p-ISSN: 2252-3006
e-ISSN: 2685-2411

Comparison of Kernel Support Vector Machine in Predicting Judges' Decisions at the Bekasi District Court

Harry Dwiyana Kartika[a1], Getah Ester Hayatulah[b2], Ali Khumaidi[a3]
[a]Faculty of Engineering, Universitas Krisnadwipayana, Indonesia
[b]Faculty of Law, Universitas Krisnadwipayana, Indonesia
e-mail: [1]harry_d@unkris.ac.id, [2]getahester@unkris.ac.id, [3]alikhumaidi@unkris.ac.id

Abstrak

Proses persidangan suatu perkara pidana di Pengadilan Negeri Bekasi tahun 2019-2021 rata-rata lama proses untuk memutuskan perkara oleh hakim adalah 65-an hari. Pada penelitian ini mengusulkan penggunaan machine learning sebagai alat bantu untuk mempercepat keputusan hakim. Data penelitian yang digunakan adalah jenis acara pidana biasa dengan status perkara minutasi yang dipublikasikan sebanyak 1.642 kasus. Proses pengolahan data mengunakan python dengan preprocessing data case folding, remove punctuation, tokenization dan removal stopword kemudian untuk pembobotan kata menggunakan TF-IDF. Untuk memprediksi putusan lama pemidanaan menggunakan pendekatan klasifikasi Support Vector Machine. Hasil perbandingan pemodelan klasifikasi menggunakan SVM dengan 4 kernel yaitu linear (89,4%), RBF (88,4%), sigmoid (88,4%), dan polynomial (89,1%). Kernel SVM terbaik adalah kernel linear dengan nilai akurasi sebesar 89,4% dan nilai error sebesar 10,6%.

Kata kunci: *Kernel, Pengadilan Bekasi, Prediksi, Putusan Hakim, SVM*

Abstract

The trial process of a criminal case at the Bekasi District Court in 2019-2021 the average length of the process for deciding cases by judges is 65 days. This study proposes the use of machine learning as a tool to speed up judge decisions. The research data used is the type of ordinary criminal procedure with the status of published cases of 1,642 minutations. The data processing uses python with preprocessing case folding data, remove punctuation, tokenization and stopword removal then for word weighting using TF-IDF. To predict the long-term sentencing decision using the Support Vector Machine classification approach. The results of the comparison of classification modeling using SVM with 4 kernels are linear (89.4%), RBF (88.4%), sigmoid (88.4%), and polynomial (89.1%). The best SVM kernel is a linear kernel with an accuracy value of 89.4% and an error value of 10.6%.

Keywords : *Bekasi Court, Judge's Decision, Kernels, Prediction, SVM*

1. Introduction

The industrial revolution 4.0 paradigm demands human change, starting with the way of life and thinking and relationships with others. This era replaced the entire old system with new ways in various human activities, not only in the field of technology but also in other fields such as economics, society, and politics so that all aspects of the line began to be automated. Automation in this era is converged with internet network technology. This is a trend of automation and data exchange in manufacturing technology that is used to achieve maximum efficiency so that it can produce a new digital-based model [1]. Technological innovation is very much needed in this era because it is an opportunity for businesses to remain competitive in the midst of increasing operational costs and shifting economic demands. In Indonesia, too, currently working on the concept of the Industrial Revolution 4.0 entitled Making Indonesia 4.0 through the Ministry of Industry.

In the era of the industrial revolution 4.0, courts were directed to provide accessibility and reuse of public sector information and to publish cases considered online. The idea of

automation is not something new in the legal field. In the 1990s there was Lexis-Nexis and Westlaw which made it easier to find data. Several technologies that support other areas of law have also been developed. In forensic linguistics using text classification, detection of a person's personality, gender identity and age [2]. Doctrinal research methods have been used for a long time, which include the interpretation of laws and practical problem solving, over time have developed towards a systematic, innovative with simplification [3]. The systematic exposition of rules on doctrinal law develops with the ability to analyze the relationship of probability, difficulty and prediction of the future [4]. With the documentation of the court decision process that can be accessed by the public and the development of machine learning, several studies that facilitate understanding cases using a quantitative approach are applied in understanding traditional law.[5], [6].

In the trial process of a criminal case at the Bekasi District Court in 2019-2021, the average length of the process required to decide a case by a judge is 65 days. The use of machine learning and technology can help ease the work of judges in deciding cases. One solution to the problem that is currently being developed is to use artificial intelligence technology. The ability of artificial intelligence related to human language processing, especially in court claims used in this study is the natural language processing (NLP) method [7]. NLP is a method that examines the interaction between humans and computers using human language [8]. The hope of this research is that it can assist judges in determining decisions so that case handling can be completed faster and more efficiently. The automation is solely to assist the performance of judges in identifying and extracting patterns that lead to the decision of a case..

2. Research Method

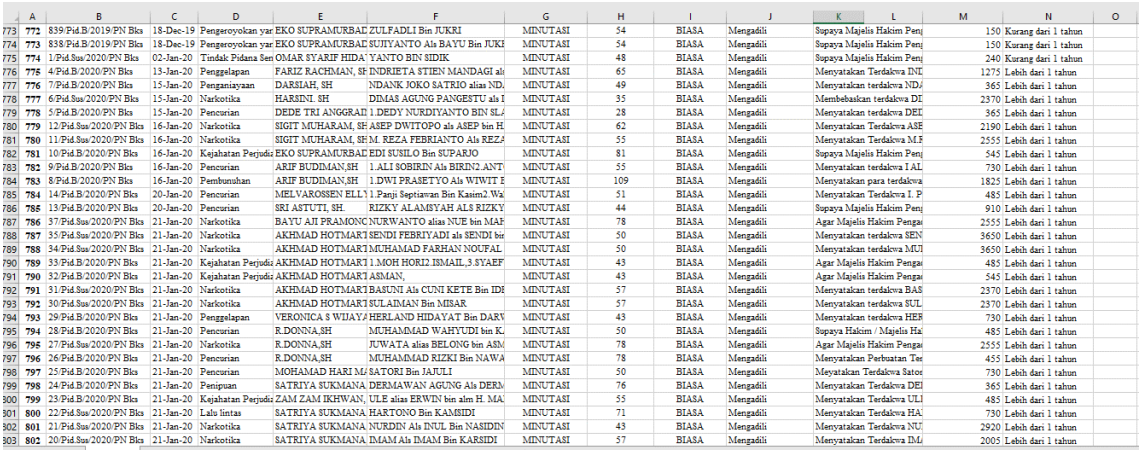
The data used in the prediction modeling are criminal cases in the Bekasi District Court for 2 years, 2019-2021. The type of data used is secondary data in the form of recapitulation of criminal cases at the Bekasi District Court from January 2019 to January 2021, totaling 1,642 cases in the form of prosecutions for ordinary crimes and decisions based on the length of a sentence with the status of a minutation case. Figure 1 is a sample of raw data before it is processed into machine learning.

The variables used in this research plan are the Letter of Demand (demand) and the Length of the Sentence (Decision). A lawsuit is a letter that makes proof of an indictment based on evidence that has been revealed at trial. In addition, it is the conclusion of the public prosecutor regarding the guilt of the defendant which is accompanied by a criminal charge. The verdict is the detention of a person's freedom for committing a crime. The class is determined based on the maximum length of sentence for each type of crime. If the sentence is 1 year, it will be included in the class for more than one year, if 1 year, it will be included in the class for less than one year.

The software used in this writing is Microsoft Excel 2021, Python, and Anaconda3-2022.05-Windows-x86_64. There are data analysis methods used in this study, namely:

1. Text Mining, in this case, the data preprocessing used is case folding, remove punctuation, stopword removal, and tokenization. Then for word weighting using TF-IDF.
2. The SVM method is used to classify and predict the length of sentencing. The SVM used is linear SVM, RBF, sigmoid, and polynomial.

Figure 2 is the research stage in the development of a judge's decision prediction classification model.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
773	772	839/Pid.B/2019/PN Bks	18-Dec-19	Pengeroyokan yar	EKO SUPRAMURBAC	ZULFADLI Bin JUKRI	MINUTASI	54	BIASA	Mengadili	Supaya Majelis Hakim Pers		150	Kurang dari 1 tahun	
774	773	838/Pid.B/2019/PN Bks	18-Dec-19	Pengeroyokan yar	EKO SUPRAMURBAC	SUJIYANTO Als BAYU Bin JUKI	MINUTASI	54	BIASA	Mengadili	Supaya Majelis Hakim Pers		150	Kurang dari 1 tahun	
775	774	1/Pid.Sus/2020/PN Bks	02-Jan-20	Tindak Pidana Ser	OMAR SYARIF HIDA	YANTO BIN SIDIK	MINUTASI	48	BIASA	Mengadili	Supaya Majelis Hakim Pers		240	Kurang dari 1 tahun	
776	775	4/Pid.B/2020/PN Bks	13-Jan-20	Penggelapan	FARIZ RACHMAN, SH	INDRIETA STIEN MANDAGI ali	MINUTASI	65	BIASA	Mengadili	Menyatakan Terdakwa IND		1275	Lebih dari 1 tahun	
777	776	7/Pid.B/2020/PN Bks	15-Jan-20	Penganiayaan	DARSIAH, SH	NDANK JOKO SATRIO alias ND	MINUTASI	49	BIASA	Mengadili	Menyatakan terdakwa NDA		365	Lebih dari 1 tahun	
778	777	6/Pid.Sus/2020/PN Bks	15-Jan-20	Narkotika	HARSINI. SH	DIMAS AGUNG PANGESTU als I	MINUTASI	35	BIASA	Mengadili	Membebaskan terdakwa DI		2370	Lebih dari 1 tahun	
779	778	5/Pid.B/2020/PN Bks	15-Jan-20	Pencurian	DEDE TRI ANGGRAI	1.DEDY NURDIYANTO BIN SL	MINUTASI	28	BIASA	Mengadili	Menyatakan terdakwa DEI		365	Lebih dari 1 tahun	
780	779	12/Pid.Sus/2020/PN Bks	16-Jan-20	Narkotika	SIGIT MUHARAM, SH	ASEP DWITOPO als ASEP bin H	MINUTASI	62	BIASA	Mengadili	Menyatakan Terdakwa ASE		2190	Lebih dari 1 tahun	
781	780	11/Pid.Sus/2020/PN Bks	16-Jan-20	Narkotika	SIGIT MUHARAM, SH	M. REZA FEBRIANTO Als REZA	MINUTASI	55	BIASA	Mengadili	Menyatakan Terdakwa M.F		2555	Lebih dari 1 tahun	
782	781	10/Pid.B/2020/PN Bks	16-Jan-20	Kejahatan Perjudi	EKO SUPRAMURBAC	EDI SUSILO Bin SUPARJO	MINUTASI	81	BIASA	Mengadili	Supaya Majelis Hakim Pers		545	Lebih dari 1 tahun	
783	782	9/Pid.B/2020/PN Bks	16-Jan-20	Pencurian	ARIF BUDIMAN,SH	1.ALI SOBIRIN Als BIRIN2.ANT	MINUTASI	55	BIASA	Mengadili	Menyatakan terdakwa LAL		730	Lebih dari 1 tahun	
784	783	8/Pid.B/2020/PN Bks	16-Jan-20	Pembunuhan	ARIF BUDIMAN,SH	1.DWI PRASETYO Als WIWIT E	MINUTASI	109	BIASA	Mengadili	Menyatakan para terdakwa		1825	Lebih dari 1 tahun	
785	784	14/Pid.B/2020/PN Bks	20-Jan-20	Pencurian	MELVAROSSEN ELLY	1.Panji Septiawan Bin Kasim2.Wa	MINUTASI	51	BIASA	Mengadili	Menyatakan Terdakwa I. P		485	Lebih dari 1 tahun	
786	785	13/Pid.B/2020/PN Bks	20-Jan-20	Pencurian	SRI ASTUTI, SH.	RIZKY ALAMSYAH ALS RIZKY	MINUTASI	44	BIASA	Mengadili	Supaya Majelis Hakim Pers		910	Lebih dari 1 tahun	
787	786	37/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	BAYU AJI PRAMONO	NURWANTO alias NUE bin MAH	MINUTASI	78	BIASA	Mengadili	Agar Majelis Hakim Penga		2555	Lebih dari 1 tahun	
788	787	35/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	AKHMAD HOTMARI	SENDI FEBRIYADI als SENDI bi	MINUTASI	50	BIASA	Mengadili	Menyatakan terdakwa SEN		3650	Lebih dari 1 tahun	
789	788	34/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	AKHMAD HOTMARI	MUHAMAD FARHAN NOUFAL	MINUTASI	50	BIASA	Mengadili	Menyatakan terdakwa MUI		3650	Lebih dari 1 tahun	
790	789	33/Pid.B/2020/PN Bks	21-Jan-20	Kejahatan Perjudi	AKHMAD HOTMARI	1.MOH HORI2.ISMAIL,3.SYAEF	MINUTASI	43	BIASA	Mengadili	Agar Majelis Hakim Penga		485	Lebih dari 1 tahun	
791	790	32/Pid.B/2020/PN Bks	21-Jan-20	Kejahatan Perjudi	AKHMAD HOTMARI	ASMAN,	MINUTASI	43	BIASA	Mengadili	Agar Majelis Hakim Penga		545	Lebih dari 1 tahun	
792	791	31/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	AKHMAD HOTMARI	BASUNI Als CUNI KETE Bin IDI	MINUTASI	57	BIASA	Mengadili	Menyatakan terdakwa BAS		2370	Lebih dari 1 tahun	
793	792	30/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	AKHMAD HOTMARI	SULAIMAN Bin MISAR	MINUTASI	57	BIASA	Mengadili	Menyatakan terdakwa SUL		2370	Lebih dari 1 tahun	
794	793	29/Pid.B/2020/PN Bks	21-Jan-20	Penggelapan	VERONICA S WIJAYA	HERLAND HIDAYAT Bin DARV	MINUTASI	43	BIASA	Mengadili	Menyatakan terdakwa HER		730	Lebih dari 1 tahun	
795	794	28/Pid.B/2020/PN Bks	21-Jan-20	Pencurian	R.DONNA,SH	MUHAMMAD WAHYUDI bin K	MINUTASI	50	BIASA	Mengadili	Supaya Hakim / Majelis Ha		485	Lebih dari 1 tahun	
796	795	27/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	R.DONNA,SH	JUWATA alias BELONG bin ASM	MINUTASI	78	BIASA	Mengadili	Agar Majelis Hakim Penga		2555	Lebih dari 1 tahun	
797	796	26/Pid.B/2020/PN Bks	21-Jan-20	Pencurian	R.DONNA,SH	MUHAMMAD RIZKI Bin NAWA	MINUTASI	78	BIASA	Mengadili	Menyatakan Perbuatan Ter		455	Lebih dari 1 tahun	
798	797	25/Pid.B/2020/PN Bks	21-Jan-20	Pencurian	MOHAMAD HARI MA	SATORI Bin JAJULI	MINUTASI	50	BIASA	Mengadili	Meyatakan Terdakwa Sator		730	Lebih dari 1 tahun	
799	798	24/Pid.B/2020/PN Bks	21-Jan-20	Penipuan	SATRIYA SUKMANA	DERMAWAN AGUNG Als DERM	MINUTASI	76	BIASA	Mengadili	Menyatakan Terdakwa DEI		365	Lebih dari 1 tahun	
800	799	23/Pid.B/2020/PN Bks	21-Jan-20	Kejahatan Perjudi	ZAM ZAM IKHWAN,	ULE alias ERWIN bin alm H. MA	MINUTASI	55	BIASA	Mengadili	Menyatakan Terdakwa ULI		485	Lebih dari 1 tahun	
801	800	22/Pid.Sus/2020/PN Bks	21-Jan-20	Lalu lintas	SATRIYA SUKMANA	HARTONO Bin KAMSIDI	MINUTASI	71	BIASA	Mengadili	Menyatakan Terdakwa HA		730	Lebih dari 1 tahun	
802	801	21/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	SATRIYA SUKMANA	NURDIN Als INUL Bin NASIDIN	MINUTASI	43	BIASA	Mengadili	Menyatakan Terdakwa NU		2920	Lebih dari 1 tahun	
803	802	20/Pid.Sus/2020/PN Bks	21-Jan-20	Narkotika	SATRIYA SUKMANA	IMAM Als IMAM Bin KARSIDI	MINUTASI	57	BIASA	Mengadili	Menyatakan Terdakwa IM		2005	Lebih dari 1 tahun	

Figure 1. Sample raw data

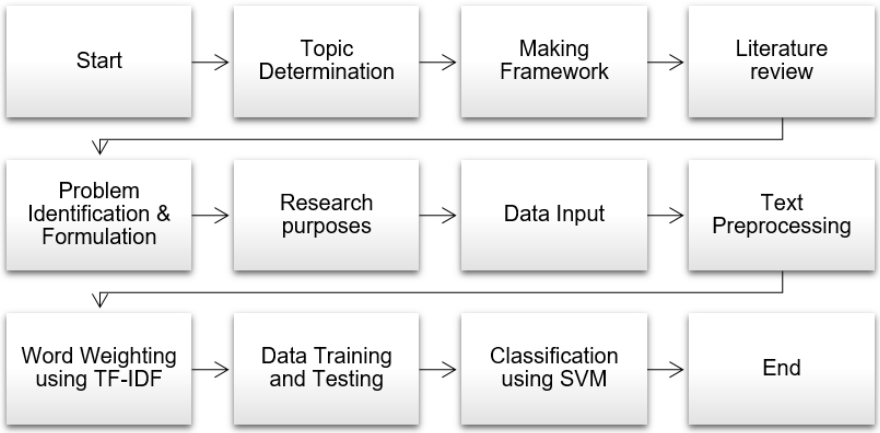


Figure 2. Research stages

SVM classification used is linear, polynomial, sigmoid, and RBF kernel [9]. Comparisons are needed to determine the accuracy of machine performance in classifying the four kernels. So the best kernel will be found. The stages can be seen in Figure 3.

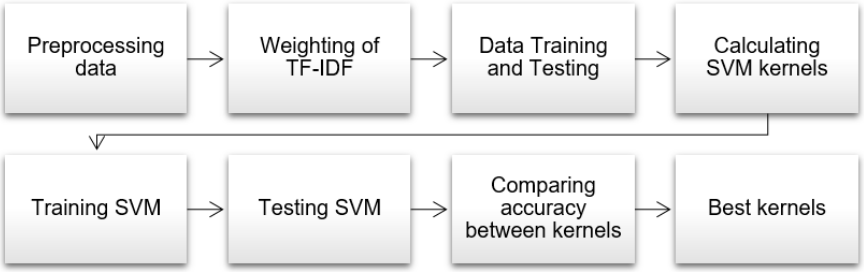


Figure 3. Stages of classification and prediction using SVM

3. Literature Study

3.1. Text mining

Text mining is a process that extracts information in large text collections and automatically identifies interesting patterns and relationships in textual data [10]. Text mining is a combination of several mathematical techniques, statistics, linguistics, data mining, machine learning, and information retrieval. The research stages include preprocessing and text extraction, analysis and statistical processing. Text mining will extract information from a collection of documents. The data used in text mining is a collection of text that has an

unstructured data format, or at least semistructured. The text mining process is divided into 3 main stages, namely, text preprocessing, pattern discovery, and text transformation [11].

1. Text Processing

   This stage is the initial stage in Text Mining. Text preprocessing is done to remove parts or text that are not needed so that they get quality data for execution or so that the mining process is more accurate [12]. In this study, the text preprocessing used is:

   a. Case folding: Converts all letters in the document to lowercase.
   b. Remove Punctuation: Remove punctuation in document
   c. Tokenization: Converts a bunch of text into words

2. Text Transformation
3. Text transformation is the process of representing the document. The concept used is the "bag of words" model and the vector space model. This process includes the formation of basic forms of words and reduction of dimensions in the document [13]. In feature generation, there is feature selection, namely the selection of features or the next stage of dimension reduction in the text transformation process. Feature selection used in this study are:

   a. Stopwords Removal: Eliminate words that are not characteristic (unique words) from a document.
   b. Stemming and Lemmatization: Transforms words contained in a document into root or base words

3.2. Weighting Term Frequency-Inverse Document Frequency

The Term Frequency-Inverse Document Frequency (TF-IDF) method is a method of assigning weight to the relationship of a word (term) to a document. Term weight is an indicator of the word, because the importance of each word is different [14]. Word weighting is influenced by the following:

1. Term Frequency (TF) is the weight of the word in the document which is determined from the occurrence of the word. The weight will increase with the number of occurrences of the word.
2. Inverse Document Frequency (IDF) is a method of checking for word dominance. Common terms can be reduced and words that rarely appear need attention.

3.3. Support Vector Machine (SVM)

The Support Vector Machine (SVM) by Boser, Guyon, Vapnik was introduced in 1992 [15]. The concept of SVM is how to form the best hyperplane in the input space. The best hyperplane is obtained when the margins and classes are at their maximum point. The outermost data distance from the class and hyperplane is called the margin. The closest pattern in each class is called the support vector [16]. SVM is a variant of the linear machine so it can only be used to solve linearly separable problems. However, in its implementation, linear data problems are rarely obtained. However, with the development of SVM, many SVM cases are non-linear. The solution to this case is to include the concept of a kernel trick in a high-dimensional workspace. These computational techniques are kernel tricks formulated in equation (1).

$$K(\vec{x}_i, \vec{x}_j) = \Phi(\vec{x}_i) . \Phi(\vec{x}_j) \quad (1)$$

The results of the classification of the data obtained equation (2).

$$f(\Phi(\vec{x})) = \vec{w} . \Phi(\vec{x}) + b \quad (2)$$

There are various types of functions that can be used as kernel K as shown in Table 1.

Table 1. Various Kernel Functions

Kernel Functions	Definition
Linear	$K(\vec{x}_i, \vec{x}_j) = \vec{x}_i, \vec{x}_j$

p-ISSN: 2252-3006
e-ISSN: 2685-2411

Radial Basic Function (RBF)	$\mathrm{K}(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2\sigma^2}\right)$
Sigmoid	$\mathrm{K}(\vec{x}_i, \vec{x}_j) = \tanh(\alpha\, \vec{x}_i \,.\, \vec{x}_j + \beta)$
Polynomial	$\mathrm{K}(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \,.\vec{x}_j + 1)^p$

4. Result and Discussion

4.1. Descriptive analysis

In this analysis, the researcher describes in general the case statistics report data (http://sipp.pn-bekasikota.go.id/statistik_perkara) of the Bekasi District Court from January 2019 to January 2021 regarding criminal cases. The number of criminal cases in the last 2 years, namely from January 2019 to January 2021, there were 18,326 cases. It is known that from 2019 to 2020 it has decreased. Meanwhile, in 2021 it cannot be explained because the data used is only 1 month data. This means that crimes and violations in Bekasi can be seen to have decreased in the last 2 years.

Criminal cases based on the type of criminal procedure are divided into 3 types of crimes, namely ordinary crimes, short crimes, and also quick crimes. Of the 3 types of criminal proceedings, it can be seen that the type of criminal procedure that has the most cases is ordinary crime. This means that in many cases the final form of the case is a decision. And in this study, researchers only used the type of ordinary criminal procedure, so the data used was 1,860 cases. Of the 1,860 cases based on the type of ordinary criminal procedure, they still had to be classified according to the variable sought with the status of a minutation case, and 123 cases were obtained with other status cases (revocation of cassation case, notification of cassation decision, notification of PK decision, notification of appeal decision, cassation decision, and etc). So that there were 1,737 cases with minutation cases. In addition, this study uses published criminal cases. The published cases are about 95% of the total criminal cases and the unpublished cases are about 5% of the total criminal cases which means the data is disguised. So that it can be seen that the cases that can be published and used by researchers are only 1,642 cases published with minutation cases or about 95% of the total ordinary criminal cases in the Bekasi District Court from January 2019 to January 2021.

The total number of data used for classification is 1642 cases. Where the amount of data for classes more than 1 year is more than the data for classes less than 1 year. This means that in the Bekasi District Court many criminal cases have been detained for more than one year. In Figure 4 is the word cloud of the demand variable. Word Cloud is a word that is often used in all documents, the larger the word size, the more often it is used. It can be seen that the words that are often used in the variable demands are the words 'article', 'paragraph', 'goods', 'evidence', 'amount', 'rp' and so on. There are quite a number of names appearing in the wordcloud because in each document the name is mentioned repeatedly.

p-ISSN: 2252-3006
e-ISSN: 2685-2411

Figure 4. Word cloud demand variables

4.2. Data Preprocessing

The first stage before data classification is preprocessing the data first. It aims to clean and eliminate unnecessary text in the data. There are several stages of data preprocessing used in this study, such as Case Folding, Removing Punctuation, Tokenize, and Stopwords Removal. Figure 5 is the initial data before preprocessing the data.

```
In [1]: import pandas as pd
        import numpy as np
        from wordcloud import WordCloud
        import matplotlib.pyplot as plt
        from nltk import FreqDist
        import seaborn as sns
        df = pd.read_excel('DATABARU.xlsx')
        df
```

Out[1]:

	TERDAKWA	STATUS PERKARA	LAMA PROSES	JENIS PERKARA	PUTUSAN	TUNTUTAN	KLASIFIKASI	LAMA KURUNGAN	LAMA PEMIDANAAN
0	NASRUL Als UDA Bin MARALI	MINUTASI	63	BIASA	Mengadili	Supaya Majelis Hakim Pengadilan Negeri Bekasi ...	Tanpa hak dan melawan hukum membawa, memiliki,...	365	Lebih dari 1 tahun
1	FETRA FAREL Als EMPET Bin UPIN	MINUTASI	35	BIASA	Mengadili	Menyatakan Terdakwa FETRA FAREL Als EMPET Bin ...	Percobaan pencurian	1825	Lebih dari 1 tahun
2	SALANTY KOESMANTO alias SALLY	MINUTASI	63	BIASA	Mengadili	Menyatakan Terdakwa SALANTY KOESMANTO aliasa S...	Tindak pidana penipuan	365	Lebih dari 1 tahun
3	AHMAD HUMAIDI Bin H SARBINI	MINUTASI	61	BIASA	Mengadili	Supaya Majelis Hakim Pengadilan Negeri Kota Be...	Tanpa hak menguasai atau memiliki Narkotika Go...	1640	Lebih dari 1 tahun
4	1.RIAN PRATAMA Als JOKER Bin YOSEP ERWANTO\n2....	MINUTASI	35	BIASA	Mengadili	Menyatakan terdakwa I Rian Pratama Als Joker B...	Pencurian dalam keadaan memberatkan	300	Kurang dari 1 tahun
...	...	...	...	...	...	...	...	...	...
354	ABDUL AZIZ AZZABADI Als REZA Bin ABDUL	MINUTASI	47	BIASA	Mengadili	Supaya Majelis Hakim Pengadilan Negeri	Tanpa Hak atau melawan hukum	2005	Lebih dari 1 tahun

Figure 5. Initial data before preprocessing

At the case folding stage, all uppercase words are changed to all lowercase letters, especially in the TUNTUTAN and LAMA PEMIDANAAN columns can be seen as in Figure 6. After case folding is done, after that, the characters in words other than letters are deleted, as shown in Figure 7. And Figure 8 is the tokenize result that is applied to the dataset.

p-ISSN: 2252-3006
e-ISSN: 2685-2411

```
In [2]: #case folding berupa lower case
        df['TUNTUTAN'] = df['TUNTUTAN'].apply(lambda x: " ".join(x.lower() for x in x.split()))
        df
```

Out[2]:

	TERDAKWA	STATUS PERKARA	LAMA PROSES	JENIS PERKARA	PUTUSAN	TUNTUTAN	KLASIFIKASI	LAMA KURUNGAN	LAMA PEMIDANAAN
0	NASRUL Als UDA Bin MARALI	MINUTASI	63	BIASA	Mengadili	supaya majelis hakim pengadilan negeri bekasi ...	Tanpa hak dan melawan hukum membawa, memiliki,...	365	Lebih dari 1 tahun
1	FETRA FAREL Als EMPET Bin UPIN	MINUTASI	35	BIASA	Mengadili	menyatakan terdakwa fetra farel als empet bin ...	Percobaan pencurian	1825	Lebih dari 1 tahun
2	SALANTY KOESMANTO alias SALLY	MINUTASI	63	BIASA	Mengadili	menyatakan terdakwa salanty koesmanto aliasa s...	Tindak pidana penipuan	365	Lebih dari 1 tahun
3	AHMAD HUMAIDI Bin H SARBINI	MINUTASI	61	BIASA	Mengadili	supaya majelis hakim pengadilan negeri kota be...	Tanpa hak menguasai atau memiliki Narkotika Go...	1640	Lebih dari 1 tahun
4	1.RIAN PRATAMA Als JOKER Bin YOSEP ERWANTO\n2....	MINUTASI	35	BIASA	Mengadili	menyatakan terdakwa i rian pratama als joker b...	Pencurian dalam keadaan memberatkan	300	Kurang dari 1 tahun
...	...	...	...	...	...	...	...	...	...
354	ABDUL AZIZ AZZABADI Als REZA Bin ABDUL BASIT	MINUTASI	47	BIASA	Mengadili	supaya majelis hakim pengadilan negeri bekasi ...	Tanpa Hak atau melawan hukum menjadi perantara...	2005	Lebih dari 1 tahun
355	FIRMANSYAH als EMPE Bin Alm ACA SUDIRJA	MINUTASI	76	BIASA	Mengadili	supaya majelis hakim pengadilan negeri bekasi ...	Tanpa hak atau melawan hukum menjual, membeli ...	2555	Lebih dari 1 tahun
356	ARDI WIBOWO ALS	MINUTASI	54	BIASA	Mengadili	supaya majelis hakim	Tindak pidana dalam	1460	Lebih dari 1

Figure 6. Case folding results

```
In [3]: #menghapus karakter selain huruf
        df['TUNTUTAN'] = df['TUNTUTAN'].str.replace('[^\w\s]','')
        df['TUNTUTAN'] = df['TUNTUTAN'].str.replace('\d+', '')
        import string
        printable = set(string.printable)
        def remove_spec_chars(in_str):
            return ''.join([c for c in in_str if c in printable])

        df['TUNTUTAN'].apply(remove_spec_chars)
        df['TUNTUTAN'].head(10)

C:\Users\Jeje\AppData\Local\Temp\ipykernel_7340\2134506378.py:2: FutureWarning: The default value of regex will change from
e to False in a future version.
  df['TUNTUTAN'] = df['TUNTUTAN'].str.replace('[^\w\s]','')
C:\Users\Jeje\AppData\Local\Temp\ipykernel_7340\2134506378.py:3: FutureWarning: The default value of regex will change from
e to False in a future version.
  df['TUNTUTAN'] = df['TUNTUTAN'].str.replace('\d+', '')

Out[3]: 0    supaya majelis hakim pengadilan negeri bekasi ...
        1    menyatakan terdakwa fetra farel als empet bin ...
        2    menyatakan terdakwa salanty koesmanto aliasa s...
        3    supaya majelis hakim pengadilan negeri kota be...
        4    menyatakan terdakwa i rian pratama als joker b...
        5    menyatakan terdakwa samuhaji bin jasmin terbuk...
        6    menyatakan terdakwa fatur rahman bin syahrudin...
        7    menyatakan terdakwa anggi banur destian bin ba...
        8    menyatakan terdakwa dedi wahyudi bin supendi t...
        9    menyatakan terdakwa aan febriansyah bin saan t...
        Name: TUNTUTAN, dtype: object
```

Figure 7. Results of removing punctuation

```
In [4]: # Menerapkan fungsi Tokenize pada Dataset
        import nltk
        nltk.download('punkt')
        def identify_tokens(row):
            text = row
            tokens = nltk.word_tokenize(text)
            # taken only words (not punctuation)
            token_words = [w for w in tokens if w.isalpha()]
            return token_words

        df['token'] = df['TUNTUTAN'].apply(identify_tokens)
        df['token'].head(5)

[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Jeje\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!

Out[4]: 0    [supaya, majelis, hakim, pengadilan, negeri, b...
        1    [menyatakan, terdakwa, fetra, farel, als, empe...
        2    [menyatakan, terdakwa, salanty, koesmanto, ali...
        3    [supaya, majelis, hakim, pengadilan, negeri, k...
        4    [menyatakan, terdakwa, i, rian, pratama, als, ...
        Name: token, dtype: object
```

Figure 8. Tokenize results

p-ISSN: 2252-3006
e-ISSN: 2685-2411

Furthermore, in the text transformation at the stopwords removal stage, words that are not important or meaningless and not included in the dictionary will be deleted, such as 'dia', 'dua', 'ia', 'seperti', 'jika', and so on, can be seen in Figure 9.

```
stop_factory = StopWordRemoverFactory()
data_stopword = stop_factory.get_stop_words()
stopword = stop_factory.create_stop_word_remover()
print(data_stopword)
```

```
Requirement already satisfied: sastrawi in c:\users\jeje\anaconda3\lib\site-packages (1.0.1)
['yang', 'untuk', 'pada', 'ke', 'para', 'namun', 'menurut', 'antara', 'dia', 'dua', 'ia', 'seperti', 'jika', 'jika', 'sehingg
a', 'kembali', 'dan', 'tidak', 'ini', 'karena', 'kepada', 'oleh', 'saat', 'harus', 'sementara', 'setelah', 'belum', 'kami', 'se
kitar', 'bagi', 'serta', 'di', 'dari', 'telah', 'sebagai', 'masih', 'hal', 'ketika', 'adalah', 'itu', 'dalam', 'bisa', 'bahwa',
'atau', 'hanya', 'kita', 'dengan', 'akan', 'juga', 'ada', 'mereka', 'sudah', 'saya', 'terhadap', 'secara', 'agar', 'lain', 'and
a', 'begitu', 'mengapa', 'kenapa', 'yaitu', 'yakni', 'daripada', 'itulah', 'lagi', 'maka', 'tentang', 'demi', 'dimana', 'keman
a', 'pula', 'sambil', 'sebelum', 'sesudah', 'supaya', 'guna', 'kah', 'pun', 'sampai', 'sedangkan', 'selagi', 'sementara', 'teta
pi', 'apakah', 'kecuali', 'sebab', 'selain', 'seolah', 'seraya', 'seterusnya', 'tanpa', 'agak', 'boleh', 'dapat', 'dsb', 'dst',
'dll', 'dahulu', 'dulunya', 'anu', 'demikian', 'tapi', 'ingin', 'juga', 'nggak', 'mari', 'nanti', 'melainkan', 'oh', 'ok', 'seh
arusnya', 'sebetulnya', 'setiap', 'setidaknya', 'sesuatu', 'pasti', 'saja', 'toh', 'ya', 'walau', 'tolong', 'tentu', 'amat', 'a
palagi', 'bagaimanapun']
```

```
In [6]: # Menerapkan fungsi Stopword pada Dataset, dan membuat kolom stopwords
def stop_list(row):
    my_list = row
    stop_list = [stopword.remove(i) for i in my_list]
    return (stop_list)
```

```
In [7]: df['stop_words'] = df['token'].apply(stop_list)
df['stop_words'].head(5)
```

```
Out[7]: 0    [, majelis, hakim, pengadilan, negeri, bekasi,...
1    [menyatakan, terdakwa, fetra, farel, als, empe...
2    [menyatakan, terdakwa, salanty, koesmanto, ali...
3    [, majelis, hakim, pengadilan, negeri, kota, b...
4    [menyatakan, terdakwa, i, rian, pratama, als, ...
Name: stop_words, dtype: object
```

Figure 9. Stopwords removal results

4.3. Weighting of TF-IDF

After preprocessing the data, in Figure 10 is a sample of the words used to show the results of the TF-IDF word weighting.

```
In [12]: #melakukan ekstraksi fitur menggunakan tf-idf
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer= TfidfVectorizer()
X = vectorizer.fit_transform(df['Text'])
vector=pd.DataFrame(X.todense().T,index=vectorizer.get_feature_names(),columns=[f'D{i+1}' for i in range(len(df['TUNTUTAN']))])
vector.tail()
```

```
C:\Users\Jeje\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:87: FutureWarning: Function get_feature_names is depreca
ted; get_feature_names is deprecated in 1.0 and will be removed in 1.2. Please use get_feature_names_out instead.
  warnings.warn(msg, category=FutureWarning)
```

Out[12]:

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	...	D1633	D1634	D1635	D1636	D1637	D1638	D1639	D1640	D1641	D1642
zulkifli	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
zulkifly	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
zulkipli	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
zumhana	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
zv	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 1642 columns

Figure 10. Weighting of TF-IDF results

4.4. SVM Modeling

Before classifying using SVM, we first determine the training data and testing data. In this study, the distribution of training data and testing data is 80:20. With 1313 training data and 329 testing data. The distribution of data can be seen in Table 2.

Table 2. Splitting of training and testing data

Class	Data Training	Data Testing	Total
More than 1 year	1198	300	1498
Less than 1 year	115	29	144
Total	1313	329	1642

After splitting training and testing data, modelling the classification using SVM. In this study, used 4 kernels, namely linear, RBF, Sigmoid, and Polynomial as well as several choices of parameters C = 0.1, 1, 10, 100, 1000 and gamma parameters = 1, 0.1, 0.01, 0.001, 0.0001. To get the best parameters from several parameter choices, uses a tuning process which then looks for the best accuracy value from the 4 kernels. The modeling process can be seen in Figure 11 and the results of the comparison of the four SVM kernels can be seen in Table 3.

p-ISSN: 2252-3006
e-ISSN: 2685-2411

Based on the accuracy value generated by the SVM model using a different kernel, a fairly good accuracy is obtained with the smallest accuracy of 88.4% using the RBF and Sigmoid kernels. The use of Linear and Polynomial kernels increases their accuracy, the use of Linear kernels has the highest accuracy of 89.4% and the Polynomial kernel has a lower accuracy of 89.1%.

Table 3. Kernels comparison results on SVM

Kernel	Accuracy
Linear	89,4 %
RBF	88,4%
Sigmoid	88,4%
Poynomial	89,1%

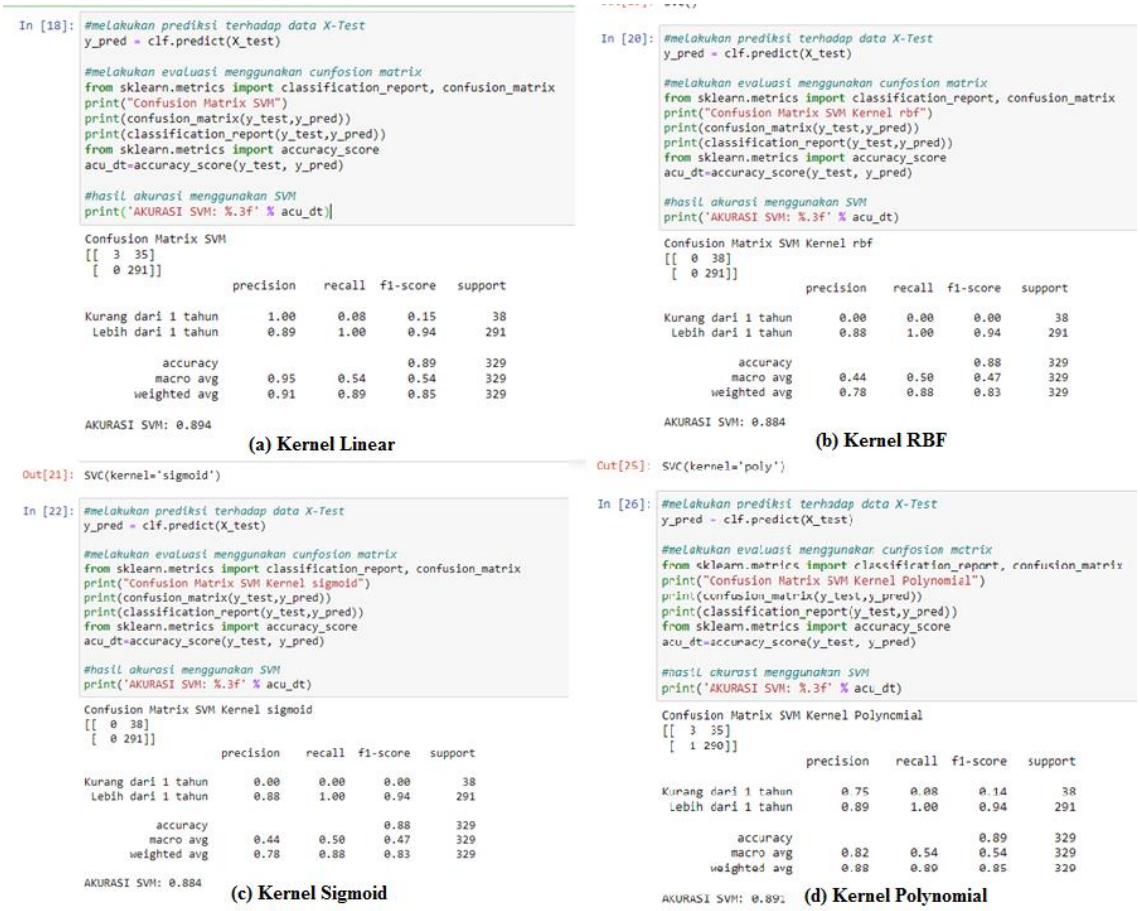


Figure 11. Process and results of SVM modeling with 4 kernels

5. Conclusion

In the research to predict the judge's decision at the Bekasi District Court, the data used is the number of criminal cases from January 2019 to January 2021. From 18,326 criminal cases based on the type of criminal procedure, they are divided into 3 types of crimes, namely ordinary crimes, short crimes, and also quick crimes. In this study using the type of ordinary criminal procedure as many as 1,860 cases, but after being classified according to the variables sought with the status of minutation cases, the number of cases became 1,737 and only 1,642 criminal cases were published. Processing data using python with preprocessing data used are case folding, remove punctuation, stopword removal, and tokenization. Then for word weighting using TF-IDF. The SVM method is used for classification and prediction of the length of punishment, before modeling the data is split with a ratio of 80:20 and the results of the comparison of classification modeling using SVM with 4 kernels are linear (89.4%), rbf (88.4%), sigmoid (88 ,4%), and polynomials (89,1%). The results obtained that have the best kernel is a

linear kernel with an accuracy value of 89.4% and an error value of 10.6%. With the results of classification modeling using SVM which is quite high, it can be used as a judge's tool in predicting the length of sentencing at trial.

Acknowledgments

The authors would like to thank the Ministry of Education, Culture, Research and Technology (Kemdikbudristek) of the Republic of Indonesia for funding research in 2022.

References

[1] M. Wijaya, "Revolusi Industri 4.0 : Implikasi terhadap Manajemen Sumberdaya Manusia," *Media Inform.*, vol. 19, no. 2, pp. 51–60, Aug. 2020, doi: 10.37595/mediainfo.v19i2.41.

[2] M. N. Angelo Basile, Gareth Dwyer, Maria Medvedeva, Josine Rawee, Hessel Haagsma, "N-gram: New groningen author-profiling model," *Noteb. PAN CLEF*, 2017.

[3] L. Mai and E. Boulot, "Harnessing the transformative potential of Earth System Law: From theory to practice," *Earth Syst. Gov.*, vol. 7, p. 100103, Mar. 2021, doi: 10.1016/j.esg.2021.100103.

[4] S. D. Kebede and Z. Tiewei, "Public work contract laws on project delivery systems and their nexus with project efficiency: evidence from Ethiopia," *Heliyon*, vol. 7, no. 3, p. e06462, Mar. 2021, doi: 10.1016/j.heliyon.2021.e06462.

[5] M. Derlln and J. Lindholm, "Serving Two Masters: CJEU Case Law in Swedish First Instance Courts and National Courts of Precedence As Gatekeepers," *SSRN Electron. J.*, 2017, doi: 10.2139/ssrn.2952783.

[6] K. A. Olsen HP, "Finding hidden patterns in ECtHR's case law: On how citation network analysis can improve our knowledge of ECtHR's Article 14 practice," *Int J Discrim Law*, vol. 17, no. 1, pp. 4–22, 2017.

[7] E. Mumcuoğlu, C. E. Öztürk, H. M. Ozaktas, and A. Koç, "Natural language processing in law: Prediction of outcomes in the higher courts of Turkey," *Inf. Process. Manag.*, vol. 58, no. 5, p. 102684, Sep. 2021, doi: 10.1016/j.ipm.2021.102684.

[8] J. E. Davidson *et al.*, "Job-Related Problems Prior to Nurse Suicide, 2003-2017: A Mixed Methods Analysis Using Natural Language Processing and Thematic Analysis," *J. Nurs. Regul.*, vol. 12, no. 1, pp. 28–39, Apr. 2021, doi: 10.1016/S2155-8256(21)00017-X.

[9] A. P. Nugraha, I. N. Piarsa, and I. M. Suwija Putra, "Comparison of Support Vector Machine and K-Nearest Neighbor for Baby Foot Identification based on Image Geometric Characteristics," *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, p. 84, Apr. 2021, doi: 10.24843/JIM.2021.v09.i01.p08.

[10] S. Kumar, A. K. Kar, and P. V. Ilavarasan, "Applications of text mining in services management: A systematic literature review," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 1, p. 100008, Apr. 2021, doi: 10.1016/j.jjimei.2021.100008.

[11] M. Lee, S. Kim, H. Kim, and J. Lee, "Technology Opportunity Discovery using Deep Learning-based Text Mining and a Knowledge Graph," *Technol. Forecast. Soc. Change*, vol. 180, p. 121718, Jul. 2022, doi: 10.1016/j.techfore.2022.121718.

[12] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, and A. Hilal, "Preprocessing Arabic text on social media," *Heliyon*, vol. 7, no. 2, p. e06191, Feb. 2021, doi: 10.1016/j.heliyon.2021.e06191.

[13] E. S. Tellez, S. Miranda-Jiménez, M. Graff, D. Moctezuma, O. S. Siordia, and E. A. Villaseñor, "A case study of Spanish text transformations for twitter sentiment analysis," *Expert Syst. Appl.*, vol. 81, pp. 457–471, Sep. 2017, doi: 10.1016/j.eswa.2017.03.071.

[14] A. Mee, E. Homapour, F. Chiclana, and O. Engel, "Sentiment analysis using TF–IDF weighting of UK MPs' tweets on Brexit," *Knowledge-Based Syst.*, vol. 228, p. 107238, Sep. 2021, doi: 10.1016/j.knosys.2021.107238.

[15] N. P. R. Apriyanti, I. K. G. D. Putra, and I. M. S. Putra, "Peramalan Jumlah Kecelakaan Lalu Lintas Menggunakan Metode Support Vector Regression," *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, p. 72, Jun. 2020, doi: 10.24843/JIM.2020.v08.i02.p01.

[16] I. P. A. P. Wibawa, I. K. A. Purnawan, D. P. S. Putri, and N. K. D. Rusjayanthi, "Prediksi Partisipasi Pemilih dalam Pemilu Presiden 2014 dengan Metode Support Vector Machine," *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, p. 182, Dec. 2019, doi: 10.24843/JIM.2019.v07.i03.p02.