Optical Character Recognition Engines Performance Comparison in Information Extraction

Tosan Wiar Ramdhani, Indra Budi, Betty Purwandari
Faculty of Computer Science
Universitas Indonesia
Depok, Indonesia

***Abstract*—Named Entity Recognition (NER) is often used to acquire important information from text documents as a part of the Information Extraction (IE) process. However, the text documents quality affects the accuracy of the data obtained, especially for text documents acquired involving the Optical Character Recognition (OCR) process, which never reached 100% accuracy. This research tried to examine which OCR engine with the highest performance for IE using NER by comparing three OCR engines (Foxit, PDF2GO, Tesseract) over 8,562 government human resources documents within six document categories, two document structures, and four measurements. Several essential entities such as name, employee ID, document number, document publishing date, employee rank, and family member's name were trying to be extracted automatically from the documents. NER processes were done using Python programming language, and the preprocessing tasks were done separately for Foxit, PDF2GO, and Tesseract. In summary, each OCR engine has its drawbacks and benefit, such as Tesseract has better NER extraction and conversion time with better accuracy but lack in the number of entities acquired.**

Keywords—Named entity recognition; information extraction; optical character recognition; government human resources documents

I. Introduction

Information extraction from scanned documents has been an issue in many countries and domains. Pathel and Bhatt in 2020 [1] used an end-to-end sequential approach for abstractive information extraction on scanned Malaysian invoices. Nguyen et al. in 2020 [2] used a rapid and convenient text-mining method to automatically extract pathology features from complex text-based scanned photocopies of Australian typewritten clinical pathology reports drawn from multiple different sources. Bures et al. in 2020 [3] proposed a system design to extract information from several countries structured scanned invoice documents by an ordinary office scanner device. Rastogi et al. in 2020 [4] used knowledge graph and Formal Concept Analysis (FCA) template detection to extract information from 1,400 scanned trade finance documents. Those issues also have been a problem in Indonesian Government.

In Indonesia, every human resources division in the government has the same problem of extracting information from scanned human resources documents. The information extraction process usually was done manually by entry operators, which is time-consuming and error-prone.

The government's human resources division records a vast number of employee documents each year. For instance, Bogor local government, one of the ninety-eight city governments in West Java, Indonesia, recorded more than 200,000 human resources documents during the 2009 to 2020 period. Those numbers are increasing each year, like in 2018, there are 10,880 recorded documents, while in 2019, the number increased to 28,784 recorded documents. With only two data operators to handle such data, this is really time-consuming, not to mention the human error factor while inserting essential pieces of information from each document into the human resources management system.

This manually extracted information is essential since it has been used for many human resources management system modules like a decision support system for talent management, executive statistics dashboard, salary budget prediction, employee formation, and many others.

Each document was acquired using scanners in PDF format. The recording process was taken manually by operators inserting important information of each document into the human resources database through the human resources management system web-based interface. With the help of IE tasks like NER, this process can be simplified using automated NER to make the data management more effective and efficient.

OCR converts the scanned images of handwritten, typewritten, or printed documents into machine-readable format [5]. IE process is used to extract structured content in entities, relations, facts, terms, and other types of information [6]. In general, NER is a subtask of IE that aims to find and categorize specific entities in text documents [7]. The documents source may vary from the web, generated PDF, and scanned documents in PDF format. In scanned documents case which involves OCR engine, to get good accuracy, precision, and recall of the entities acquired is a challenge for this research.

Three different OCR engines with different environments were selected for the experiment. Foxit as the desktop-based OCR engine, PDF2GO as the web-based OCR engine, and Tesseract as an open-source programable OCR engine. Among those OCR engines, we would like to examine which engine has the highest measurements score to extract essential information from government human resources documents using four measurements.

The measurement results will determine which engine is the suitable solution to handle IE tasks involving NER and OCR process in two document structures. The results will also help any organization to manage scanned documents more effectively and efficiently. As for Indonesian government, an effective and efficient document management would really help any government division to work more efficient with less or even none human resources involved with the help of automated IE from this study.

The following sections of this paper are the related works in Section 2, the materials and methods in Section 3, experiments results and discussion in Section 4. Lastly, the conclusion is in Section 5.

II. Related Work

IE using scanned documents is often noisy and often suffering from blur effects, faded text, watermarks, scanning artifacts, and wrinkles. Those noises often caused the downstream OCR and other errors [4]. Those noises are the cause of low accuracy in IE using scanned documents.

In some cases, preprocessing tasks, such as image contrast improvement, noise reduction, binarization, and image deskewing, are required to get a visual improvement of the scanned documents [3]. High accuracy and low latency for processing large numbers of documents are required to have the best result of OCR [1]. IE in a medical domain has proven that keyword trigger-based automation with OCR correction and negation handling is rapid and convenient and provides consistent and reliable data abstractions from scanned clinical records [2]. Those cases showed that preprocessing has an important role to get a better result in IE using scanned documents.

Taghva et al. in 2006 [8] and Pereda in 2011 [9] proved that the IE task is significantly influenced by OCR errors, while Vijayarani and Sakila in 2015 [10] tried to compare 8 OCR tools using image and PDF documents. NER previous research on PDF documents like legal documents had been done to extract information for specific entities. Solihin and Budi, in 2018 [11], researched the extraction of data from general criminal court decision documents using the rule-based method. Meanwhile, Leitner et al. in 2019 [12] used machine learning methods like CRF (Conditional Random Fields) and deep learning methods like BiLSTM (Bidirectional Long Short-Term Memory) to extract information using NER on legal documents. Nuranti and Yulianti in 2020 [13] also tried the same method in Indonesian legal documents. Those research [11,12,13] used generated PDF legal documents as their data source, while our study used scanned PDF documents which may have lower accuracy influenced by OCR errors.

Tesseract library is often used in several research of IE involving OCR. Patel and friends produced 70% accuracy using 20 sample images in 2012 [14]. Kumar and friends produced 97% accuracy for small scanned bill documents and 83% accuracy for small scanned bill documents using Tesseract OCR on 25 scanned bills in 2020 [15]. Akinbade and friends produced 81.9% character accuracy and 69.7% word accuracy on 11 sample images in 2020 [16]. Haraj and Raissouni produced an average of 95.77% charcater accuracy using tesseract and opencv library over 4 sample images in 2015 [17]. Those research [14,15,16,17] only used relatively small samples (less than 50 documents), while our study used more documents (8,562 documents in 6 Categories and two document structures). Previous research [14,15,16,17], which also employed the Tesseract library, only used string matching to measure the OCR. On the other hand, our study used four measurements, i.e., conversion time, NER time, string match accuracy as precision, and the number of entities acquired as recall.

III. Material and Methods

The proposed method we use in this study is an adaptation of the general framework for text analysis in mining text data literature as shown in Fig. 1 [18]. The framework start with text corpus as an input source, followed by preprocessing phase. Then followed by the text representation process and it ends with the knowledge discovery process as shown in Fig. 1.

Fig. 2 shows the four main phases of the proposed method. The phase began with data collection, followed by the preprocessing step by converting PDF to text. Then, the NER process used rule-based entity recognition written in Python, followed by the end's evaluation process.

The reason we use the prosed method as an adaptation from the general framework for text analysis is because it has suitable phases that also can be implement for IE using NER in this study. Nevertheless the original framework has different examples in each phase since the example is used for text data analytics in social media.

The data collection refers to the text corpus phase, the preprocessing phase is similar, the NER phase refers to the representation phase, and evaluation refers to the knowledge discovery phase.

A. Data Collection

Eight thousand five hundred sixty-two government human resources documents within 6 document categories were collected as data samples. Those documents were downloaded from the human resources server of Bogor local government using PHP script for each document category selected. Table I demonstrates the number of entities and the number of records of each class.

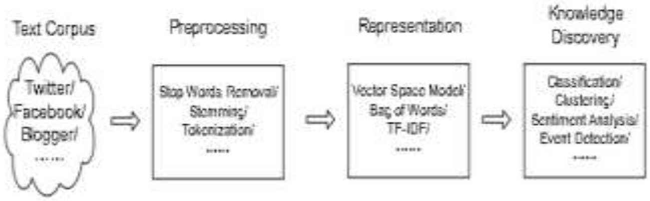

Text Corpus
Preprocessing
Representation
Knowledge Discovery
Twitter/ Facebook/ Blogger/ ......
Stop Words Removal/ Stemming/ Tokenization/ ......
Vector Space Model/ Bag of Words/ TF-IDF/ ......
Classification/ Clustering/ Sentiment Analysis/ Event Detection/ .....

Fig. 1. The General Framework for Text Analysis in Mining Text Data [17].

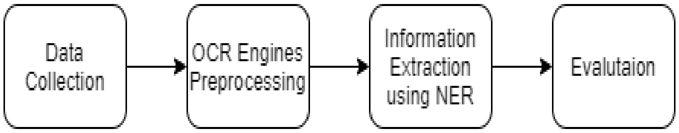


Fig. 2. The Main Phases of the Proposed Method.

TABLE I. DOCUMENTS AND ENTITIES OF EACH DOCUMENT CATEGORY

Category	Average Size	Samples	Entity	Structure
Retirement	505 Kb	110	6	Tabular
Recruitment	448 Kb	453	6	Non table
CV	39 Kb	5,964	7	Tabular
ID conversion	419 Kb	1,168	5	Tabular
Employee position	501 Kb	265	5	Non-table
Family allowance	332 Kb	602	6	Tabular

The category selection was based on the head of National Civil Service Agency regulation number 18 in 2011 about manuscript layout management of the national human resources document [19]. From those categories, four categories (curriculum vitae, ID conversion, family allowance, retirement) have a tabular data structure, and the other two categories (recruitment, employee position) do not have a table data structure.

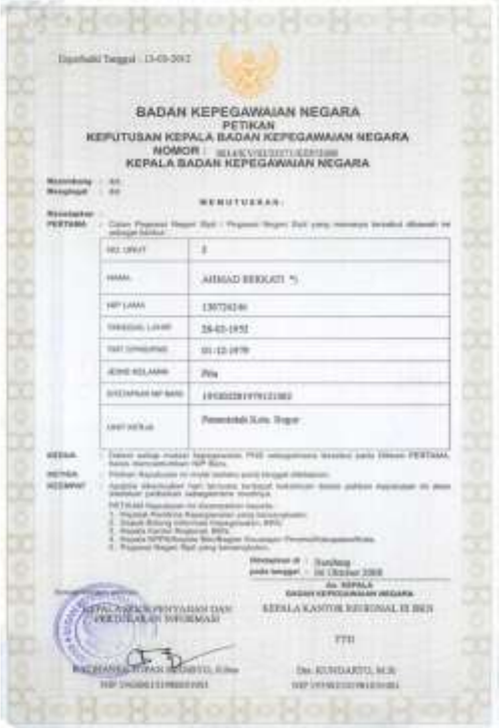
Each document category has a similar average size and similar number type of entities except for Curriculum Vitae, which has only 39 Kb average size per document. Retirement documents are the fewest, with 110 documents, and Curriculum Vitae is the most significant sample with 5,964 documents.Each document type has at least two same entities: name and employee ID. They are employed as primary keys in the rule-based NER written in Python language to compare the actual values of each entity within the human resources database with the entities acquired from the OCR process results.

The entity selection for each document category were based on the existing information stored in the human resources database. Therefore we can validate and measure the accuracy of each entity acquired by comparing it with the actual value from the human resource database. We used word level accuracy as a measurement for each entity acquired. Therefore we only compare some part of each document as entities with the actual value from the human resources database instead of using the whole text from each document.

Each category has different important entities to extract. Each entity in every type has its own character, as shown in Table II.

TABLE II. RETIREMENT DOCUMENT CHARACTER

Entity	Format
Name	Free text
New ID number	18 digits number
Old ID number	Nine digits number
Document number	Nine digits number + year
Working period	Two digits year and month
Retirement date	dd-mm-YYYY



Diperbaiki Tanggal : 13-03-2012

BADAN KEPEGAWAIAN NEGARA
PETIKAN
KEPUTUSAN KEPALA BADAN KEPEGAWAIAN NEGARA
NOMOR : [illegible]
KEPALA BADAN KEPEGAWAIAN NEGARA

Menimbang : dst
Mengingat : dst

MEMUTUSKAN:

Menetapkan :
PERTAMA : Calon Pegawai Negeri Sipil / Pegawai Negeri Sipil yang namanya tersebut dibawah ini sebagai berikut:

NO. URUT	2
NAMA	AHMAD BERKATI *)
NIP LAMA	130724240
TANGGAL LAHIR	26-02-1952
TMT [illegible]	01-12-1978
JENIS KELAMIN	Pria
DITETAPKAN NIP BARU	195202261978121002
UNIT KERJA	Pemerintah Kota Bogor

KEDUA : [illegible] PERTAMA, [illegible] NIP Baru.
KETIGA : [illegible]
KEEMPAT : [illegible]

PETIKAN Keputusan ini disampaikan kepada:
1. [illegible]
2. [illegible]
3. Kepala Kantor Regional BKN
4. [illegible]
5. Pegawai Negeri Sipil yang bersangkutan.

Ditetapkan di : Bandung
pada tanggal : 06 Oktober 2008

An. KEPALA
BADAN KEPEGAWAIAN NEGARA
KEPALA KANTOR REGIONAL III BKN

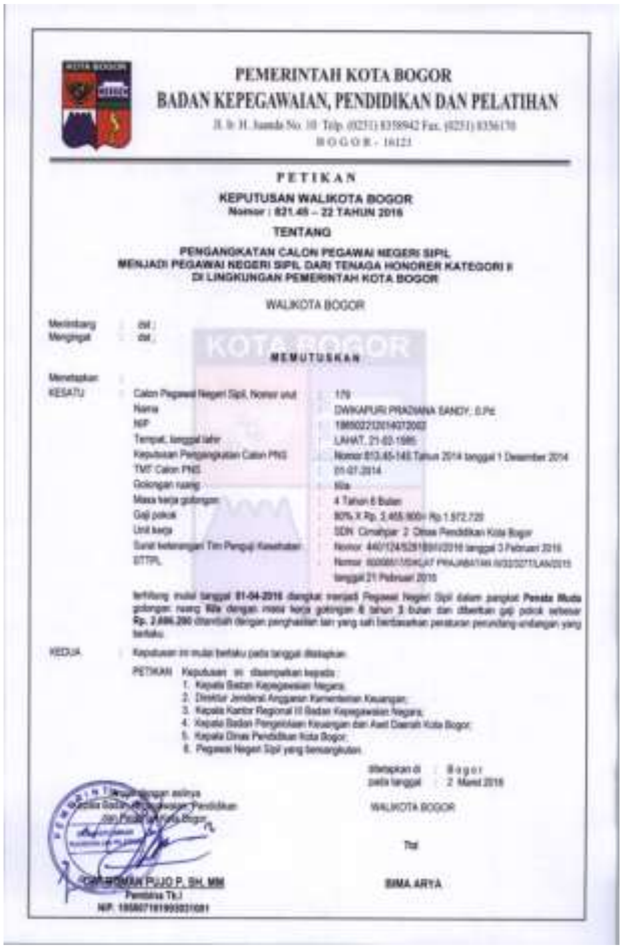
TTD

Drs. KUNDARTO, M.Si
NIP [illegible]

KEPALA SEKSI PENYAJIAN DAN PERTUKARAN INFORMASI

[illegible], S.Sos
NIP [illegible]

Fig. 3. ID Conversion Document Sample.



PEMERINTAH KOTA BOGOR
BADAN KEPEGAWAIAN, PENDIDIKAN DAN PELATIHAN
Jl. Ir. H. Juanda No. 10 Telp. (0251) 8358942 Fax. (0251) 8356170
BOGOR - 16121

PETIKAN
KEPUTUSAN WALIKOTA BOGOR
Nomor : 821.45 – 22 TAHUN 2016
TENTANG
PENGANGKATAN CALON PEGAWAI NEGERI SIPIL
MENJADI PEGAWAI NEGERI SIPIL DARI TENAGA HONORER KATEGORI II
DI LINGKUNGAN PEMERINTAH KOTA BOGOR

WALIKOTA BOGOR

Menimbang : dst ;
Mengingat : dst ;

MEMUTUSKAN

Menetapkan :

KESATU	Calon Pegawai Negeri Sipil, Nomor urut	179
	Nama	DWIKAPURI PRADIANA SANDY, S.Pd
	NIP	198502212014072002
	Tempat, tanggal lahir	LAHAT, 21-02-1985
	Keputusan Pengangkatan Calon PNS	Nomor 813.45-145 Tahun 2014 tanggal 1 Desember 2014
	TMT Calon PNS	01-07-2014
	Golongan ruang	II/a
	Masa kerja golongan	4 Tahun 6 Bulan
	Gaji pokok	80% X Rp. 2.465.900= Rp 1.972.720
	Unit kerja	SDN Cimahpar 2 Dinas Pendidikan Kota Bogor
	Surat keterangan Tim Penguji Kesehatan	Nomor 440/124/5281/BBI/2016 tanggal 3 Februari 2016
	STTPL	Nomor [illegible]/DIKLAT PRAJABATAN [illegible]/LAN/2015 tanggal 21 Februari 2015

terhitung mulai tanggal **01-04-2016** diangkat menjadi Pegawai Negeri Sipil dalam pangkat **Penata Muda** golongan ruang **II/a** dengan masa kerja golongan **6** tahun **3** bulan dan diberikan gaji pokok sebesar **Rp. 2.886.200** ditambah dengan penghasilan lain yang sah berdasarkan peraturan perundang-undangan yang berlaku.

KEDUA : Keputusan ini mulai berlaku pada tanggal ditetapkan.

PETIKAN Keputusan ini disampaikan kepada:
1. Kepala Badan Kepegawaian Negara;
2. Direktur Jenderal Anggaran Kementerian Keuangan;
3. Kepala Kantor Regional III Badan Kepegawaian Negara;
4. Kepala Badan Pengelolaan Keuangan dan Aset Daerah Kota Bogor;
5. Kepala Dinas Pendidikan Kota Bogor;
6. Pegawai Negeri Sipil yang bersangkutan.

Ditetapkan di : Bogor
pada tanggal : 2 Maret 2016

WALIKOTA BOGOR

Ttd

BIMA ARYA

Salinan sesuai dengan aslinya
Kepala Badan Kepegawaian, Pendidikan dan Pelatihan Kota Bogor,

[illegible] PUJO P, SH, MM
Pembina Tk.I
NIP. 195807181993031081

Fig. 4. Recruitment Document Sample.

There are four document types with tabular structure data: retirement, curriculum vitae, ID conversion, and family allowance. There are only two document types with non-table document structures, which are recruitment and employee position.

The following Fig. 3 is an example of an ID conversion document as the tabular structure document, and Fig. 4 is an example of a recruitment document as the non-table structure document.

B. OCR Engines Preprocessing

The next phase after collecting the scanned PDF documents is to convert them into text format. Three OCR engines from three different environments were used in this phase. An offline desktop-based application called Foxit, an online-based application called PDF2GO, and an open-source OCR library called Tesseract were used to convert all documents. Patel et al. in 2012 [14] produce 70% accuracy, Kumar et al. in 2020 [15] produce 97% accuracy and, Akinbade [16] produce 81.9% accuracy using the Tesseract library on scanned documents.

Each engine has its benefits and drawbacks. For Foxit, it has better reading for tabular data and preserves spaces from the original document. However, it has no batch conversion feature. Hence, it must be done manually, one by one. Another drawback for tabular data is the required RTF conversion to get a better result before converting to a TXT file.

On PDF2GO, it has a 500 document conversion per batch feature and supports the Indonesian language feature on the OCR process. However, it does not preserve spacing from the original document, and it takes more time to upload and download the document since it is an online application.

It has unlimited batch conversion on Tesseract since we can customize the process, supporting the Indonesian language OCR process. However, it does not preserve spacing from the original document. The Foxit OCR engine has better OCR quality for tabular data. Still, since it has no batch conversion feature and needs an RTF conversion first before converted to text files, it makes it not efficient in terms of processing time. PDF2GO has 500 documents per batch conversion, and Tesseract has an unlimited document batch. Still, even though it takes 5 to 10 seconds to convert a document for Tesseract, it takes a longer time in PDF2G0 since we have to upload and download the paper first, which consumes more time. In terms of conversion time, Tesseract is the best option. In terms of the OCR feature, Foxit can preserve the space and better handle tabular data. Even though Tesseract included this feature in their library, the page segmentation mode will help extract information from tabular data.

C. Information Extraction using NER

The IE process using NER was written using Python programming language since it has features for NLP tasks. The rule-based method was used since OCR results from the documents are not 100% accurate, unlike the generated PDF documents.

The employee ID number is the primary key used to connect to each entity's other actual value. If we get the wrong OCR result for the employee ID number, it will get the wrong or even empty weight from the human resource database. Each document category from each OCR engine has a different extraction time, as shown in Table III. The table shows that each document category has the various best time for each OCR engine. Foxit has better time records on the retirement category with 0.017 seconds per document and the family allowance category with 0.077 seconds per document. Tesseract has better time records on recruitment category with 0.088 seconds per document, employee position category with 0.0104 seconds per document, and Curriculum Vitae with 0.040 seconds per document. PDF2GO only has a better time record on the ID conversion category with 0.011 seconds per document. On average, Tesseract has the best extraction time with 0.044 seconds per document, followed by PDF2GO with 0.046 seconds per document and Foxit with 0.050 seconds per document.

TABLE III. NER TIME FOR EACH DOCUMENT CATEGORY

Category	**Foxit**	**PDF2GO**	**Tesseract**
Retirement	0.017	0.019	0.023
Recruitment	0.096	0.089	0.088
Curriculum Vitae	0.074	0.071	0.040
ID conversion	0.022	0.011	0.012
Employee position	0.016	0.010	0.010
Family allowance	0.077	0.078	0.092
Average	0.050	0.046	0.044

D. Evaluation

In terms of quality measurement, the string match function from the Python library was used by comparing the entity string acquired with the actual data from the human resources database. This function is based on the G*estalt* pattern matching, which measures accuracy by this equation:

$$D_{\boldsymbol{\gamma}\mathbf{0}} = \frac{2K_m}{|S_1|+|S_2|} \tag{1}$$

The D$\boldsymbol{\gamma}$0 value ranged from 0 to 1, where one value means 100% the same and 0 value means vice versa. The S1 and S2 refer to the character number from the first and second strings, respectively. Km refers to the number of the same character between two strings being compared [20]. We used percentages ranged from 0 to 100 to represent the D$\boldsymbol{\gamma}$0 in our experiment.

The quantity measurement used a simple counter parameter on the rule-based NER Python code each time an entity was acquired. The actual quantity for each entity was obtained using the SQL count function from the human resources database.

IV. RESULTS AND DISCUSSION

A. Results

After the NER process, precision and recall measurement divided for each document category to see which OCR engine text result has better precision, recall, and F1-Score percentages shown in Table IV.

TABLE IV. PRECISSION, RECALL AND F1 MEASUREMENT

Document Type	**Structure**	**Foxit (%)**			**PDF2GO (%)**			**Tesseract (%)**		
		Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
Retirement	Tabular	93.99	85.06	89.30	93.62	86.58	89.62	88.19	52.42	65.76
Recruitment	Non Table	83.66	73.66	78.34	86.85	92.20	89.45	86.86	90.91	88.83
Curriculum Vitae	Tabular	86.28	86.64	86.46	80.13	97.91	88.13	84.14	80.98	82.53
ID Conversion	Tabular	68.85	84.33	75.81	74.45	77.84	76.11	74.99	54.40	63.06
Employee position	Non Table	87.59	100	93.38	86.46	99.62	92.57	92.8	99.62	96.09
Family allowance	Tabular	95.22	75.87	84.45	92.16	72.84	81.37	91.17	71.42	80.10
Average		85.93	84.26	84.62	85.61	**87.83**	**86.27**	**86.36**	74.96	79.39

Based on Table IV, we can see that in terms of precision by string matching, Tesseract has the highest accuracy with 86.36%. However, in terms of recall, PDF2GO has more identified entities with 87.83%. PDF2GO has the highest F1-Score with 86.27%.

Even though it seems PDF2GO has the highest F1-Score on average, but in terms of document structures, the highest F1-Score are different. Foxit has the highest F1-Score average for tabular data structure with 84.01%, followed by PDF2GO with 83.89% and Tesseract with 72.85%. Tesseract has the highest F1-Score average for non-table data structure with 92.46%, followed by PDF2GO with 91.01% and Foxit with 85.86%.

The results show that the tabular data structure F1-Score average for Tesseract is very low (72.85%). Consequently, it caused the total average of 6 document types also low at 79.39%, even though for non-table data structure Tesseract has the highest F1-Score with 92.46%. Even though Foxit has the highest F1-Score average for tabular data structure with 84.01%, it also has the lowest F1-Score average for non-table data structure with only 85.86%.

In general, PDF2GO has the highest F1-Score average with 86.27%. In terms of tabular document structure, Foxit has the highest F1-Score average with 84.01%. Tesseract has the highest F1-Score average with 92.46% for non-table document structure.

Even though Tesseract has the best precision, however, it has the worst recall. Therefore it also has the worst F1-Score since the precision gap between each OCR engine is slightly different, and the recall gap for Tesseract is far.

In the IE phase, we recorded the processing time of OCR engines for each document category. We can compare which OCR has the best OCR processing time, as shown in Fig. 5.

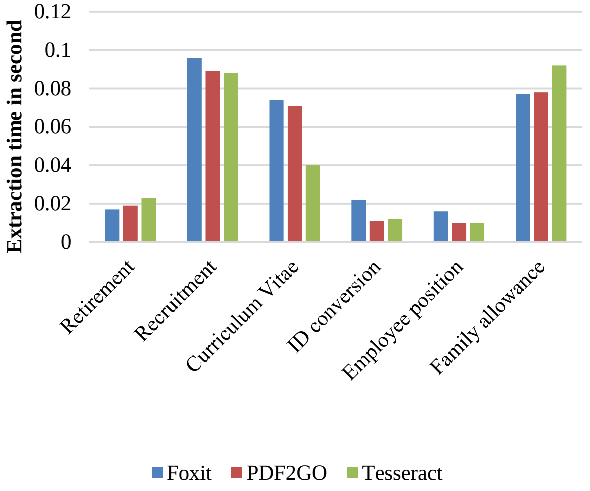


Fig. 5. Extraction Time per Document for each Category.

Tesseract is dominant in three document categories with 0.004 seconds per Curriculum Vitae document, 0.008 seconds per recruitment document, and 0.01 seconds per employee position document. On the other hand, Foxit is dominant in two document categories with 0.017 seconds per retirement document and 0.077 seconds per family allowance document. PDF2GO is only prevalent in one document category with 0.011 seconds per ID conversion document.

On average, Tesseract is the fastest OCR engine with 0.044 seconds per document, followed by PDF2GO with 0.046 seconds per document and Foxit with 0.050 seconds per document. Even though the NER times are slightly different, the impact is enormous since the volume of the document to extract is also enormous. We can say that in terms of NER information extraction time, Tesseract has the best time record, not to mention the benefit of having unlimited batch conversion. In contrast, in Foxit, you have to do the conversion one by one, and in PDF2GO, the batch conversion is limited to 500 documents in the preprocessing phase.

In the precision measurement phase, we recorded the precision of each OCR engine for each document category. Each acquired entity was compared automatically with the actual value from the human resources database records using the *Gestalt* string matching function described previously. From those records, we can compare which OCR has the best string match accuracy, as shown in Fig. 6.

Tesseract has the highest precisions in three document categories with 86.86% in the recruitment category, 92.8% in the employee position category, and 74.99% in the ID conversion category. Foxit has better precision in three document categories with 86.28% in the Curriculum Vitae category, 95.22% in the family allowance category, and 93.99% in the retirement category.Tesseract is more dominant in three documents. Besides, it has the best precision average with 86.36%, followed by Foxit with 85.93% and PDF2GO with 85.61%.

We recorded the number of entities acquired for each OCR engine in each document category in the recall measurement phase. The number of entities acquired for each entity compared to the number of entities in the human resources database using the employee ID number as the primary key. We can compare which OCR has the most entities acquired for the recall measurement from those entity numbers, as shown in Fig. 7.

We can see that PDF2GO has more entities acquired in three document categories, with 86.58% recall in the retirement category, 92.2 % recall in the recruitment category, and 97.91% in the Curriculum Vitae category. Foxit has more entities acquired in three document categories with 84.33% recall in the ID conversion category, 100% accuracy in the employee position category, and 75.87% recall in the family allowance category.

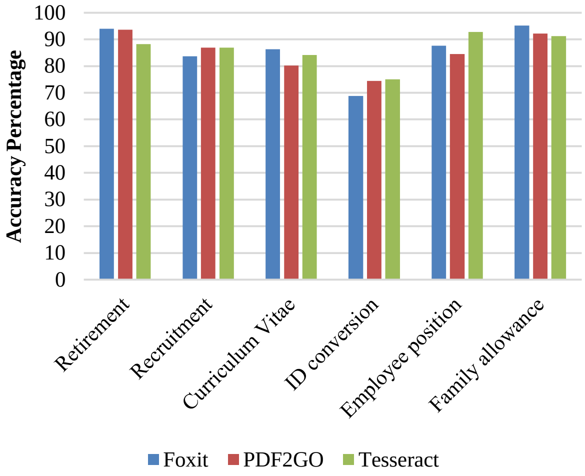


Fig. 6. The Precision Percentage for each Document Category.

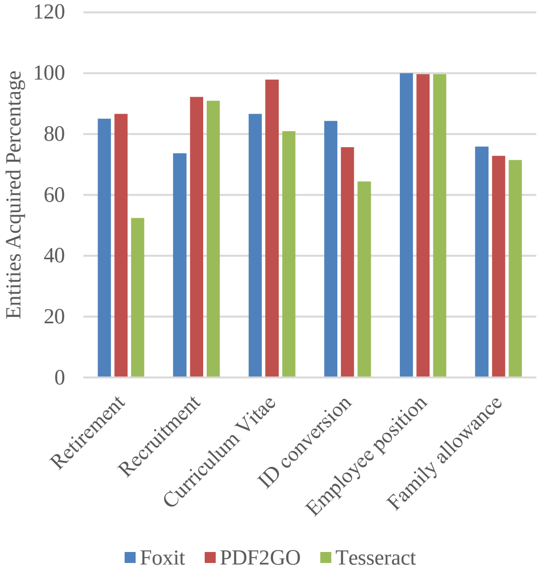


Fig. 7. Recall Percentage for each Category.

Tesseract has the lowest number of entities acquired with only 74.96% recall. PDF2GO and Foxit each have three dominant categories in the number of entities gained. On average, PDF2GO has the highest entities acquired with 87.83%, followed by Foxit with 84.26% and Tesseract with 74.96% entities acquired.

B. Discussion

We found a trade-off selection of the OCR engine from the measurements we conducted since Tesseract is good on preprocessing, information extraction time, and accuracy but lacks entity acquirement. On the other hand, PDF2GO and Foxit are dominant for entity acquirement. It lacks accuracy, extraction time, and preprocessing phase since Foxit has no batch conversion feature, and PDF2GO limits the batch conversion of 500 documents per batch.

It shows that Tesseract seems to be the best option for NER in scanned documents regardless of the number of entities acquired. The dominant measurement for Tesseract in preprocessing time, NER time, and precision proved that this OCR engine is reliable to get time efficiency and string match accuracy even though it has a low percentage of recall. The gap between Tesseract and PDF2GO on entities acquired is 13%.

The number of entities acquired is significant since having fast preprocessing and NER time would be meaningless if we have low numbers of information extracted. Writing a better algorithm in rule-based NER, having an automated text normalization, or using deep learning methods for NER might increase the number of entities acquired.

PDF2GO has the best F1-Score with 86.27%, regardless it has a slower NER time with 0.046 seconds per document and slower preprocessing time with 500 documents limit per batch and additional download and upload time to the PDF2GO web.

Even though PDF2GO has the best F1-Score, we found that Foxit has better precision in tabular document structure (retirement, CV, ID conversion, family allowance), as shown in Table IV. For tabular document structure, Foxit has an 84.01% average F1-Score followed by PDF2GO with 83.89% and Tesseract with 72.85%. For non-table document structure (retirement and employee position), Tesseract has the best F1-Score with 92.46%, followed by PDF2GO with 91.01% and Foxit with 85.86%.

Even though the F1, precision, and recall, and OCR engines are less than the previous research [10], we processed many more documents, categories, and documents structure. It consists of 8,562 documents within six categories and two document structures to prove that the OCR comparison using more documents in two different structures may have different results. The previous research [10] only employed string matching as a measurement to compare OCR engine results. In contrast, our study performed at least five measurements, which are preprocessing time, NER time, precision, recall, and F1-Score. Those measurements are more comprehensive than just string matching for an end-to-end information extraction system to manage many documents.

Similar research employed the Tesseract library [16] with only 11 images as input yielding 69.7% precision. On the other hand, our study produced 83.07% precision with 8,562 documents as the same library input. It may not be a fair comparison since the previous research using English text documents while our documents are in the Indonesian language. Tesseract support both English and Indonesian language, therefore even using more variety of document categories and different languages, Tesseract can have a better accuracy result.

Previous research that also performed rule-based NER [11] has 89% F1-Score while our study has 86.27% F1-Score. The preceding experiment employed generated PDF documents with no OCR involves, while ours used scanned documents, and it has the worst text result after the OCR process. In this case, there is a 2.73% gap between scanned documents and generated PDF using the same rule-based NER method. The experiments' results are a potential reference for an end-to-end information extraction system using a vast number of scanned documents involving an OCR engine.

C. Limitations

There are several limitaions on this study such as:

- We had only use tabular and non-tabular document structures in six categories.
- Six document categories were selected from ten available categories based on the head of National Civil Service Agency regulation number 18 in 2011.
- The number of entities for each document category were selected based on esential informations that stored in the human resources database of Bogor local government.
- We only measure the precision through the NER results, therefore we do not use the whole text of the scanned documents since only essential entities such as names, dates, organizations are being extracted and compared to the ground truth from the database.
- We only compared three OCR engines which are Foxit as an offline engine, PDF2GO as an online engine and Tesseract as an opensource engine.

V. Conclusion

This research analyses which OCR engine is the most suitable for IE using rule-based NER written in Python over 8,562 scanned PDF government human resources documents within six document categories. Those categories are retirement, recruitment, Curriculum Vitae, ID conversion, employee position, and family allowance documents with tabular and non-tabular structures. Involving more document structures such as watermark and handwriting scanned documents may have a better data representation for future research.

Three OCR engines from three different environments were compared during the experiment. Foxit as an offline desktop OCR engine, PDF2GO as an online OCR engine, and Tesseract as a free and open-source multiplatform OCR engine library. Tesseract is the most suitable solution in terms of preprocessing since it provides an unlimited document conversion batch. Tesseract is also the fastest OCR engine in NER extraction time; nevertheless, numbers of entities acquired Foxit and PDF2GO are more dominant on three documents each. In terms of string match entity accuracy as precision, the Tesseract OCR engine is dominant on three document categories.

On average, PDF2GO has the highest F1-score (86.27%). In terms of tabular document structure, Foxit has the highest F1-Score (84.01%). Tesseract has the highest F1-Score (92.46%) for non-table document structure. Those scores show that Tesseract is more suitable for non-table documents and Foxit is more appropriate for tabular documents.

Acknowledgment

This study was supported by research grants from Universitas Indonesia (Hibah Publikasi Doktoral Tahun Anggaran 2020 and Publikasi Terindeks International (PUTI) Doktor 2020 No: NKB-3226/UN2.RST/HKP.05.00/2020).

References

[1] S. Pattel, D. Bhatt. Abstractive Information Extraction from Scanned Invoices (AIESI) using end-to-end sequential approach. arXiv preprint arXiv:2009.05728.

[2] A. Nguyen, J. O'Dwyer, T. Vu, P.M Webb,S.E Johnatty, A.B Spurdle. "Generating high-quality data abstractions from scanned clinical records: text-mining-assisted extraction of endometrial carcinoma pathology features as proof of principle". BMJ Open. June 2020.

[3] L. Bures, P. Neduchal, L. Müller. "Automatic information extraction from scanned documents" in SPECOM: International Conference on Speech and Computer, Lecture Notes in Computer Science , vol. 12335, p. 87-96, Springer, Cham, 2020.

[4] M. Rastogi, S.A. Ali, M. Rawat, L. Vig, P. Agarwal, G. Shroff, A. Srinivasan. "Information extraction from document images via FCA based template detection and knowledge graph rule induction" in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

[5] RSS. Kumari, R. Sangeetha. Optical character recognition for document and newspaper. International Journal of Applied Engineering Research, Vol 10, pp. 15279-15285, 2015.

[6] K. Adnan. R Akbar. "Limitations of information extraction methods and techniques for heterogeneous unstructured big data". International Journal of Engineering Business Management., vol. 11, pp. 1-23, 2019.

[7] D. Nadeau, S. Sekine. "A survey of named entity recognition and classification". Lingvistica Investigationes"., vol 30., pp 3-26, 2007.

[8] K. Taghva, R. Beckleyy, J. Coombs. "The effects of OCR error on the extraction of private information" in Bunke H., Spitz A.L. (eds) Document Analysis Systems VII. DAS 2006. Lecture Notes in Computer Science, vol 3872. Springer, Berlin, Heidelberg.

[9] R. Pereda . Information Extraction in an Optical Character Recognition Context. University of Nevada, Las Vegas. 2011.

[10] S. Vijayarani, A. Sakila. "Performance comparison of OCR tools" in International Journal of UbiComp (IJU), Vol.6, No.3, July 2015

[11] F. Solihin, I. Budi. "Recording of law enforcement based on court decision document using rule-based information extraction" in 2018 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2018.

[12] E. Leitner , G. Rehm , J. Moreno-Schneider. "Fine-grained named entity recognition in legal documents" in Semantic Systems. The Power of AI and Knowledge Graphs, 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9–12, 2019.

[13] E.Q. Nuranti, E. Yulianti. "Legal entity recognition in Indonesian court decision documents using Bi-LSTM and CRF approaches" in 2020 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2020.

[14] C.I Patel, A. Patel, D. Patel. "Optical character recognition by open-source OCR tool tesseract: a case study". International Journal of Computer Applications. pp 50-56. 2012.

[15] V. Kumar , P. Kaware, P. Singh, R. Sonkusare. "Extraction of information from bill receipts using optical character recognition" in Proceedings of the International Conference on Smart Electronics and Communication, ICOSEC 2020.

[16] D. Akinbade , A.O. Ogunde, M.O. Odim, B.O. Oguntunde. "An adaptive thresholding algorithm-based optical character recognition system for information extraction in complex images". Journal of Computer Science. Science Publications. pp. 784-801. 2020.

[17] A.E. Harraj, N. Raaissouni. "OCR accuracy improvement on document images through a novel preprocessing approuch" in Signal & Image Processing : An International Journal (SIPIJ) Vol.6, No.4, August 2015.

[18] C.C. Aggarwal, C. Zhai. Mining text data. Springer ISBN 978-1-4614-3222-7. 2012.

[19] National Civil Service Agency. The head of National Civil Service Agency regulation number 18 in 2011 about manuscript layout management of the national human resources document. National Civil Service Agency of Indonesia. 2011.

[20] J. Ratcliff, D. Metzener, "Pattern Matching: The Gestalt Approach", Dr. Dobb's Journal, Vol. 13, No. 7, pp. 46-72 1988.