An Artificial Intelligence Based Approach to Automate Document Processing in Business Area

by

Ta Hang Chen

B.B.A., Information Management, National Central University, 2007

Submitted to the System Design and Management Program and
the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of
Master of Science in Engineering and Management
and
Master of Science in Electrical Engineering and Computer Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2021

Author____

Ta Hang Chen
System Design and Management Program
Department of Electrical Engineering and Computer Science
May 20, 2021

Certified by____

Amar Gupta
Research Scientist, Computer Science and Artificial Intelligence Lab
Thesis Supervisor

Certified by____

Peter Szolovits
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Certified by____

Donna H. Rhodes
Principal Research Scientist, Sociotechnical Systems Research Center
Thesis Supervisor

Accepted by____

Joan S. Rubin
Executive Director, System, Design and Management

Accepted by____

Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

An Artificial Intelligence Based Approach to Automate Document Processing in Business Area

By

Ta Hang Chen

Submitted to the System Design and Management Program and
the Department of Electrical Engineering and Computer Science
on May 20, 2021 in Partial Fulfillment of the
Requirements for the Degrees of Master of Science in Engineering and Management
and
Master of Science in Electrical Engineering and Computer Science

Abstract

Automatic document processing is always a strategy for business executives to improve operational efficiency. With Optical Character Recognition (OCR) and machine learning techniques, businesses are able to apply Artificial Intelligence (AI) to automate the process. However, introducing an AI application to business is challenging; it is easy to fail because of the complexity between the technical and organizational components. This thesis considers document processing from a sociotechnical system perspective and leverages a four-step system analysis approach to identify the critical components.

This research also proposes a machine learning model using Support Vector Machine (SVM) as the classifier and Word2vec embeddings as document features to classify business documents. The proposed model reaches a 0.872 Macro F1-score using scanned business documents from the RVL-CDIP dataset. The proposed model outperforms the other commonly used rule-based algorithms, RIPPER and PART, showing that the proposed model is potentially suitable to be deployed into business to classify the documents.

Thesis supervisor: Amar Gupta
Title: Research Scientist, Computer Science and Artificial Intelligence Lab
Thesis supervisor: Peter Szolovits
Title: Professor of Electrical Engineering and Computer Science
Thesis supervisor: Donna H. Rhodes
Title: Principal Research Scientist, Sociotechnical Systems Research Center

Acknowledgement

I would like to take this chance to thank my advisors, Dr. Amar Gupta, Professor Peter Szolovits, and Dr. Donna H. Rhodes, for their guidance on my thesis work and writing. They always provide me detailed and critical feedback and encourage me to evolve this work to the next level. I am truly grateful to have them as my thesis advisors in my MIT journey.

I also want to thank my wife and my daughter for their support when I am writing this thesis. It was a tough year in 2020 as my family had to stay in Taiwan because of the Covid-19 pandemic. It was a huge loss that I missed my daughter's first birthday, but finally, things get better now.

I also want to thank the MIT community. I learned so much from the students, faculties, and friends in the campus in these two years. I enjoyed being a student again, and this incredible journey has certainly changed my life in some perspectives and my future career direction.

Lastly, I want to thank the students in the document processing lab, where I got to know so many great MIT undergraduate students. It was really my pleasure to work with you to deliver research results to the sponsors. My best wishes to all of you for a bright future.

Contents

List of Figures

List of Tables

Chapter 1

1 Introduction

In business operations, companies use documents to communicate ideas, transact business, and store agreements with external and internal parties. Typical business document categories are invoices, purchase orders, sales agreements, and tax forms. Processing business documents still relies heavily on manual effort to classify the documents and extract the information -- a costly operation. The function of document processing becomes a key driver to improve operational efficiency and reduce cost.

Current approaches for reducing the cost of document processing can be performed through business process outsourcing [1] or in-house [2]. Through outsourcing, businesses can have several additional advantages, such as focusing on core strategic areas, but face some challenges, like risks of exposing confidential data and management difficulties [3]. A typical document processing cycle within an organization includes receiving documents, sorting documents, pre-processing documents, and dispatching documents [4]. The process owner receives the document and conducts the corresponding transactions. The goal for businesses is to automate this repetitive operation.

Optical Character Recognition (OCR), workflow system, and machine learning techniques are the key technologies to build automatic document processing [5]. Additionally, Natural Language Processing (NLP) techniques are widely used to understand the content of business documents [6]. Lastly, Computer vision and image processing techniques are often necessary preprocessing tools for building an AI-based automatic document processing [5].

However, businesses often face challenges in deploying an automatic document process. A *Forbes* article in 2020 reported that fewer than 5% of AI applications were successfully deployed into business organizations [7]. The reason is that, for AI applications, businesses need to set up new policies, processes, and teams to continuously maintain the model for supporting business operations [8]. The key for a successful AI application is to have both social and technology elements working together properly in a system [8].

1.1 Motivations

Designing an automatic document processing process is not simply a technical issue but a complex organizational problem. Employees must be involved from end-to-end to process information correctly. Moreover, companies change their operations from time to time, so ongoing updating and maintenance are essential for deploying a sustainable automatic document processing model.

To understand the complex organizational and technical challenges when deploying an automatic document process, system analysis tools offer potential for generating insights for document processing. While system analysis approaches have been applied to different business domains, the focus of this dissertation is to apply a system analysis framework to identify the organizational challenges when deploying an AI-based document process.

My personal interest in this thesis is based on my experience in working in document processing centers in Dalian (China) and Bratislava (Slovakia). Both places still rely on human employees to classify documents and send them to downstream applications for processing. My personal experience is that there is a huge potential for improvement in this area, which motivates my investigation into understanding the challenging components of implementing an AI-based document processing solution.

1.2 Research Objective

The goal of this thesis is to identify the challenges when implementing an automatic document process in business and propose a machine learning model that is easy to implement and maintain. This thesis also incorporates a performance evaluation of the proposed model by comparing it with commonly used rule-based classifiers. Finally, this thesis aims to address the following research questions:

I. What are the critical elements that need additional attention when implementing automatic document processing?

II. What is the performance of the proposed model compared to other commonly used rule-based algorithms in the industry?

1.3 Research Approach

This thesis adopts Object Process Methodology (OPM) [9] to describe system components and uses Engineering System – Multidomain Decision Matrix (ES-MDM) [10] to analyze the critical elements in the system. For the machine learning model, this thesis uses SVM as the classifier, and Word2vec to approximate the document embeddings as the feature. RVL-CDIP (Ryerson Vision Lab Complex Document Information Processing) [11] provides the source for training and testing data. RVL-CDIP contains scanned business document images from the tobacco industry in the 1990s. This thesis uses Tesseract [12] OCR engine to extract text from digital documents in the dataset. The performance evaluation is based on the F1-scores between the proposed model and two other rule-based classifiers.

1.4 Thesis Structure

The thesis is organized into the following sections:

Chapter 1: This chapter introduces current document processing approaches, the research motivation, research objective, research approach, and thesis structure.

Chapter 2: This chapter provides the background of document processing, the concept of the system analysis tools, such as OPM and ES-MDM, and the relevant background information for the proposed machine learning classifiers used in this research.

Chapter 3: This chapter presents a system analysis for document processing as a system using Stakeholder Analysis, OPM, and ES-MDM. A network representation of ES-MDM is presented that shows the critical components with the highest degree of centrality.

Chapter 4: This chapter presents implementation details for the proposed machine learning model. The chapter describes details about the data source (RVL-CDIP), data transformation logic, program structure, proposed model, and performance evaluation.

Chapter 5: This chapter summarizes key findings from the research and analysis and discusses potential future work and the research questions.

Chapter 2

2 Background and Literature Review

This chapter provides the background of document processing, the concept of the system analysis tools, such as Object Processes Methodology (OPM) and Engineering System - Multi Domain Matrix (ES-MDM), and the relevant background information for the proposed machine learning classifiers used in this research.

2.1 Background

Business entities use documents to communicate ideas and conduct transactions. Although the technology is more advanced than it was a decade ago, business entities receive more documents than before due to business operation complexity. More and more types of business documents are created for various reasons, such as regulations and audit controls. Also, the business documents change constantly. The tax form is an excellent example of how often the form may be changed. Lastly, business entities often change their document processing organization every few years to optimize the processes. All these factors make it difficult for the business to deploy a fully automatic document process.

Document Process Automation is increasingly a goal for businesses. Several applications were developed in the 1990s [13], [14]; however, these applications are not scalable, and they are limited to particular organizations. Recently, machine learning has been used widely in document process automation [15]. The applications achieved good performance but still did not bring large-scale adoption. The challenges of implementing machine learning applications into business are on both the technical and business organizational levels [16], [17]. This thesis performs a system analysis to identify the challenges for deploying document process automation in business.

2.2 System Analysis

2.2.1 Sociotechnical System Approach

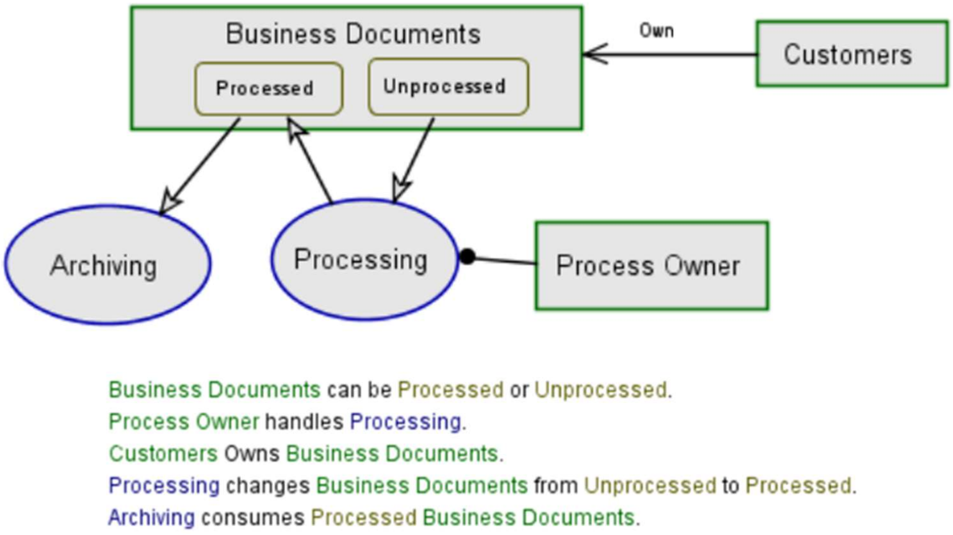
Document processing involves both social and technical elements in the system and can be viewed as part of a complex sociotechnical system. A sociotechnical system is characterized by substantial uncertainty because of human factors [18]. The sociotechnical system considers humans as assets that enhance a technical system through learning and adapting. Technology should support humans to achieve the goals but not replace them [19], [20].

In the software development domain, DevOps are being considered as a sociotechnical system [21], [22] as the development process involves different teams and there are uncertainties between the social factor (team) and technical components (software). In the business domain, sociotechnical system approaches have been used to improve the sales forecast process by identifying the issues between People, Tasks, Structure, and Technology domains using socio-technical matrix [23]. The other application is to view supply chain as a complex sociotechnical system and use System Dynamics and agent-based simulation to evaluate the performance [24].

For the AI-based application, an intelligence system can be viewed as a complex sociotechnical system because people use technology to perform tasks under organizational structure [19]. This thesis uses ES-MDM to analyze the system as one of the approaches that is appropriate for use in analyzing the sociotechnical system [25].

2.2.2 Object Processing Methodology

Object Processing Methodology (OPM) is a modeling language to conceptualize the system. OPM was first published in 2002 [9], and standardized as ISO 19450 in 2015 [26]. OPM contains two main components, Object Process Diagram (OPD) to visualize relationships and Object Process Language (OPL) to represent entity relationships in a text. OPD uses three basic diagram objects -- objects, states of an object, and processes -- to represent the system components [9]. An object can be physical or informational. Links between objects represent static and dynamic relationships between objects. Through all these symbols and relationships, OPM can construct a clear system description. An example of OPD and OPL is shown in Figure 1.



Business Documents can be Processed or Unprocessed.
Process Owner handles Processing.
Customers Owns Business Documents.
Processing changes Business Documents from Unprocessed to Processed.
Archiving consumes Processed Business Documents.

Figure 1. Example of an OPD with corresponding OPL
OPD visualizes abstract systems by object, status, and process.

OPM can present the system as a visual diagram and as text language at the same time. Grobshtein et al. compare the OPM with SysML and suggest that OPM has the advantage of navigating the system, including zooming in and zooming out [27]. In terms of OPM and UML comparison, Reinhartz-Berger and Dori conducted an experiment and showed that OPM is better in understanding system dynamics in modeling a web application system [28]. OPM has been used widely in industry. For example, Hiekata et al. used OPM to architect a maritime internet of things system to monitor equipment status [29], and Mordecai used OPM to capture the Cyber-Physical gap in the air traffic control system [30].

In the business application domain, Casebolt et al. use OPM to optimize the business process as a system in the enterprise by three steps: decomposing, rationalizing, and optimizing [31]. Wang et al. model a big data system using OPM [32]. Mordecai et al. use OPM to model the enterprise architecture in a digital transformation project [33]. All these examples support that OPM is an appropriate tool to describe the complex system in this research.

2.2.3 Engineering System - Multi Domain Matrix

Engineering System Multiple Domain Matrix (ES-MDM) is a Design Structure Matrix framework specially designed for complex engineering systems [10]. Engineering system relies on the factors from different domains to work cohesively to achieve the predefined tasks. To analyze the multidisciplinary interaction, ES-MDM contains six domains DSM: System Driver, Stakeholders, Objective, Function, Objects, and Activities [34]. From the six domains, ES-MDM constructs a holistic DSM and documents all the cross-domain interactions. ES-MDM is suitable for analyzing sociotechnical systems. The comprehensive matrix provides excellent raw data for system analysis. The information stored in ES-MDM can support the system modeler to identify critical components in the system.

ES-MDM has been adopted to analyze sociotechnical systems in different industries. Okami and Kohtake use ES-MDM to analyze the malaria surveillance system in Cambodia [35]. Alkhaldi and Alouani use ES-MDM to analyze healthcare organizations and proposed a generic conceptual model [36]. Songhori et al. use ES-MDM to analyze the Dutch Railway system, focusing on the change propagation perspective [37], [38].

2.3 Optical Character Recognition (OCR)

The original idea of character recognition was to aid the visually handicapped, and the first successful attempts were made by the Russian scientist Tyurin in the 1900s [39]. In the middle of the 1940s, the first version of character recognizers was invented with the development of digital computers [39].

The commercial OCR systems appeared in the 1960s. They were often considered first-generation OCR. The first-generation OCR systems used cut and try methodology with constrained letter shapes as input characters [40]. The second-generation of OCR systems appeared from the middle of the 1960s to the early 1970s. The second-generation of OCR systems were characterized by hand-printed character recognition capabilities [40]. The third-generation of OCR systems emerged between 1975 and 1985. The OCR systems started to process poor quality characters, and hand-printed characters for languages having an extensive character set, such as Chinese [40].

In the 1990s, image processing techniques, pattern recognition techniques, and artificial intelligence methodologies successfully enabled researchers to develop complex OCR algorithms [41]. Artificial Neural Network, Markov models, and

Natural Language Processing were applied to the OCR systems, and the OCR performance gain improvement [41].

Recently, deep learning models, such as convolutional neural network (CNN) and long short-term memory (LSTM) model, were applied to the OCR area [42]. Currently, OCR technology is widely used in industrial applications. This research uses Tesseract OCR Engine as the OCR engine to extract text information from data. Tesseract is an open-source OCR engine maintained by Google [12].

Tesseract performance is comparable to other commercial OCR packages. Tafti et al. conduct a performance evaluation for Tesseract, Google Doc OCR, ABBY FineReader, and Transym [43]. From that research, Tesseract's performance is comparable to that of other OCR packages in all the fields except for hand-written characters, multi-oriented text strings and noisy documents.

2.4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a robust supervised machine learning model for classification problems. SVM is a binary classification but can be expanded to a multi-class classifier with one-against-one or one-against-rest methods [44]. This research addresses a multi-class classification problem and uses one-against-rest implementation from the sklearn package [45]. The SVM model can handle both linear and non-linear classification tasks. The SVM kernel function is the key for classifying non-linear tasks. The kernel reduces the dimensionality of the input space for the classification. The commonly used kernel functions are linear, radial basis, polynomial, and sigmoid functions [46]. SVM has been widely used in applications in different domains. For the text classification, SVM has been used to perform sentiment analysis for Twitter data with word embedding as features and reaches a 0.8082 F1-score [47].

2.5 Word2vec Word Embeddings

Word embedding is a Natural Language Processing (NLP) technique using vectors to represent the semantic meanings of the words [48]. Word2vec was proposed by Google in 2013. There are two types of the training model for Word2vec: continuous bag-of-words (CBOW) and continuous skip-gram. CBOW uses surrounding words to predict the current word. Skip-gram uses the current word to predict the surrounding words. There are two common ways, average and sum, to construct document embeddings from individual word embeddings [49]. The average operator is used to approximate the document embeddings.

2.6 Rule-Based classifiers

This thesis evaluates the performance for three rule-based classifiers, OneR [50], RIPPER [51], and PART [52] using the RVL-CDIP dataset. OneR was proposed by R.C. Holte (1993) and is a simple algorithm to learn one classification rule for each class. RIPPER is the abbreviation for Repeated Incremental Pruning to Produce Error Reduction algorithm, proposed by William W. Cohen (1995). The RIPPER version used is the optimized version from the WEKA package [53], an improved version of the original algorithm [54]. PART is a partial decision tree which is generated from the C4.5 algorithm [52]. Although rule-based classifiers are simple, researchers have applied rule-based classifiers with embedding techniques to classify documents and get acceptable results [52]. The reason for selecting these classifiers is that these three classifiers have been used widely in text classification in the industry [55] [56] and generate stable performance with different datasets [57].

Chapter 3

3 System Analysis

This chapter presents a system analysis for document processing as a system using Stakeholder Analysis, Object Process Methodology (OPM), and Engineering System - Multi Domain Matrix (ES-MDM). A network representation of ES-MDM is presented that shows the critical components in the system.

3.1 Overview

Implementing an AI-based document processing into business is a complex task that involves both technical and organizational challenges. There are hidden efforts in the intersecting domains of business and technology. If the enterprise is viewed as a complex sociotechnical system, the use of a system analysis tool would be helpful to understand the critical components in the system boundary.

This thesis conducts a system analysis by using OPM and ES-MDM to understand the system components in the scope of automating document processing. Network analysis is presented as the final step for analyzing the network centrality to identify the critical components in the system. The detailed steps are listed below:

- The first step is to explore the social factors for the system; stakeholder analysis is performed based on need and value exchanges.
- The second step is to define the system boundary; OPM is the modeling language used in this thesis.
- The third step is to analyze the interactions between components; ES-MDM is the framework used in this thesis.

- The fourth step is to present the network visualization and perform centrality analysis for each node.

3.2 Document Processing as a System

Document processing is a dynamic function in business. It is a function that requires sub-components to work cohesively to achieve the goal and therefore, can be represented as a system. Businesses onboard new processes into document processing centers based on their changing needs. Regardless of how the document processing changes, the core functions, such as receiving, classifying, extracting, and validating the document, remain the same.

This thesis uses the case study from Swiss Post Solution [58] and the author's past relevant work experience in related fields as the input data for the system analysis. The Swiss Post Solution use case is available on the internet from IDC, International Data Corporation, and it represents a straightforward but effective document processing center design. Figure 2 is an example of a global delivery platform, which involves different teams cooperating to complete the tasks. Customers, document receiver, process owner, sub processes owner, and communicator are major teams participating in the document processing flow.

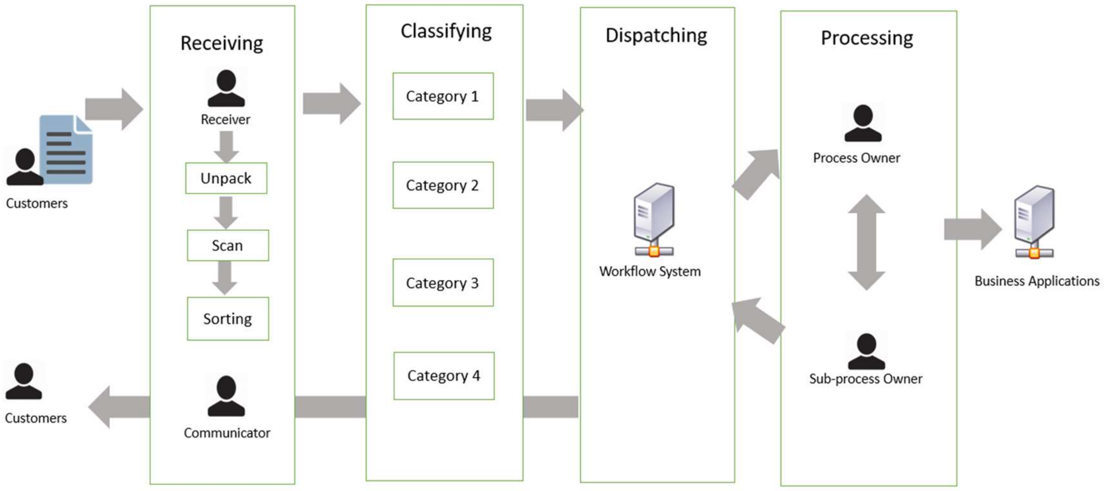


Figure 2. A typical process for Global Document Delivery Platform.
This process diagram illustrates a typical document processing center for mailroom operations.

3.3 Stakeholder Analysis

A definition for a system is an integrated set of elements, each with specified capabilities, which enable specific behaviors to achieve desired outcomes [59]. Stakeholder analysis is performed as part of understanding a system. If document processing is viewing in context of the sociotechnical system, the key beneficiary stakeholders are customers, suppliers, regulators, local communities, a business team, an IT team, and the enterprise.

Each of the stakeholders has different needs, and needs are exchanged value with another party within the system. Customers and suppliers are grouped into one object as they provide identical input to the document processing system. The stakeholder groups and their value exchanges are shown in Figure 3. The useful finding is that enterprise and customer/supplier are the main stakeholders for this system. They should receive additional attention when implementing the automatic document processing system.

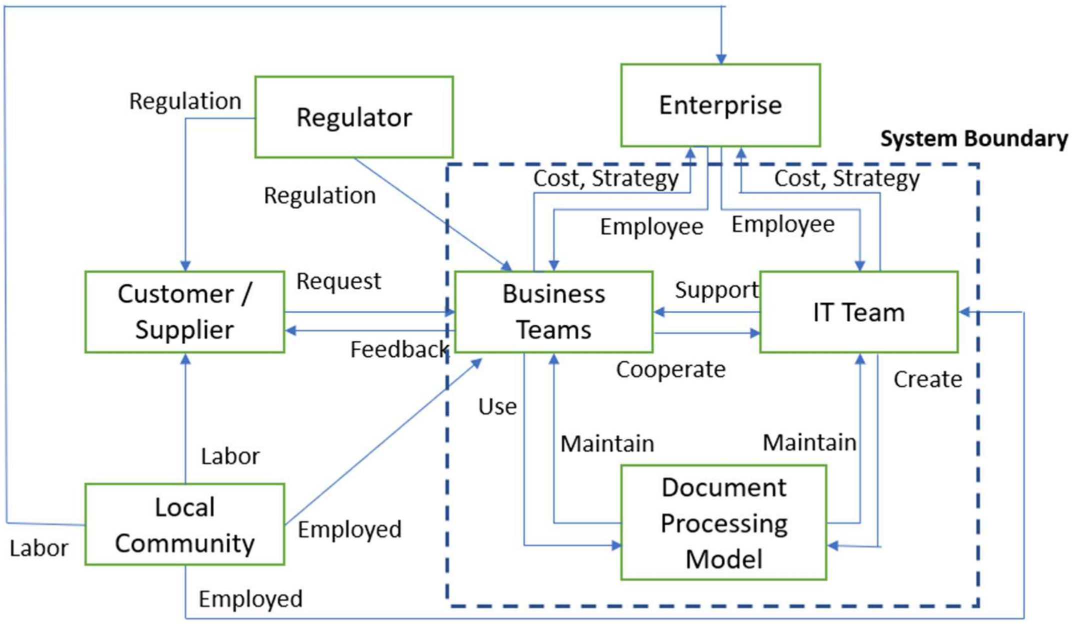


Figure 3. Stakeholder Analysis for Document Processing System.
This diagram shows the value and need exchange between stakeholders

3.4 Modeling the System

Several methods, for instance, SysML, are used widely to conceptualize the system. Among them, OPM has the advantage of describing holistic relationships. This thesis chose OPM to model document processing as a system, and OPCAT [60] is the software tool for developing OPM. OPM is a modeling tool to describe systems. It helps engineers to visualize an abstract system and analyze component-level relationships. OPM uses object, status, and process to describe the relationship between system components.

A typical document process has seven processes: Receiving Document, Classifying Document, Extracting Document, Processing Document, Communicating, Handling Errors, and Maintaining. The process starts from the Receiver processing the document and ends when either the error process is triggered or sends the communication back to customers or supports. The system OPD is depicted in Figure 4.

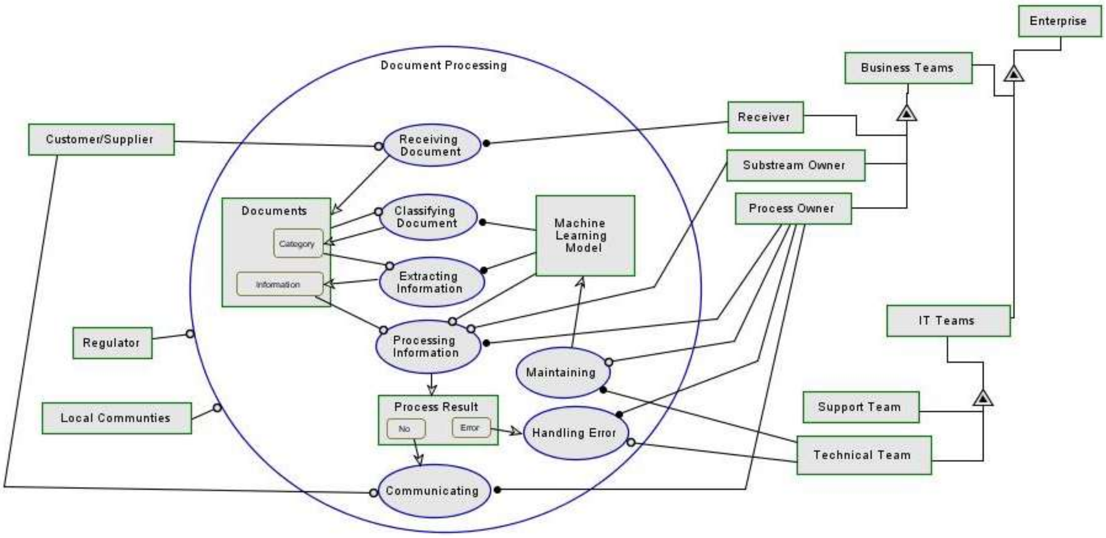


Figure 4. Object Process Diagram (OPD) for Document Processing as a System. The OPD diagram illustrates the system boundary of document processing.

Figure 4 shows that the Process Owner and the Artificial Intelligence Model are the two key agent objects for the system. They initiate all the processes except for receiving. The document category drives extracting information, which has the downstream effect on the Processing Information and Error Handling. In addition, the information technology team is the other key object in the system. The IT team maintains the Model and handles the documents flow to Error Handling for the process owner to resolve the issue. The subsequent discussion will be based on the system defined in this section.

3.5 ES-MDM

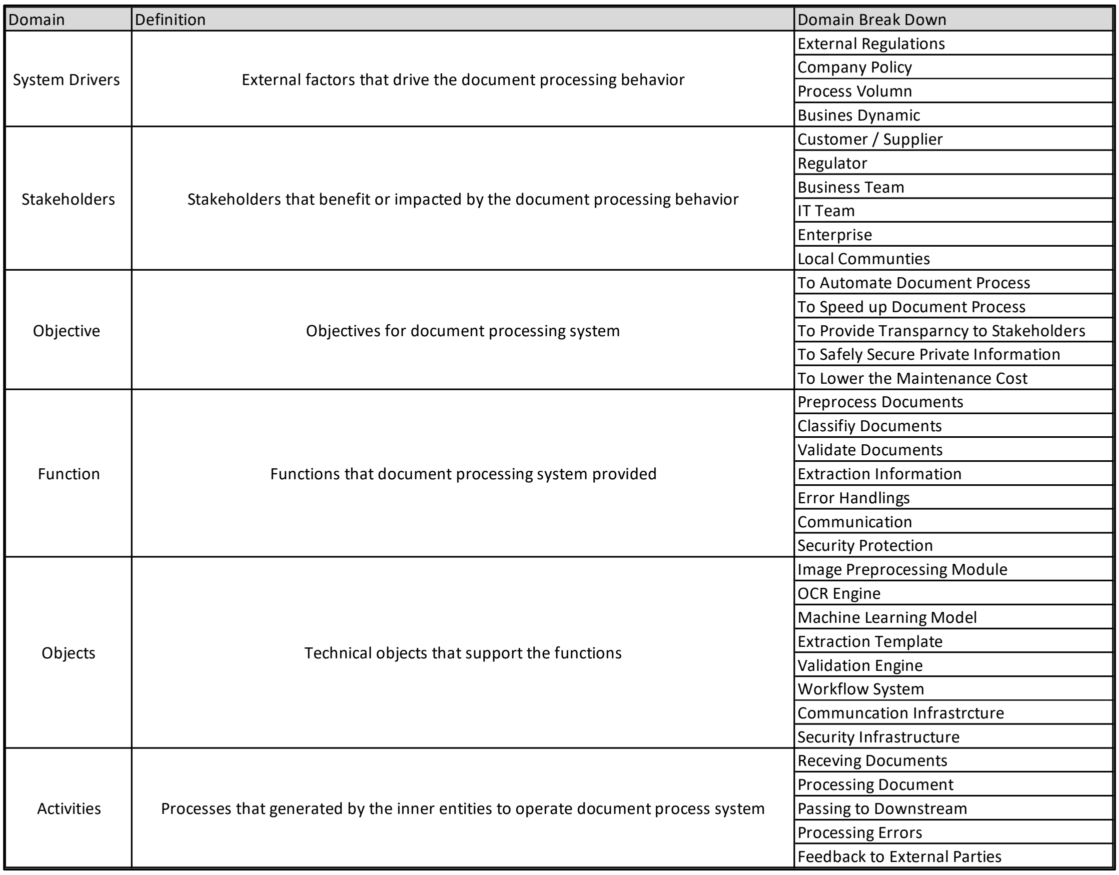
ES-MDM is a framework specialized in analyzing systems purpose developed by Bartolomei et al. (2009) [10]. The first step for constructing ES-MDM is to select the proper breakdown components for the six main domains in the system: System

Driver, Stakeholders, Objective, Function, Objects, and Activities. The definitions of six domains and their domain breakdown are listed in Table 1.

Since ES-MDM analyzes both social factors – system drivers and stakeholders, and engineering factors – functions and objects, it is a suitable tool to analyze document processing as a system. The six domains are broken down to the following sub-domains:

- **System Drivers:** External Regulations, Company Policy, Business Dynamic, and Process Volume and Categories
- **Stakeholders**: customers, suppliers, regulators, local communities, business team, IT team, and the enterprise are the stakeholder groups.
- **Objectives**: To Automate Document Process, To Speed up Document Process, To Lower the Maintenance Cost, To Provide Transparency to Stakeholders, and To Safely Secure Private Information are the objectives in the system.
- **Functions**: Preprocess Documents, Classify Documents, Validate Documents, Extract Information, Communication, Security Protection, and Error Handling.
- **Objects**: Image Preprocessing Module, Optical Character Recognition Engine, Machine Learning Model, Validation Engine, Workflow System, Communication Infrastructure, and Security Infrastructure.
- **Activities**: Receiving Documents, Processing Documents, Processing Errors, and Communicating to External Party.

Table 1: Sub-domains for ES-MDM to Analyze Document Processing



Domain	Definition	Domain Break Down
System Drivers	External factors that drive the document processing behavior	External Regulations
		Company Policy
		Process Volumn
		Busines Dynamic
Stakeholders	Stakeholders that benefit or impacted by the document processing behavior	Customer / Supplier
		Regulator
		Business Team
		IT Team
		Enterprise
		Local Communties
Objective	Objectives for document processing system	To Automate Document Process
		To Speed up Document Process
		To Provide Transparncy to Stakeholders
		To Safely Secure Private Information
		To Lower the Maintenance Cost
Function	Functions that document processing system provided	Preprocess Documents
		Classifiy Documents
		Validate Documents
		Extraction Information
		Error Handlings
		Communication
		Security Protection
Objects	Technical objects that support the functions	Image Preprocessing Module
		OCR Engine
		Machine Learning Model
		Extraction Template
		Validation Engine
		Workflow System
		Communcation Infrastrcture
		Security Infrastructure
Activities	Processes that generated by the inner entities to operate document process system	Receving Documents
		Processing Document
		Passing to Downstream
		Processing Errors
		Feedback to External Parties

3.5.1 ES-MDM Matrix

The final ES-MDM is a 76 * 76 matrix consists of break-down components from six domains. This matrix describes the interdependent relationship between entities. From looking at the degree of cluster of an object, the ES-MDM matrix shows that the process owners and classification model are two key factors in this system. The classification model links to almost all the technical objects across domains, and process owner is the driver of function. Another important factor is model owner. Without proper maintenance, the automating document process cannot be sustained. The screenshot ES-MDM is shown below, Figure 5, and the full picture is attached in Appendix B.

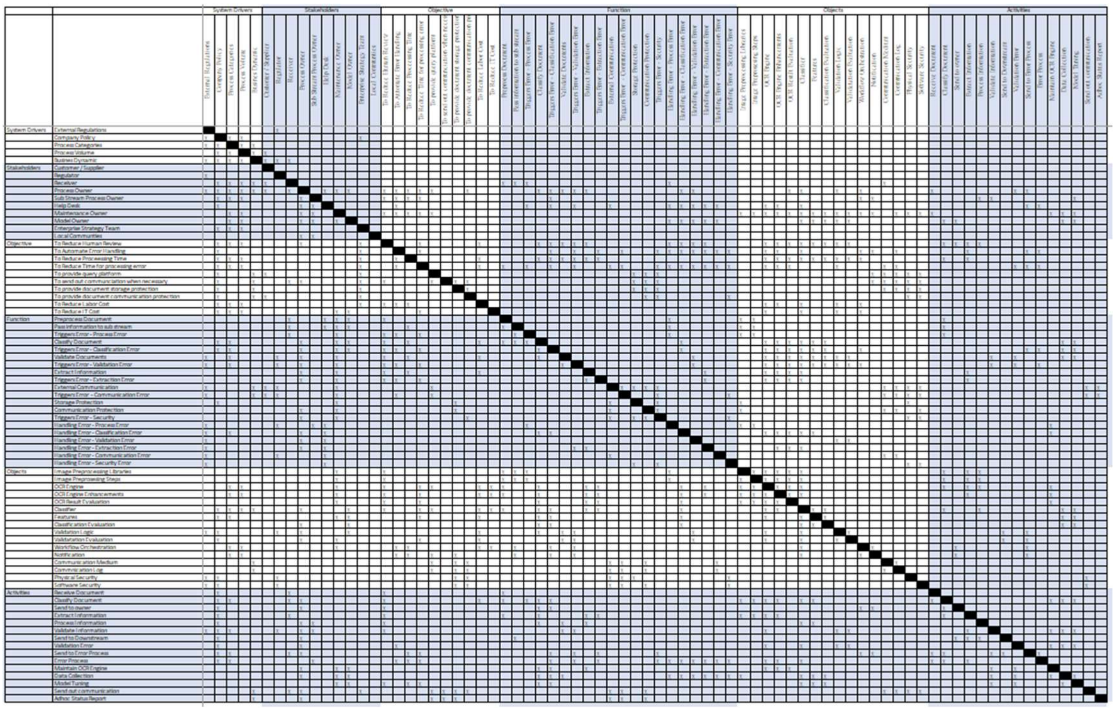


Figure 5. ES-MDM.
The cell value represents the binary relationship between the row and the column objects.

3.5.2 Network Representation and Analysis

An ES-MDM can be transformed into a network representation. A graph is a mathematical representation of connected nodes, which are the objects in the matrix. Nodes are connected by edges, which are the relationships in the matrix. This thesis transforms the ES-MDM to a directed network graph and uses betweenness centrality to identify the critical components in the system. Betweenness centrality measures how often a node is on the shortest path to connecting two other nodes. The network diagram is shown in Figure 6. The six nodes with the highest betweenness centrality are highlighted by red color.

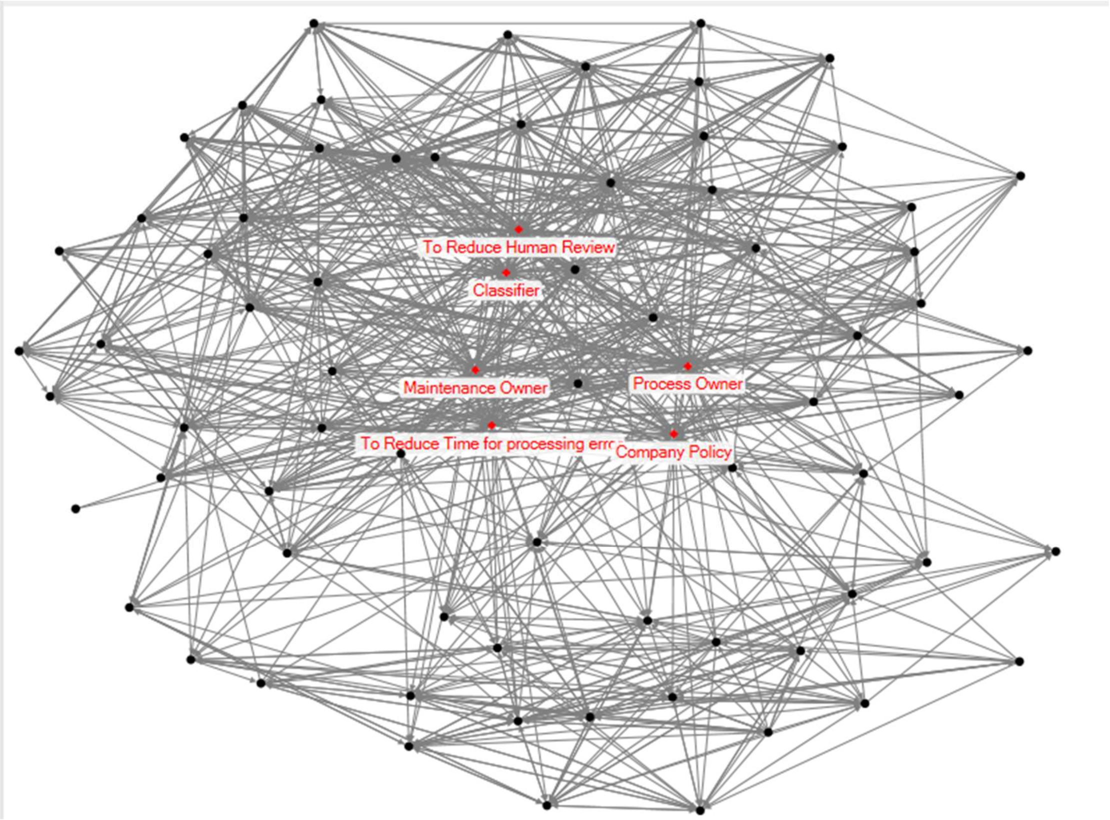


Figure 6. ES-MDM Network Representation.
The nodes (components) with the highest betweenness centrality are marked with red color.

The diagram shows that To Reduce Processing Time, Maintenance Owner, Process Owner, and Company Policy are the key factors for this network. It is rather surprising that company policy has such a significant interrelated relationship in this system. Its betweenness centrality is 333.053 based on calculation from NodeXL [61]. Usually, a high-level system driver would not have a comprehensive effect. For other nodes, Process and Maintenance owners are the key persons in this system. Two objectives, to Reduce Time for Processing Abnormal Cases and to Reduce Time for Processing Documents are other two critical areas from the analysis. Also, Classifier is a key technical object that will have a significant impact on the system.

Table 2: Top six nodes with highest Betweenness Centrality

Nodes (System Components)	Domain*	Betweenness Centrality
Process Owner	Stakeholder	678.522
Maintenance Owner	Stakeholder	470.903
Company Policy	System Driver	333.053
To Reduce Time for Processing Abnormal Cases	System Objective	287.349
To Reduce Time for Processing Documents	System Objective	203.700
Classifier	Technical Object	173.525

** Domain category was defined in section 3.5*

These identified nodes (system components) are heavily connected with other entities in the system. Any changes to them may cause a cascade effect. For example, changing a company policy may impact the process owners' operation and logic for Classifier. This thesis suggests finalizing these entities before introducing automatic document processing into organizations. Lastly, constructing ES-MDM is an iterative process. Companies that use this framework should continuously update the matrix based on the system situation.

3.6 Chapter Summary

This chapter presents a system analysis of automatic document processing, using a multi-step approach. In the first step, stakeholder analysis shows that enterprise and customer/supplier are the main stakeholders for this system as they have both needs and value exchange. In the second step, the OPD visualizes the document processing as a system. In the third step, the ES-MDM is created which describe the interdepend relationship between the components. In the fourth step, the network representation is created and shows the top six nodes with the highest betweenness centrality. All these nodes(components), Process Owner, Maintenance Owner, Company Policy, To Reduce Time for Processing Abnormal Cases, To Reduce Time for Processing

Documents, and Classifiers, can be considered as the key elements in the document processing system.

The next chapter proposes a stable and easy-to-train machine learning model to classify business documents. Having a stable and low-maintenance model addresses the two critical system components found in this chapter, To Reduce Time for Processing Abnormal Cases, and To Reduce Time for Processing Documents. The proposed model reaches a 0.872 F1-score, outperforming two other commonly used rule-based classifiers. The detail for the implementation and evaluation experiment is described in next chapter.

Chapter 4

4 Machine Learning Model

This chapter presents implementation details for the proposed machine learning model and the performance evaluation. The chapter describes details about the data source (RVL-CDIP), data transformation logic, program structure, proposed model, and evaluation metrics.

4.1 Overview

Classifying business documents is key to automating document center operations. Because of business complexity, a stable and easy to train classification model is desirable. With the advance of Natural Language Processing (NLP) techniques, recent embedding models can encode the semantic meaning into vector space. This enables the possibility to use embeddings to classify business documents with classical machine learning classifiers.

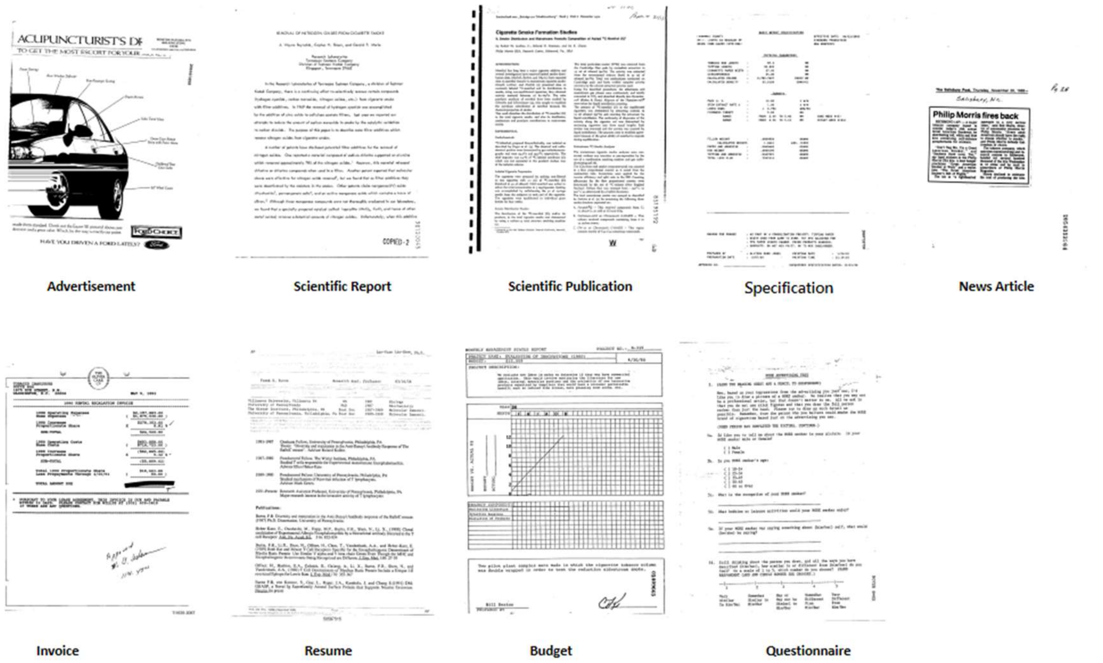
This research implements a machine learning model using SVM and Word2vec [48] to construct document embedding as features to classify business documents. In the business area, the same type of business documents conveys a similar meaning; therefore, this thesis proposes to use Word2vec to approximate the semantic meaning of the document and classify the document based on it.

The proposed model is compared with RIPPER [51] and PART [52] rule-based classifiers for the performance evaluation. RIPPER and PART are two commonly used rule-induction algorithms in various tasks. RIPPER and PART generate rules from the bag-of-words as document representation. RVL-CDIP [11], which is an image dataset from the tobacco industry, is the data used in performance evaluation in this work.

4.2 Data Preparation

RVL-CDIP (Ryerson Vision Lab Complex Document Information Processing) is an image dataset consisting of scanned business documents from the tobacco industry. RVL-CDIP contains 16 categories; each has 25,000 image files. The 16 categories are letter, form, email, handwritten, advertisement, scientific report, scientific publication, specification, file folder, news article, budget, invoice, presentation, questionnaire, resume, and memo. All the images are stored in grayscale, and the largest dimension does not exceed 1000 pixels.

For this research, letter, form, email, handwritten, file folder, presentation, questionnaire and memo categories are excluded from the dataset because those categories serve the general purpose, and do not convey specific business meaning. The sample images used in this research are shown in Figure 7.



Advertisement

Scientific Report

Scientific Publication

Specification

News Article

Invoice

Resume

Budget

Questionnaire

Figure 7. Sample Images Used in the Research.

In addition, some advertisement documents only contain images without text. An additional data transformation is applied to make sure the image contains enough text for analysis. The threshold value is 40 words in the OCR output. This data transformation keeps all the words, including noise, from OCR output to simulate the real scenario. Also, to make sure the data has a balanced distribution, the limit of the number of documents per category is 4000 records. The records are randomly sampled from the original RVL-CDIP dataset. The final data distribution is shown in Figure 8.

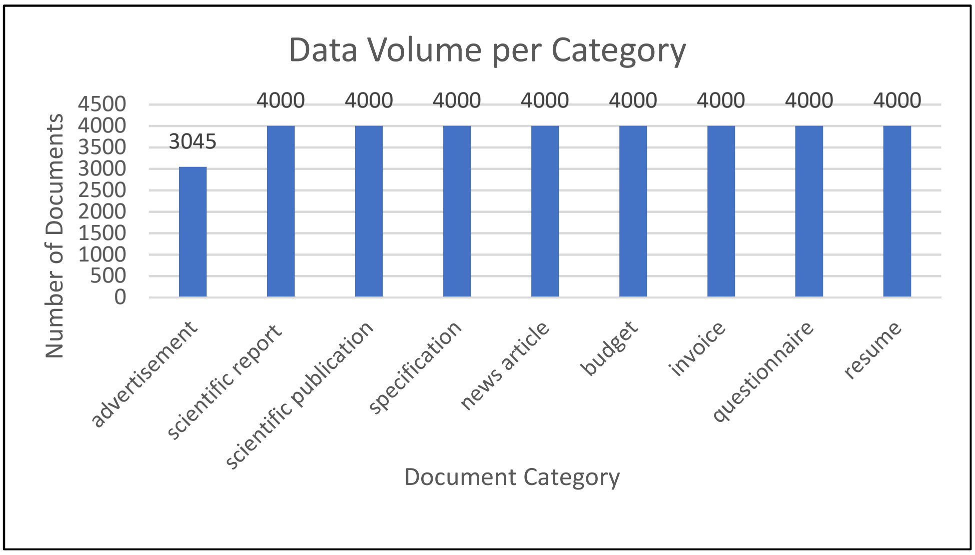


Figure 8. Data Volume per Category.
Data were randomly sampled from RVL-CDIP dataset with the transformation logic described in section 4.2.

4.3 Implementation Structure

The first step is to preprocess the images from the dataset. Tesseract OCR works best on images which have at least 300 dpi [62]. Additionally, there is some noise, such as ink dots, in the documents. Based on a discussion in Tesseract's online forum [63], Gaussian blur [64], Morphological Opening and Morphological Closed [65] are recommended for denoising images. Below image processing techniques are applied to have better OCR output performance:

- Rescale the image to at least 300 dpi
- Perform Gaussian blur with kernel size 5x5
- Perform Morphological Opening and Morphological Closed with kernel size 5x5

In addition, if the image's width is less than 1024 pixels after applied above preprocessing steps, the image is resized to 1024 pixels wide. Resize operation is for better visual observation to the input documents. OpenCV [66] is the main package used in the preprocessing stage.

The second step is the extraction process. Tesseract OCR is used to convert TIFF format to text files. The extracted text files are stored in a file folder for the program to access. py-tesseract is the main package in the extraction step. Text processing is also performed in this stage. Punctuation, numeric character, stop words, and short words are removed from the raw text. A filter is applied to exclude the words that are not actual English words to avoid the noise from OCR output. After all these processes, the raw text is ready for building the features. Gensim [67] is the main package for processing text.

The next step is to construct embeddings for the documents. Facebook's pre-trained Word2vec embedding is the basic model. The embedding is trained by Wikipedia pages and contains 300-dimensional vectors for 1 million vocabularies. In terms of constructing paragraph vectors, the program gets the sum of encodings for the document and divides it by the number of words in the documents. Gensim is the main package for building paragraph embedding.

For the rule-based feature, the porter-stemming is applied to the raw text. A bag-of-words model is constructed, and the document vector is based on the occurrence and non-occurrence of the words. The top 300 most frequent words are selected to represent the document vector. WEKA package is the main package in this stage.

The program splits the data into training and testing sets. An 80% and 20% ratio is used to split the data. The last step is to train the classifier and run the experiment.

The sklearn package is used for SVM implementation, and WEKA is used for rule-based classifier implementation. The result is evaluated in Precision, Recall and F measure. The overall implementation structure is depicted at Figure 9.

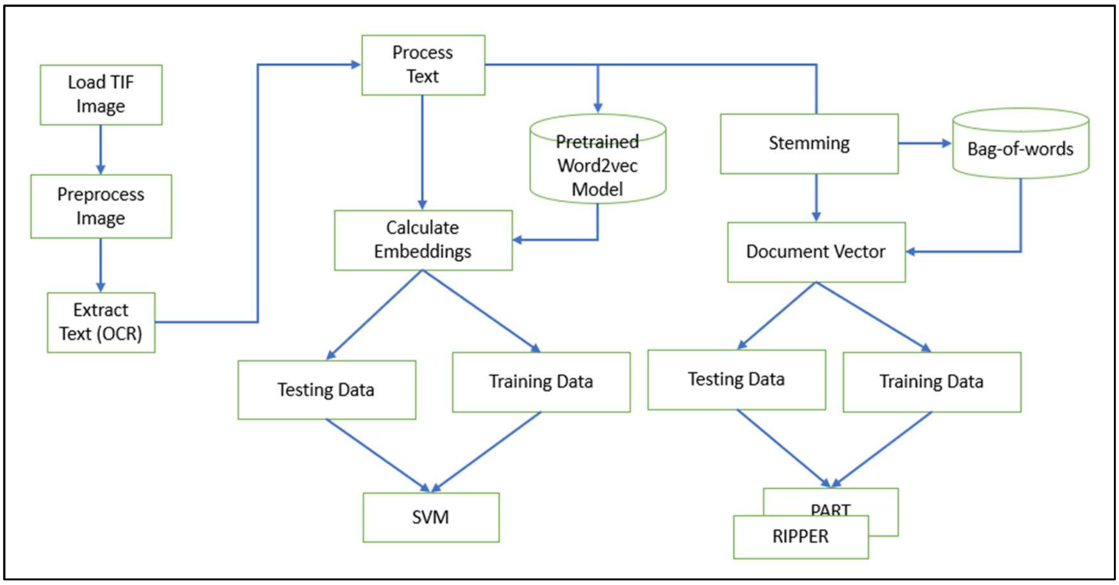


Figure 9. Implementation Structure.
The proposed model uses Word2vec to approximate the document embeddings; the rule base classifiers use bag-of-words model to construct document vectors.

4.4 Optical Character Recognition

Optical Character Recognition (OCR) is the key technology to automate document processing. In this thesis, Tesseract OCR is selected as the backend engine. From the literature review, the performance of Tesseract OCR is comparable to the commercial OCR packages [43] and is widely used in industry applications. In this work, the parameters for the OCR engine are set up as “config = --oem 1 --psm 3 --l eng”. The parameter explanation is listed below:

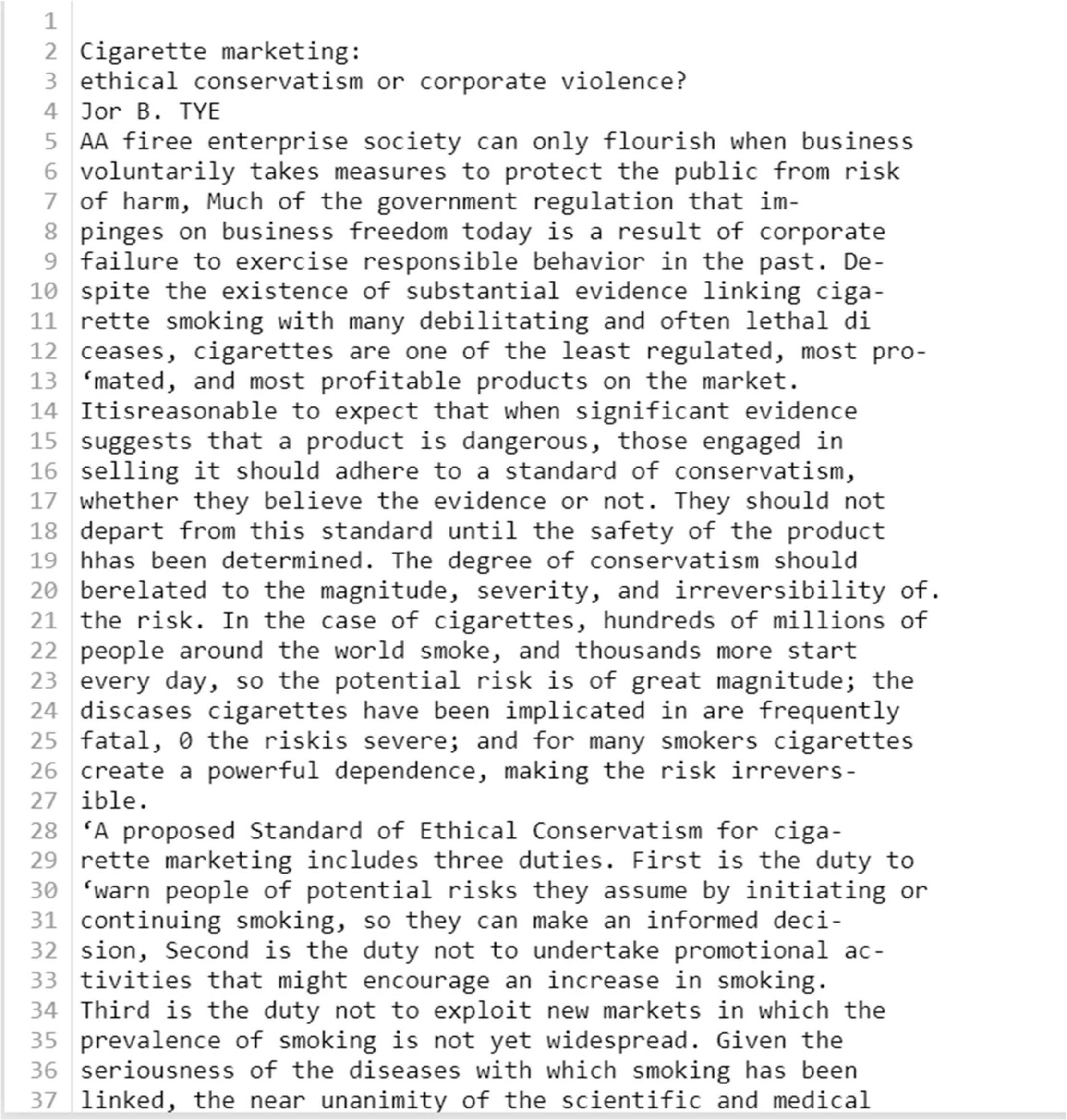
- **Engine Mode** (--oem) : Tesseract has several engine modes which are trained by different models. The latest Tesseract version added a new engine which uses the

LSTM model. –oem 1 enables Tesseract to use the LSTM engine to extract the text

- **Page Segmentation Mode** (--psm)**:** The page segmentation parameter affects how Tesseract separates lines and text. --psm 3 represents the auto detection for page segmentation which suits scanned documents.
- **Language** (--l) – This parameter specifies the language of the input document. English is the target language in this thesis.

Finally, to efficiently process the files, the program can process multiple files in a single directory. All the documents are processed by alphabet order. Multiple-page PDFs are allowed. The program assigns a sequence number to each page and reconstructs the document content based on the ID number. The sample result is shown in Figure 10.

.



```

Cigarette marketing:
ethical conservatism or corporate violence?
Jor B. TYE
AA firee enterprise society can only flourish when business
voluntarily takes measures to protect the public from risk
of harm, Much of the government regulation that im-
pinges on business freedom today is a result of corporate
failure to exercise responsible behavior in the past. De-
spite the existence of substantial evidence linking ciga-
rette smoking with many debilitating and often lethal di
ceases, cigarettes are one of the least regulated, most pro-
‘mated, and most profitable products on the market.
Itisreasonable to expect that when significant evidence
suggests that a product is dangerous, those engaged in
selling it should adhere to a standard of conservatism,
whether they believe the evidence or not. They should not
depart from this standard until the safety of the product
hhas been determined. The degree of conservatism should
berelated to the magnitude, severity, and irreversibility of.
the risk. In the case of cigarettes, hundreds of millions of
people around the world smoke, and thousands more start
every day, so the potential risk is of great magnitude; the
discases cigarettes have been implicated in are frequently
fatal, 0 the riskis severe; and for many smokers cigarettes
create a powerful dependence, making the risk irrevers-
ible.
‘A proposed Standard of Ethical Conservatism for ciga-
rette marketing includes three duties. First is the duty to
‘warn people of potential risks they assume by initiating or
continuing smoking, so they can make an informed deci-
sion, Second is the duty not to undertake promotional ac-
tivities that might encourage an increase in smoking.
Third is the duty not to exploit new markets in which the
prevalence of smoking is not yet widespread. Given the
seriousness of the diseases with which smoking has been
linked, the near unanimity of the scientific and medical
```

Figure 10. Sample OCR Result from a news article in the dataset.

4.5 Document Representation

This research uses Facebook pre-trained word embedding. The word embedding has 300 dimensions and is trained by Wikipedia text corpus. In this research, the document vector is represented by the average embedding for all the words in the document. Assume a document D consists of a total of n words $w_1, w_2, \ldots, w_n$. Each word has a word2vec embedding $E_{w1}, E_{w2}, \ldots, E_{wn}$. The document vector is represented as

$$E_D = \frac{1}{n}\sum_{i=1}^{n} E_{wi}$$

This research also performs an experiment to evaluate the impact of having incremental training on pre-trained embeddings. The result shows that the model performance is improved after conducting the incremental training to pre-trained Word2vec embedding; however, based on the literature search, it is not recommended to perform incremental training on the pre-trained word2vec model [68]. The evaluation result is described in section 4.10.2.

For the rule-based algorithms, the document is presented as a 300-dimension vector based on bag-of-words style. Each dimension is a word with possible values 1 and 0 by occurrence and non-occurrence of the words in the document. The top 300 most frequently used words are selected to construct vector.

4.6 Support Vector Machine

The sklearn package is used for SVM implementation. One-vs-rest parameter is used; this parameter enables the SVM to achieve multi-category classification. For the Penalty parameter, L2 parameter is selected. The L2 parameter changes the objective function's regularization term and makes the classifier less sensitive to outliers. For the loss function, hinge square is used. Value of Class_Weight is None since the data category is balanced. Random State parameter does not have any impact on multi-category. The maximum iteration is 1000, and all the training does not reach the maximum iteration. Lastly, the GridSearch function is used to optimize the value for parameter C with a range from 10^{-1} to 10^{1} and kernel function. The full parameter table is shown in Table 3.

Table 3: Implementation Parameters for SVM

Parameters	Parameter Value
Penalty	L2
Loss	Square Hinge
Dual	False
Tol	1e-4
C	Optimized by GridSearchCV function; value range from 10^{-1} to 10^{1}
Kernel	Optimized by GridSearchCV function, Value range [linear, rbf, poly]
multi_class	OVR
fit_intercept	True
intercept_scaling	1
class_weight	None
random_state	None
max_iter	1000

4.7 Rule-Based Classifier

For the rule-based classifier, WEKA's implementation of RIPPER and PART is used to generate the rules. The feature is a 300-dimension vector based on bag-of-words model. For RIPPER, the parameters are set up as minimal_weights_of_instance 2.0, number_of_optmization 2, and pruning True. For PART algorithm, pruning confidence is 0.25. The smaller the confidence, the more pruning generated. The batch size is 100, and minimum number of objects is 2. The full parameter table is shown in Table 4.

Table 4: Implementation Parameters for RIPPER and PART

Algorithm	Parameters	Parameter Value
RIPPER	weights_of_instance	2.0
RIPPER	number_of_optmization	2
RIPPER	Pruning	True
PART	pruning confidence	0.25
PART	minimum number	2

4.7.1 RIPPER Algorithm

RIPPER (Repeated Incremental Pruning to Produce Error Reduction) algorithm has two stages: grow and prune. RIPPER grows the best rules from training set using Sequential Covering algorithm, which continuously builds the rules until all data are associated with a specific rule. The algorithm uses the testing set to calculate the threshold value to determine whether the rule should be removed. Lastly, the global optimization steps are performed.

In this experiment, the document vector is represented as a 300-dimension vector by occurrence and non-occurrence of words. Each dimension is a word with 1 or 0 value. The rules generated for Invoice class is shown in Appendix B. The interesting finding is that “invoic”, which is the stem result from “invoice”, is a strong word that can classify invoice documents. The first rules using “invoic” covers 2234 records with only 90 records incorrectly classified.

4.7.2 PART Algorithm

PART is an algorithm that builds the rules without global optimization. It was developed by Frank and Witten. PART adopts a divide and conquer strategy by building a rule, removing the data covered by the rule, and creating rules until no data are left. PART generates rules by a partial C4.5 decision tree which expands the tree based on the estimated information gained. The leaf with the largest coverage is made as a rule. In this experiment, the documents are represented as a 300-dimension vector. Each word is a feature dimension. The selected rules generated for resume category are shown in Appendix C. In resume category, “univers” is a strong word to classify resume documents. “univers” is the stem result of university, which is a commonly used word in the resume.

4.8 Evaluation

Precision, Recall, and F-measure are used for performance evaluation. The proposed model is compared to RIPPER and PART, two commonly used rule-based classifiers, using RVL-CDIP dataset. The document categories used in the evaluation are advertisement, scientific report, scientific publication, specification, news article, budget, invoice, questionnaire, and resume from the dataset. Precision is defined as the number of true positives divided by the sum of true positives and false positives. The formal representation is below:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$where\ TP\ is\ True\ Positive\ and\ FP\ is\ False\ Positive$$

Recall is defined as true positive divided by sum of true positive and false negative:

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$where\ TP\ is\ Ture\ Positive\ and\ FN\ is\ False\ Negative$$

F1-score is defined as two times of precision times recall and divided by precision and recall:

$$F_1 = \frac{2\ (TP)}{FP + FN + 2(TP)}$$

$$where\ TP\ is\ Ture\ Positive, FP\ is\ False\ Positive\ and\ FN\ is\ False\ Negative$$

Since this classifier is a multi-class classifier, the Macro F1-score is used to evaluate the overall model performance. Marco F1-score is the unweighted mean of all the categories. This thesis selected the Macro F1-score because the dataset is balanced, and all the categories are equally important to the measurement.

4.9 Result

This thesis compares the proposed model, which uses SVM as the classifier and document embedding to represent the feature, to the other two commonly used rule-based classifiers, RIPPER and PART.

Figure 11 is the F1-score for the proposed model in all the categories. The model has the weakest performance in news article category, which is 0.84. The potential reason is that news article has wide text usage and does not have concentrated document embeddings.

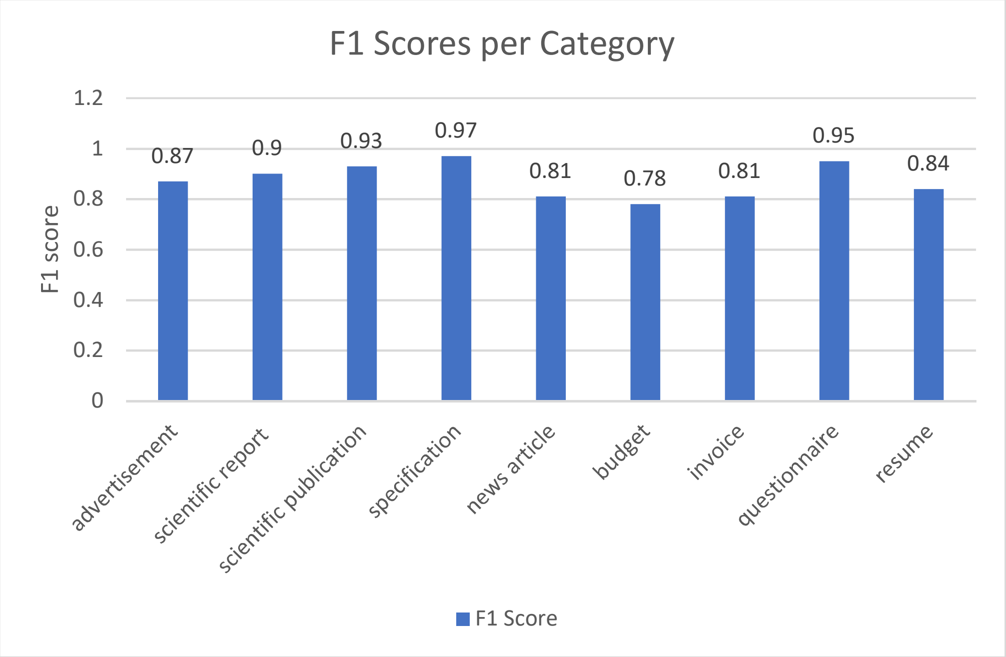


Figure 11. F1-score per Document Category for the Proposed Model. The proposed model has rather poor performance in news article.

Figure 12 is the F1-score comparison between the proposed model and the other two rule-based models. In the resume category, rule-based classifiers outperform the proposed model. The potential reason is that resume category has several distinct words such as “univers”, which is the stem result of “university”, and “biograph”, which is a stem result from “biography”. These two words are commonly used in resume writing. The generated rules for resume categories are listed in Appendix D. Most of the rules contain “univers” and “biograph” as the criteria.

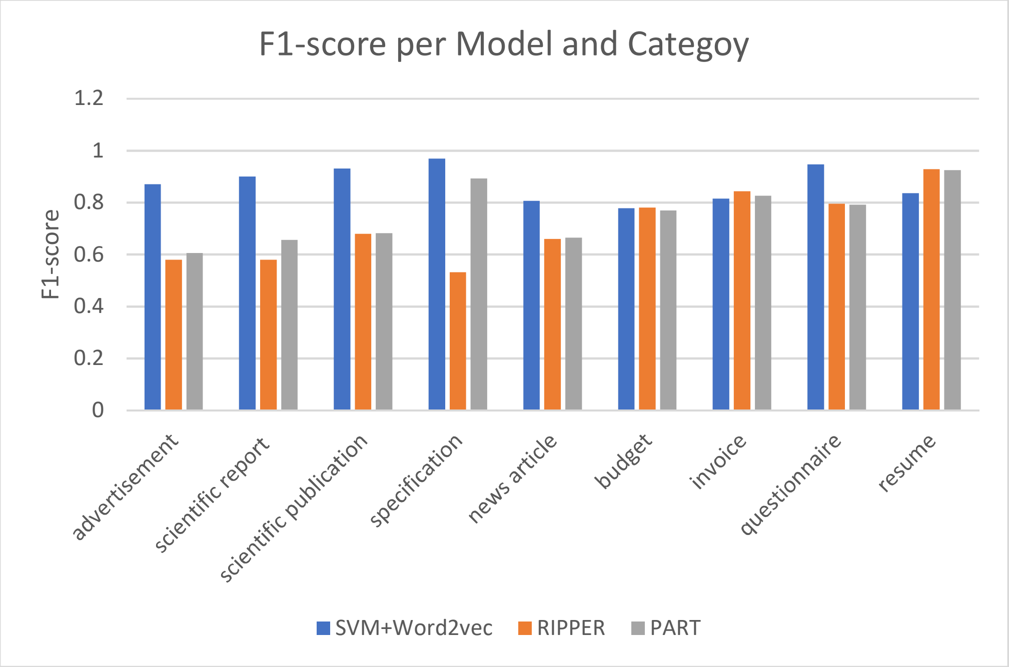


Figure 12. F1-score Comparison between SVM, RIPPER and PART

The overall model performance is described in Table 5. F1-scores for the proposed model, RIPPER, and PART are 0.872, 0.708, and 0.756. This thesis uses Marco average score to evaluate the performance. The proposed SVM model with Word2vec feature representation achieves better precision, and recall, reaching 0.873 and 0.872.

Table 5: Performance Matrix for SVM, RIPPER and PART

	Avg. Precision	Avg. Recall	F-Measure
SVM	0.873	0.872	0.872
RIPPER	0.773	0.696	0.708
PART	0.756	0.757	0.756

* *Marco Average is adopted*

4.10 Additional Discussion

4.10.1 OneR Algorithm

In this thesis, one additional classifier, OneR, was implemented. OneR builds a classification rule based on the frequency table to all the possible categories for each feature and selects the rule with a minimum error rate. OneR is known for its simplicity and stable performance. OneR is usually chosen as the baseline model.

In this experiment, the word occurrence and non-occurrence are used to represent the document vector. Each word becomes a dimension of feature and has only two values, 1 and 0. In this experiment, OneR can only classify two categories, specification and resume. The generated rules are listed in Appendix E. The model output is shown in Figure 13.

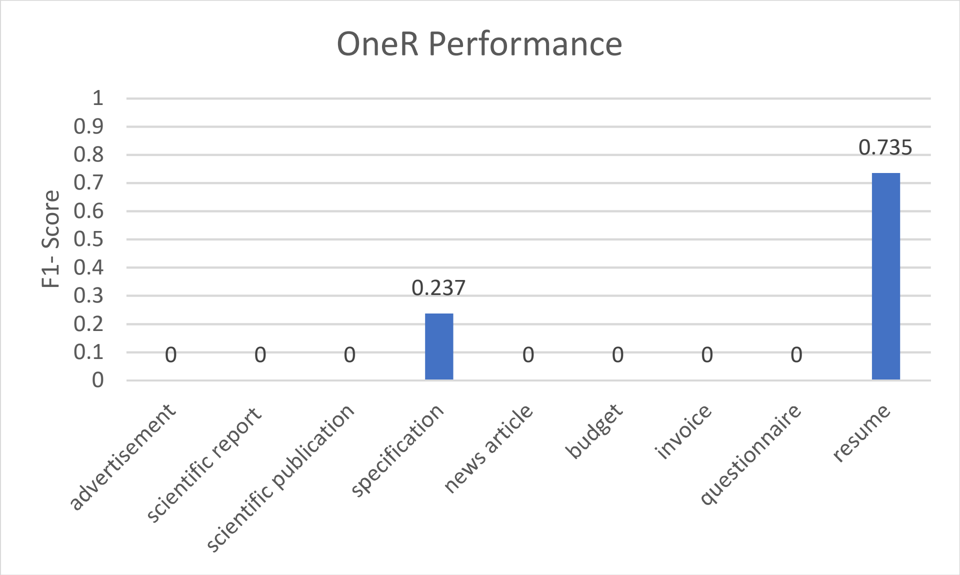


Figure 13. OneR Performance Result.

4.10.2 Incremental Training for Pre-trained Embeddings

This thesis also completed an experiment about whether incremental training to the pre-trained Word2vec model can improve the performance. In this work, the Word2vec embedding is fine-tuned by the corpus collected from the documents used in the evaluation. The result shows that there is a performance improvement by using the tuned embeddings. The changes for similarity in embeddings are described in Table 6. The performance comparison between original pre-trained embeddings and incremental training embeddings is described in Table 7.

Table 6: Similarity Result after Incremental Training to the model

Word Similarity	Pre-trained Embeddings	Pre-trained Embeddings with incremental Training
"invoice" and "tobacco "	0.27182385	0.30306637
“questionnaire” and “tobacco”	0.28011847	0.31092876
“news” and “tobacco”	0.36573988	0.38157657

* After incremental training, the frequent used words move closer to “tobacco”.

Table 7: Performance Comparison for incremental training.

Category	F1-score	
	Pre-trained Embeddings without Incremental Training	Pre-trained Embeddings with Incremental Training
advertisement	0.8	0.87
scientific report	0.83	0.90
scientific publication	0.91	0.93
specification	0.95	0.97
news article	0.75	0.81
budget	0.72	0.78
invoice	0.78	0.81
questionnaire	0.94	0.95
resume	0.8	0.84

4.11 Chapter Summary

This chapter describes the performance evaluation for the proposed machine learning model with two other rule-based classifiers using RVL-CDIP dataset. The proposed model reaches a 0.872 F1-score, outperforming RIPPER and PART algorithms. In the additional experiment, OneR algorithm with the bag-of-words feature shows poor performance. The incremental training for pre-trained Word2vec increases the proposed model's F1-score from 0.831 to 0.872.

This result shows that our proposed model, using SVM and Word2vec, can effectively classify business documents from scanned documents. As mentioned in chapter2, a stable and robust classifier is a key element to automate document processing. The proposed model could be a suitable machine learning classifier used in business environments. The next chapter summarizes the findings and revisits the research objectives.

Chapter 5

5 Summary and Future Work

This chapter summarizes the present research, discusses the findings from the system analysis and performance evaluation, and makes recommendations for future research.

5.1 Summary

Due to the COVID-19 pandemic, the need for automatic document processing is increased as fewer employees works in the office. It is important for businesses to understand the challenges while implementing an AI-based automatic document processing and a performance benchmark of a machine learning classifier.

To understand the challenges, this thesis presents a system analysis by viewing document processing in context of the sociotechnical system. The stakeholder analysis shows that the enterprise and customers are the two key stakeholder groups. The OPM defines the system boundary, and Process Owner and Machine Learning Classifiers are two major components in the diagram. Lastly, the network representation shows that Process Owner, Maintenance Owner, Company Policy, To Reduce Time for Processing Abnormal Cases, To Reduce Time for Processing Documents, and Classifier are critical system components in this system as they have the highest betweenness centrality.

A machine learning classification model has been implemented using SVM and Word2vec to classify business documents. In the experiment, the proposed model reached a 0.872 Marco F1-score. The other useful finding is that if the document category has unique keywords, rule-based classifiers can also provide good performance.

These findings provide the answers to the research questions in section 1.2. First, the critical system components are identified through the system analysis. The most valuable finding is that deploying an AI-based document processing system is not only a technical project but also an organizational transformation project. This finding is consistent with the literature review that a successful AI application deployment needs comprehensive planning on the organizational level [69]. Second, the proposed model outperforms two commonly used rule-based algorithms and can potentially be deployed into businesses to classify documents.

5.2 Limitation and Future Work

To expand this work to other types of documents and business organizations, a case study should be conducted. A case study can validate our system analysis result and improve the analysis model. For the machine learning classifier, the future work can explore to use the language models to approximate the document meaning.

Due to time limitations of the research, the system analysis is conducted based on the author’s relevant working experience and online case study data. Future research could conduct a complete case interview with the business owner and IT department to validate the system analysis results from this work. Such a deeper discussion could explore what is the best way to integrate the development process into the organization.

The other limitation is that the model performance evaluation is based on scanned business documents from the tobacco industry. It is recommended to perform the evaluation with the scanned documents from other industries, such as financial institutions and legal institutions, to improve the model.

It is also recommended that future research explore using language models, such as BERT [70], to classify business documents. Language models consider the context information more than the word embeddings and achieve superior results s in various tasks. Using document embeddings generated from language models could potentially achieve better performance in classifying business documents.

Reference

[1] M. Kobayashi-Hillary, Outsourcing to India: The Offshore Advantage, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2005. doi: 10.1007/b106280.

[2] W.P. Hsu, “Intelligent Document Recognition on Financial Process Automation,” in 2020 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), Aug. 2020, pp. 1–1. doi: 10.1109/VLSI-DAT49148.2020.9196318.

[3] S. Somjai, “Advantages and disadvantages of outsourcing,” The Business and Management Review, vol. 9, no. 1, p. 4, 2017.

[4]J. Fulcher, “A Feasibility Study on the Implementation of a Digital Mailroom for UTi,” 2015, Accessed: May 18, 2021. [Online]. Available: https://repository.up.ac.za/handle/2263/52845

[5] X. Ling, M. Gao, and D. Wang, “Intelligent document processing based on RPA and machine learning,” in 2020 Chinese Automation Congress (CAC), Nov. 2020, pp. 1349–1353. doi: 10.1109/CAC51589.2020.9326579.

[6] T. XU, Y. ZHANG, X. WU, and W. MING, “Intelligent Document Processing : Automate Business with Fluid Workflow,” Konica Minolta technology report, vol. 18, pp. 89–94, Jan. 2021.

[7] J. Wu, “ModelOps Is The Key To Enterprise AI,” Forbes. https://www.forbes.com/sites/cognitiveworld/2020/03/31/modelops-is-the-key-to-enterprise-ai/ (accessed May 18, 2021).

[8] ModelOp, "ModelOps Essentials," [Online]. Available: https://www.modelop.com/wp-content/uploads/2020/05/ModelOps_Essential_Guide.pdf. [Accessed 13 05 2021].

[9] Dov Dori, Object-Process Methodology - A Holistic Systems Paradigm. Springer, Berlin, Heidelberg. Accessed: May 18, 2021. [Online]. Available: https://doi.org/10.1007/978-3-642-56209-9

[10] J. E. Bartolomei, D. E. Hastings, R. de Neufville, and D. H. Rhodes, "Engineering Systems Multiple-Domain Matrix: An organizing framework for modeling large-scale complex systems," Systems Engineering, vol. 15, no. 1, pp. 41–61, 2012, doi: https://doi.org/10.1002/sys.20193.

[11] A. W. Harley, A. Ufkes, and K. G. Derpanis, "Evaluation of Deep Convolutional Nets for Document Image Classification and Retrieval," arXiv:1502.07058 [cs], Feb. 2015, Accessed: May 18, 2021. [Online]. Available: http://arxiv.org/abs/1502.07058

[12] R. Smith, "An Overview of the Tesseract OCR Engine," in Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Sep. 2007, vol. 2, pp. 629–633. doi: 10.1109/ICDAR.2007.4376991.

[13] J. Cardenosa and J. A. Espinosa, "Document classification intelligent system in complex organizations," in 1997 IEEE International Conference on Intelligent Processing Systems (Cat. No.97TH8335), Oct. 1997, vol. 2, pp. 1885–1888 vol.2. doi: 10.1109/ICIPS.1997.669385.

[14] J. T. L. Wang and P. A. Ng, "Texpros: an intelligent document processing system," in The Impact of Case Technology on Software Processes, vol. Volume 3, 0 vols., WORLD SCIENTIFIC, 1994, pp. 103–135. doi: 10.1142/9789812798053_0006.

[15] B. Vajgel et al., "Development of Intelligent Robotic Process Automation: A Utility Case Study in Brazil," IEEE Access, pp. 1–1, 2021, doi: 10.1109/ACCESS.2021.3075693.

[16] L. Baier, F. Jöhren, and S. Seebacher, “Challenges in Deploying Machine Learning: a Survey of Case Studies,” May 2019.

[17] “Key challenges of Intelligent Document Processing - KPMG Belgium,” KPMG, Sep. 07, 2020. https://home.kpmg/be/en/home/insights/2020/09/ta-key-challenges-of-intelligent-document-processing.html (accessed May 18, 2021).

[18] W. Pasmore, S. Winby, S. A. Mohrman, and R. Vanasse, “Reflections: Sociotechnical Systems Design and Organization Change,” Journal of Change Management, vol. 19, no. 2, pp. 67–85, Apr. 2019, doi: 10.1080/14697017.2018.1553761.

[19] R. Oosthuizen and M. C. Van ’t Wout, “Sociotechnical system perspective on artificial intelligence implementation for a modern intelligence system,” Oct. 2019, Accessed: May 18, 2021. [Online]. Available: https://researchspace.csir.co.za/dspace/handle/10204/11347

[20] G. J. M. Read, P. M. Salmon, N. Goode, and M. G. Lenné, “A sociotechnical design toolkit for bridging the gap between systems-based analyses and system design,” Human Factors and Ergonomics in Manufacturing & Service Industries, vol. 28, no. 6, pp. 327–341, 2018, doi: https://doi.org/10.1002/hfm.20769.

[21] M. Rajkumar, A. K. Pole, V. S. Adige, and P. Mahanta, “DevOps culture and its impact on cloud delivery and software development,” in 2016 International Conference on Advances in Computing, Communication, Automation (ICACCA) (Spring), Apr. 2016, pp. 1–6. doi: 10.1109/ICACCA.2016.7578902.

[22]B. B. N. de França, H. Jeronimo, and G. H. Travassos, “Characterizing DevOps by Hearing Multiple Voices,” in Proceedings of the 30th Brazilian Symposium on Software Engineering, New York, NY, USA, Sep. 2016, pp. 53–62. doi: 10.1145/2973839.2973845.

[23] I. Bider and V. Klyukina, “Using a Socio-Technical Systems Approach for a Sales Process Improvement,” in 2018 IEEE 22nd International Enterprise Distributed Object Computing Workshop (EDOCW), Oct. 2018, pp. 48–58. doi: 10.1109/EDOCW.2018.00019.

[24] B. Behdani, “Evaluation of paradigms for modeling supply chains as complex socio-technical systems,” in Proceedings of the 2012 Winter Simulation Conference (WSC), Dec. 2012, pp. 1–15. doi: 10.1109/WSC.2012.6465109.

[25] A. M. Madni, M. Spraragen, and C. C. Madni, “Exploring and assessing complex systems’ behavior through model-driven storytelling,” in 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Oct. 2014, pp. 1008–1013. doi: 10.1109/SMC.2014.6974045.

[26] ISO, “ISO/PAS 19450:2015 - Automation systems and integration — Object-Process Methodology.” https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/22/62274.html (accessed May 18, 2021).

[27] Y. Grobshtein, V. Perelman, E. Safra, and D. Dori, “Systems Modeling Languages: OPM Versus SysML,” in 2007 International Conference on Systems Engineering and Modeling, Mar. 2007, pp. 102–109. doi: 10.1109/ICSEM.2007.373339.

[28] I. Reinhartz-Berger and D. Dori, “OPM vs. UML--Experimenting with Comprehension and Construction of Web Application Models,” Empirical Software Engineering, vol. 10, no. 1, pp. 57–80, Jan. 2005, doi: 10.1023/B:EMSE.0000048323.40484.e0.

[29] K. Hiekata, S. Wanaka, T. Mitsuyuki, R. Ueno, R. Wada, and B. Moser, “Systems analysis for deployment of internet of things (IoT) in the maritime industry,” J Mar Sci Technol, Jul. 2020, doi: 10.1007/s00773-020-00750-5.

[30]Y. Mordecai, “Conceptual Modeling of Cyber-Physical Gaps in Air Traffic Control,” Procedia Computer Science, vol. 140, pp. 21–28, Jan. 2018, doi: 10.1016/j.procs.2018.10.288.

[31] J. M. Casebolt, A. Jbara, and D. Dori, “Business process improvement using Object-Process Methodology,” Systems Engineering, vol. 23, no. 1, pp. 36–48, 2020, doi: https://doi.org/10.1002/sys.21499.

[32] T. Wang, F. Yang, Y. Zhu, X. Li, and X. Zhou, “Extending the Object-Process Methodology to Big Data Systems,” DEStech Transactions on Computer Science and Engineering, vol. 0, no. msota, Art. no. msota, 2018, doi: 10.12783/dtcse/msota2018/27545.

[33] Y. Mordecai, O. L. de Weck, and E. F. Crawley, “Towards an Enterprise Architecture for a Digital Systems Engineering Ecosystem,” presented at the Conference on Systems Engineering Research, 2020, Oct. 2020.

[34] J. E. Bartolomei, “Qualitative knowledge construction for engineering systems : extending the design structure matrix methodology in scope and procedure,” Thesis, Massachusetts Institute of Technology, 2007. Accessed: May 18, 2021. [Online]. Available: https://dspace.mit.edu/handle/1721.1/43855

[35] S. Okami and N. Kohtake, “Modeling and analysis of health-information system of systems for managing transitional complexity using engineering systems multiple-domain matrix,” in 2017 Annual IEEE International Systems Conference (SysCon), Apr. 2017, pp. 1–8. doi: 10.1109/SYSCON.2017.7934805.

[36]F. Alkhaldi and A. Alouani, “Development of a Generic Model for Large-Scale Healthcare Organizations,” in 2019 IEEE 19th International Symposium on High Assurance Systems Engineering (HASE), Jan. 2019, pp. 200–207. doi: 10.1109/HASE.2019.00038.

[37] M. J. Songhori, A. M. Leo van Dongen, and M. Rajabalinejad, “A Multi-domain Approach Toward Adaptations of Socio-technical Systems: The Dutch Railway Case-Part 1,” in 2020 IEEE 15th International Conference of System of Systems Engineering (SoSE), Jun. 2020, pp. 99–104. doi: 10.1109/SoSE50414.2020.9130512.

[38] M. J. Songhori, L. A. M. van Dongen, and M. Rajabalinejad, “A Multi-domain Approach Toward Adaptations of Socio-technical Systems: The Dutch Railway Case-Part 2,” in 2020 IEEE 15th International Conference of System of Systems Engineering (SoSE), Jun. 2020, pp. 223–228. doi: 10.1109/SoSE50414.2020.9130531.

[39]J. Mantas, “An overview of character recognition methodologies,” Pattern Recognition, vol. 19, no. 6, pp. 425–430, Jan. 1986, doi: 10.1016/0031-3203(86)90040-3.

[40] S. Mori, C. Y. Suen, and K. Yamamoto, “Historical review of OCR research and development,” Proceedings of the IEEE, vol. 80, no. 7, pp. 1029–1058, Jul. 1992, doi: 10.1109/5.156468.

[41] A. Chaudhuri, K. Mandaviya, P. Badelia, and S. K. Ghosh, “Optical Character Recognition Systems,” in Optical Character Recognition Systems for Different Languages with Soft Computing, A. Chaudhuri, K. Mandaviya, P. Badelia, and S. K Ghosh, Eds. Cham: Springer International Publishing, 2017, pp. 9–41. doi: 10.1007/978-3-319-50252-6_2. https://www.springer.com/gp/book/9783319502519 (accessed May 18, 2021).

[42]M. Namysl and I. Konya, “Efficient, Lexicon-Free OCR using Deep Learning,” arXiv:1906.01969 [cs], Jun. 2019, Accessed: May 18, 2021. [Online]. Available: http://arxiv.org/abs/1906.01969

[43] A. P. Tafti, A. Baghaie, M. Assefi, H. R. Arabnia, Z. Yu, and P. Peissig, “OCR as a Service: An Experimental Evaluation of Google Docs OCR, Tesseract, ABBYY

FineReader, and Transym," in Advances in Visual Computing, Cham, 2016, pp. 735–746. doi: 10.1007/978-3-319-50835-1_66.

[44] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," IEEE Transactions on Neural Networks, vol. 13, no. 2, pp. 415–425, Mar. 2002, doi: 10.1109/72.991427.

[45]F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, no. 85, pp. 2825–2830, 2011.

[46]A. Patle and D. S. Chouhan, "SVM kernel functions for classification," in 2013 International Conference on Advances in Technology and Engineering (ICATE), Jan. 2013, pp. 1–9. doi: 10.1109/ICAdTE.2013.6524743.

[47]I. Kaibi, E. H. Nfaoui, and H. Satori, "A Comparative Evaluation of Word Embeddings Techniques for Twitter Sentiment Analysis," in 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), Apr. 2019, pp. 1–4. doi: 10.1109/WITS.2019.8723864.

[48]T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," arXiv:1310.4546 [cs, stat], Oct. 2013, Accessed: Apr. 28, 2021. [Online]. Available: http://arxiv.org/abs/1310.4546

[49]"machine learning - Does sum of embeddings make sense?," Data Science Stack Exchange. https://datascience.stackexchange.com/questions/44635/does-sum-of-embeddings-make-sense (accessed Apr. 30, 2021).

[50]R. C. Holte, "Very Simple Classification Rules Perform Well on Most Commonly Used Datasets," Machine Learning, vol. 11, no. 1, pp. 63–90, Apr. 1993, doi: 10.1023/A:1022631118932.

[51] W. W. Cohen, "Fast Effective Rule Induction," in In Proceedings of the Twelfth International Conference on Machine Learning, 1995, pp. 115–123.

[52] E. Frank and I. H. Witten, “Generating Accurate Rule Sets Without Global Optimization,” in Proceedings of the Fifteenth International Conference on Machine Learning, San Francisco, CA, USA, Jul. 1998, pp. 144–151.

[53]M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software: an update,” SIGKDD Explor. Newsl., vol. 11, no. 1, pp. 10–18, Nov. 2009, doi: 10.1145/1656274.1656278.

[54]“JRip (weka-dev 3.9.5 API).” https://weka.sourceforge.io/doc.dev/weka/classifiers/rules/JRip.html (accessed Apr. 28, 2021).

[55]Y. Diao, H. Jamjoom, and D. Loewenstern, “Rule-Based Problem Classification in IT Service Management,” in 2009 IEEE International Conference on Cloud Computing, Sep. 2009, pp. 221–228. doi: 10.1109/CLOUD.2009.80.

[56]P. Xu, Z. Ding, and M. Pan, “An improved credit card users default prediction model based on RIPPER,” in 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Jul. 2017, pp. 1785–1789. doi: 10.1109/FSKD.2017.8393037.

[57]D. R. Kawade and K. S. Oza, “News Classification: A Data Mining Approach,” Indian Journal of Science and Technology, vol. 9, p. 46, 2016.

[58]“Knowledge Center - Swiss Post Solutions,” @SPSGlobal. https://www.swisspostsolutions.com/en/knowledge-center (accessed Apr. 28, 2021).

[59]C. S. Wasson, System Engineering Analysis, Design, and Development: Concepts, Principles, and Practices, 2nd edition. Hoboken, New Jersey: Wiley, 2015.

[60]D. Dori, C. Linchevski, R. Manor, and O. M. Opm, “OPCAT–An Object-Process CASE Tool for OPM-Based Conceptual Modelling,” in 1st International Conference on Modelling and Management of Engineering Processes, 2010, pp. 1–30.

[61]M. A. Smith, “NodeXL: Simple network analysis for social media,” in 2013 International Conference on Collaboration Technologies and Systems (CTS), May 2013, pp. 89–93. doi: 10.1109/CTS.2013.6567211.

[62]“Improving the quality of the output,” tessdoc. https://tesseract-ocr.github.io/tessdoc/ImproveQuality.html (accessed May 18, 2021).

[63] “Help extracting text from images.” https://groups.google.com/g/tesseract-ocr/c/AcguXNGznJs/m/T9gKFxKdUkoJ (accessed May 19, 2021).

[64] “OpenCV: Smoothing Images.” https://docs.opencv.org/master/d4/d13/tutorial_py_filtering.html (accessed May 19, 2021).

[65]“OpenCV: Morphological Transformations.” https://docs.opencv.org/master/d9/d61/tutorial_py_morphological_ops.html (accessed May 19, 2021).

[66]Bradski, Gary and Kaehler, Adrian, “Learning OpenCV [Book].” https://www.oreilly.com/library/view/learning-opencv/9780596516130/ (accessed May 18, 2021).

[67]R. Rehurek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” undefined, 2010, Accessed: May 18, 2021. [Online]. Available: /paper/Software-Framework-for-Topic-Modelling-with-Large-Rehurek-Sojka/83a6cacc126d85c45605797406262677c256a6af

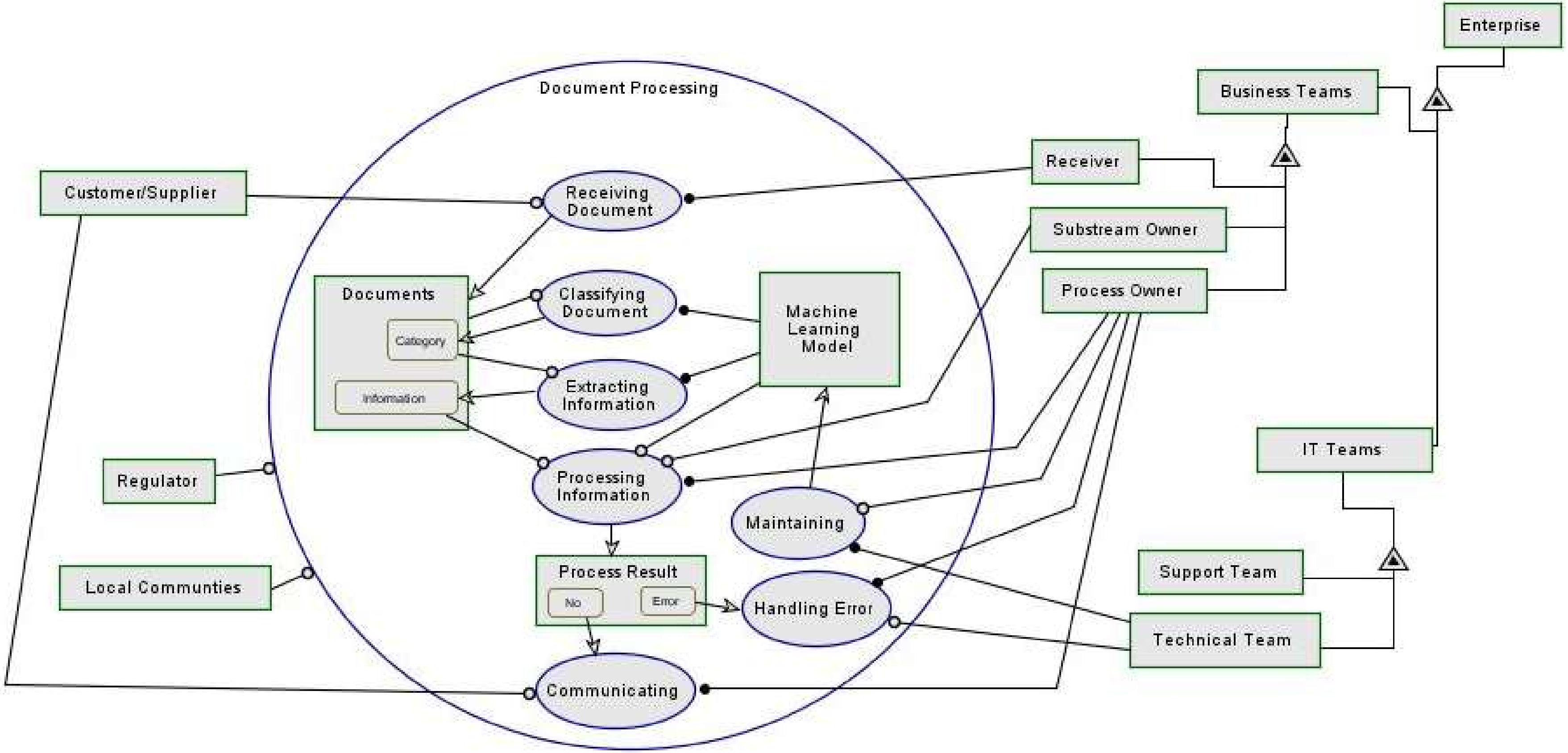
[68]“retraining Word2Vec model.” https://groups.google.com/g/gensim/c/iyba5Ni6q4k/m/KzNFHSztAQAJ (accessed May 18, 2021).

[69]J. Jöhnk, M. Weißert, and K. Wyrtki, “Ready or Not, AI Comes— An Interview Study of Organizational AI Readiness Factors,” Bus Inf Syst Eng, vol. 63, no. 1, pp. 5–20, Feb. 2021, doi: 10.1007/s12599-020-00676-7.

[70]J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proceedings of

the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota, Jun. 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

Appendix A


Enterprise
Business Teams
Document Processing
Receiver
Customer/Supplier
Receiving Document
Substream Owner
Process Owner
Documents
Classifying Document
Machine Learning Model
Category
Extracting Information
Information
IT Teams
Regulator
Processing Information
Maintaining
Local Communties
Process Result
No
Error
Handling Error
Support Team
Technical Team
Communicating

Documents can be Category or Information.

Enterprise exhibits Business Teams and IT Teams.

Business Teams exhibits Process Owner, Substream Owner, and Receiver.

Process Owner handles Communicating, Processing Information, and Handling Error.

Receiver handles Receiving Document.

IT Teams exhibits Support Team and Technical Team.

Technical Team handles Maintaining.

Machine Learning Model handles Extracting Information and Classifying Document.

Process Result can be Error or No.

Document Processing requires Local Communties and Regulator.

Handling Error requires Technical Team.

Handling Error consumes Error Process Result.

Classifying Document requires Documents.

Classifying Document yields Category Documents.

Processing Information requires Machine Learning Model, Information Documents, and Substream Owner.

Processing Information yields Process Result.

Extracting Information requires Category Documents.

Extracting Information yields Information Documents.

Maintaining requires Process Owner.

Maintaining yields Machine Learning Model.

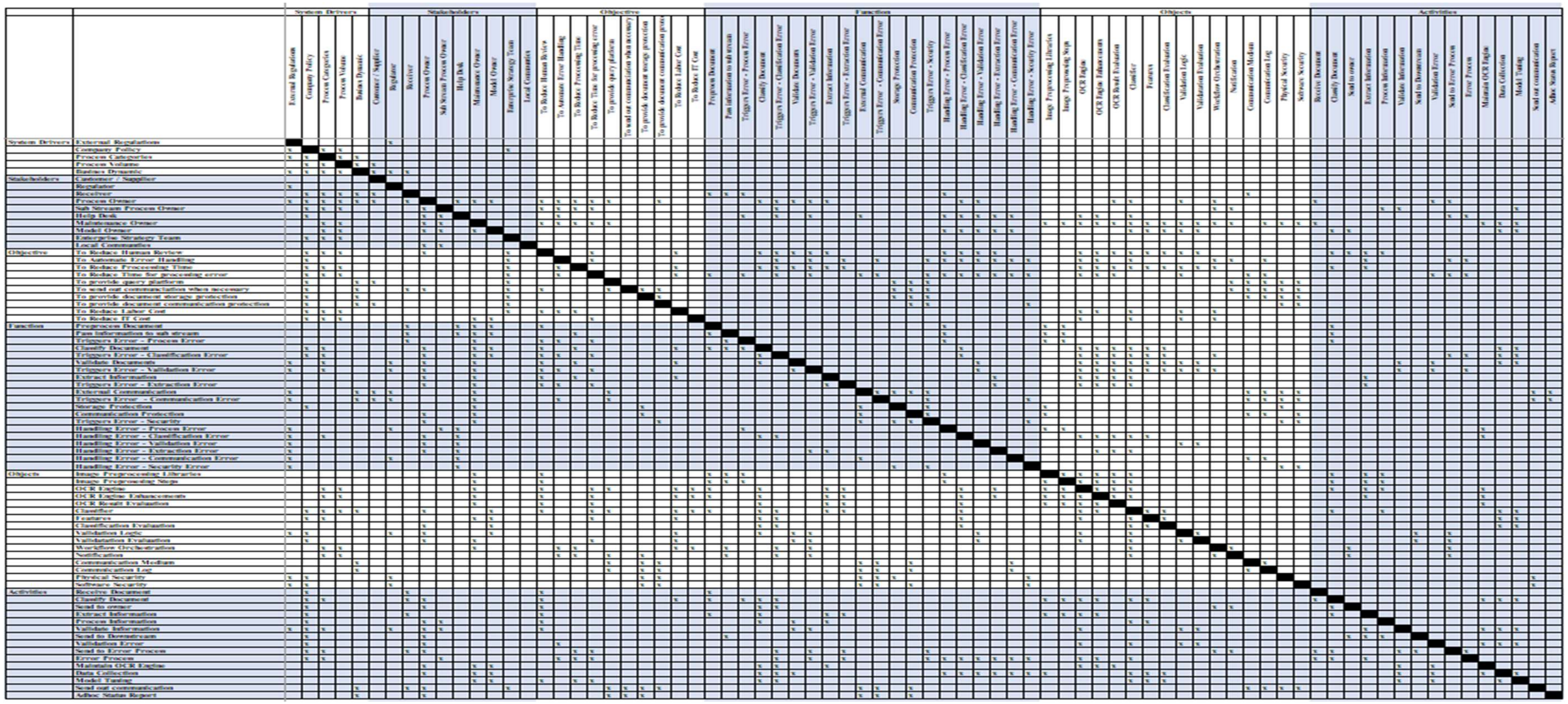
Receiving Document requires Customer/Supplier.

Receiving Document yields Documents.

Communicating requires Customer/Supplier.

Appendix B

ES-MDM full matrix



System Drivers | Stakeholders | Objective | Function | Objects | Activities

System Drivers: External Regulation, Company Policy, Process Categories, Process Volume, Business Dynamic

Stakeholders: Customer / Supplier, Regulator, Receiver, Process Owner, Sub Stream Process Owner, Help Desk, Maintenance Owner, Model Owner, Enterprise Strategy Team, Local Communities

Objective: To Reduce Human Review, To Automate Error Handling, To Reduce Processing Time, To Reduce Time for processing error, To provide query platform, To send out communciation when necessary, To provide document storage protection, To provide document communication protection, To Reduce Labor Cost, To Reduce IT Cost

Function: Preprocess Document, Pass information to sub stream, Triggers Error - Process Error, Classify Document, Triggers Error - Classification Error, Validate Documents, Triggers Error - Validation Error, Extract Information, Triggers Error - Extraction Error, External Communication, Triggers Error - Communication Error, Storage Protection, Communication Protection, Triggers Error - Security, Handling Error - Process Error, Handling Error - Classification Error, Handling Error - Validation Error, Handling Error - Extraction Error, Handling Error - Communication Error, Handling Error - Security Error

Objects: Image Preprocessing Libraries, Image Preprocesing Steps, OCR Engine, OCR Engine Enhancements, OCR Result Evaluation, Classifier, Features, Classification Evaluation, Validation Logic, Validatation Evaluation, Workflow Orchestration, Notification, Communication Medium, Commnication Log, Physical Security, Software Security

Activities: Receive Document, Classify Document, Send to owner, Extract Information, Process Information, Validate Information, Send to Downstream, Validation Error, Send to Error Process, Error Process, Maintain OCR Engine, Data Collection, Model Tuning, Send out communication, Adhoc Status Report

Appendix C

RIPPER Algorithm – Invoice Rule

In the generated ruleset, the condition is whether a stem of a word appears in the document. If the stem appears in the document, the condition is ">=1'; if the stem does not appear in the document, the condition is "<=0". An example of a stem word is that "invoic" is from "invoice." The parameter: "attribute_0" is the result of classification. 11 is the value of the invoice category. The last part of the rule is represented by the format: (the number of records covered / the number of records misclassified).

(invoic >= 1) => attribute_0=11 (2234.0/90.0)

(account >= 1) and (the <= 0) and (payment >= 1) and (code >= 1) => attribute_0=11 (110.0/0.0)

(account >= 1) and (the <= 0) and (for >= 1) => attribute_0=11 (139.0/12.0)

(avenu >= 1) and (univers <= 0) and (total >= 1) => attribute_0=11 (182.0/19.0)

(box >= 1) and (amount >= 1) and (date >= 1) => attribute_0=11 (116.0/14.0)

(pleas >= 1) and (avenu >= 1) and (on >= 1) => attribute_0=11 (39.0/0.0)

(the <= 0) and (univers <= 0) and (account >= 1) and (pleas >= 1) and (busi <= 0) => attribute_0=11 (35.0/2.0)

(studi <= 0) and (date >= 1) and (account >= 1) and (report <= 0) and (york >= 1) => attribute_0=11 (29.0/1.0)

(box >= 1) and (amount >= 1) and (method <= 0) and (product <= 0) => attribute_0=11 (29.0/5.0)

(the <= 0) and (univers <= 0) and (inc >= 1) and (total >= 1) and (page <= 0) => attribute_0=11 (67.0/17.0)

(studi <= 0) and (avenu >= 1) and (univers <= 0) and (copi >= 1) => attribute_0=11 (42.0/1.0)

(the <= 0) and (univers <= 0) and (box >= 1) and (date >= 1) and (laboratori <= 0) and (menthol <= 0) and (mai <= 0) and (address <= 0) and (filler <= 0) and (inform <= 0) => attribute_0=11 (51.0/8.0)

```
(studi <= 0) and (total >= 1) and (charg >= 1) and (inc >= 1) => attribute_0=11 (8.0/2.0)
(studi <= 0) and (account >= 1) and (date >= 1) and (end <= 0) and (health <= 0) => attribute_0=11 (146.0/53.0)
(pleas >= 1) and (payment >= 1) and (locat <= 0) => attribute_0=11 (37.0/5.0)
(studi <= 0) and (total >= 1) and (cigarett <= 0) and (print >= 1) and (compani >= 1) => attribute_0=11 (36.0/1.0)
(the <= 0) and (univers <= 0) and (york >= 1) and (cigarett <= 0) and (avenu >= 1) and (depart <= 0) and (and <= 0) => attribute_0=11 (35.0/6.0)
(the <= 0) and (univers <= 0) and (inc >= 1) and (york >= 1) and (cigarett <= 0) and (associ >= 1) => attribute_0=11 (15.0/1.0)
(studi <= 0) and (total >= 1) and (amount >= 1) and (check >= 1) => attribute_0=11 (22.0/1.0)
(compani >= 1) and (smoke <= 0) and (box >= 1) and (us <= 0) and (order >= 1) => attribute_0=11 (16.0/0.0)
(studi <= 0) and (inc >= 1) and (pleas >= 1) and (with <= 0) and (issu <= 0) and (busi <= 0) => attribute_0=11 (20.0/4.0)
(studi <= 0) and (street >= 1) and (univers <= 0) and (smoke <= 0) and (pleas >= 1) and (name <= 0) and (citi <= 0) and (good <= 0) => attribute_0=11
(19.0/1.0)
(studi <= 0) and (order >= 1) and (compani >= 1) and (year <= 0) and (descript >= 1) => attribute_0=11 (21.0/1.0)
(the <= 0) and (univers <= 0) and (avenu >= 1) and (servic >= 1) and (produc <= 0) and (cost <= 0) => attribute_0=11 (14.0/1.0)
(studi <= 0) and (smoke <= 0) and (compani >= 1) and (the <= 0) and (total >= 1) and (advertis >= 1) => attribute_0=11 (13.0/0.0)
(the <= 0) and (univers <= 0) and (expens >= 1) and (work >= 1) => attribute_0=11 (22.0/4.0)
```

Appendix D

PART Algorithm – Resume Rule

In PART algorithm, the rule representation is different. If the stem of a word appears in the document, the condition is "> 0'; if the stem does not appear in the document, the condition is "<=0". An example of a stem word is that " univers " is the result of " university." 14 is the value of the resume category. The last part of the rule is represented by the format: (the number of records covered / the number of records misclassified).

```
professor > 0 AND
said <= 0 AND
reason <= 0 AND
smoker <= 0 AND
suggest <= 0 AND
smoke <= 0 AND
univers > 0 AND
report <= 0 AND
tobacco <= 0 AND
import <= 0 AND
nicotin <= 0 AND
with <= 0 AND
caus <= 0 AND
all <= 0 AND
```

```
design <= 0: 14 (2169.0/9.0)

postdoctor > 0 AND
payment <= 0 AND
receiv <= 0 AND
these <= 0 AND
charg <= 0 AND
produc <= 0 AND
support <= 0: 14 (501.0)

fellow > 0 AND
you <= 0 AND
thei <= 0 AND
account <= 0 AND
that <= 0 AND
week <= 0 AND
suggest <= 0 AND
question <= 0 AND
requir <= 0 AND
issu <= 0 AND
show <= 0 AND
appear <= 0 AND
result <= 0 AND
```

```
smoker <= 0 AND
cigarett <= 0 AND
period <= 0 AND
increas <= 0: 14 (353.0/8.0)

protein > 0 AND
honor <= 0 AND
educ > 0 AND
result <= 0 AND
indic <= 0: 14 (46.0)

biologi > 0 AND
educ > 0 AND
at <= 0: 14 (157.0/4.0)

protein > 0 AND
honor > 0: 14 (30.0)

cell > 0 AND
profession > 0 AND
advertis <= 0 AND
smoker <= 0: 14 (64.0/3.0)
```

```
univers > 0 AND
honor > 0 AND
month <= 0 AND
but <= 0: 14 (92.0/2.0)

univers > 0 AND
educ > 0 AND
but <= 0 AND
indic <= 0 AND
caus <= 0 AND
result <= 0 AND
expens <= 0 AND
ad <= 0 AND
see <= 0 AND
check <= 0 AND
percent <= 0 AND
total <= 0 AND
evid <= 0 AND
determin <= 0 AND
all <= 0 AND
at <= 0 AND
cigarett <= 0: 14 (128.0)
```

Appendix E

RIPPER Algorithm – Resume Rule

In the generated ruleset, the condition is whether a stem of a word appears in the document. If the stem appears in the document, the condition is ">=1'; if the stem does not appear in the document, the condition is "<=0". An example of a stem word is that " univers " is from "university." The parameter: "attribute_0" is the result of classification. 11 is the value of the invoice category. The last part of the rule is represented by the format: (the number of records covered / the number of records misclassified).

```
(univers >= 1) and (assist >= 1) and (fellow >= 1) => attribute_0=14 (1553.0/5.0)
(univers >= 1) and (educ >= 1) and (smoke <= 0) and (institut >= 1) and (report <= 0) => attribute_0=14 (783.0/1.0)
(univers >= 1) and (assist >= 1) and (the <= 0) => attribute_0=14 (297.0/3.0)
(univers >= 1) and (profession >= 1) and (smoke <= 0) and (experi >= 1) => attribute_0=14 (307.0/3.0)
(univers >= 1) and (professor >= 1) and (smoke <= 0) and (biologi >= 1) and (import <= 0) => attribute_0=14 (98.0/3.0)
(univers >= 1) and (educ >= 1) and (smoke <= 0) and (year <= 0) and (import <= 0) and (end <= 0) => attribute_0=14 (196.0/5.0)
(univers >= 1) and (the <= 0) and (intern >= 1) and (associ >= 1) => attribute_0=14 (33.0/0.0)
(fellow >= 1) and (research >= 1) and (profession >= 1) => attribute_0=14 (60.0/1.0)
(univers >= 1) and (the <= 0) and (colleg >= 1) and (research >= 1) and (produc <= 0) => attribute_0=14 (40.0/4.0)
(univers >= 1) and (assist >= 1) and (group <= 0) and (honor >= 1) => attribute_0=14 (14.0/0.0)
(univers >= 1) and (the <= 0) and (fellow >= 1) => attribute_0=14 (25.0/1.0)
(univers >= 1) and (thi <= 0) and (director >= 1) and (said <= 0) => attribute_0=14 (122.0/40.0)
```

```
(assist >= 1) and (smoke <= 0) and (professor >= 1) and (for <= 0) => attribute_0=14 (42.0/0.0)
(univers >= 1) and (the <= 0) and (presid >= 1) => attribute_0=14 (25.0/5.0)
(scienc >= 1) and (the <= 0) and (educ >= 1) and (from <= 0) => attribute_0=14 (22.0/2.0)
(school >= 1) and (the <= 0) and (biologi >= 1) => attribute_0=14 (15.0/2.0)
(univers >= 1) and (thi <= 0) and (smoke <= 0) and (award >= 1) and (report <= 0) and (depart <= 0) => attribute_0=14 (17.0/1.0)
(school >= 1) and (the <= 0) and (member >= 1) and (medicin <= 0) => attribute_0=14 (13.0/2.0)
(research >= 1) and (profession >= 1) and (tobacco <= 0) and (experi >= 1) => attribute_0=14 (25.0/1.0)
(smoke <= 0) and (research >= 1) and (thi <= 0) and (fellow >= 1) and (support <= 0) => attribute_0=14 (24.0/3.0)
(the <= 0) and (manag >= 1) and (director >= 1) and (compani >= 1) => attribute_0=14 (14.0/1.0)
(director >= 1) and (smoke <= 0) and (experi >= 1) and (time <= 0) => attribute_0=14 (13.0/1.0)
(scienc >= 1) and (the <= 0) and (present >= 1) and (measur <= 0) and (contain <= 0) and (patholog <= 0) => attribute_0=14 (17.0/5.0)
(univers >= 1) and (result <= 0) and (chemistri >= 1) and (section >= 1) and (increas <= 0) => attribute_0=14 (9.0/1.0)
(univers >= 1) and (thi <= 0) and (and <= 0) and (smoke <= 0) and (work >= 1) and (report <= 0) and (program <= 0) and (chang <= 0) and (us
<= 0) => attribute_0=14 (23.0/6.0)
(biologi >= 1) and (thi <= 0) and (public >= 1) and (the <= 0) => attribute_0=14 (8.0/0.0)
(smoke <= 0) and (colleg >= 1) and (societi >= 1) and (scienc >= 1) => attribute_0=14 (10.0/2.0)
```

Appendix F

Generated Rule from OneR Algorithm

In this algorithm, the rule is whether a stem of a word appears in the document. If the word appears in the document, the condition is ">=0.5'; if the word does not appear in the document, the condition is "<=0.5". An example of a stem word is that "univers" is the result of "universe." The parameter: "attribute_0" is the result of classification. 7 is the value of the specification category and 14 is the value of resume category.

If “univers <0.5”, then attribute_0 = 7
If “univers >= 0.5”, then attribute_0 = 14
(7643/35045 instance correct)