Knowledge Engineering and Data Science (KEDS)
Vol 5, No 2, December 2022, pp. 150–159

pISSN 2597-4602
eISSN 2597-4637

Can Multinomial Logistic Regression Predicts Research Group using Text Input?

Harits Ar Rosyid [a,1,*], Aulia Yahya Harindra Putra [a,2], Muhammad Iqbal Akbar [a,3], Felix Andika Dwiyanto [b,4]

[a] *Department of Electrical Engineering, Universitas Negeri Malang,
Malang 65145, Indonesia*
[b] *Faculty of Computer Science, Electronics, and Telecommunications, AGH University of Science and Technology,
30-059 Kraków, Poland*
[1] *harits.ar.ft@um.ac.id*; [2] yahya.harindraputra.1905356@students.um.ac.id; [3]iqbal.akbar.ft@um.ac.id;
[4]dwiyanto@agh.edu.pl*
* corresponding author*

ARTICLE INFO

Article history:
Received 11 November 2022
Revised 29 November 2022
Accepted 9 December 2022
Published online 30 December 2022

Keywords:
Classification
Logistic Regression
Title
Abstract
Research Group
Thesis

ABSTRACT

While submitting proposals in SISINTA, students often confuse or falsely submit their proposals to the less relevant or incorrect research group. There are 13 research groups for the students to choose from. We proposed a text classification method to help students find the best research group based on the title and/or abstract. The stages in this study include data collection, preprocessing data, classification using Logistic Regression, and evaluation of the results. Three scenarios in research group classification are based on 1) title only, 2) abstract only, and 3) title and abstract. Based on the experiments, research group classification using title-only input is the best overall. This scenario gets the most optimal results with accuracy, precision, recall, and f1-score successively at 63.68%, 64.91%, 63.68%, and 63.46%. This result is sufficient to help students find the best research group based on the text titles. In addition, lecturers can comment more elaborately since the proposals are relevant to the research group's scope.

I. Introduction

The Department of Electrical Engineering and Informatics (DEEI), Universitas Negeri Malang, has a thesis and final project management site, SISINTA UM. Every student submitting a thesis title must adjust the title and abstract of the thesis to match the research group. Based on a short survey of 25 students who have submitted titles and abstracts to SISINTA UM, the results show that students feel confused and have difficulty adjusting the proposed thesis's title and abstract. Most lecturers from the target research group usually respond briefly to any mismatch between the proposal and the research group. This subjective response could lead to more confusion for the students.

The traditional solution would be to consult their topic with lecturers or academic supervisors. This approach is somewhat complex and not straightforward. Factors like time and place arrangements between students and lecturers are too dynamic. The system should be able to recommend the best research group based on the information referring to a thesis or final project. This approach is adapted from [1], which shows a Lexile Level within an article posted on a website. This straightforward information will help readers to find the preferable articles.

We propose a text classification technique to construct a research group recommendation based on text input: title and/or abstract. The main idea is driven by the abundant text information stored in the SISINTA database. Once this text data is retrieved, we apply a text mining process initialized by text preprocessing to clean and restructure the text. Then, the term weighting stage applies to convert text into a computable form: numbers. Subsequently, resampling is essential to tackle the imbalanced distribution of classes. In the next stage, we applied the Logistic Regression (LR) algorithm [2] that will learn to distinguish research groups based on the title and or abstract. LR is a classification algorithm to predict the probability of the target variable [3]. This algorithm is useful in text

https://doi.org/10.17977/um018v5i22022p150-159

classification, such as sentiment analysis [4]. Finally, we evaluate how well the LR predicts the research group based on the text input.

II. Method

In this research, several stages of the research methodology are described in Figure 1. We collected raw data from DEEI's SISINTA database at the data collection stage by dumping the SQL data into a Microsoft Excel file. No personal information such as students, supervisors, grades, or logs was included during data exporting. The main content we retrieved was text information relevant to these and the final projects. Data obtained from 16 April 2016 to 4 October 2022 contained 2164 samples, and the SISINTA administrator confirmed that these data are accurate. Each sample has independent variables: the title, abstract, and research group class. Thirteen research groups and their class distributions are shown in Table 1. From this table, we can see an imbalanced distribution of research groups. A challenge to be tackled by Resampling Technique in our proposed method.

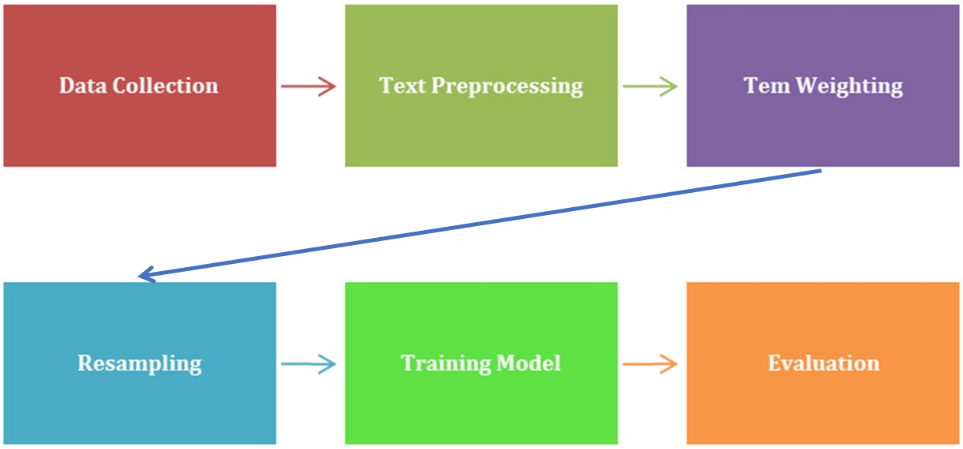


Fig 1. Research Methodology

Table 1. Number of rows in each research group of the data studied

Research Group	Total
Pengembangan Aplikasi dan Media Pembelajaran Teknologi dan Kejuruan	463
Strategi Pembelajaran Teknologi dan Kejuruan	395
Kurikulum Pendidikan Teknologi dan Kejuruan	200
Rekayasa pengetahuan dan ilmu data (Knowledge Engineering and Data Science)	174
Evaluasi dan Pengelolaan Pendidikan Kejuruan	155
Ketenegakerjaan Teknologi dan Kejuruan	142
Teknologi Digital Cerdas (Ubiquitous Computing Technique)	132
Intelligent Power and Advanced Energy System (IPAES)	121
Intelligent Power Electronics and Smart Grid (IPESG)	104
Game Technology and Machine Learning Applications	90
Telematics loT System and Devices	88
Biomedic and Intelligent Assistive Technology (TAT)	55
Sistem Dinamis, Kendali, dan Robotika (Dynamic Systems, Control, and Robotics)	45

Text preprocessing is carried out to ensure text data is ‘clean’ and the algorithm can learn from it [5]. Text preprocessing involves stages to make text information more structured [6], which include text cleaning, removing missing values, removing duplicate rows, tokenization, stopword removal, and stemming.

Text cleaning consists of four steps. First, tag removal aims to remove HTML tags contained in the document [7]. Many of the text data contains HTML tags. This often happens when students copy-paste text from the document processor to the SISINTA input form. We use regular expression filtering (a.k.a regex) to remove HTML tags and keep informative text. Say, `inputText = “<h1> Hello </h1>”`. By applying `regex = re.compile(r'<[^>]+>')`, the function `regex.sub('', inputText)` will ouput → `Hello`. Second, case folding aims to convert capital letters to lowercase. It is helpful to prevent the computer from interpreting the same word with different meanings [8]. For instance, `Python case fold (“Case”)` will output the `case`. The third stage, trim text, aims to remove white space at the beginning and end of the text [9]. In Python, it is achieved by running the `strip()` function to remove spaces from both ends. The last stage removes punctuation, special characters, double white space, and the number [10]. We apply the regex for this purpose by adding more memorable characters to be removed.

The second stage of text preprocessing is to remove missing values. This step is carried out to handle missing data by removing columns or rows whose data is not available or NaN (Not a Number). This deletion's purpose is to reduce data bias [11]. This study's third stage of text preprocessing is to remove duplicates or redundant samples [12]. This will minimize the overfitting effect due to duplicates [13].

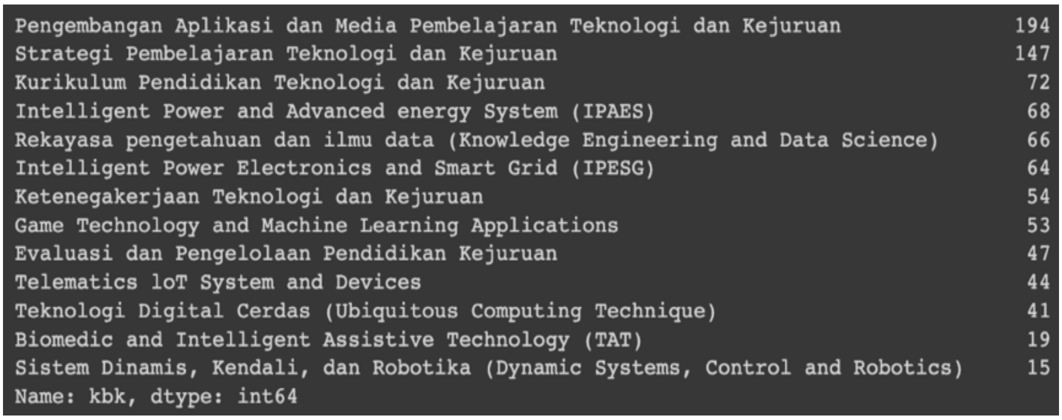
We use the Natural Language Toolkit (NLTK) for this step specifically the `nltk.tokenize` package. The goal is to break down sentences into words or tokens [14]. In this study, tokenization applies to the title and abstract into word fragments to identify words and the separators. Hence, tokenization helps extract meaning from text.

This study's fifth stage of text preprocessing is stopword removal or text filtering. We use `nltk.corpus` → `stopwords`, to filter out stop words such as *'diperlukan'*, *'hendaknya'*, and *'tapi'*.

The final text preprocessing stage stems [15]. Stemming is used to cut prefixes, suffixes, inserts, combinations of prefixes and endings, and remove affixes [16]. Besides that, it can also eliminate word inflection to its basic form. The steaming process can be done using a particular Indonesian language streamer library, Sastrawi. This process aims to make the computer interpret a word constructed from the same root word with a different meaning [17]. For instance, when stemming is applied, the word “*kecepatan*” will produce “`cepat`”.

Once the text data is clean and ready, term weighting converts data into a numeric form [18]. We apply the Term Frequency-Inverse Document Frequency (TF-IDF) method in this study. TF-IDF assigns a weight to each word that frequently appears to quantitatively measure how strong the relationship between the word and the document is [19]. When a word appears more frequently in a document, its weight increases proportionally. In contrast, the weight decreases if the word appears more regularly in many documents [20]. We apply the sci-kit-learn library, `sklearn.feature_extraction.text.TfidfVectorizer` for this purpose.

Until the resampling stage, the dataset was distributed unevenly between research groups. Although there are significant sample drops within each research group, the distribution is not balanced, as seen in Figure 2. The imbalanced dataset can cause bias in the data, where partial data tends to make the classifier performs best only when predicting dominant classes [21]. Therefore, we applied the resampling method, the Synthetic Minority Oversampling Technique (SMOTE). SMOTE iteratively generates artificial samples based on the original neighboring samples. This phase stops until all classes have the same number of samples, 194 samples each.



```
Pengembangan Aplikasi dan Media Pembelajaran Teknologi dan Kejuruan        194
Strategi Pembelajaran Teknologi dan Kejuruan                               147
Kurikulum Pendidikan Teknologi dan Kejuruan                                 72
Intelligent Power and Advanced energy System (IPAES)                        68
Rekayasa pengetahuan dan ilmu data (Knowledge Engineering and Data Science)  66
Intelligent Power Electronics and Smart Grid (IPESG)                        64
Ketenegakerjaan Teknologi dan Kejuruan                                      54
Game Technology and Machine Learning Applications                           53
Evaluasi dan Pengelolaan Pendidikan Kejuruan                                47
Telematics loT System and Devices                                           44
Teknologi Digital Cerdas (Ubiquitous Computing Technique)                   41
Biomedic and Intelligent Assistive Technology (TAT)                         19
Sistem Dinamis, Kendali, dan Robotika (Dynamic Systems, Control and Robotics)  15
Name: kbk, dtype: int64
```

Fig 2. Class distribution on the raw dataset

This study used Multinomial Logistic Regression (MLR) due to 13 research group classes. Before modeling, we separated the dataset into 70% training and 30% test sets. The training set was then used to train and optimize the MLR via Grid Search Cross Validation (GSCV) method. This tuning method aims to find a combination of parameters from the model that produces the most optimal and effective predictions [22]. The GSCV method heuristically constructs and evaluates the MLR model using all parameter value combinations in Table 2 in a cross-validated environment (we use 10-fold). The GSCV method produces insights into using different parameter combinations regarding classification performances. Then, we refitted the MLR using the parameters that produce the highest classification performance.

Table 2. MLR parameters for grid search CV

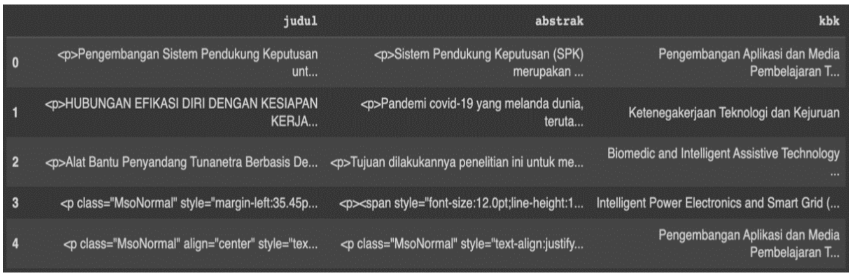
Parameter	Spesification
multi_class	multinomial
solver	saga
penalty	[‘l1’, ‘l2’, ‘none’]
C	[‘0.1’, ‘1.0’, ‘5’, ‘10’]

Since there are two types of input relevant to the research group: title and abstract, we ran three scenarios of MLR prediction based on: 1) a title, 2) an abstract, and 3) a combination of a title and abstract. The goal is to identify which classifier performs best. Hence, the GSCV method is applied within each scenario producing 12 model candidates. In total, there are 36 candidates for the research group prediction model.

In the evaluation stage, the best model from each scenario was tested using 30% test data. The metrics used were accuracy, precision, recall, and f1-score. The goal was to test how effective the MLR was based on the classification performance or correctness level [23]. From there, we can choose which MLR is best applied for SISINTA.

III. Results and Discussion

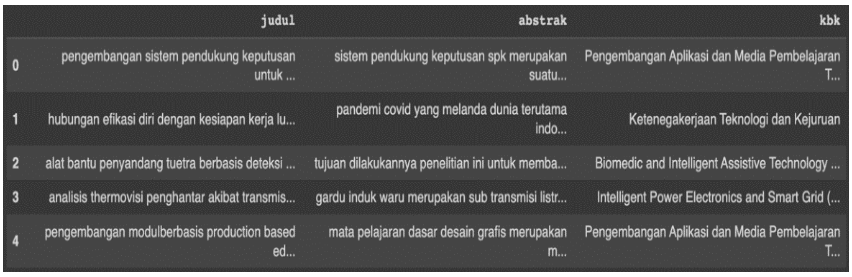
The retrieved 2164 rows of data were raw text structured into columns: title, abstract, and research group. Figure 3 shows the rawness of the dataset.



	judul	abstrak	kbk
0	<p>Pengembangan Sistem Pendukung Keputusan unt...	<p>Sistem Pendukung Keputusan (SPK) merupakan ...	Pengembangan Aplikasi dan Media Pembelajaran T...
1	<p>HUBUNGAN EFIKASI DIRI DENGAN KESIAPAN KERJA...	<p>Pandemi covid-19 yang melanda dunia, teruta...	Ketenegakerjaan Teknologi dan Kejuruan
2	<p>Alat Bantu Penyandang Tunanetra Berbasis De...	<p>Tujuan dilakukannya penelitian ini untuk me...	Biomedic and Intelligent Assistive Technology ...
3	<p class="MsoNormal" style="margin-left:35.45p...	<p><span style="font-size:12.0pt;line-height:1...	Intelligent Power Electronics and Smart Grid (...
4	<p class="MsoNormal" align="center" style="tex...	<p class="MsoNormal" style="text-align:justify...	Pengembangan Aplikasi dan Media Pembelajaran T...

Fig 3. Example of data collection results

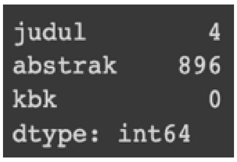
The process of tag removal, case folding, small text, and removal of punctuation marks, special characters, double spaces, and numbers is carried out at the next cleaning stage. The processing results of this stage can be seen in Figure 4.



	judul	abstrak	kbk
0	pengembangan sistem pendukung keputusan untuk ...	sistem pendukung keputusan spk merupakan suatu...	Pengembangan Aplikasi dan Media Pembelajaran T...
1	hubungan efikasi diri dengan kesiapan kerja lu...	pandemi covid yang melanda dunia terutama indo...	Ketenegakerjaan Teknologi dan Kejuruan
2	alat bantu penyandang tuetra berbasis deteksi ...	tujuan dilakukannya penelitian ini untuk memba...	Biomedic and Intelligent Assistive Technology ...
3	analisis thermovisi penghantar akibat transmis...	gardu induk waru merupakan sub transmisi listr...	Intelligent Power Electronics and Smart Grid (...
4	pengembangan modulberbasis production based ed...	mata pelajaran dasar desain grafis merupakan m...	Pengembangan Aplikasi dan Media Pembelajaran T...

Fig 4. Example of text cleaning results

The next step is to remove the missing values. There are four rows of missing values in the title column and 896 rows of missing values in the abstract column, where the number of missing values in the dataset can be seen in Figure 5.

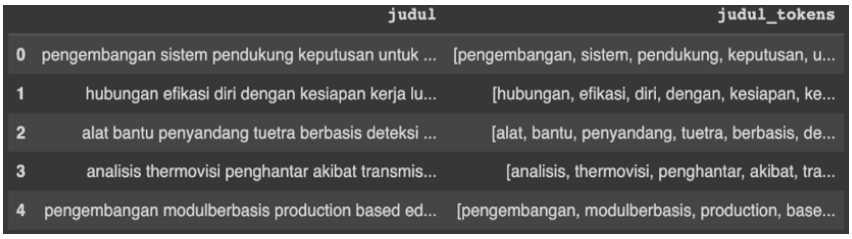


```
judul        4
abstrak    896
kbk          0
dtype: int64
```

Fig 5. Number of missing values in each dataset column

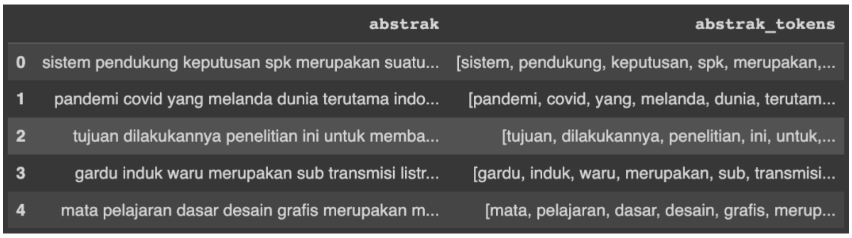
Furthermore, we identified one data duplication from the title column but none from the abstract. As a result of text preprocessing, the distribution of the dataset falls short, but there are imbalanced distributions of research group classes, see Figure 2.

The tokenization stage is carried out to separate text into tokens or words [24]. Figure 6 and Figure 7 show examples of the tokenization result in the title and abstract columns.



	judul	judul_tokens
0	pengembangan sistem pendukung keputusan untuk ...	[pengembangan, sistem, pendukung, keputusan, u...
1	hubungan efikasi diri dengan kesiapan kerja lu...	[hubungan, efikasi, diri, dengan, kesiapan, ke...
2	alat bantu penyandang tuetra berbasis deteksi ...	[alat, bantu, penyandang, tuetra, berbasis, de...
3	analisis thermovisi penghantar akibat transmis...	[analisis, thermovisi, penghantar, akibat, tra...
4	pengembangan modulberbasis production based ed...	[pengembangan, modulberbasis, production, base...

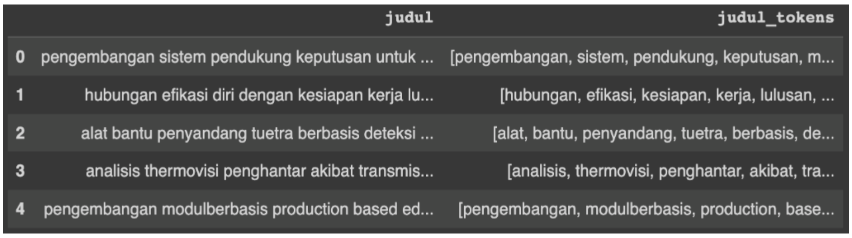
Fig 6. Tokenization results in the title column



	abstrak	abstrak_tokens
0	sistem pendukung keputusan spk merupakan suatu...	[sistem, pendukung, keputusan, spk, merupakan,...
1	pandemi covid yang melanda dunia terutama indo...	[pandemi, covid, yang, melanda, dunia, terutam...
2	tujuan dilakukannya penelitian ini untuk memba...	[tujuan, dilakukannya, penelitian, ini, untuk,...
3	gardu induk waru merupakan sub transmisi listr...	[gardu, induk, waru, merupakan, sub, transmisi...
4	mata pelajaran dasar desain grafis merupakan m...	[mata, pelajaran, dasar, desain, grafis, merup...

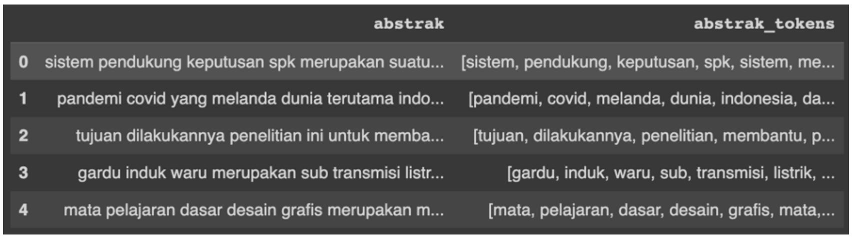
Fig 7. Tokenization results in the abstract column

The stopwords removal stage is carried out to remove words or tokens that appear frequently and have no critical meaning in the text [25]. The results of the stopwords removal process in the title and abstract columns can be seen in Figure 8 and Figure 9.



	judul	judul_tokens
0	pengembangan sistem pendukung keputusan untuk ...	[pengembangan, sistem, pendukung, keputusan, m...
1	hubungan efikasi diri dengan kesiapan kerja lu...	[hubungan, efikasi, kesiapan, kerja, lulusan, ...
2	alat bantu penyandang tuetra berbasis deteksi ...	[alat, bantu, penyandang, tuetra, berbasis, de...
3	analisis thermovisi penghantar akibat transmis...	[analisis, thermovisi, penghantar, akibat, tra...
4	pengembangan modulberbasis production based ed...	[pengembangan, modulberbasis, production, base...

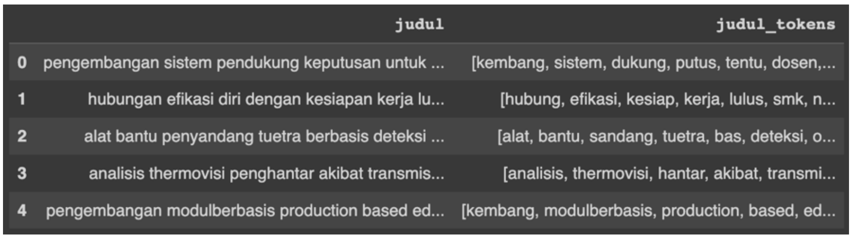
Fig 8. Stopwords removal results in the title column



	abstrak	abstrak_tokens
0	sistem pendukung keputusan spk merupakan suatu...	[sistem, pendukung, keputusan, spk, sistem, me...
1	pandemi covid yang melanda dunia terutama indo...	[pandemi, covid, melanda, dunia, indonesia, da...
2	tujuan dilakukannya penelitian ini untuk memba...	[tujuan, dilakukannya, penelitian, membantu, p...
3	gardu induk waru merupakan sub transmisi listr...	[gardu, induk, waru, sub, transmisi, listrik, ...
4	mata pelajaran dasar desain grafis merupakan m...	[mata, pelajaran, dasar, desain, grafis, mata,...

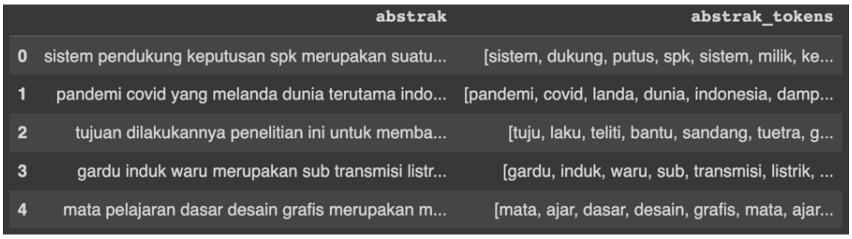
Fig 9. Stopwords removal results in the abstract column

The stemming stage is carried out to remove all affixes in words, such as suffixes, inserts, prefixes, and combinations between prefixes and suffixes [26]. The results of the steaming process in the title and abstract columns can be seen in Figure 10 and Figure 11.



	judul	judul_tokens
0	pengembangan sistem pendukung keputusan untuk ...	[kembang, sistem, dukung, putus, tentu, dosen,...
1	hubungan efikasi diri dengan kesiapan kerja lu...	[hubung, efikasi, kesiap, kerja, lulus, smk, n...
2	alat bantu penyandang tuetra berbasis deteksi ...	[alat, bantu, sandang, tuetra, bas, deteksi, o...
3	analisis thermovisi penghantar akibat transmis...	[analisis, thermovisi, hantar, akibat, transmi...
4	pengembangan modulberbasis production based ed...	[kembang, modulberbasis, production, based, ed...

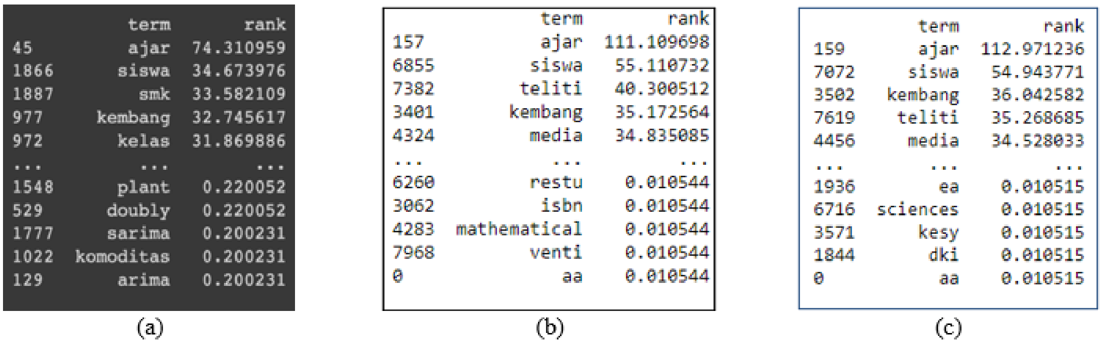
Fig 10. Stemming results in the title column



	abstrak	abstrak_tokens
0	sistem pendukung keputusan spk merupakan suatu...	[sistem, dukung, putus, spk, sistem, milik, ke...
1	pandemi covid yang melanda dunia terutama indo...	[pandemi, covid, landa, dunia, indonesia, damp...
2	tujuan dilakukannya penelitian ini untuk memba...	[tuju, laku, teliti, bantu, sandang, tuetra, g...
3	gardu induk waru merupakan sub transmisi listr...	[gardu, induk, waru, sub, transmisi, listrik, ...
4	mata pelajaran dasar desain grafis merupakan m...	[mata, ajar, dasar, desain, grafis, mata, ajar...

Fig 11. Stemming results in the abstract column

TF-IDF produced a matrix in the training set of the title scenario in the form of a vector of 884 samples x 2300 columns. Meanwhile, the matrix test set of the title scenario makes a vector of 380 samples x 2300 columns. For the second and third scenarios, the remaining scenarios produced nearly quadrupled columns: 8218 and 8485 columns. An example view of term weighting using TF-IDF can be seen in Figure 12.



```
           term       rank
45         ajar  74.310959
1866      siswa  34.673976
1887        smk  33.582109
977     kembang  32.745617
972       kelas  31.869886
...         ...        ...
1548      plant   0.220052
529      doubly   0.220052
1777     sarima   0.200231
1022  komoditas   0.200231
129       arima   0.200231
```

(a)

```
              term        rank
157           ajar  111.109698
6855         siswa   55.110732
7382        teliti   40.300512
3401       kembang   35.172564
4324         media   34.835085
...            ...         ...
6260         restu    0.010544
3062          isbn    0.010544
4283  mathematical    0.010544
7968         venti    0.010544
0               aa    0.010544
```

(b)

```
          term        rank
159       ajar  112.971236
7072     siswa   54.943771
3502   kembang   36.042582
7619    teliti   35.268685
4456     media   34.528033
...        ...         ...
1936        ea    0.010515
6716  sciences    0.010515
3571      kesy    0.010515
1844       dki    0.010515
0           aa    0.010515
```

(c)

Fig 12. Term weighting examples using TF-IDF: (a) Title scenario, (b) Abstract scenario, and (c) Combination of title and abstract

We applied the default configuration of the SMOTE in generating synthetic samples (*n_neighbors = 5*). There are 194 data on each RESEARCH GROUP after the resampling process using SMOTE. In total, there are 2522 samples ready for model training.

In title scenario, using the Grid Search Cross-Validation (GSCV) method, the best parameter configurations for the MLR were *C=0.1* and using a *'none'* penalty. Fig. 13 depicts the comparison between the candidates' performances (in dots) that applies various regularization parameters (x-axis) and penalty (colored line). This graph shows that the MLR performs best when the C value is high, ignoring the penalty type. The result of MLR in the green line is suspect of overfitting because the other MLRs (orange and blue lines) underperformed when the C is lowest. This means that regularization is essential for the MLR to perform generically. From Figure 13, the L2-type regularization (orange line) should be the best since it performs better even using a low C value compared to the L1-type. The higher the C value, the MLR using L2-type is always on top of the MLR with L1-type. Therefore, the MLR was refitted in this scenario using the Penalty=L2 with C=5 as the most optimal one.

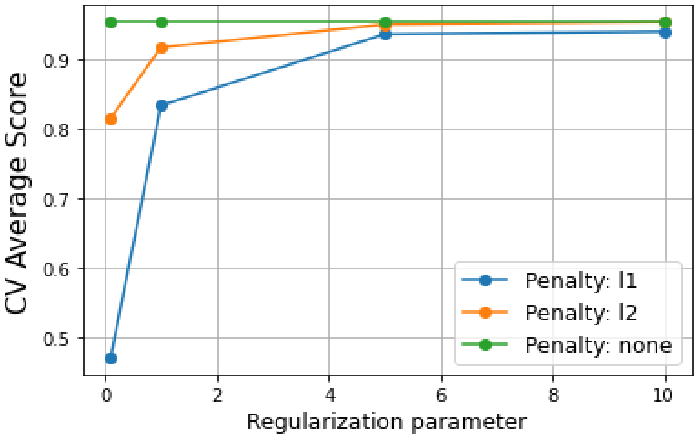


Fig 13. Grid search CV results on title scenario

In the abstract scenario, the results of the most optimal combination of parameters can be seen in Figure 14. Our analysis in this second scenario is similar to the first one. The difference appears only slightly in the resulting scores. From this graph, the MLR using abstract as input is refitted with Penalty=L2 and C=5.

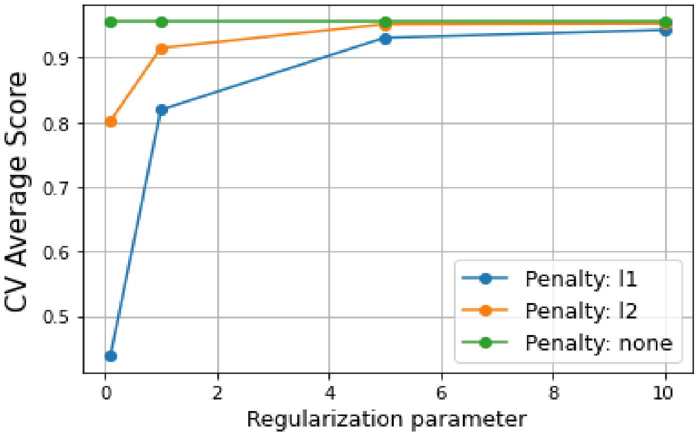


Fig 14. Grid search CV results on the abstract scenario

GSCV results for the third scenario can be seen in Figure 15. Our analysis in this third scenario is similar to the former two. The difference appears only slightly in the resulting scores. From this graph, the MLR using abstract as input is refitted with Penalty=L2 and C=5.

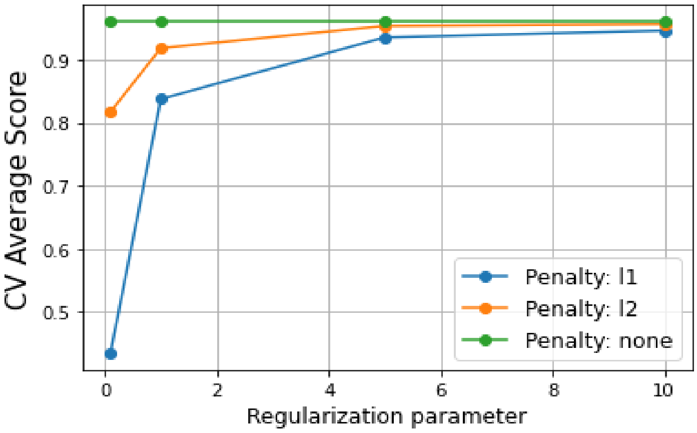


Fig 15. Grid Search CV results on the title and abstract scenario

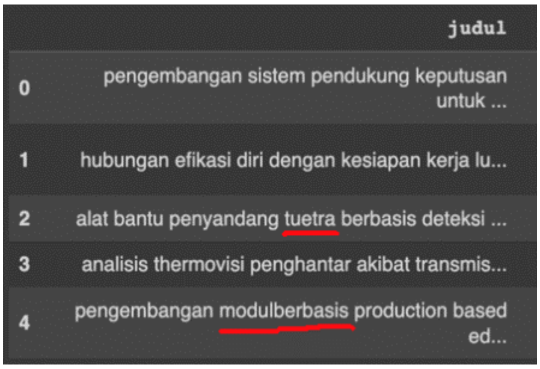
From the three scenarios using GSCV, there were no significant differences between the effect of input used. Even the performances were relatively identical. However, we tested each using the test data to delve deeper into how the three MLR model performs. We measured each scenario's performance metrics; the results can be seen in Table 3.

Table 3. Performance comparison

No	Input Type	Accuracy	Precision	Recall	F1-Score
1.	Title	63.68%	64.91%	63.68%	63.46%
2.	Abstract	61.05%	61.16%	61.05%	60.73%
3.	Title+Abstract	62.89%	63.17%	62.89%	62.57%

The evaluation results show that the title scenario is the best and optimal scenario. Although this result is insignificant compared to the other two scenarios, it is more efficient since the input size for MLR is way smaller if using the title only. As such is a way to reduce the curse of dimensionality in research group classification. Hence, a minor computation power is available. In addition, there will be a slight chance of repeated words in the titles (except stopwords) compared to the abstract. Hence, we argue that using the title is more concise for the classification's performance.

We also pointed out the overall metrics that are below 70%. We identified the causes: typographical error (TYPO) within the title or abstract, coupled words, and the lack of a validation process to check for these errors. Examples of errors contained in the dataset can be seen in Figure 16. The words highlighted were only a few in a brief observation. However, these words are not core or root words that highly correlate with the research group. The classification model will lose some accuracy if this word is mistyped while contributing to a particular research group. The solution is applying a policy in the SISINTA that any typo entered in the title or abstract will dismiss the students to get comments from the research group. Either manual observation or automatic one is feasible. Alternatively, by applying additional text preprocessing to identify these typos and decide whether to correct or remove them.



	judul
0	pengembangan sistem pendukung keputusan untuk ...
1	hubungan efikasi diri dengan kesiapan kerja lu...
2	alat bantu penyandang tuetra berbasis deteksi ...
3	analisis thermovisi penghantar akibat transmis...
4	pengembangan modulberbasis production based ed...

Fig16. Writing errors on the dataset

In addition, great topics overlap between research group classes. For instance: the research group "Game Technology and Machine Learning" and "Knowledge Engineering and Data Science". Both research groups contain research with the keywords “machine learning”, “data mining”, “classification”, etc. Too many terms were shared between these two examples of research groups. Only a few keywords disparate the two research groups, for instance, “game” and “text”. To overcome the problem of shared words by looking at the linked words, we can use n-grams that decompose a text into n-character chunks so that linked words can be parsed. However, using the n-gram feature significantly enlarges the dimension. Hence, more complex algorithms like Deep Learning should fit the task.

Finally, our proposed method is applicable in different departments as long as the digital storage of the student's research is organized in the research group (web-based information system and the database). Based on our findings, the future implementation may only need to structure the data into the title column and research group. Then, additional text preprocessing to identify and replace typos in the content is also essential to ensure the dataset's quality for the learning algorithm. Other learning algorithms are available depending on the target classes and the size of the dataset provided. Parameter tuning should be performed using GSCV with more combinations since the dataset's target case differs from our research. The remaining stages of research group recommendation are repeatable as is.

When SISINTA implements a recommendation of a research group based on user input, the initial procedure of the thesis or final project proposal can be done in seconds. This can also help lecturers in the research group to provide more elaborated and comprehensive comments within their scope of knowledge regarding the proposals. If there are revisions required for the proposal are relevant and constructive to make their research go in the right direction. Overall, this automatic instruction in SISINTA can make it an intelligent information system for educational purposes. Not only applicable in DEEI, but this approach should also be applicable in other departments as long as there are good platforms and data.

IV. Conclusion

This research showed that we successfully applied Multinomial Logistic Regression (MLR) Algorithm to predict the research group based on text input, either the title or thesis abstract. The stages we followed in the text mining technique were straightforward, and MLR performed adequately well to classify 13 research groups. The best scenario in this study was the MLR with the input variable from the title. Using title data as a model training scenario is considered adequate, optimal, and efficient. This is because there will be rare to write repeated words within a thesis title, except stopwords. With performances just above 63% in overall metrics, we argue that this MLR model with title text input is optimal due to its small dimensionality. However, the relatively low performances below the 70% threshold were limited because research groups shared similar keywords and typos inside the dataset. These typos can become noise or must be extracted from the core word. Therefore, additional text preprocessing should consider these typos.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no known conflict of financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission information are available at http://journal2.um.ac.id/index.php/keds.

Publisher’s Note: Department of Electrical Engineering - Universitas Negeri Malang remains neutral with regard to jurisdictional claims and institutional affiliations.

References

[1] H. A. Rosyid, U. Pujianto, and M. R. Yudhistira, “Classification of Lexile Level Reading Load Using the K-Means Clustering and Random Forest Method,” Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control, pp. 139–146, May 2020.

[2] M. Taddy, “Multinomial inverse regression for text analysis,” J. Am. Stat. Assoc., vol. 108, no. 503, pp. 755–770, 2013.

[3] H. Chai, Y. Liang, S. Wang, and H. Shen, “A novel logistic regression model combining semi-supervised learning and active learning for disease classification,” Sci. Rep., vol. 8, no. 1, p. 13009, Aug. 2018.

[4] W. P. Ramadhan, S. T. M. T. Astri Novianty, and S. T. M. T. Casi Setianingsih, “Sentiment analysis using multinomial logistic regression,” in 2017 International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC), Sep. 2017, pp. 46–49.

[5] S. A. Salloum, M. Al-Emran, A. A. Monem, and K. Shaalan, “Using Text Mining Techniques for Extracting Information from Research Articles,” in Studies in Computational Intelligence, 2018, pp. 373–397.

[6] V. Dogra, A. Singh, S. Verma, Kavita, N. Z. Jhanjhi, and M. N. Talib, “Understanding of Data Preprocessing for Dimensionality Reduction Using Feature Selection Techniques in Text Classification,” in Intelligent Computing and Innovation on Data Science, 2021, pp. 455–464.

[7] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, “The influence of preprocessing on text classification using a bag-of-words representation,” PLoS One, vol. 15, no. 5, p. e0232525, May 2020.

[8] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, “Sentiment Analysis Using Word2vec And Long Short-Term Memory (LSTM) For Indonesian Hotel Reviews,” Procedia Comput. Sci., vol. 179, pp. 728–735, 2021.

[9] J. Lever et al., “PGxMine: Text mining for curation of PharmGKB Jake,” Pac Symp Biocomput, no. 25, pp. 611–622, 2020.

[10] S. Vijayaraghavan et al., “Fake News Detection with Different Models,” ArXiv, 2020.

[11] ReLearn: A Robust Machine Learning Framework in Presence of Missing Data for Multimodal Stress Detection from Physiological Signals,” in 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Nov. 2021, pp. 535–541.

[12] P. R. Vishnu, P. Vinod, and S. Y. Yerima, “A Deep Learning Approach for Classifying Vulnerability Descriptions Using Self Attention Based Neural Network,” J. Netw. Syst. Manag., vol. 30, no. 1, p. 9, Jan. 2022.

[13] H. Inoue, “Multi-Sample Dropout for Accelerated Training and Better Generalization,” ArXiv, 2019.

[14] G. N. R Prasad Sr Asst professor, “Identification of Bloom’s Taxonomy level for the given Question paper using NLP Tokenization technique,” Turkish J. Comput. Math. Educ., vol. 12, no. 13, pp. 1872–1875, 2021.

[15] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. Abd Elaziz, and A. Dahou, “A Study of the Effects of Stemming Strategies on Arabic Document Classification,” IEEE Access, vol. 7, pp. 32664–32671, 2019.

[16] M. Adriani, J. Asian, B. Nazief, S. M. M. Tahaghoghi, and H. E. Williams, “Stemming Indonesian,” ACM Trans. Asian Lang. Inf. Process., vol. 6, no. 4, pp. 1–33, Dec. 2007.

[17] M. A. Rosid, A. S. Fitrani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, “Improving Text Preprocessing For Student Complaint Document Classification Using Sastrawi,” IOP Conf. Ser. Mater. Sci. Eng., vol. 874, no. 1, p. 012017, Jun. 2020.

[18] J. M.-T. Wu, G. Srivastava, J. C.-W. Lin, and Q. Teng, “A Multi-Threshold Ant Colony System-based Sanitization Model in Shared Medical Environments,” ACM Trans. Internet Technol., vol. 21, no. 2, pp. 1–26, Jun. 2021.

[19] S. Qaiser and R. Ali, “Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents,” Int. J. Comput. Appl., vol. 181, no. 1, pp. 25–29, Jul. 2018.

[20] N. S. Mohd Nafis and S. Awang, “An Enhanced Hybrid Feature Selection Technique Using Term Frequency-Inverse Document Frequency and Support Vector Machine-Recursive Feature Elimination for Sentiment Classification,” IEEE Access, vol. 9, pp. 52177–52192, 2021.

[21] M. Umer et al., “Scientific papers citation analysis using textual features and SMOTE resampling techniques,” Pattern Recognit. Lett., vol. 150, pp. 250–257, Oct. 2021.

[22] G. S. K. Ranjan, A. Kumar Verma, and S. Radhika, “K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries,” in 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Mar. 2019, pp. 1–5.

[23] B. H. Shekar and G. Dagnew, “Grid Search-Based Hyperparameter Tuning and Classification of Microarray Cancer Data,” in 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), Feb. 2019, pp. 1–8.

[24] M. P. Geetha and D. Karthika Renuka, “Improving the performance of aspect based sentiment analysis using fine-tuned Bert Base Uncased model,” Int. J. Intell. Networks, vol. 2, pp. 64–69, 2021.

[25] A. W. Pradana and M. Hayaty, “The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 375–380, Oct. 2019, doi: 10.22219/kinetik.v4i4.912.

[26] J. Jumadi, D. S. Maylawati, L. D. Pratiwi, and M. A. Ramdhani, “Comparison of Nazief-Adriani and Paice-Husk algorithm for Indonesian text stemming process,” IOP Conf. Ser. Mater. Sci. Eng., vol. 1098, no. 3, p. 032044, Mar. 2021.