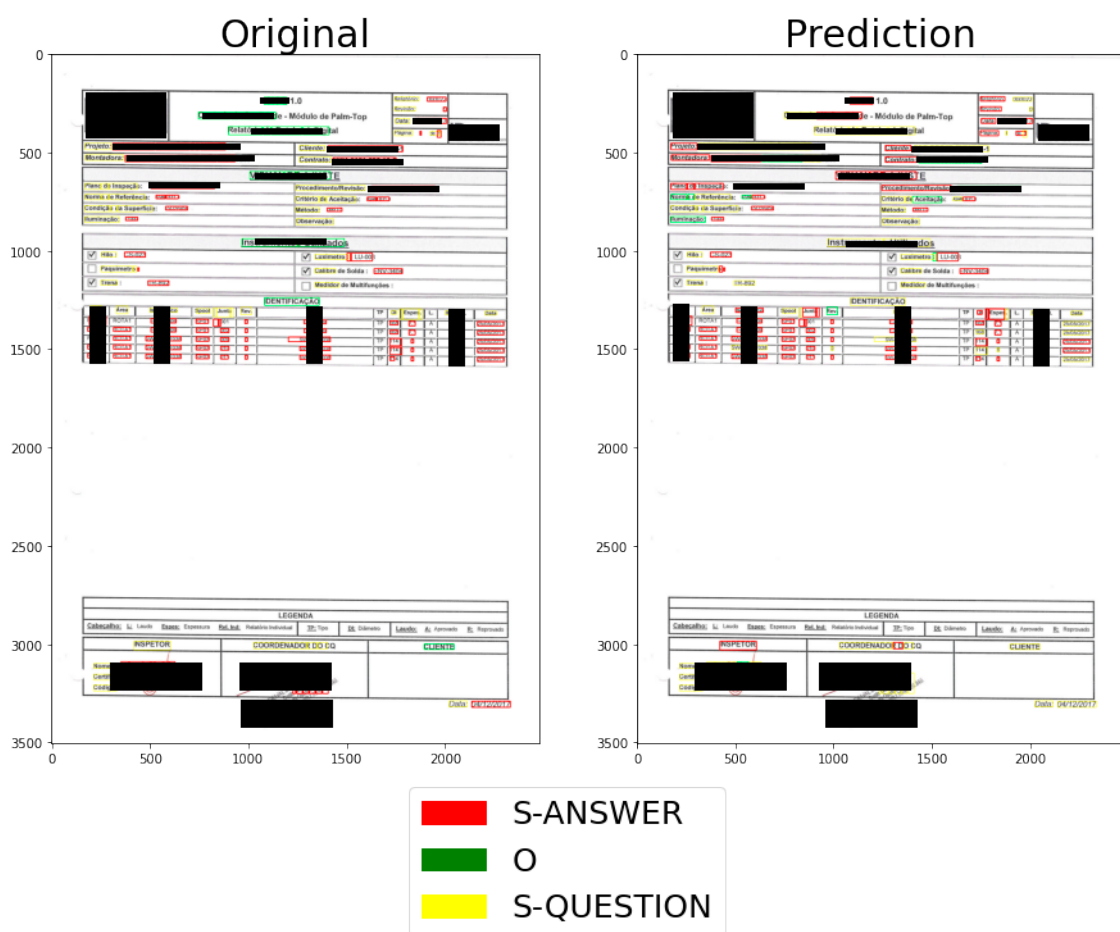
Figuras referentes aos experimentos do *fine-tuning* do modelo *LayoutLM* utilizando o *dataset* próprio desenvolvido.

Figura 50: Comparação entre as classificações originais e as obtidas pelo experimento com 100 páginas do *dataset* próprio.



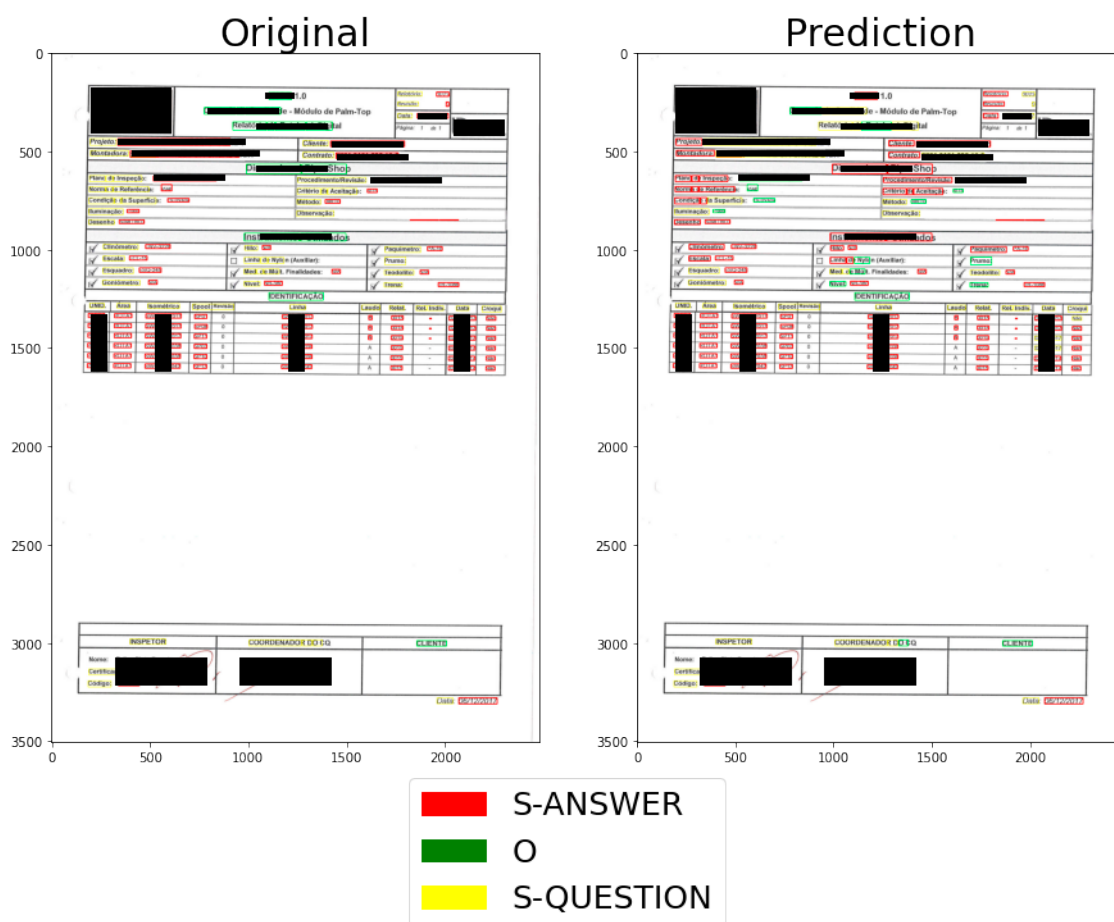
Fonte: Elaborado pelo autor.

Figura 51: Comparação entre as classificações originais e as obtidas pelo experimento com 150 páginas do *dataset* próprio.



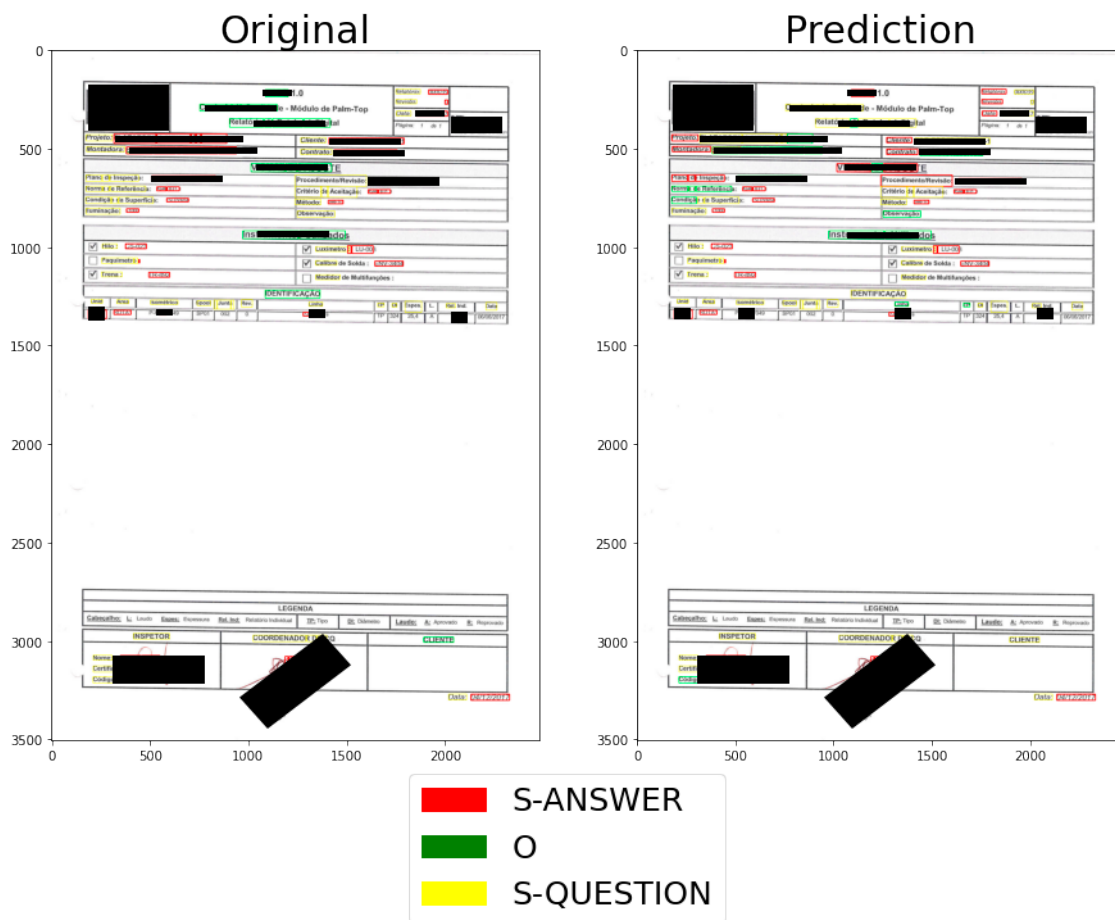
Fonte: Elaborado pelo autor.

Figura 52: Comparação entre as classificações originais e as obtidas pelo experimento com 200 páginas do *dataset* próprio.



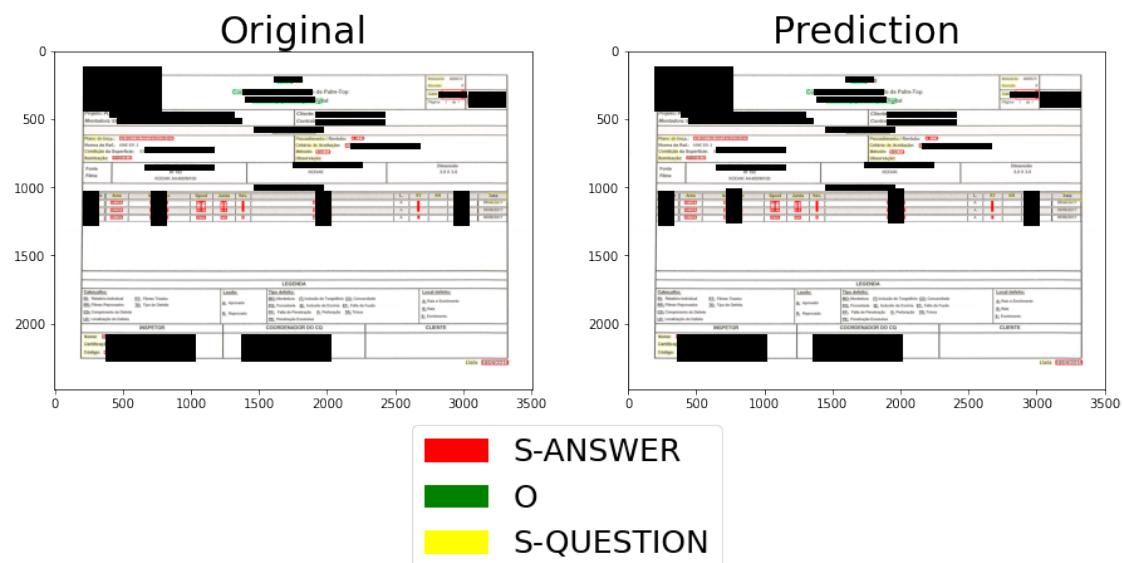
Fonte: Elaborado pelo autor.

Figura 53: Comparação entre as classificações originais e as obtidas pelo experimento com 250 páginas do *dataset* próprio.



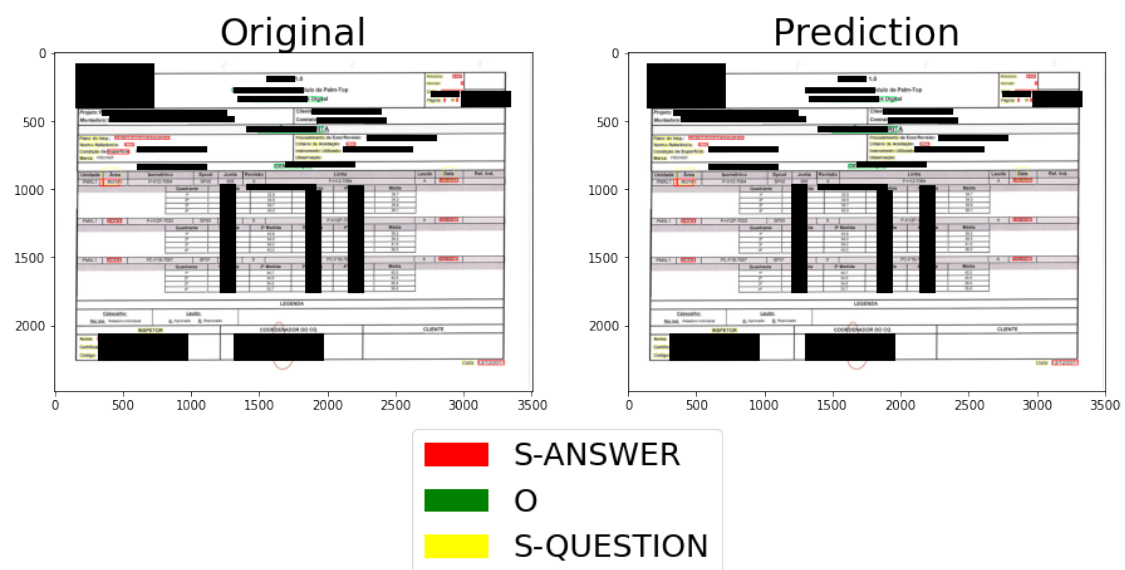
Fonte: Elaborado pelo autor.

Figura 54: Comparação entre as classificações originais e as obtidas pelo experimento com 400 páginas do *dataset* próprio.



Fonte: Elaborado pelo autor.

Figura 55: Comparação entre as classificações originais e as obtidas pelo experimento com 450 páginas do *dataset* próprio.



Fonte: Elaborado pelo autor.

REFERÊNCIAS

- Agbemenu, A. S., Yankey, J. & Addo, E. O. (2018), 'An automatic number plate recognition system using opencv and tesseract ocr engine', *International Journal of Computer Applications* **180**(43), 1–5.
- Ahmed Awan, M. N., Khan, S., Latif, K. & Khattak, A. M. (2019), 'A new approach to information extraction in user-centric e-recruitment systems', *Applied Sciences* **9**(14), 2852.
- Ahmed, S., Malik, M. I., Liwicki, M. & Dengel, A. (2012), Signature segmentation from document images, in '2012 International Conference on Frontiers in Handwriting Recognition', IEEE, pp. 425–429.
- Aiello, M., Monz, C., Todoran, L., Worring, M. et al. (2002), 'Document understanding for a broad class of documents', *International Journal on Document Analysis and Recognition* **5**(1), 1–16.
- Alam, H., Kumar, A., Nakamura, M., Rahman, A. F. R., Tarnikova, Y. & Wilcox, C. (2003), Structured and unstructured document summarization: Design of a commercial summarizer using lexical chains., in 'ICDAR', Vol. 3, p. 1147.
- Allen, J. (1988), *Natural language understanding*, Benjamin-Cummings Publishing Co., Inc.
- Appalaraju, S., Jasani, B., Kota, B. U., Xie, Y. & Manmatha, R. (2021), 'Docformer: End-to-end transformer for document understanding'.
- Binmakhashen, G. M. & Mahmoud, S. A. (2019), 'Document layout analysis: a comprehensive survey', *ACM Computing Surveys (CSUR)* **52**(6), 1–36.
- Bstieler, L., Gruen, T., Akdeniz, B., Brick, D., Du, S., Guo, L., Khanlari, M., McIlroy, J., O'Hern, M. & Yalcinkaya, G. (2018), 'Emerging research themes in innovation and new product development: insights from the 2017 pdma-unh doctoral consortium'.
- Cha, H. & Lee, D. (2018), 'Framework based on building information modelling for information management by linking construction documents to design objects', *Journal of Asian Architecture and Building Engineering* **17**(2), 329–336.

Cho, H.-C., Okazaki, N., Miwa, M. & Tsujii, J. (2013), ‘Named entity recognition with multiple segment representations’, *Information Processing & Management* **49**(4), 954–965.

Chowdhury, G. G. (2003), ‘Natural language processing’, *Annual review of information science and technology* **37**(1), 51–89.

Chuang, Z., Ming, W., Guang, L. C., Bo, X. & Zhi-qing, L. (2009), Resume parser: Semi-structured chinese document analysis, in ‘2009 WRI world congress on computer science and information engineering’, Vol. 5, IEEE, pp. 12–16.

Das, S., Banerjee, P., Seraogi, B., Majumder, H., Mukkamala, S., Roy, R. & Chaudhuri, B. B. (2018), Hand-written and machine-printed text classification in architecture, engineering & construction documents, in ‘2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)’, IEEE, pp. 546–551.

Deng, X., Sun, H., Lees, A., Wu, Y. & Yu, C. (2020), ‘Turl: Table understanding through representation learning’, *arXiv preprint arXiv:2006.14806* .

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2018), ‘Bert: Pre-training of deep bidirectional transformers for language understanding’, *arXiv preprint arXiv:1810.04805* .

Eikvil, L. (1993), ‘Ocr-optical character recognition’.

Embley, D. W., Campbell, D. M., Smith, R. D. & Liddle, S. W. (1998), Ontology-based extraction and structuring of information from data-rich unstructured documents, in ‘Proceedings of the seventh international conference on Information and knowledge management’, pp. 52–59.

Forcmański, P. (2010), ‘Stamp detection in scanned documents’, *Annales Universitatis Mariae Curie-Sklodowska, sectio AI-Informatica* **10**(1), 61–68.

GIURA, G. (n.d.), ‘Information extraction and social network analysis of criminal sentences’.

Grishman, R. (2015), ‘Information extraction’, *IEEE Intelligent Systems* **30**(5), 8–15.

Han, X. & Sun, L. (2012), An entity-topic model for entity linking, in ‘Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning’, pp. 105–115.

Hasnat, M., Chowdhury, M. R., Khan, M. et al. (2009), ‘Integrating bangla script recognition support in tesseract ocr’.

- Hobbs, J. R. & Riloff, E. (2010), ‘Information extraction.’, *Handbook of natural language processing* **15**, 16.
- Hwang, W., Yim, J., Park, S., Yang, S. & Seo, M. (2020), ‘Spatial dependency parsing for semi-structured document information extraction’, *arXiv preprint arXiv:2005.00642* .
- Jaume, G., Ekenel, H. K. & Thiran, J.-P. (2019), Funsd: A dataset for form understanding in noisy scanned documents, in ‘2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)’, Vol. 2, IEEE, pp. 1–6.
- Katti, A. R., Reisswig, C., Guder, C., Brarda, S., Bickel, S., Höhne, J. & Faddoul, J. B. (2018), ‘Chargrid: Towards understanding 2d documents’, *arXiv preprint arXiv:1809.08799* .
- Kipf, T. N. & Welling, M. (2016), ‘Semi-supervised classification with graph convolutional networks’, *arXiv preprint arXiv:1609.02907* .
- Kise, K. (2014), ‘Page segmentation techniques in document analysis.’.
- Koistinen, M., Kettunen, K. & Kervinen, J. (2017), ‘How to improve optical character recognition of historical finnish newspapers using open source tesseract ocr engine’, *Proc. of LTC* pp. 279–283.
- Lassoued, Y. & Deleris, L. (2016), Thesaurus-based hierarchical semantic grouping of medical terms in information extraction, in ‘Exploring Complexity in Health: An Interdisciplinary Systems Approach’, IOS Press, pp. 446–450.
- Lee, C.-Y., Li, C.-L., Wang, C., Wang, R., Fujii, Y., Qin, S., Popat, A. & Pfister, T. (2021), ‘Rope: Reading order equivariant positional encoding for graph-based document information extraction’, *arXiv preprint arXiv:2106.10786* .
- Li, C., Bi, B., Yan, M., Wang, W., Huang, S., Huang, F. & Si, L. (2021), ‘Structurallm: Structural pre-training for form understanding’, *arXiv preprint arXiv:2105.11210* .
- Liu, X., Gao, F., Zhang, Q. & Zhao, H. (2019), ‘Graph convolution for multimodal information extraction from visually rich documents’, *arXiv preprint arXiv:1903.11279* .
- MacGillivray, C. & Reinsel, D. (2019), ‘Worldwide global datasphere iot device and data forecast 2019–2023’, *Int. Data Corp.(IDC), Framingham, MA, USA, Tech. Rep. US45066919* .
- Majumder, B. P., Potti, N., Tata, S., Wendt, J. B., Zhao, Q. & Najork, M. (2020), Representation learning for information extraction from form-like documents, in ‘proceedings of the 58th annual meeting of the Association for Computational Linguistics’, pp. 6495–6504.

Manoharan, S. (2019), 'A smart image processing algorithm for text recognition, information extraction and vocalization for the visually challenged', *Journal of Innovative Image Processing (JIIP)* **1**(01), 31–38.

Mao, J., Abayan, M. & Mohiuddin, K. (1996), A model-based form processing subsystem, in 'Proceedings of 13th International Conference on Pattern Recognition', Vol. 3, IEEE, pp. 691–695.

Martinez-Rodriguez, J. L., Hogan, A. & Lopez-Arevalo, I. (2020), 'Information extraction meets the semantic web: a survey', *Semantic Web* **11**(2), 255–335.

Matsubayashi, K., Yamashita, A., Nonaka, H. & Konno, Y. (2018), A research on document summarization and presentation system based on feature word extraction from stored informations, in '2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)', IEEE, pp. 60–63.

Mebout, I. (2021), 'Parsing pdfs using yolov3', Disponível em: <https://github.com/ismail-mebout/Parsing-PDFs-using-YOLOV3>. Acesso em: 20 Junho 2021.

Medvet, E., Bartoli, A. & Davanzo, G. (2011), 'A probabilistic approach to printed document understanding', *International Journal on Document Analysis and Recognition (IJDAR)* **14**(4), 335–347.

Micenkova, B. & van Beusekom, J. (2011), Stamp detection in color document images, in '2011 International Conference on Document Analysis and Recognition', IEEE, pp. 1125–1129.

Mitra, M. & Chaudhuri, B. (2000), 'Information retrieval from documents: A survey', *Information retrieval* **2**(2), 141–163.

Namoodiri, A. M. & Jain, A. K. (2007), Document structure and layout analysis, in 'Digital Document Processing', Springer, pp. 29–48.

National Coordinator for Health Information Technology (2017), 'Hospitals Participating in the CMS EHR Incentive Programs,' Health IT Quick-Stat', <https://dashboard.healthit.gov/quickstats/pages/FIG-Hospitals-EHR-Incentive-Programs.php>. Acessado em 20 de Setembro de 2021.

Norgeot, B., Glicksberg, B. S. & Butte, A. J. (2019), 'A call for deep-learning healthcare', *Nature medicine* **25**(1), 14–15.

Paliwal, S. S., Vishwanath, D., Rahul, R., Sharma, M. & Vig, L. (2019), Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned

document images, in ‘2019 International Conference on Document Analysis and Recognition (ICDAR)’, IEEE, pp. 128–133.

Patel, C., Patel, A. & Patel, D. (2012), ‘Optical character recognition by open source ocr tool tesseract: A case study’, *International Journal of Computer Applications* **55**(10), 50–56.

Patel, S. & Bhatt, D. (2020), ‘Abstractive information extraction from scanned invoices (aiesi) using end-to-end sequential approach’, *arXiv preprint arXiv:2009.05728*.

Peng, X., Setlur, S., Govindaraju, V., Sitaram, R. & Bhuvanagiri, K. (2009), Markov random field based text identification from annotated machine printed documents, in ‘2009 10th International Conference on Document Analysis and Recognition’, IEEE, pp. 431–435.

Pham, M., Alse, S., Knoblock, C. A. & Szekely, P. (2016), Semantic labeling: a domain-independent approach, in ‘International Semantic Web Conference’, Springer, pp. 446–462.

Prasad, D., Gadpal, A., Kapadni, K., Visave, M. & Sultanpure, K. (2020), Cascadetabnet: An approach for end to end table detection and structure recognition from image-based documents, in ‘Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops’, pp. 572–573.

Promin, E. & Suriyachai, P. (2019), Improvement of scanned medical document management system, in ‘2019 11th International Conference on Knowledge and Smart Technology (KST)’, IEEE, pp. 126–131.

Pudasaini, S., Shakya, S., Lamichhane, S., Adhikari, S., Tamang, A. & Adhikari, S. (2022), Application of nlp for information extraction from unstructured documents, in ‘Expert Clouds and Applications’, Springer, pp. 695–704.

Qi, Y., Huang, W. R., Li, Q. & Degange, J. (2020), Deep erase: Weakly supervised ink artifact removal in document text images, in ‘Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision’, pp. 3522–3530.

Rani, N. S., Vineeth, P. & Ajith, D. (2016), ‘Detection and removal of graphical components in pre-printed documents’, *International Journal of Applied Engineering Research* **11**(7), 4849–4856.

Ruan, W., Appasani, N., Kim, K., Vincelli, J., Kim, H. & Lee, W.-S. (2018), Pictorial visualization of emr summary interface and medical information extraction of clinical notes, in ‘2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)’, IEEE, pp. 1–6.

Santos, G. L., Silva, V. T., Dalmolin, L. A., Rodrigues, R. N., Drews, P. L. & Duarte Filho, N. L. (2021), A form understanding approach to printed and structured engineering documentation, *in* ‘2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)’, IEEE, pp. 330–337.

Sarawagi, S. (2008), *Information extraction*, Now Publishers Inc.

Shafait, F. & Smith, R. (2010), Table detection in heterogeneous documents, *in* ‘Proceedings of the 9th IAPR International Workshop on Document Analysis Systems’, pp. 65–72.

Singh, S. (2018), ‘Natural language processing for information extraction’, *arXiv preprint arXiv:1807.02383*.

Smith, R. (2007), An overview of the tesseract ocr engine, *in* ‘Ninth international conference on document analysis and recognition (ICDAR 2007)’, Vol. 2, IEEE, pp. 629–633.

Soto, C. & Yoo, S. (2019), Visual detection with context for document layout analysis, *in* ‘Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)’, pp. 3464–3470.

Tafti, A. P., Baghaie, A., Assefi, M., Arabnia, H. R., Yu, Z. & Peissig, P. (2016), Ocr as a service: an experimental evaluation of google docs ocr, tesseract, abbyy finereader, and transym, *in* ‘International Symposium on Visual Computing’, Springer, pp. 735–746.

Tang, G., Xie, L., Jin, L., Wang, J., Chen, J., Xu, Z., Wang, Q., Wu, Y. & Li, H. (2021), ‘Matchvie: Exploiting match relevancy between entities for visual information extraction’, *arXiv preprint arXiv:2106.12940*.

Thakare, S., Kamble, A., Thengne, V. & Kamble, U. (2018), Document segmentation and language translation using tesseract-ocr, *in* ‘2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS)’, IEEE, pp. 148–151.

Trabucchi, D., Baganza, T., Dell’Era, C. & Pellizzoni, E. (2018), ‘Exploring the inbound and outbound strategies enabled by user generated big data: Evidence from leading smartphone applications’, *Creativity and Innovation Management* **27**(1), 42–55.

Uçkan, T. & Karıcı, A. (2020), ‘Extractive multi-document text summarization based on graph independent sets’, *Egyptian Informatics Journal*.

Wang, Y., Wang, L., Rastegar-Mojarad, M., Moon, S., Shen, F., Afzal, N., Liu, S., Zeng, Y., Mehrabi, S., Sohn, S. et al. (2018), ‘Clinical information extraction applications: a literature review’, *Journal of biomedical informatics* **77**, 34–49.

Wang, Z., Zhan, M., Liu, X. & Liang, D. (2020), ‘Docstruct: A multimodal method to extract hierarchy structure in document for general form understanding’, *arXiv preprint arXiv:2010.11685* .

Wang, Z., Zhan, M., Ren, H., Hou, Z., Wu, Y., Zhang, X. & Liang, D. (2021), ‘Group-link: An end-to-end multitask method for word grouping and relation extraction in form understanding’, *arXiv preprint arXiv:2105.04650* .

Wu, X., Zheng, Y., Ma, T., Ye, H. & He, L. (2021), ‘Document image layout analysis via explicit edge embedding network’, *Information Sciences* **577**, 436–448.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K. et al. (2016), ‘Google’s neural machine translation system: Bridging the gap between human and machine translation’, *arXiv preprint arXiv:1609.08144* .

Xie, S., Girshick, R., Dollár, P., Tu, Z. & He, K. (2017), Aggregated residual transformations for deep neural networks, in ‘Proceedings of the IEEE conference on computer vision and pattern recognition’, pp. 1492–1500.

Xu, Y., Li, M., Cui, L., Huang, S., Wei, F. & Zhou, M. (2020), Layoutlm: Pre-training of text and layout for document image understanding, in ‘Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining’, pp. 1192–1200.

Xu, Y., Lv, T., Cui, L., Wang, G., Lu, Y., Florencio, D., Zhang, C. & Wei, F. (2021), ‘Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding’.

Xu, Y., Xu, Y., Lv, T., Cui, L., Wei, F., Wang, G., Lu, Y., Florencio, D., Zhang, C., Che, W., Zhang, M. & Zhou, L. (2021), ‘Layoutlmv2: Multi-modal pre-training for visually-rich document understanding’.

Yujian, L. & Bo, L. (2007), ‘A normalized levenshtein distance metric’, *IEEE transactions on pattern analysis and machine intelligence* **29**(6), 1091–1095.

Zhang, P., Xu, Y., Cheng, Z., Pu, S., Lu, J., Qiao, L., Niu, Y. & Wu, F. (2020), ‘Trie: End-to-end text reading and information extraction for document understanding’, *arXiv preprint arXiv:2005.13118* .

Zhong, X., Tang, J. & Yepes, A. J. (2019), Publaynet: largest dataset ever for document layout analysis, in ‘2019 International Conference on Document Analysis and Recognition (ICDAR)’, IEEE, pp. 1015–1022.

Zhu, G., Jaeger, S. & Doermann, D. (2006), A robust stamp detection framework on degraded documents, *in* 'Document Recognition and Retrieval XIII', Vol. 6067, International Society for Optics and Photonics, p. 60670B.

Zhu, G., Zheng, Y., Doermann, D. & Jaeger, S. (2008), 'Signature detection and matching for document image retrieval', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(11), 2015–2031.