khazanah
informatika

Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika is a national scientific journal that publishes scientific research papers/articles or reviews in the field of Computer Systems and Informatics. The journal is accredited "Sinta 2" according to the decree of Directorate of DPRP (Dirjen Penguatan Riset dan Pengembangan), Ministry of Higher Education number 200/M/KPT/2020 dated December 30, 2020. The decree is valid from vol. 6 issue 2 (2020) until vol. 11 no. 2 (2025). The scope of this journal includes software engineering, information systems development, computer systems, and computer networking. The journal is published by Muhammadiyah University Press (MUP), Universitas Muhammadiyah Surakarta.

ISSN: 2621-038X e-ISSN: 2477-698X

khazanah informatika

Vol. 8, No. 2 | October 2022

Table of Contents

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Herbal Compound Screening with GPU Computation on ZINC Database through Similarity Comparison Approach

Refianto Damai Darmawan, Wisnu Ananta Kusuma*, Hendra Rahmawan
Computer Science Department
IPB University
Bogor, Indonesia
*ananta@apps.ipb.ac.id

Abstract-Covid-19 is a global pandemic that drives many researcher strive to look for its solution, especially in the field of health, medicine, and total countermeasures. Early screening with in-silico processes is crucial to minimize the search space of the potential drugs to cure a disease. This research aims to find potential drugs of covid-19 disease in ZINC database to be further investigated through in-vitro method. About 997.402.117 chemical compounds are searched about its similarity to some of confirmed drugs to combat coronavirus. Sequential computation would take months to accomplish this task. General programming graphic processing unit approach is used to implement similarity comparison algorithm in parallel, in order to speed up the process. The result of this study shows the parallel algorithm implementation can speed-up the computation process up to 55 times faster, and also that some of the chemical compounds have high similarity score and can be found in nature.

Keywords: covid-19, GPU programming, parallel programming, similarity comparison

Article info: *submitted November 19, 2021, revised January 22, 2022, accepted February 24, 2022*

1. Introduction

a. Background

Covid-19 is a disease that is so phenomenal in 2020. This disease is caused by infection with the SARS COV-2 virus. Due to the nature of this virus that spreads quickly and is assisted by easy access for humans to carry out cross-country transportation, the problem of local viruses in China has become a global pandemic that has an impact on all countries in the world. During 12 to 18 July, 32 out of 34 provinces in Indonesia reported an increase in cases of which 17 provinces experienced an increase of 50% or more [1]. Researchers around the world are working on developing vaccines and drugs for COVID-19 (coronavirus disease). The way this virus enters the human body is through the ACE-2 receptor. The ligand referred to in the picture is a molecule that binds to another molecule, in this case the spike protein of the corona virus with the ACE-receptor.

The human need for drugs and means of prevention has prompted researchers to conduct research to find alternative drugs and vaccines for the corona virus. One alternative developed is the use of herbal plants to prevent and treat this disease [2]. The drug repurposing strategy is very useful considering that the conventional drug discovery process takes a long time. Drug repurposing is done by finding new benefits or efficacy of drug compounds that have been registered. Drug repurposing is usually done by analyzing the interaction of drug compounds with related proteins of a disease, then predicting new drug-target interactions that have not been known before [3].

The herbal medicine approach, or commonly called herbal medicine, is considered very useful for the prevention and treatment of COVID-19 disease due to several factors, namely the level of drug availability, drug safety, and the level of trust of the Indonesian people. As a traditional medicine made from plants, the level of availability of herbal medicines in Indonesia is so abundant, especially after being declared as one of the countries with a very wide and large plant biodiversity [4]. The safety of drugs from herbs or herbal plants has been tested from time to time because this method has been used for generations in traditional societies. The level of Indonesian people's trust in herbal medicine is also quite high, considering that more and more people are using herbs as an alternative treatment for various diseases.

The large amount of compound data available makes the sequential similarity search process take a very long time, thus requiring a more efficient approach. In addition to the use of adequate hardware, computing speed is also affected by the algorithm or how the computer works. The concept of parallelization and the use of Graphics Processing Unit (GPU) to speed up calculations make computation time faster than the Central Processing Unit (CPU) for certain types of computing [5]. The search for the similarity of these compounds can use the parallelization concept provided by the GPU to speed up the process. In [5], the GPU-assisted version of Support Vector Machine (SVM) is developed to significantly decrease the processing time of SVM training for large scale training data. The result showed that the use of GPU is proven to be significantly decrease the training time. The bigger the dataset, the more training time reduction it gets from using GPU.

Parallel drug-target interaction (DTI) research has been carried out using several schemes, including breadth-first search (BFS) [6], molecular docking with GPU [7], and BINDSURF which is a virtual screening methodology to find a protein binding site for a ligand [8]. In [6], BFS is used to predict drug-target interaction in a graph and is optimized by parallelization using CUDA, which gained a speed-up of 51.33 times by using 4 threads. In [7], a novel molecular docking approach is proposed and optimized by using heterogeneous implementation based on multicore CPUs and multiple GPUs. The result shows that this novel approach is able to perform blind docking simulations in a scenario where the two prominent docking programs, i.e., AutoDock 4 and AutoDock Vina, are not able to perform. In addition to that, the average real-mean-square deviation (RMSD) score of the proposed method is lower than the average RMSD score of the other two. In [8], a virtual screening method is presented to find new hotspot, an area in a protein where ligands might interact with. The GPU parallelism is used to allow fast processing of large ligand database.

This study aims to find out the potential of GPU in the search for compound similarity by utilizing its parallelization potential. By knowing GPU performance in the search for this medicinal compound, it is hoped that it can be a reference for the next in-silico research. Aside from that, this study also aims to find the potential of herbal compounds that exist in nature as Covid-19 drugs, through searching for similarity with several existing drugs. The herbal compounds identified as similar to the Covid-19 drugs can be carried out by further in-vitro research for verification, and if scientifically proven they will be able to assist the public in finding alternative drugs to deal with the Covid-19 pandemic.

b. Problem Formulation

Problem formulation in this research is:

a) How to apply a similarity comparison algorithm to molecular compound data with GPU computing to determine the level of similarity?

b) What are the herbal plants that have a high potential to become drug candidates for the COVID-19 disease?

c. Aims

Aims of this research is:

a) Design, implement, and evaluate parallel computing solutions similarity comparison using GPU.

c) Finding candidate compounds that have the potential to prevent or treat Covid-19.

d. Benefit

The results of this study are expected to provide benefits to improve the quality and speed of the drug discovery and drug repurposing process in the world of herbal medicine or herbal medicines so that in the end it can help the community in overcoming relatively new diseases at affordable prices.

e. Research Scope

The scope of this research is to look for compound data in-silico without being accompanied by in-vitro and in-vivo tests. The data sought is limited to the compound SMILES (simplified molecular input line entry system) string information.

2. Literature Review

a. Ligand-based Compound Screening

Virtual screening is a chemoinformatics technology designed to evaluate a large number of compounds computationally, with the aim of quickly identifying the desired structure so that it can be submitted as a bioassay [2]. Traditionally, the screening process is carried out through high-throughput screening (HTS), which is testing several compounds in bulk to find compounds that hit (interact) with the target protein. However, post-HTS analyzes are often disrupted [9] by the presence of protein-reactive compounds [10] or optically interfering components, which are the result of sample degradation from biochemical assays [11], or the tendency of chemicals to conduct aggregation [12]. To overcome the weakness of the HTS, virtual screening was developed in order to obtain more accurate screening results.

In this study, the approach used is ligand-based compound screening, namely the selection of compounds based on the level of similarity (similarity) of a compound that has successfully bound to the desired ligand (active protein region).

b. Herbal Compound

Herbal medicine is defined as a collection of therapeutic experiences from generation to generation by traditional healers over hundreds of years [13]. Most of the sources of herbal medicines are plants, so that in their development herbal medicines are identical to medicines derived from plants. Because herbal medicines are used by people based on people's habits, the scientific evidence they have is not strong enough. Further research on the real efficacy needs to be done in the process of using this herbal medicine in order to have strong evidence that the drug is safe for human use. To be accepted as a viable alternative to modern medicine, rigorous scientific and clinical validation methods must also be carried out to prove the safety and effectiveness of an herbal product [14].

In this study, the herbal compounds referred to refer to all active substances found in plants and have been used as medicine for generations in Indonesian society. One example that can be mentioned is curcumin in turmeric which is commonly used to relieve inflammation because it is anti-inflammatory [15].

c. Graphic Processing Unit

Graphic Processing Unit (GPU) is an electronic circuit hardware designed to manipulate memory quickly to accelerate image creation in a frame buffer that is intended as output on a display screen [16]. Although the original purpose of GPUs was to improve graphics performance, researchers often use them for the purpose of accelerating data processing. This happens because the GPU has many computational cores that can be used for parallel computing processes. The thing that users need to prepare is how to separate the data so that the GPU can work on it in parallel, then combine the results of the calculations so that there are no errors and the results are valid.

In Figure 1, you can see the differences in the architecture of the Central Processing Unit (CPU) and GPU. In the CPU architecture drawing, it can be seen that quite a lot of space is used for the control unit and cache. This makes sense because the CPU will receive a lot of data and commands that tend to be unique for each data to be processed, so it needs to be accommodated with an adequate cache and control unit. On the GPU side, it can be seen that the allocation of space for cache and control unit is relatively small and minimal, and mostly consists of relatively small but large number of arithmetic and logic units (ALU). This happens because the initial purpose of the GPU is to improve computer performance in processing graphic data so that it is processed faster so that it can appear on the screen faster. And this is achieved by increasing the number of ALU data processing units because the data processed is quite large and the instructions for processing the data are in the form of simple arithmetic, such as adding and subtracting times [17].

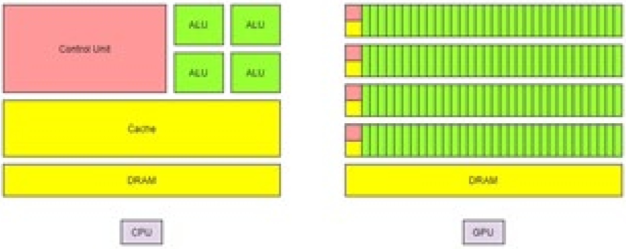
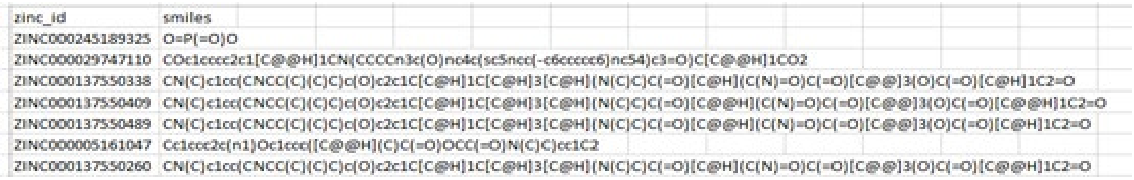


Figure 1. CPU and GPU architecture design illustration

Departing from the analogy of graphics processing, where each pixel will be processed in parallel, the GPU has many threads that are used as a place to process data. A collection of threads, called a block, has a shared cache or memory. A collection of blocks in the same place will form a grid [18]. On the GPU, the processing element used to process data is a thread. These abstractions will facilitate parallel programming when implementing research.

d. ZINC Database

ZINC is a commercially available molecular database. Basically, the molecular data contained in this database is in the form of a simplified molecular input line entry system (SMILES), but in its development, two-dimensional and three-dimensional representations of molecules are also available [19]. The ZINC database is often used in research to find ligands, which are ions or molecules that can attach to metal atoms by covalent bonds. These ligands are often used to find a cure for a disease or virus by disrupting the life cycle of the virus, or directly destroying the virus structurally. To find out the form of data from the ZINC database, please see Figure 2.



zinc_id	smiles
ZINC000245189325	O=P(=O)O
ZINC000029747110	COc1cccc2c1[C@@H]1CN(CCCCn3c(O)nc4c(sc5ncc(-c6ccccc6)nc54)c3=O)C[C@@H]1CO2
ZINC000137550338	CN(C)c1cc(CNCC(C)(C)C)c(O)c2c1C[C@H]1C[C@H]3[C@H](N(C)C)C(=O)[C@H](C(N)=O)C(=O)[C@@]3(O)C(=O)[C@H]1C2=O
ZINC000137550409	CN(C)c1cc(CNCC(C)(C)C)c(O)c2c1C[C@H]1C[C@H]3[C@H](N(C)C)C(=O)[C@@H](C(N)=O)C(=O)[C@@]3(O)C(=O)[C@@H]1C2=O
ZINC000137550489	CN(C)c1cc(CNCC(C)(C)C)c(O)c2c1C[C@H]1C[C@H]3[C@H](N(C)C)C(=O)[C@@H](C(N)=O)C(=O)[C@@]3(O)C(=O)[C@H]1C2=O
ZINC000005161047	Cc1ccc2c(n1)Oc1ccc([C@@H](C)C(=O)OCC(=O)N(C)C)cc1C2
ZINC000137550260	CN(C)c1cc(CNCC(C)(C)C)c(O)c2c1C[C@H]1C[C@H]3[C@H](N(C)C)C(=O)[C@H](C(N)=O)C(=O)[C@@]3(O)C(=O)[C@@H]1C2=O

Figure 2. Sample file download from ZINC database

e. Similarity Comparison Algorithm

Similarity comparison algorithm is an algorithm to determine the similarity between two data sequences. In bioinformatics, this algorithm has been found and used for quite a long time, namely since the 20th century [20]. The use and implementation of this algorithm is also wide, and depends on the type of data to be compared, such as protein sequence data, fingerprint data, binary data, and others. In this study, the data being compared is binary data in the form of fingerprint output from compound molecules.

Broadly speaking, the comparison of binary similarity carried out in this study aims to provide similarity values for two different binary strings [21]. For example, the binary strings being compared are 0110 and 1010. Each index in the string indicates the presence or absence of a feature in the referenced compound. Tanimoto's algorithm will find the number of digits of 1 in both strings that are at the same index (in this example it is the third index), and divide it by the number of digits of 1 in both strings at different indices (in this example, there is a value of 1 at the first index, second, and third). Tanimoto's algorithm will give 1/3 value for both binary strings above. In other words, Tanimoto's algorithm will divide the number of characteristics that are the same in both compounds by the total number of characteristics that exist in both compounds, whether only one compound has one or both.

Besides Tanimoto, there are several other algorithms to determine the value of binary similarity, namely Dice and Cosine. Dice's algorithm multiplies the number of traits that are the same in both compounds by 2, and then divides by the number of traits that exist in both compounds. The cosine algorithm will divide the number of features that are the same in both compounds by the root value of the product of the number of features that only one compound has. This study uses the Tanimoto algorithm because it is quite simple and has proven to be suitable for use as a basis for calculating the similarity of chemical compounds according to [21].

3. Methods

a. Research Stages

Figure 3 describes the stages and research methods used. As initial data, data on ZINC compounds were collected in the biogenic sub-group which amounted to 308.035 compounds. The fingerprint calculations used are MACCS and PubChem, and are carried out using CPU sequential and CPU parallel methods. After the fingerprint data is formed, parameter tuning is carried out on the GPU parallelization scheme, namely determining the block size and determining the number of streams. After the block size and the appropriate number of streams are determined, a similarity comparison process is carried out which is implemented using two methods, namely GPU sequential and GPU parallel. After calculating the time, the evaluation of the two methods is carried out by calculating the speed-up value.

After the best model was found, the overall ZINC data, which amounted to 997,402,117 compounds and the PubChem fingerprint was calculated, which was applied to the model to find compounds similar to Covid-19 drug compounds.

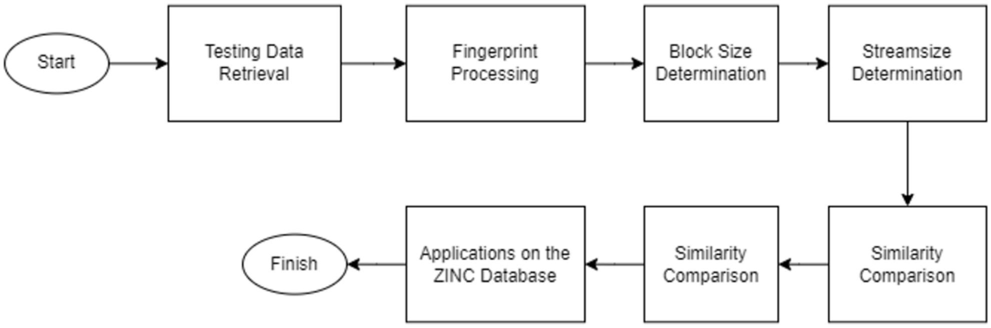


Figure 3. Flowchart of research stages

b. Research Data

The data used and analyzed in this study are all compound SMILES data in the ZINC database. Due to the large amount of data, the researchers took some of the ZINC data in the biogenic subset of 308.035 compounds as the basis for developing the algorithm. The data on these compounds were sought for the level of similarity to eight Covid-19 drug compounds that have been approved or are considered strong candidates in the medical world. To facilitate the search, a list of medicinal compounds used as a reference can be seen in Table 1.

Table 1. List of medicinal compounds used as a reference

Compound Name	How It Works	Reference
Remdesivir	Stops the replication of coronavirus RNA[a] inside the host cell.	[22]
Favipiravir	RNA-dependent RNA-polymerase inhibitors in common cold viruses.	[23]
Lopinavir	Antiretroviral agents, protease inhibitors.	[24]
Hydroxichloroquine	Causes alkalization in cells, preventing the acidization needed by viruses for replication.	[25]
Chloroquine	Causes alkalization in cells, preventing the acidization needed by viruses for replication.	[25]
Nitazoxanide	Suppress inflammation during a cytokine storm.	[26]
Oseltamivir	Inhibitors on the corona virus 3CLpro protein.	[27]

[a] Ribonucleic Acid, carrier of genetic information in virus.

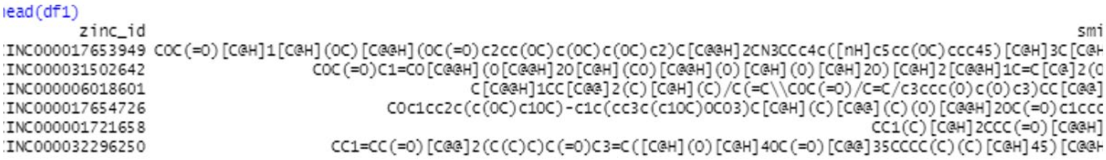
c. Fingerprint Processing

As can be seen in Figure 4, the data obtained from the ZINC database consists of zinc_id, which is a unique code for indexing each compound, and

SMILES. SMILES is a form of writing compounds in the form of strings so that they can be represented concisely and clearly. However, this representation cannot be used as a reference that a compound is similar or not, because SMILES is just a simple string form of a compound.

This SMILES representation needs to be converted into fingerprint form which is a collection of compound representations in binary form that is used to determine the level of similarity of a compound with other compounds. The use of fingerprints as data representation of a compound makes similarity comparison results more accurate because we judge from the properties brought by the structure of the compound itself.

This study uses two types of SMILES fingerprints that are commonly used, namely MACCS and PubChem fingerprints. MACCS encodes a compound into 166 binary bits, while PubChem fingerprint will encode a compound into 881 binary bits. Each binary bit present in MACCS and PubChem fingerprints is a molecular identifier that chemists commonly use to group compounds. Some examples of the characteristics used are the number of C atoms in the compound which is more than 4, as well as the presence of a cis/trans structure.



```
ıead(df1)
        zinc_id                                                                                                                                     smi
:INC000017653949 COC(=O)[C@H]1[C@H](OC)[C@@H](OC(=O)c2cc(OC)c(OC)c(OC)c2)C[C@@H]2CN3CCc4c([nH]c5cc(OC)ccc45)[C@H]3C[C@H
:INC000031502642                  COC(=O)C1=CO[C@@H](O[C@@H]2O[C@H](CO)[C@@H](O)[C@H](O)[C@H]2O)[C@H]2[C@@H]1C=C[C@]2(O
:INC000006018601                                   C[C@@H]1CC[C@@]2(C)[C@H](C)/C(=C\\COC(=O)/C=C/c3ccc(O)c(O)c3)CC[C@@]
:INC000017654726                        COc1cc2c(c(OC)c1OC)-c1c(cc3c(c1OC)OCO3)C[C@H](C)[C@@](C)(O)[C@@H]2OC(=O)c1ccc
:INC000001721658                                                                          CC1(C)[C@H]2CCC(=O)[C@@H]
:INC000032296250                 CC1=CC(=O)[C@@]2(C(C)C)C(=O)C3=C([C@H](O)[C@H]4OC(=O)[C@@]35CCCC(C)(C)[C@H]45)[C@@H
```

Figure 4. Data downloaded from ZINC database in Rstudio environment

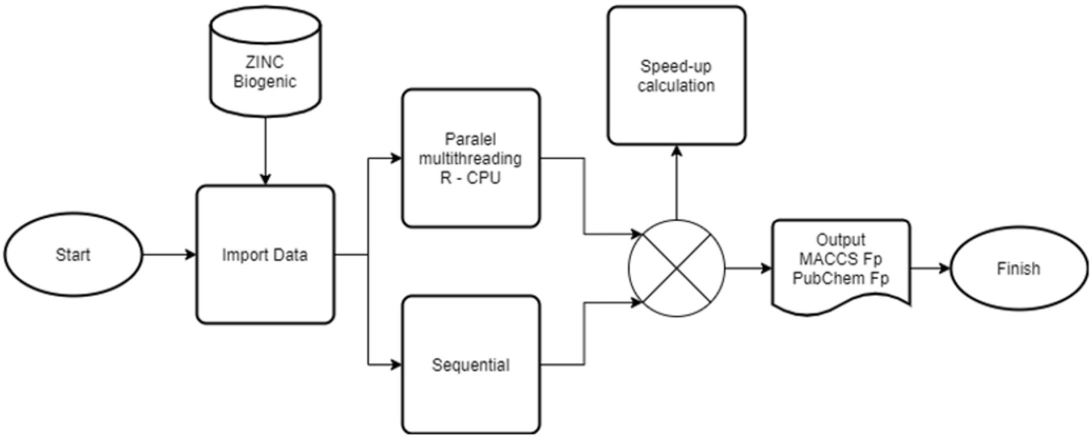
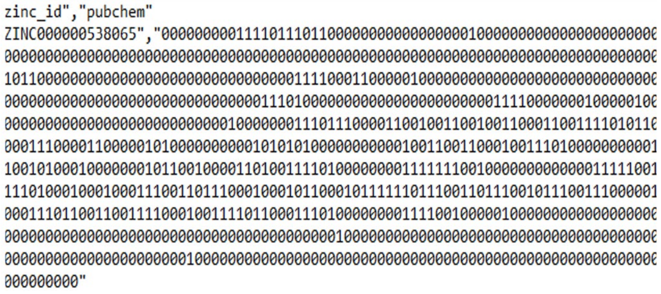


Figure 5. Fingerprint processing flow chart

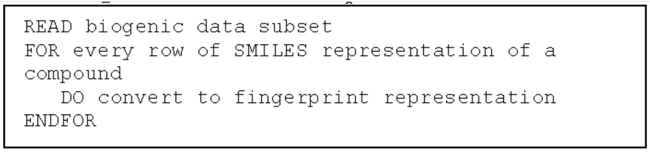


```
zinc_id","pubchem"
ZINC000000538065","0000000001111011101100000000000000000001000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000
1011000000000000000000000000000000000111100011000001000000000000000000000000000000
0000000000000000000000000000000001110100000000000000000000000001111000000000100000100
0000000000000000000000000000100000001110111000011001001100100110001100111101011
0001110000110000010100000000001010101000000000000100110011000100111010000000000001
1001010001000000010110010000110100111101000000001111111001000000000000001111001
1110100010001000111001101110001000101100010111111011100110111001011100111000001
0001110110011001111000100111101100011101000000001111001000001000000000000000000
0000000000000000000000000000000000000000000001000000000000000000000000000000000
0000000000000000000000001000000000000000000000000000000000000000000000000000000
000000000"
```

Figure 6. PubChem fingerprint representation example

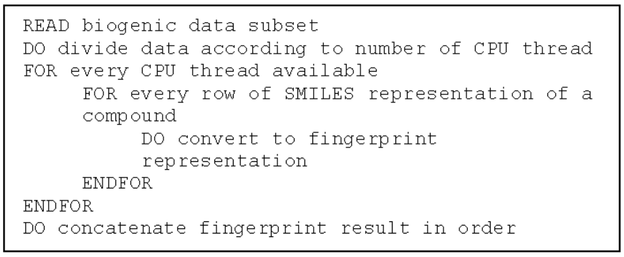
In general, the flowchart of the fingerprint processing process is depicted in Figure 6. This process begins by importing a subset of ZINC data into the RStudio software, then performs two kinds of fingerprint processing, namely CPU sequential and CPU parallel. The sequential process will process MACCS and PubChem fingerprints one by one. The parallel process will divide this fingerprint processing task into all available CPU cores, i.e. 32 cores. The parallelization process is carried out with the CPU, because it is more practical and already supports the required library, i.e. rcdk.

After the fingerprint processing is complete, then the speed-up value is calculated as a result of sequential and parallel processing time. The output produced is a MACCS fingerprint of 166 bits and a PubChem fingerprint of 881 fingerprints. This output will be used in the next process, namely calculating the similarity or similarity using CUDA. The flowchart of this process can be seen in Figure 5. An example of the resulting fingerprint can be seen in Figure 6.



```
READ biogenic data subset
FOR every row of SMILES representation of a
compound
   DO convert to fingerprint representation
ENDFOR
```

Figure 7. Pseudocode for fingerprint processing CPU sequential algorithm



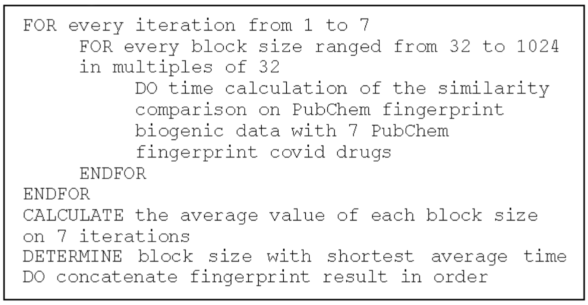
```
READ biogenic data subset
DO divide data according to number of CPU thread
FOR every CPU thread available
     FOR every row of SMILES representation of a
     compound
          DO convert to fingerprint
          representation
     ENDFOR
ENDFOR
DO concatenate fingerprint result in order
```

Figure 8. Pseudocode for parallel CPU fingerprint processing algorithm

Figures 7 and 8 show the algorithm used to process fingerprints, both MACCS and PubChem fingerprints. The parallelization scheme used is CPU parallelization with Single Instruction Multiple Data (SIMD).

d. Block Size Determination

Determining the block size is important to determine the optimal value of threads per block that will be used in the GPU parallelization process. Figure 9 shows the process of determining the most optimal block size to use. Since the CUDA architecture has a warp size of 32, the block size will also be determined in multiples of 32, up to a maximum value of 1024 threads per block.

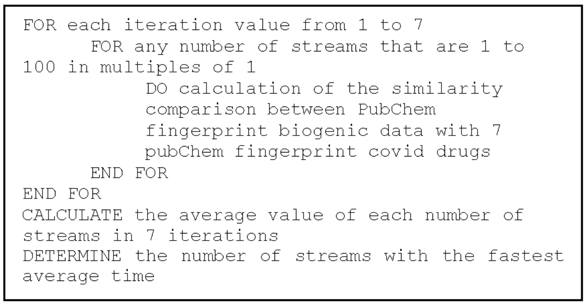


```
FOR every iteration from 1 to 7
     FOR every block size ranged from 32 to 1024
     in multiples of 32
          DO time calculation of the similarity
          comparison on PubChem fingerprint
          biogenic data with 7 PubChem
          fingerprint covid drugs
     ENDFOR
ENDFOR
CALCULATE the average value of each block size
on 7 iterations
DETERMINE block size with shortest average time
DO concatenate fingerprint result in order
```

Figure 9. Pseudocode for block size determination algorithm

e. Streamsize Determination

Determining the streamsize is needed to find out



```
FOR each iteration value from 1 to 7
      FOR any number of streams that are 1 to
100 in multiples of 1
           DO calculation of the similarity
           comparison between PubChem
           fingerprint biogenic data with 7
           pubChem fingerprint covid drugs
      END FOR
END FOR
CALCULATE the average value of each number of
streams in 7 iterations
DETERMINE the number of streams with the fastest
average time
```

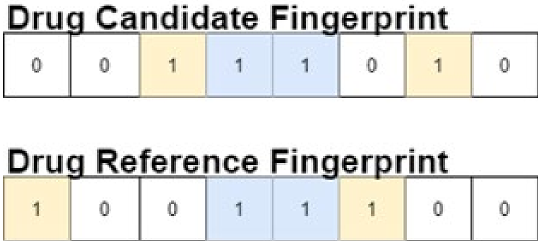
Figure 10. Pseudocode for streamsize determination algorithm

f. Similarity Comparison

1) Tanimoto Similarity

Tanimoto similarity algorithms will compare each bit in the same location. This is done considering that fingerprints at the same position show the same compound markers, so we can see whether or not a pair of compounds from these markers is similar.

For example, in Figure 11 there is one pair of compound fingerprints being compared. Tanimoto similarity is calculated by counting the number of all bits in a position where the two compounds have the same value, which is one, then divided by the number of all bit positions where one of the two compounds is worth one. In the example above, there are two positions where the two compounds have the same value of one, namely the fourth and fifth positions. And there are six positions where one of the two compounds is worth one, namely positions one, three, four, five, six, and seven. So that the similarity value of the two compounds is two-sixth, or in other words one-third.



Drug Candidate Fingerprint

0	0	1	1	1	0	1	0

Drug Reference Fingerprint

1	0	0	1	1	1	0	0

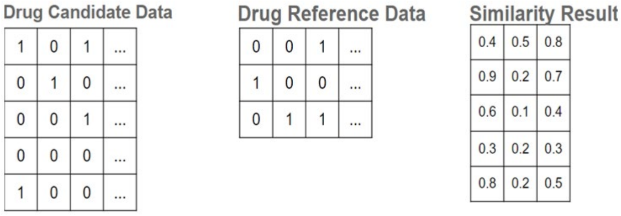
Figure 11. Illustration of Tanimoto similarity calculation

2) Sequential Algorithm

The sequential algorithm takes one candidate compound data represented by one data line, then calculates its similarity with several drug compounds that have been selected, and stores the results in a table that is sized according to the number of candidate compounds and drug compounds.

In the illustration in Figure 12, as well as the pseudocode in Figure 13, we take one fingerprint data on the top row, then compare it with the three existing drug fingerprints. The results of the Tanimoto similarity calculation are written in the top row Similarity Results table, so that each row of the results table represents the candidate compound, and each column represents the drug compound being compared. The table of Similarity Results in the first row and third column shows the Tanimoto similarity value between the first candidate compound and the third drug compound.

As the name implies, this algorithm is performed sequentially in a CPU thread, so the processing time depends on the CPU's ability to process data. This implementation is written in a Python language program using the pandas library to perform data import, data export, and dataframe management, as well as the NumPy library to perform operations on two-dimensional arrays.



Drug Candidate Data

1	0	1	...
0	1	0	...
0	0	1	...
0	0	0	...
1	0	0	...

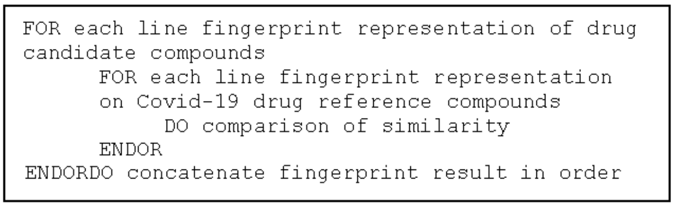
Drug Reference Data

0	0	1	...
1	0	0	...
0	1	1	...

Similarity Result

0.4	0.5	0.8
0.9	0.2	0.7
0.6	0.1	0.4
0.3	0.2	0.3
0.8	0.2	0.5

Figure 12. Representation of the similarity comparison process



```
FOR each line fingerprint representation of drug
candidate compounds
      FOR each line fingerprint representation
      on Covid-19 drug reference compounds
           DO comparison of similarity
      ENDOR
ENDORDO concatenate fingerprint result in order
```

Figure 13. Pseudocode for sequential algorithm for similarity comparison on CPU

3) Parallelization Algortihm

Parallelization of the similarity comparison process is carried out by utilizing the Nvidia CUDA General Purpose GPU (GPGPU) as a processor. Each thread in the GPU is used to process one candidate compound with the eight drug compounds. After calculating the similarity value, this GPU writes the value to the result table, like the sequential process above. This term is known as single instruction, multiple data (SIMD).

Figure 14 also shows that apart from dividing each compound into one thread, multistreaming is also carried out in this process, namely dividing the parallel process into several different streams. This is achieved by dividing the compound data equally into each stream, and copying the same drug reference data to each stream to be used, so that each stream does not need to communicate with other streams and can run optimally.

This mechanism can speed up the parallelization process because each stream has its own clock, allowing these processes to run asynchronously. Of course, this process only occurs until the data is written into the similarity table. At the end of the process, each stream collects a table of results and rewrites it in a table of similarity results sequentially, starting from the first stream to the last. The writing of these results is carried out in synchronization, so that there is no overlap, and the order of the resulting data tables is in accordance with the order of the compounds used. This algorithm is briefly described in the pseudocode in Figure 15.

In GPU parallelization abstraction, apart from being known as a thread that does real work, there is also a term called a block which is a collection of threads that are in the same container. Programmers need to determine the appropriate block size so as to achieve maximum performance gain. Therefore, in this study, various parallelization schemes with different block sizes were used, then repeated seven times, so as to find a block size that could process the training data optimally. The block size values used are in the range of 32, 64, 96, and so on up to 1024. This is because the maximum number of threads in one block on the CUDA architecture is only up to 1024 threads per block, and the warp size on the CUDA architecture is 32, so that to achieve optimal efficiency it is necessary to have a block size that has multiples of 32.

As mentioned above, the multistreaming mechanism is implemented so that each parallel process runs asynchronously, so it can run faster. This study also calculates similarity with a number of different streams, then repeated seven times, so that researchers can determine the best number of streams to use in parallelizing the entire ZINC database. The value of the number of streams used is in the range of one to one hundred streams.

This parallel similarity comparison process is written in the Python programming language as the basis, with the help of the PyCUDA library which is used as programming code that occurs on the GPU. On each GPU thread, the program code is written in C++, according to the needs of the CUDA library in order to execute its commands.

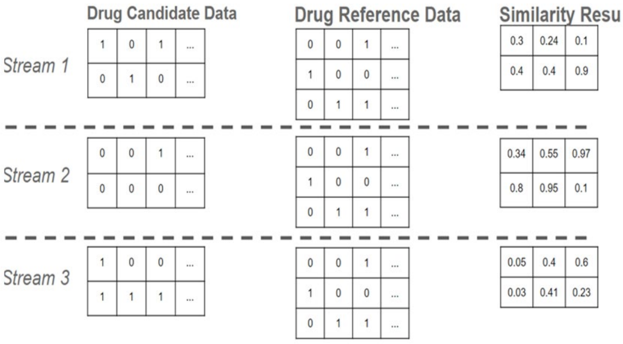


Figure 14. Representation of the parallel comparison process

```
DO stream creation according to the optimal
number of streams
DO duplicate Covid-19 drug fingerprint data
DO distribution of drug candidate compound data
to each stream according to the number of
existing streams
FOR every stream on GPU
        FOR each GPU thread that calculates
        similarity values line fingerprint
        representation on drug candidate compounds
              FOR each row fingerprint
              representation of the reference drug
              data
                    DO comparison of similarity
              ENDOR
        ENDOR
ENDOR
DO concatenation of similarity results in order
```

Figure 15. Pseudocode for parallel GPU similarity comparison algorithm

g. Algorithm Evaluation

Parallel algorithms and sequential algorithms are compared by means of calculating the speed-up value. The speed-up value is the ratio of the time it takes to run the sequential algorithm to the time it takes to run the parallel algorithm. For example, if the sequential algorithm takes 9 seconds and the parallel algorithm takes 5 seconds, then the speed-up value obtained is 1.8.

$$speed\text{-}up = t_{sequential} / t_{parallel} \quad (1)$$

Equation (1) is a general formula for calculating speed-up values, where $t_{sequential}$ is the time it takes to do something sequentially, and $t_{parallel}$ is the time it takes to do something in parallel. The greater the difference in sequential and parallel time, the higher the speed-up value, with a note that the sequential time is longer than the parallel time.

h. Application on the ZINC Database

After obtaining a good parallelization model from the biogenic subset ZINC training data, the model was applied to all datasets in the ZINC database. This parallelization process takes a long time so that by implementing parallelization, data processing time can be accelerated. The results of processing this entire database show information about several potential compounds that are similar to Covid-19 drug compounds. This study uses the PubChem fingerprint to process the entire ZINC database because it has more descriptors, so it can identify similar compounds more accurately. For comparison, the training data used were 308,035 compounds, while the ZINC database totaled 997,402,117 compounds.

i. Development Environment

The software and hardware specifications used in this study are as follows:

Device type	: Desktop PC
Operation System	: Xubuntu 18.04
Processor	: Intel Xeon Silver 4110 2,2 GHz
Memory	: 64 GB
GPU	: NVIDIA RTX 2080
Storage Drive	: SSD 512 GB, HDD 7 TB
Programming language	: C++, Python, dan R
Library support	: PyCUDA dan rcdk
Software	: RStudio dan Visual Studio Code

4. Results and Discussion

a. Fingerprint Processing

Figure 16 and Figure 18 show the difference in processing time of the sequential and parallel algorithms for both MACCS and PubChem fingerprints. The graphs shown in Figure 17 and Figure 19 show the speed-up values obtained for the conversion process from the SMILES representation to MACCS and PubChem fingerprints. Overall, the speed-up value obtained is quite large and significant. This shows that the use of parallel processing with multithreaded CPUs to process fingerprints can have a significant impact on processing time.

In Figure 17, there is a very significant increase in the speed-up value of the number of drug candidates. In the number of candidate drug compounds, which amounted to a thousand and under, an insignificant increase was seen. The increase in the speed-up value is starting to be large and can be seen in the amount of data as much as five thousand and above. By looking at this graph, it can be seen that the scalability potential of MACCS fingerprint processing is still very high, or in other words, the -speed-up value can still increase again as the number of candidate compounds processed increases. The graph also shows that further research is needed on the potential limitations of parallel processing for MACCS fingerprint processing to get the optimal speed-up value. So far, the highest speed-up value of 27.73 was obtained from a total of 300,000 drug candidates. In other words, if we parallel processing MACCS fingerprints on 300,000 compound data, it will be 27 times faster than if we process them sequentially. Taking into account the number of processor threads, which are 32, the efficiency of the MACCS parallelization process with 300,000 drug candidates is 86.66%.

In Figure 19 it can be seen that the speed-up value that occurs in the PubChem fingerprint processing of 15.27 tends to be smaller when compared to the MACCS fingerprint which can reach a value of 27. Taking into account the number of threads, an efficiency value of 47.72% is also obtained. There was a significant increase in the number of candidates from ten, fifty, one hundred, and five hundred. For the number of candidates of a thousand and above, the speed-up value looks increasingly sloping, until the values of 50,000, 100,000, and 300,000 appear to have gone up and down, so it is quite possible that the threshold has been reached in this area. The speed-up difference between MACCS and PubChem fingerprint processing is probably caused by the number of bits in MACCS and PubChem, namely MACCS with 166 bits and PubChem with 881 bits. Of course this will affect the work of each thread that processes the fingerprint, so the work done by a thread to process the PubChem fingerprint will be greater than the MACCS fingerprint. This also explains the speed-up value that tends not to increase in MACCS fingerprint processing for the number of candidate compounds of ten, fifty, one hundred, five hundred, and one thousand, because the time used to process fingerprints is shorter than the time used to divide the fingerprints raw data and collect processing data from each working thread.

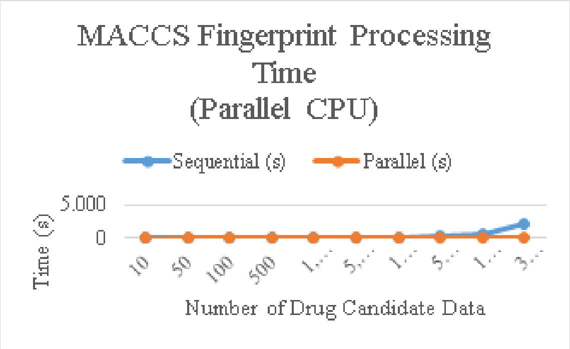


Figure 16. Graph of MACCS fingerprint processing time (Parallel CPU)

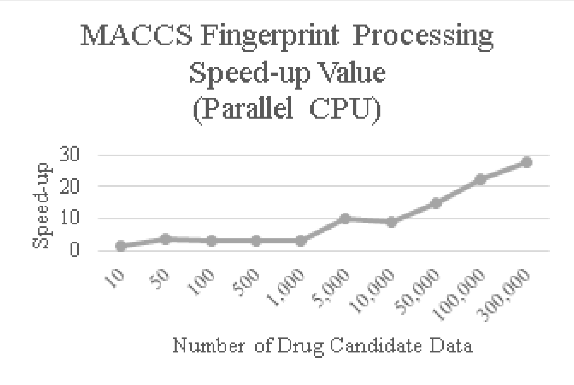


Figure 17. Graph of MACCS fingerprint processing speed-up value (parallel CPU)

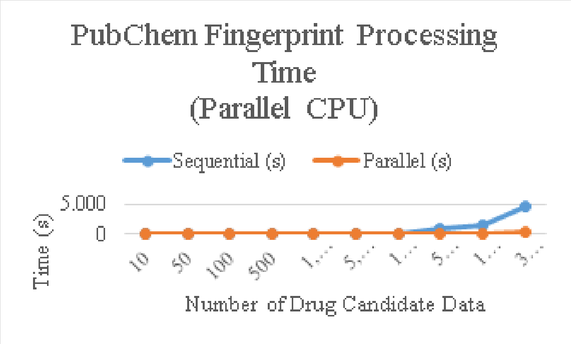


Figure 18. Graph of PubChem fingerprint processing time (parallel CPU)

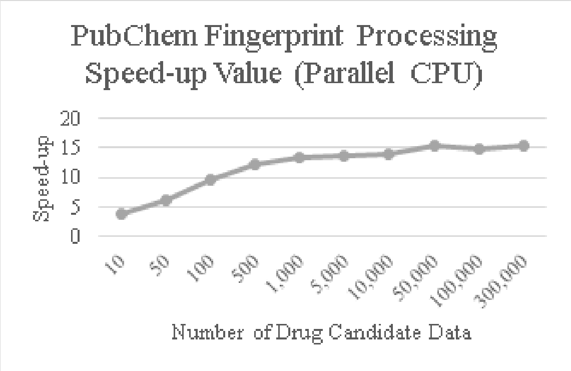


Figure 19. Graph of PubChem fingerprint processing speed-up value (parallel CPU)

b. Block Size Determination

Figure 20 shows that the run times for various block sizes are dynamic and have a global minimum of around 600 threads per block. The dark yellow line shows the average processing time of seven replicates. After looking for the least average value, we get a block size of 640 threads per block with an average time of 1.176 seconds.

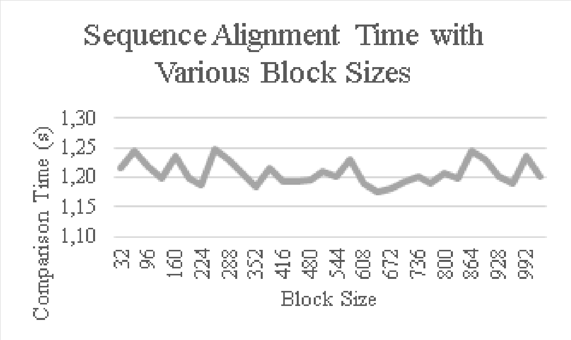


Figure 20. Graph of block size value with comparison time on the similarity comparison algorithm

c. Streamsize Determination

The average value indicated by the dark yellow line in Figure 21 shows the wave-like oscillation or looping. It can be seen in the number of flows from one to five, the processing time tends to decrease, then continues at the number of flows above five which slowly shows an increase. The lowest value refers to the number of flows of five with a processing time of 0.828 seconds. This shows that the use of the number of streams less than five is included in a non-optimal state, because the number of streams is not proportional to the amount of data processed. A value of more than five, which indicates an increase in processing time, indicates that the number of streams is too large for the size of the data being processed, so the computer does the unnecessary work of creating and allocating data divisions into these additional streams.

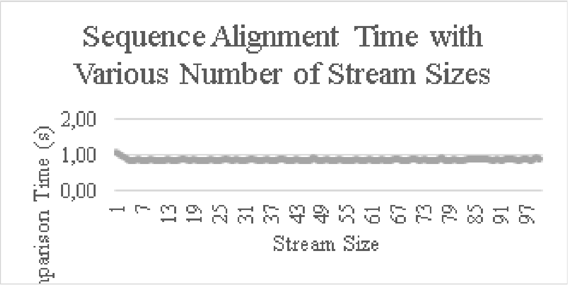


Figure 21. Graph of the amount of stream size against the comparison time on the similarity comparison algorithm

d. Similarity Comparison

Figure 22 and Figure 24 show the time difference between the sequential and parallel similarity comparison processes, for both MACCS and PubChem fingerprints. The graphs shown in Figure 23 and Figure 25 show that the sequential and parallel comparisons of the similarity comparison algorithms, for both MACCS and PubChem fingerprints, result in significant speed-up values. For MACCS fingerprint, the highest speed-up value is at 19.05 which is in the number of drug candidates of 300,000, while for PubChem fingerprint, the highest speed-up value is at 55.51 which is in the number of drug candidates of 100,000. This difference in speed-up values indicates that the tasks that can be parallelized on the PubChem fingerprint tend to be more, given the number of bits that reach 881.

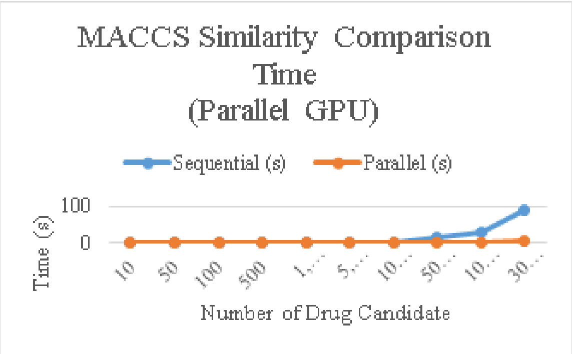


Figure 22. Graph of the similarity comparison processing time with the MACCS fingerprint (parallel GPU)

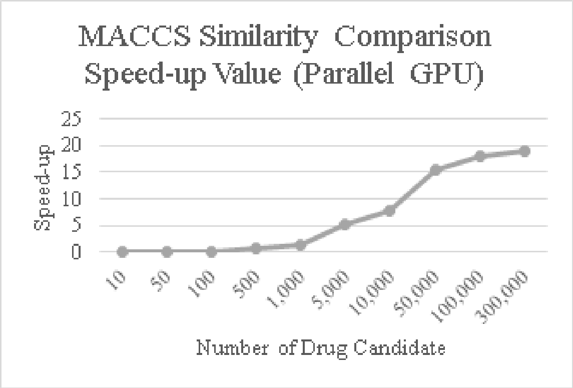


Figure 23. Graph of speed-up value for comparison of similarity with MACCS fingerprint (parallel GPU)

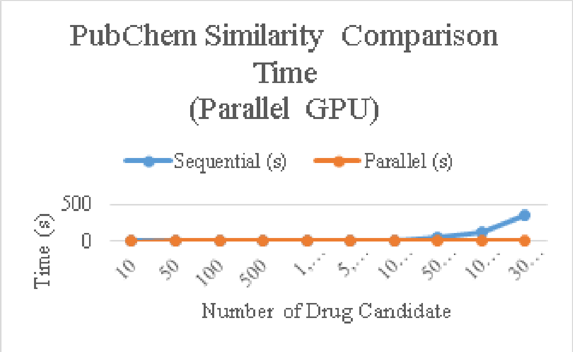


Figure 24. Graph of the similarity comparison processing time with the PubChem fingerprint (parallel GPU)

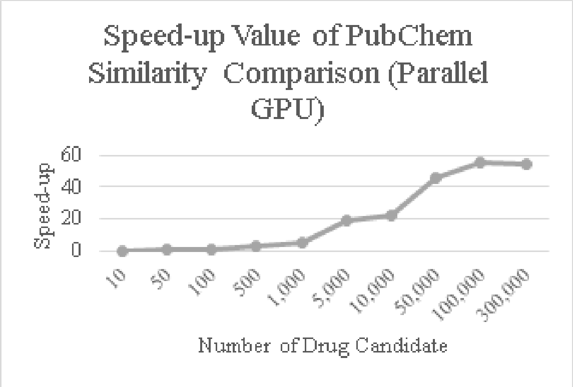


Figure 25. Graph of speed-up value comparing similarity with PubChem fingerprint (GPU parallel)

e. Application Result on ZINC Database

After applying the similarity comparison algorithm in parallel to the entire ZINC database, several compounds have the potential to be drug candidates. Most of the compound data that has a high similarity value are included in the unnamed compound category, so further identification cannot be done. In summary, the results of the similarity search can be seen in Table 2.

Table 2. Caption of my table (inf_tableHead)

Drug Name	*Siimilarity Value*	*Drug Candidate ID*	*Drug/Compound Candidate Name*	*Natural Source*
Remde-sivir	0.7799	ZINC- 29134440	Brivanib Alaninate	-
	0.7567	ZINC- 13684256	Brivanib	-
	0.7391	ZINC- 53147179	Puromycin	*Streptomyces alboninger* bacteria [28]
Favipi-ravir	0.7500	ZINC- 5116994	5-hydroxypy-razinamide	-
	0.7478	ZINC- 1081066	2-carboxy-pyrazine	-
	0.6850	ZINC- 500059	Pyrazine Carboxylic Acid Hydrazide	-
Lopi-navir	0.8214	ZINC- 49889244	Gliotide	*Gliocladium sp.* fungi [29]
	0.8214	ZINC- 95607016	Paecilodepsi-peptide C	*Paecilomyces cinnamomeus* fungi [30]
	0.8083	ZINC- 230078061	Adouetine Y	*Discaria Americana* tree bark [31]
Hydro-xichlo-roquine	1.000	ZINC- 1530654	Hydroxy-chloroquine	*Peruvia cinchona* tree bark [32]
	0.9934	ZINC- 1843038	Cletoquine	-
	0.9334	ZINC- 19144231	Chloroquine	*Peruvia cinchona* tree bark [32]
Chloro-quine	1.000	ZINC-19144231	Chloroquine	*Peruvia cinchona* tree bark [32]
	0.9929	ZINC- 1873617	Desethyl-chloroquine	-
	0.9858	ZINC- 6036375	Bidesethyl-chloroquine	-
Nitazo-xanide	0.8971	ZINC- 5924265	Tizoxanide	-
	0.8839	ZINC- 29124339	Tizoxanide Glucuronide	-
	0.7273	ZINC- 2257	Zolamine	-
Oselta-mivir	0.7593	ZINC- 13370140	Antillatoxin	*Lyngbya majuscule* [33]
	0.7265	ZINC-169367255	Aspochalasin J	*Aspergillus flavipes* [34]
	0.7248	ZINC-169290233	Cespitulactam J	*Cespitularia taeniata* [35]

Starting from the first drug, a candidate drug compound similar to remdesivir is puromycin, but this is not feasible for further research because of the low similarity value of 0.7391. The other two compounds cannot be found in natural sources, and their production can only be carried out in the laboratory.

The second drug, favipiravir, had no naturally-derived compounds in its top search results. It can also be seen that the most similar to favipiravir is 5-hydroxypyrazinamide which has a similarity level of 0.75, low enough to be considered similar.

The third drug, lopinavir, has several candidate compounds that are quite similar, namely gliotide, paecilopdepsipeptice C, and adouetine Y. These three compounds can be found sequentially in the fungus *Gliocladium sp.*, the fungus *Paecilomyces cinnamomeus*, and the bark of *Discaria americana*. It should also be noted that the similarity value of these three candidate compounds is not too high, which is in the range of 0.80 – 0.82.

For the fourth and fifth drug candidates, namely hydroxychloroquine and chloroquine, both can be found in the bark of the *Peruvia cinchona* plant. The thing to note is that this plant is not native to Indonesia, so it will be difficult to find it.

The sixth drug, nitazoxanide, had no naturally-derived compounds in its top search results. Compounds such as tizoxanide, tizoxanide glucuronide, and zolamine can only be produced in the laboratory.

The seventh drug, namely oseltamivir, had similarity to natural sources in the similarity range of 0.70 to 0.75. 75% similarity was found in antillatoxin compounds, namely toxin compounds obtained from the marine cyanobacteria *Lyngbya majuscula*. 72% similarity was found in aspochalasin J and Cespitulactam B compounds, which were found in the fungus *Aspergillus flavipes* and the coral *Cespitularia taeniata*, respectively.

Finally, because the limitation of this research is to search for herbal ingredients, namely natural ingredients derived from plants, this search resulted in *Gliocladium sp.*, *Paecylomyces cinnamomeus*, *Discaria americana*, and *Peruvia cinchona*.

5. Conclusion and Suggestion

a. Conclusion

The similarity comparison algorithm can be applied to find the similarity value in molecular compound data in the form of SMILES by finding the descriptor in the form of a fingerprint, then comparing the two compound fingerprints with Tanimoto similarity. The use of GPU computing can increase the performance of this process up to 55 times faster.

There are several candidate herbal plants, and even the coronavirus drug compounds themselves can be found in nature and become candidates for Covid-19 drug compounds. Some of the herbal ingredients are difficult to find in Indonesia because they are from abroad.

b. Suggestion

Although this research focuses on herbal medicines, it does not rule out the possibility that non-herbal compounds obtained from the ZINC database screening can be a more practical alternative to become Covid-19 drugs. Researchers hope that the results of this study can be suggestions, input, and motivation to carry out further analysis in the laboratory to identify herbal ingredients that can be used to treat the corona virus.

In addition, researchers also hope that the results of research on the content of herbal compounds present in plants and organisms from Indonesia can be collected into a single database so that a comprehensive in silico analysis can be carried out. This suggestion arises from the difficulty of researchers in further identifying whether a compound is present in certain herbal plants in Indonesia.

Reference

[1] World Health Organization Indonesia, "Weekly Epidemiological Update on COVID-19," World Health Organization Indonesia, Jakarta, Indonesia, 23 Jul. 2021.

[2] C. G. Bologa, O. Ursu, and T. I. Oprea, "How to prepare a compound collection prior to virtual screening," *Methods in Molecular Biology,* vol. 1939, pp. 119–138, 2019.

[3] T. T. Ashburn and K. B. Thor, "Drug repositioning: Identifying and developing new uses for existing drugs," *Nature Reviews Drug Discovery*, vol. 3, no. 8, pp. 673–683, 2004.

[4] F. Medail and P. Quezel, "Biodiversity hotspots in the Mediterranean Basin: Setting global conservation priorities," *Conservation Biology*, vol. 13, no. 6, pp. 1510–1513, 1999.

[5] A. Athanasopoulos, A. Dimou, V. Mezaris, I. Kompatsiaris, Z. Wen, J. Shi, et al., "Performance of deep learning computation with TensorFlow software library in GPU-capable multi-core computing platforms," in *International Workshop on Image Analysis for Multimedia Service,* vol. 19, pp. 240–242, 2017.

[6] A. Reinaldo, W. A. Kusuma, H. Rahmawan, and Y. Herdiyeni, "Implementation of breadth-first search parallel to predict drug-target interaction in plant-disease graph," in *International Conference on Computer Science and Its Application in Agriculture (ICOSICA)*, pp. 1-5, 2020.

[7] B. Imbernon, A. Serrano, A. Bueno-Crespo, J. L. Abellan, H. Perez-Sanchez, and J. M. Cecilia, "METADOCK 2: a high-throughput parallel metaheuristic scheme for molecular docking," *Bioinformatics,* vol. 37, no. 11, pp. 1515-1520, 2021.

[8] I. Sanchez-Linares, H. Perez-Sanchez, J. M. Cecilia, and J. M. Garcia, "High-throughput parallel blink virtual screening using BINDSURF," *BMC Bioinformatics*, vol. 13, no. 14, pp. 1-14, 2012.

[9] T. I. Oprea, C. G. Bologa, B. S. Edwards, E. R. Prossnitz, and L. A. Sklar, "Post-high-throughput screening analysis: An empirical compound prioritization scheme," *Journal of Biomolecular Screening*, vol. 10, no. 5, pp. 419–426, 2005.

[10] G. M. Rishton, "Reactive compounds and in vitro false positives in HTS," *Drug Discovery Today*, vol. 2, no. 9, pp. 382–384, 1997.

[11] G. M. Rishton, "Nonleadlikeness and leadlikeness in biochemical screening," *Drug Discovery Today*, vol. 8, no. 2, pp. 86–96, 2003.

[12] S. L. McGovern, E. Caselli, N. Grigorieff, and B. K. Shoichet, "A common mechanism underlying promiscuous inhibitors from virtual and high-throughput screening," *Journal of Medicinal Chemistry*, vol 45, no. 8, pp. 1712–1722, 2002.

[13] V. P. Kamboj, "Herbal medicine," *Current Science*,

vol. 78, no. 1, pp. 35-39, 2000.

[14] S. K. Pal and Y. Shukla, "Herbal medicine: current status and the future," *Asian Pacific Journal of Cancer Prevention*, vol. 4, no. 4, pp. 281-288, 2003.

[15] P. Basnet and N. Skalko-Basnet "Curcumin: An anti-inflammatory molecule from a curry spice on the path to cancer treatment," *Molecules*, vol. 16, no. 6, pp. 4567–4598, 2011.

[16] D. Luebke, M. Harris, N. Govindaraju, A. Lefohn, M. Houston, J. Owens, et al., "GPGPU: general-purpose computation on graphics hardware," in *ACM/IEEE Conference on Supercomputing*, 2006.

[17] F. Li, Y. Ye, Z. Tian, and X. Zhang, "CPU versus GPU: which can perform matrix computation faster—performance comparison for basic linear algebra subprograms," *Neural Computing and Applications*, vol. 31, no. 8, pp. 4353–4365, 2019.

[18] NVIDIA Corporation, "CUDA Toolkit Documentation - v11.4.0," NVIDIA Corporation, Santa Clara, United States, 2021.

[19] J. J. Irwin, T. Sterling, M. M. Mysinger, E. S. Bolstad, and R. G. Coleman, "ZINC: a free tool to discover chemistry for biology," *Journal of Chemical Information and Modeling*, vol. 52, no. 7, pp. 1757–1768, 2012.

[20] F. Corpet, "Multiple sequence alignment with hierarchical clustering," *Nucleic Acids Research*, vol. 16, no. 22, pp. 10881–10890, 1984.

[21] D. Bajusz, A. Racz, and K. Heberger, "Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations?," *Journal of Cheminformatics*, vol. 7, no. 1, pp. 1-13, 2015.

[22] Y. C. Cao, Q. X. Deng, and S. X. Dai, "Remdesivir for severe acute respiratory syndrome coronavirus 2 causing COVID-19: An evaluation of the evidence," *Travel Medicine and Infectious Disease*, vol. 35, pp. 101647, 2020.

[23] Y. Furuta, T. Komeno, and T. Nakamuba, "Favipiravir (T-705), a broad spectrum inhibitor of viral RNA polymerase," in *Proceedings of the Japan Academy, Series B*, vol. 93, no. 7, pp. 449–463, 2017.

[24] R. S. Cvetkovic and K. L. Goa, "Lopinavir/ Ritonavir", *Drugs*, vol. 63, no. 8, pp. 769–802, 2003.

[25] Z. N. Lei, Z. X. Wu, S. Dong, D-H. Yang, L. Zhang, Z. Ke, C. Zou, and Z-S. Chen, "Chloroquine and hydroxychloroquine in the treatment of malaria and repurposing in treating COVID-19," *Pharmacology & Therapeutics*, vol. 216, no. 1, pp. 107672, 2020.

[26] D. B. Mahmoud, Z. Shitu, and A. Mostafa, "Drug repurposing of nitazoxanide: can it be an effective therapy for COVID-19?," *Journal of Genetic Engineering and Biotechnology*, vol. 18, no. 35, pp. 1–10, 2020.

[27] A. Tan, L. Duan, Y. Ma, Q. Huang, K. Mao, W. Xiao, et al., "Is oseltamivir suitable for fighting against COVID-19: In silico assessment, in vitro and retrospective study," *Bioorganic Chemistry*, vol. 104, pp. 104257, 2020.

[28] T. F. de Koning-Ward, A. P. Waters, and S. Brendan, "Puromycin-N-acetyltransferase as a selectable marker for use in *Plasmodium falciparum*," *Molecular and Biochemical Parasitology*, vol. 117, no. 2, pp. 155-160, 2001.

[29] G. Lang, M. I. Mitova, G. Ellis, S. van der Sar, R. K. Phipps, J. W. Blunt, et al., "Bioactivity profiling using HPLC/microtiter-plate analysis: application to a New Zealand marine alga-derived fungus, *Gliocladium sp.*," *Journal of Natural Products*, vol. 69, no. 4, pp. 621-624, 2006.

[30] M. Isaka, S. Palasarn, S. Lapanun, and K. Sriklung, "Paecilodepsipeptide A, an antimalarial and antitumor cyclohexadepsipeptide from the insect pathogenic fungus *Paecilomyces cinnamomeus* BCC 9616," *Journal of Natural Products*, vol. 70, no. 4, pp. 675-678, 2007.

[31] S. R. Giacomelli, F. C. Missau, M. A. Mostardeiro, U. F. da Silva, I. I. Dalcol, N. Zanatta, and A. Morel, "Cyclopeptides from the bark of *Disaria americana*," *Journal of Natural Products*, vol. 64, no. 7, pp. 997-999, 2001

[32] A. G. Tolkushin, E. A. Luchinin, M. E. Kholovnya-Voloskova, and A. A. Zavyalov, "History of aminoquinoline preparations: from cinchona bark to chloroquine and hydroxychloroquinon," *Problemy Sotsial'noi Gigieny, Zdravookhraneniia i Istorii Meditsiny*, vol. 28 [special issue], pp. 1118-1122, 2020.

[33] R. Goto, K. Okura, H. Sakazaki, T. Sugawara, S. Matsuoka, and M. Inoue, "Synthesis and biological evaluation of triazole analogues of antillatoxin," *Tetrahedron*, vol. 67, no. 35, pp. 6659-6672, 2011.

[34] G. X. Zhou, E. K. Wijeratne, D. Bigelow, L. S. Pierson, H. D. VanEtten, and A. L. Gunatilaka, "Aspochalasins I, J, and K: three new cytotoxic cytochalasans of *Aspergillus flavipes* from the rhizosphere of *Ericameria laricifolia* of the Sonoran Desert," *Journal of Natural Products*, vol. 67, no. 3, pp. 328-332, 2004.

[35] Y. C. Shen, Y. B. Cheng, J. Kobayashi, T. Kubota, Y. Takahashi, Y. Mikami, et al., "Nitrogen-containing verticillene diterpenoids from the taiwanese soft coral *Cespitularia taeniata*," *Journal of Natural Products*, vol. 70, no. 12, pp. 1961-1965, 2007.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Job Scheduling on Grid Computing Using First Fit, Best Fit, and Worst Fit

Ardi Pujiyanta*, Fiftin Novianto
Informatics Study Program
Universitas Ahmad Dahlan, Yogyakarta
*ardipujiyanta@tif.uad.ac.id

Abstract-Grid computing can be considered as a large-scale distributed cluster computing and parallel distributed network processing. The two most important issues in managing user works are resource allocation and scheduling of required resources. When user jobs are submitted, they are managed by resource intermediaries which find and allocate the right resources. After the resource allocation stage, work is scheduled on the existing resources according to the user's required resources. In most grid systems with traditional scheduling, jobs are submitted and placed in waiting room queues to wait for the required resources to become available. Each grid system can use a different scheduling algorithm to execute jobs based on other parameters, such as resources, delivery time, and execution duration. There is no guarantee that these traditional scheduling algorithms will get the job done. The First Come First Serve Left Right Hole Scheduling (FCFS-LRH) reservation strategy improves resource utilization in a grid system by using a local scheduler, compared to traditional strategies. There are two objectives of this research. First, comparing the first fit, best fit, and worst fit algorithms to find empty timeslots and place them in a virtual view. Second, reducing the idle time value. The results showed that the FCFS-LRH method could reduce the idle time value of the FCFS-EDF and FCFS methods. The overall execution time of the first fit with the FCFS-LRH strategy is better than the FCFS-EDF.

Key Word: Grid computing, Scheduling, FCFS-LRH, FCFS-EDF

Article info: *submitted January 8, 2022, revised May 17, 2022, accepted May 27, 2022*

1. Introduction

In general, a grid computing system is used to increase the utilization of homogeneous or heterogeneous resources so that workload management will be optimal [1][2][3][4]. Computing resources facilitate organization formation such as servers, network nodes, storage elements [5]. Resources clustered together will result in a robust computing environment. Grid computing allows independent users and organizations to utilize untapped CPU cycles, such as databases, scientific tools, and storage elements. Millions of computer systems will be interconnected, placed on a global network with minimal access costs[6]. Grid computing is similar to Power Gridlines, as in power company operations. The grid system model provides the sharing of data and computing resources regardless of the location and origin of the resources. Grid users will submit their work to the Grid operating system via an interface. Then, the Grid system decides and finds computing resources that can serve the user's needs.[7]. Complete research on Grid done in the reference [8][9][10][11].

Grid computing is a promising next-generation science, engineering, and research problem-solving technology. Grid computing differs from conventional distributed computing in that it focuses on large-scale resources, sharing innovative applications. Grid computing is a problem-solving environment that leverages unused resources and maximizes resource capability. Grid computing uses an innovative approach in leveraging existing information technology infrastructure to optimize computing resources in managing data and computing workloads [12][13]. The grid computing platform enables the sharing, selection, and combination of geographically distributed heterogeneous resources (data sources and computers), belonging to different managerial organizations (virtual organizations) to answer large-scale engineering, commerce, and science problems.[14][15][16]. The primary purpose of parallel computers is to overcome the single processor speed blockage [17]. There are three approaches to creating parallel applications. The first approach is based on automatic parallelization, with this approach, the programmer does not have to worry about parallelizing jobs. The second approach is based on the use of parallel libraries. This approach has the same parallel code for multiple applications placed in the parallel library. The third approach is re-coding or writing code from scratch in making parallel applications. Programmers are free to choose the language and programming model used to create similar applications[18].

Jobs from users are submitted and managed by a resource broker who must find and allocate the right resources for the job. After the resource allocation stage, the work must be scheduled on the existing resources according to the user's required resources, in most of the grid systems with traditional scheduling, the work is submitted and placed in a waiting room queue to wait for the required resources to become available. Each grid system can use a different scheduling algorithm to execute jobs based on different parameters, such as the number of resources, delivery time, and execution duration. With this traditional scheduling algorithm(FCFS), there is no guarantee the job will be executed. FCFS-EDS is proposed to provide guaranteed jobs executed on grid computing[19]. First fit algorithm is used by FCFS-EDS to place jobs in empty spaces in virtual views. Once the job is placed in the virtual view, the user will be notified that the job has been accepted. The job to be executed will be mapped to the physical view. The weakness of FCFS-EDS is that user-submitted jobs are not placed on the left side of the virtual view used. By not placing a job on the left side of the virtual view, it is suspected that it can cause a high delay. The reservation strategy First Come First Serve Left Right Hole Scheduling (FCFS-LRH)[20] is proposed to improve resource utilization in the grid system. Job requests are sent based on number of jobs, initial start time, execution time. Incoming user requests will be sorted by priority of execution start time, execution time, and required amount of resources. The accepted job will be placed in the virtual view and sent to the physical view when it is executed. The purpose of this study: first, first fit, best fit, and worst fit algorithms will be used on FCFS-LRH then compared with FCFS-EDS first fit. Which algorithm has the best timing when the job is placed in the virtual view. Second, comparing the FCFS-LRH method with FCFS-EDS and FCFS, can the FCFS-LRH method reduce idle time?.

2. Methods

The steps in this research are as follows: first, determine the tools used in the study. Second, determine the amount of data to be used obtained from randomly generated data. Generate data using usability factors 2 and 3 and flexibility values from 25% to 100%. The three data generated results will use as input to the first fit, best fit, and worst fit algorithms to get the execution time value for the virtual node.

a. Tools and materials

1) Tools

Hardware and software requirements needed to run the simulation and test the proposed reservation scheduling algorithm in this study:

Hardware

a.	Prosesor	: Amd A 10-5750 M APU 2.50 GHz
b.	*Ram*	: 16 GB.
c.	*Disk drive*	: 320 GB.
d.	*Display*	: 12" *Wide-screen.*

Software

a. Operating Systems windows 8 64 bit.

b. *Eclipse Kepler Build id*:20130614-0229AppServ v2.5.8 : *Web Server.*

2) Materials

The data collection method used is a literature study method that refers to research data [19][21][22][23]. Figure 1 shows the use of a workload generator. The user submits a job description (1) based on the user's job description and grid description information, which will be used as input to the workload generator (2). The output of the workload generator is then submitted or sent back to the grid (3). The network environment is responsible for carrying out the work, returning the user output (4), and generating a detailed work report. The user processes all the results in a post-production step (5).

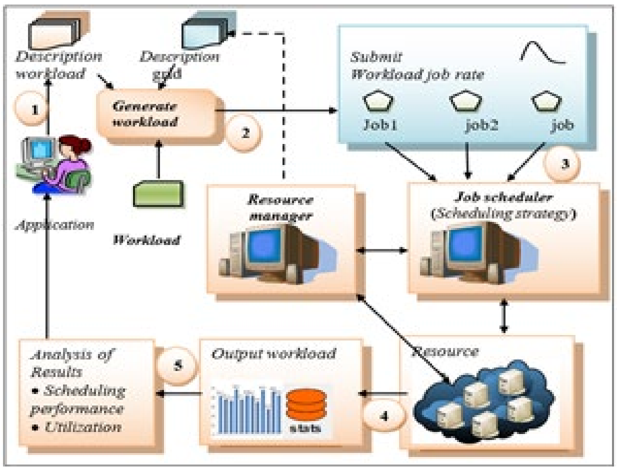


Figure 1. Generate Workload Process Model on Grid Computing [24]

b. Workload Generator

The scheduling performance proposed in this study checked using data generated from the workload generator. The workload generator output used as input to the proposed reservation scheduling. Characteristics of the workload generator in this study[19][21][22][23]:

1) The arrival rate of incoming jobs follows the Poisson distribution [21].
2) The execution period of each reservation request is uniformly distributed.

3) The earliest start time of each reservation is uniformly distributed.
4) The flexible reservation percentage is randomly selected.
5) Relax time range for each flexible reservation is uniformly distributed.
6) The required amount of resources is uniformly distributed.
7) The width of the timeslot in this study is 5 minutes [2].
8) The number of jobs generated is 800.

c. Method FCFS-LRH

An empty timeslot will be found in the virtual view when a user submits a job to the grid. If an empty timeslot is found, the job will be allocated to the virtual view. The user will be notified that the job is accepted. If no empty timeslot is found the job will be rejected. Contains a description of how to carry out research. Figure 2. below shows the parameters used by the FCFS-LRH Method. User will submit (jobId, $tesr$, $tlsr$, te, $numCN$). The function of each parameter can be explained as follows:

jobId : job number.
tes : earliest start time the job can executed.
tls : the last start time the job can executed
te : Job execution time
$numCN$: The number of resources needed by the job.
tf : Total time flexibility
$tr1$, $tr2$: left and right side flexibility time.

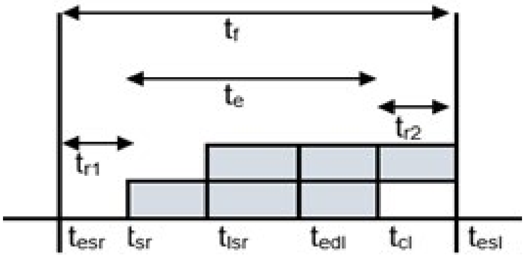
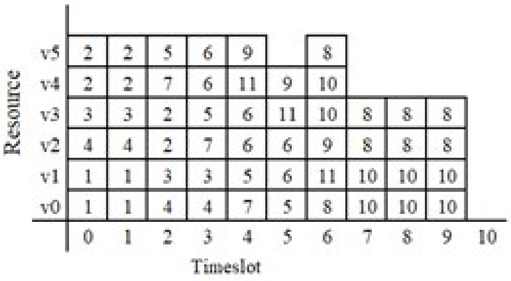


Figure 2. Job scheduling allocation.

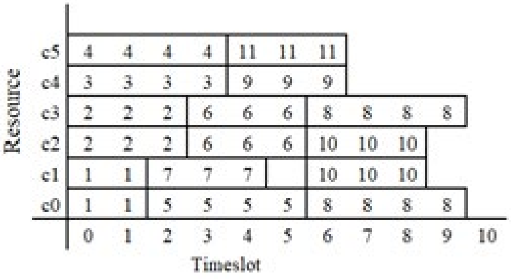
d. Virtual view and Physical View

All jobs sent to the grid will first find a place in the virtual view, whether there is an empty slot or not[25]. If an empty slot found, the job will be placed in the virtual view. The user will be notified that the job will executed. Figure 2 an example of randomly placing 10 jobs placed on 6 virtual view resources. Figure 3 shows job placement in physical view after recombination.


Resource
Timeslot

Resource	0	1	2	3	4	5	6	7	8	9	10
v5	2	2	5	6	9		8				
v4	2	2	7	6	11	9	10				
v3	3	3	2	5	6	11	10	8	8	8	
v2	4	4	2	7	6	6	9	8	8	8	
v1	1	1	3	3	5	6	11	10	10	10	
v0	1	1	4	4	7	5	8	10	10	10	

Figure 3 job scheduling in virtual view


Resource
Timeslot

Resource	0	1	2	3	4	5	6	7	8	9	10
c5	4	4	4	4	11	11	11				
c4	3	3	3	3	9	9	9				
c3	2	2	2	6	6	6	8	8	8	8	
c2	2	2	2	6	6	6	10	10	10		
c1	1	1	7	7	7		10	10	10		
c0	1	1	5	5	5	5	8	8	8	8	

Figure 4 job scheduling in physical view

e. Performance Metrics FCFS-LRH

The resource may be idle despite a reservation request. This occurs when the idle time does not match the allocation policy. RIT is calculated by applying the formula below.

RIT = Finishprevious – startcurrent

When there is a reservation request with a conflict. The following equation calculates the total idle time of the resource.

$$\text{Total RIT}(T_{\text{RIT}}) = \sum_{i=1}^{size} RIT$$

f. Algoritme FCFS-LRH

```
Input: Job (jobId, $tesr$,$lsr$,$te$, $numCN$)
Output: IdleTime
For j=0:numSlot
    sort arrival jobs based on priority $tesr$, $te$, $numCN$
Endfor
For i=0:numSlot // numSlot is the amount of job/ timeslot
    calculate the value of $d2=tesr+te-1$
    Search timeslot free with First fit,Best fit, Worst fit strategy
    IF (timeslot==0) then insert Jobid value
    IF (timeslot!=0) then execution procedure moveSlot().
Endfor
Procedure moveSlot();
Initialization; finish=0,suc=false, start=$tesr$, finish=$tesr+te$-1.
relax=start–$tesr$, $tr=tlsr-tesr$,CNs=0.
while (!suc and relax <=$tr$)
For cek=start:finish
set the variable CNs=0
```

```
16. For s=0:atrans.size()
17. IF atrans.get(s,cek)!=0 then
18. variable CNs increases by 1
19. Endif
20. Endfor
21. calculate the variable sel=maxC-CNs // maxC is the
    number of physical nodes
22. IF (sel>=CN) then
23. calculate the variable t=start, suc=true
24. Else
25. calculate the variable t=cek, finish=start+$te$-1,
    relax=start-$tesr$, suc=true
26. IF (start>=$tlsr$) then continuous to line 4
27. Endif
28. Endfor
29. Endwhile
30. IF (suc==true) then
31. calculate the variable start=t+1, finish=start+$te$-1,
    relax=start-$tesr$
32. insert JobID with the first fit, best fit, worst fit
    strategy
33. calculate IdleTime
34. Endif
```

The explanation of the FCFS-LRH algorithm is as follows: user submits Job (jobId, ,*lsr*,*te*, N). Lines 1-3 show the sorting of jobs by priority. Lines 5-6 look for empty timeslots in the virtual view using First fit,Best fit, Worst fit. If there is an empty timeslot do line 7. If there is no empty timeslot move the job, shown in line 8 and call the moveslot procedure. The function of the moveslot procedure on lines 11-34 is to shift the job if there is an empty timeslot. If the job can be shifted then allocate the job to the timeslot and calculate the idle time.

3. Result

Experimental stages in this study: (1) Setting parameters whose values fixed and changes shown in table 1; (2) Setting the flexibility parameters and usability factors, are shown in table 2; (3) Generating jobs randomly with usability factors 2 and 3, and determining the percentage of flexibility from 25% to 100%, is shown in table 3. The results of the job generation in table 3 used as input to the FCFS-LRH, FCFS-EDS method. (4) The results of data input processing were tested using the FCFS-LRH, FCFS-EDS methods with the first fit, best fit, and worst fit strategies. (5) The first fit, best fit and worst fit strategy with the best value will be used to find the idle time value in the FCFS, FCFS-EDF and FCFS-LRH methods.

The user sends his work to the resource in the form of JobId, execution start time, execution time, execution end time and the number of resource nodes needed. The FCFS-LRH strategy will respond by finding an empty slot in the virtual view. If an empty slot is found, the job will be allocated to the virtual view, and the user will be notified that the job has been accepted. If no vacant slot found, the job will be rejected. Table 4, Figure 5, shows the search time and job allocation using FCFS-LRH and FCFS-EDS with the first fit, best fit, and worst fit strategies. Table 4, figure 5 uses the flexibility of 25% to 100%; Utilization factor =2 and =3 ; the number of jobs is between 300 and 795. The average result of the search time and job placement in the virtual view for the FCFS-LRH method with the first fit algorithm is 146.61; the best fit of 153.21; the worst fit is 150.66. The search and job allocation results using FCFS-EDS with a first fit obtained 181.30. These results show that the average job search and placement time in virtual view first fit is faster than best fit and worst fit. Figure 5 shows that using =2 and =3 the average search time and job allocation in the FCFS-LRH virtual view with first fit is faster than FCFS-EDS with first fit. These results indicate that FCFS-LRH notifications to users are better than FCFS-EDS.

Figure 6 compares idle time between FCFS-LRH with FCFS and FCFS-EDF. FCFS-LRH average idle time is lower than FCFS and FCFS-EDF.

Table 5 shows that with the utilization factor of 2, the idle time value of FCFS-LRH is lower than FCFS and FCFS-EDF. Likewise, for the utilization factor 3, the idle time value of FCFS-LRH is lower than FCFS and FCFS-EDF.

Table 1 Jobs Experiment Parameters

Parameter name	Nilai parameter
Job execution time	constant
Amount of resources required	constant
Flexibility time	Changed
Execution start time	Changed
Execution end time	Changed

Table 2 Parameters of Utilization Factors and Percent Flexibility

Load	μ	Percent flexibility (%)
Small	μ=2	25, 50, 75, 100
Moderate	μ=3	25, 50, 75, 100

Table 3 Generate jobs

Factor utilization(μ)	Flexibility(β) (%)	Number of jobs
2	25	383
2	50	421
2	75	459
2	100	553
3	25	627
3	50	711
3	75	764
3	100	795

Table 4. Comparison of the execution time of first fit, best fit and worst fit

Number of jobs	μ	β(%)	FCFS-LRH first fit	FCFS-LRH best fit	FCFS-LRH worst fit	FCFS-EDS first fit
383	2	25	99.73	102.58	98.95	130.19
421	2	50	116.90	128.21	123.13	151.66
459	2	75	128.08	132.49	132.02	156.65
553	2	100	142.47	148.18	145.94	174.32
627	3	25	152.66	157.3	158.53	187.23
711	3	50	169.38	175.83	174.57	207.79
764	3	75	177.56	186.71	181.49	214.19
795	3	100	186.11	194.36	190.61	228.35
	Average		146,61	153,21	150,66	181.30

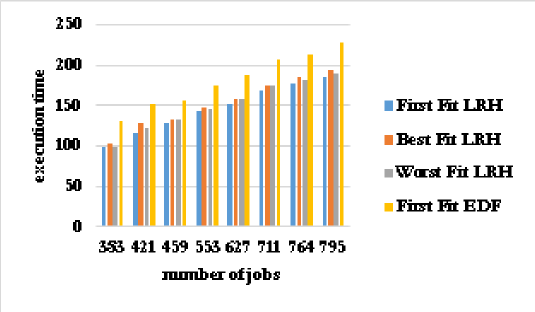


Figure 5 Comparison of Execution Time of First Fit, Best Fit, Worst Fit Based on Number of Jobs

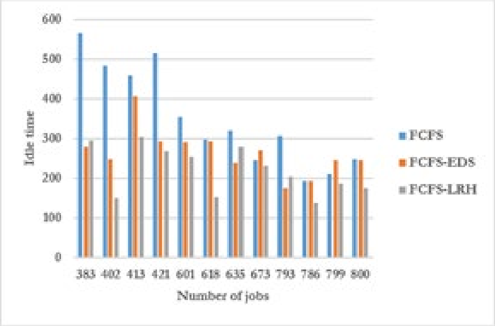


Figure 6 FCFS-LRH idle time comparison with FCFS-EDS and FCFS

Table 5. average idle time based on utilization factor

Method	μ=2	μ=3
FCFS	505,25	303,8
FCFS-EDS	305,75	272,3
FCFS-LRH	254,25	228,8

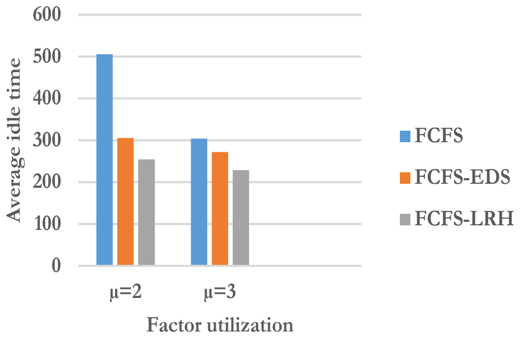


Figure 7. FCFS-LRH average idle time comparison with FCFS-EDS and FCFS

4. Discussion

Reservation of resources in advance ensures the availability of resources when needed, increases the efficient utilization of resources, and reduces the execution time of a process. There are various approaches. There is no guarantee that most of the conventional methods will execute the work because the result is placed in a waiting room. e.g. the FCFS approach. The FCFS-LRH approach proposes that users can be sure that their work will be executed and reduce idle time.

Based on the research results, the use of first fit is better than best fit and worst fit when used in the FCFS-LRH method. The FCFS-LRH method using first fit is faster than FCFS-EDF, which results in faster notifications to users. The experimental results on the FCFS-LRH method using a usability factor of 2 and a flexibility of 25% to 100% resulted in a reduction in the idle time value of FCFS-LRH compared to FCFS of 49.68%. Meanwhile, when compared with FCFS-EDF, the idle time reduction of FCFS-LRH is 16.84%. If you use a benefit factor of 3 and flexibility of 25% to 100%, the result is a reduction in the idle time value of FCFS-LRH compared to FCFS of 24.69%.

Meanwhile, when compared with FCFS-EDF, the idle time reduction of FCFS-LRH is 15.98%. The average idle time reduction of FCFS-LRH compared to FCFS is 40.3%. The average idle time reduction of FCFS-LRH compared to FCFS-EDF is 16.44%. The FCFS-LRH method can reduce the idle time value due to the job scheduling policy by sorting incoming jobs by priority. As well as allocating incoming jobs starting from the left side of the timeslot.

5. Conclusion

From the study results, it can be concluded that the average idle time of FCFS-LRH is lower than FCFS by 24.39% and FCFS-EDF by 16.89%. The FCFS-LRH idle time value is lower because the FCFS-LRH scheduling policy is carried out by sorting incoming jobs by priority. As well as placing jobs starting from the far left of the timeslot. This is not done in the FCFS and FCFS-EDF methods.

Reference

[1] S. Kumari and G. Kumar, “Survey on Job Scheduling Algorithms in Grid Computing,” *Int. J. Comput. Appl.*, vol. 115, no. 15, pp. 17–20, Apr. 2015, doi: 10.5120/20227-2511.

[2] A. Sulistio, K. H. Kim, and R. Buyya, “On incorporating an on-line strip packing algorithm into elastic grid reservation-based systems,” *Proc. Int. Conf. Parallel Distrib. Syst. - ICPADS*, vol. 1, 2007, doi: 10.1109/ICPADS.2007.4447738.

[3] A. Sulistio, U. Cibej, S. K. Prasad, and R. Buyya, “GarQ: An efficient scheduling data structure for advance reservations of grid resources,” *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 24, no. 1, pp. 1–19, 2009, doi: 10.1080/17445760801988979.

[4] A. B.Patel, “Modeling and Simulation of Grid Resource Brokering Algorithms,” *Int. J. Comput. Appl.*, vol. 42, no. 8, pp. 31–36, 2012, doi: 10.5120/5715-7774.

[5] H. B. Prajapati and V. A. Shah, “Scheduling in Grid Computing Environment,” in *2014 Fourth International Conference on Advanced Computing & Communication Technologies*, Feb. 2014, pp. 315–324, doi: https://doi.org/10.1109/ACCT.2014.32.

[6] A. Sulis, “GRID computing approach for multireservoir operating rules with uncertainty,” *Environ. Model. Softw.*, vol. 24, no. 7, pp. 859–864, Jul. 2009, doi: https://doi.org/10.1016/j.envsoft.2008.11.003.

[7] C. Castillo, G. N. Rouskas, and K. Harfoush, “On the design of online scheduling algorithms for advance reservations and QoS in grids,” *Proc. - 21st Int. Parallel Distrib. Process. Symp. IPDPS 2007; Abstr. CD-ROM*, 2007, doi: 10.1109/IPDPS.2007.370226.

[8] I. Foster and C. Kesselman, “The history of the grid,” *Adv. Parallel Comput.*, vol. 20, pp. 3–30, 2011, doi: 10.3233/978-1-60750-803-8-3.

[9] I. Foster, C. Kesselman, and S. Tuecke, “The Anatomy of the Grid,” *Grid Comput.*, pp. 169–197, 2003, doi: 10.1002/0470867167.ch6.

[10] A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, and S. Tuecke, “The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets,” *J. Netw. Comput. Appl.*, vol. 23, no. 3, pp. 187–200, 2000, doi: 10.1006/jnca.2000.0110.

[11] R. Buyya, D. Abramson, and J. Giddy, “Nimrod/G: An architecture for a resource management and scheduling system in a global computational grid,” *Proc. - 4th Int. Conf. High Perform. Comput. Asia-Pacific Reg. HPC-Asia 2000*, vol. 1, pp. 283–289, 2000, doi: 10.1109/HPC.2000.846563.

[12] C. T. Yang, W. C. Shih, and C. H. Hsu, “On utilization of the grid computing technology for video conversion and 3D rendering,” *Comput. Stand. Interfaces*, vol. 32, no. 1–2, pp. 29–37, 2010, doi: 10.1016/j.csi.2009.06.003.

[13] K. Al Tabash, A. Barradah, and R. Al Shaikh, “Empirical Utilization Analysis for High Performance and Grid Computing,” in *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, Mar. 2014, pp. 392–398, doi: https://doi.org/10.1109/UKSim.2014.47.

[14] S. Sahhaf, M. Barshan, W. Tavernier, H. Moens, D. Colle, and M. Pickavet, “Resilient algorithms for advance bandwidth reservation in media production networks,” *Proc. 2016 12th Int. Conf. Des. Reliab. Commun. Networks, DRCN 2016*, no. Drcn, pp. 130–137, 2016, doi: 10.1109/DRCN.2016.7470847.

[15] A. A. Haruna, B. Z. Nordin, and H. Narleeni, “Grid Resource Allocation: A Review,” *Res. J. Inf. Technol.*, vol. 4, no. 2, pp. 38–55, 2012, [Online]. Available: http://www.maxwellsci.com/print/rjit/v4-38-55.pdf.

[16] M. B. Qureshi, M. A. Alqahtani, and N. Min-Allah, “Grid resource allocation for real-time data-intensive tasks,” *IEEE Access*, vol. 5, pp. 22724–22734, 2017, doi: 10.1109/ACCESS.2017.2760801.

[17] Kai Hwang and Zhiwei Xu, “Scalable parallel computers for real-time signal processing,” *IEEE Signal Process. Mag.*, vol. 13, no. 4, pp. 50–66, Jul. 1996, doi: https://doi.org/10.1109/79.526898.

[18] M. P. Tiemeyer and J. S. K. Wong, “A task migration algorithm for heterogeneous distributed computing systems,” *J. Syst. Softw.*, vol. 41, no. 3, pp. 175–188, Jun. 1998, doi: https://doi.

org/10.1016/S0164-1212(97)10018-8.

[19] R. Umar, A. Agarwal, and C. R. Rao, "Advance Planning and Reservation in a Grid System," *Commun. Comput. Inf. Sci.*, vol. 293 PART 1, pp. 161–173, 2012, doi: 10.1007/978-3-642-30507-8_15.

[20] A. Pujiyanta, L. E. Nugroho, and Widyawan, "Resource allocation model for grid computing environment," *Int. J. Adv. Intell. Informatics*, vol. 6, no. 2, pp. 185–196, 2020, doi: https://doi.org/10.26555/ijain.v6i2.496.

[21] A. Iosup, D. H. J. Epema, J. Maassen, and R. Van Nieuwpoort, "Synthetic grid workloads with Ibis, KOALA, and GRENCHMARK," in *Integrated Research in GRID Computing - CoreGRID Integration Workshop 2005, Selected Papers*, 2007, pp. 271–283, doi: 10.1007/978-0-387-47658-2_20.

[22] A. Sulistio, K. H. Kim, and R. Buyya, "Using revenue management to determine pricing of reservations," in *Third IEEE International Conference on e-Science and Grid Computing (e-Science 2007)*, 2007, pp. 396–404, doi: 10.1109/E-SCIENCE.2007.83.

[23] M. Carvalho and F. Brasileiro, "A user-based model of grid computing workloads," in *2012 ACM/IEEE 13th International Conference on Grid Computing*, 2012, pp. 40–48, doi: 10.1109/Grid.2012.13.

[24] A. Pujiyanta, L. E. Nugroho, and Widyawan, "Planning and Scheduling Jobs on Grid Computing," *Proceeding - 2018 Int. Symp. Adv. Intell. Informatics Revolutionize Intell. Informatics Spectr. Humanit. SAIN 2018*, pp. 162–166, 2019, doi: https://doi.org/10.1109/SAIN.2018.8673372.

[25] A. Pujiyanta, L. E. Nugroho, and Widyawan, "Advance Reservation for Parametric Job on Grid Computing," *Proc. 2019 4th Int. Conf. Informatics Comput. ICIC 2019*, pp. 0–4, 2019, doi: https://doi.org/10.1109/ICIC47613.2019.8985978.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Risk Management in Software Development Projects A Systematic Literature Review

Marzuki Pilliang, Munawar*
Computer Science Faculty
Esa Unggul University
Jakarta
*moenawar@gmail.com

Abstract-Risk Management is an integral part of every project. Risk management must estimate the risks' significance, especially in the SDLC process, and mitigate those risks. Since 2016, many papers and journals have researched planning, design, and risk control in software development projects over the last five years. This study aims to find the most exciting topics for researchers in risk management, especially in software engineering projects. This paper takes a systematic approach to reviewing articles containing risk management in software development projects. This study collects papers and journals included in the international online library database, then summarizes them according to the stages of the PICOC methodology. This paper results in the focus of research in the last five years on Agile methods. The current issue is that many researchers are trying to explicitly integrate risk management into the Agile development process by creating a comprehensive risk management framework. This SLR helps future research get a theoretical basis to solve the studied problem. The SLR explains the focuses of previous research, analysis of research results, and the weaknesses of the investigation. For further study, take one of the topic papers, do a critical review, and find research gaps.

Keywords: project, risk management, software development, systematic literature review, SLR

Article info: *submitted January 8, 2022, revised April 25, 2022, accepted May 27, 2022*

1. Introduction

Uncertainty and extreme competition in the information systems industry increase new challenges and problems in today's growing companies. Cost, deadline, and implementation of development methodologies are severe factors in software development project failure [1]. Risk is part of the project, and managing risk leads to success. Most software development companies view risk differently and less comprehensively [2]. This failure is why risk management in software projects has become a significant concern for many companies. Organizations that adopt risk management strategies positively affect the outcomes of their software projects and typically result in reduced costs, fewer delays, and improved performance [3].

Software engineering is a discipline that covers everything related to the software development process, from the design stage to the implementation stage and post-implementation, so that the software life cycle can take place efficiently and measurably [4]. In (Rudy 2016), the definition of a project, according to PMBOK (Project Management Body of Knowledge), is a temporary effort to produce specific/unique products, services, or results [5]. Risk is an integral part of every project, and risk management is an essential part of the decision-making process at every stage of the project. The success or failure of a project is highly dependent on the approach to the potential emergence of risks that can affect the productivity, quality, timeliness, and or cost of the project [6]. Joshua Partogi also says the extra work that causes software development costs to be more expensive is a risk that can eliminate in a fail-safe environment [7].

Risk management in software development projects describes an integrated engineering approach with methods, processes, and artifacts that continuously identify, analyze, control, and pool risks, to reduce the risk of project failure. The risk management process consists of all the activities necessary to identify risks that may potentially impact the software project [8].

The importance of risk management in software development projects encourages researchers to conduct studies in this field to find a novelty for knowledge and the software industry. However, in every research, it is often asked why the field was chosen and whether the lot is outdated or has the potential to find elements of novelty. Who researched the area (risk management in software development projects), and what were the results?

To answer these problems, a systematic literature review was carried out to identify and evaluate the research, with the object of study in the form of papers published in the last five years until September 2021, when this research was conducted. This paper describes the research focus, analysis of research results, and weaknesses of previous studies so that the results of this literature review are used as a theoretical basis for further research.

2. Method

This paper takes an approach systematically to reviewing the literature on risk management in software development projects. The Systematic Literature Review (SLR) method is well established in medical research and deeper in information technology [9]. The SLR used is an approach by Kitchenham and Charters to identify, assess, and interpret findings on a research topic to answer predetermined research questions [10].

a. Research Question

Research questions are obtained from the PICOC (Population, Intervention, Comparison, Outcomes, and Context), which contains the criteria and scope of the papers included in the literature study, as shown in Table 1. The PICOC method is used to build an evidence-based practice by asking well-structured practical questions.

Table 1 Scope of formulating research questions

	Criteria	Scope
P	Population	Risk management and software development projects
I	Intervention	Limited to research on risk management in software development projects
C	Comparison	n/a
O	Outcomes	Risk management in software development projects dominates trends and topics of research concern.
C	Context	A review of all research containing risk management in software development projects

Based on these criteria and scope, five Research Questions (RQ) were generated or shown in Table 2.

Table 2 Research Questions

	Research Question
RQ1	Does the paper discuss risk management?
RQ2	Does the paper discuss software development projects?
RQ3	What is the main focus of the research?
RQ4	What is the result of the research?
RQ5	Who has researched the most in this field?

b. Search strategy

The search strategy was carried out by determining the search string formulation, searching for data sources from the online database literature, defining the inclusion and exclusion criteria as shown in Table 3, and extracting papers based on the RQ in Table 2.

- Search string is English and combines keywords using Boolean ANDs & ORs [9]. This paper is used search strings like the following: risk management AND (software develop* OR project manage*).
- Literature from the most popular Internet is explored to the broadest possible range for study dan research. The following is a list of digital repository indexes:
  - Springer (link.springer.com)
  - Research Gate (researchgate.net)
  - IEEE Xplore (ieeexplore.ieee.org)
  - Elsevier (elsevier.com)
  - ACM Digital Library (dl.acm.org)
- Inclusion criteria as requirements of relevant research, and exclusion is used to exclude studies or research those not pertinent.

Table 3 Inclusion and exclusion criteria

Inclusion	Exclusion
Articles published in English	Articles published not in English.
Articles published between January 1st, 2016, and September 30th, 2021	Articles published before 1st 2016 and outside inclusion period
Articles included in international journals.	Articles included not in international journals.
Fully accessible papers	Fully inaccessible papers
Articles belonging to the risk management category and software development projects	Articles not belonging to the risk management category and software development projects

3. Result

The following steps are carried out in the search for papers:

1. Enter keywords in the search field in each online repository (link.springer.com, researchgate.net, ieeexplore.ieee.org, elsevier.com, dl.acm.org).
2. Limit search years (2016 - 2021).
3. Perform downloads for articles that can be accessed.

After browsing and searching for sources from online repositories, then extracting papers based on inclusion and exclusion criteria, 54 articles were determined. The list of titles, year of publication, and repository sources can be seen in Table 4.

Table 4 List of Search Results

	Year	Title	Source	Seq
[11]	2016	Causes of Human Errors in Early Risk assessment in Software Project Management	dl.acm.org	1
[12]	2018	Open data standards for open source software risk management routine	dl.acm.org	2
[13]	2019	Risk management in projects based on open-source software	dl.acm.org	3
[14]	2019	Risking: A game for teaching risk management in software projects	dl.acm.org	4
[15]	2020	Risk Management for Software Projects in Banking	dl.acm.org	5
[16]	2016	Categorization and standardization of accidental risk-criticality levels of human error to develop risk and safety management policy	elsevier.com	6
[17]	2017	Climate-Agriculture-Modeling and Decision Tool (CAMDT): A software framework for climate risk management in agriculture	elsevier.com	7
[18]	2017	Framework for risk management software system for SMEs in the engineering construction sector	elsevier.com	8
[19]	2017	A risk management framework for distributed agile projects	elsevier.com	9
[20]	2019	Risk management framework for distributed software team: A case study of telecommunication company	elsevier.com	10
[2]	2019	A framework for risk management in Scrum development process	elsevier.com	11
[21]	2020	Project planning and risk management as a success factor for IT projects in agricultural schools in Serbia	elsevier.com	12
[22]	2021	A risk prediction model for software project management based on similarity analysis of context histories	elsevier.com	13
[23]	2016	Expert's opinions on software project effective risk management	ieeexplore.ieee.org	14
[24]	2016	Experimental evaluation of a novel ISO 14971 risk management software for medical devices	ieeexplore.ieee.org	15
[25]	2016	Corporate risk estimation by combining machine learning technique and risk measure	ieeexplore.ieee.org	16
[26]	2017	Quantitative planning and risk management of Agile Software Development	ieeexplore.ieee.org	17
[27]	2017	Decision support system for risk assessment and management strategies in distributed software development	ieeexplore.ieee.org	18
[28]	2018	A critical analysis of software risk management techniques in large scale systems	ieeexplore.ieee.org	18
[29]	2018	A Software System for Risk Management of Information Systems*	ieeexplore.ieee.org	20
[30]	2018	Agile Software Risk Management Architecture for IoT-Fog based systems	ieeexplore.ieee.org	21
[31]	2018	Exploring Experiential Learning Model and Risk Management Process for an Undergraduate Software Architecture Course	ieeexplore.ieee.org	22
[32]	2018	Modeling information security threats for smart grid applications by using software engineering and risk management	ieeexplore.ieee.org	23
[33]	2018	Intelligent Software Platform and End-Point Software for Risk Management	ieeexplore.ieee.org	24
[34]	2019	Risk Management in Agile Software Development: A Survey	ieeexplore.ieee.org	25
[35]	2019	Agile risk management for multi-cloud software development	ieeexplore.ieee.org	26
[36]	2019	Risk Management Technology of Software Project Sustainability in Fuzzy Conditions	ieeexplore.ieee.org	27
[37]	2019	Risk Catalogs in Software Project Management	ieeexplore.ieee.org	28
[38]	2020	Data-driven Risk Management for Requirements Engineering: An Automated Approach based on Bayesian Networks	ieeexplore.ieee.org	29
[39]	2020	Risk Management in Software Engineering Using Big Data	ieeexplore.ieee.org	30
[40]	2021	Artificial Intelligence based Risk Management Framework for Distributed Agile Software Development	ieeexplore.ieee.org	31
[41]	2021	Adapting a Software Acquisition Curriculum to Instruct Supply Chain Risk Management in a Project-Based Software Development Course	ieeexplore.ieee.org	32

	Year	Title	Source	Seq
[42]	2021	Assessing the Risk of Software Development in Agile Methodologies Using Simulation	ieeexplore.ieee.org	33
[43]	2016	Software risk management: Using the automated tools	link.springer.com	34
[44]	2016	A study on software risk management strategies and mapping with SDLC	link.springer.com	35
[45]	2016	Risk Management During Software Development: Results of a Survey in Software Houses from Germany, Austria and Switzerland	link.springer.com	36
[46]	2016	Software Testing in Clinical Risk Management	link.springer.com	37
[47]	2016	Risk Factor Classification GEMIO in the Planning Phase of Logistic Project Management	link.springer.com	38
[48]	2016	Improving Project Risk Management of Cloud CRM Using DANP Approach	link.springer.com	39
[49]	2017	Concept implementation of decision support software for the risk management of complex technical system	link.springer.com	40
[50]	2017	3PR Framework for Software Project Management: People, Process, Product, and Risk	link.springer.com	41
[51]	2018	Agile risk management using software agents	link.springer.com	42
[52]	2018	Risk Management in Software Engineering: What Still Needs to Be Done	link.springer.com	43
[53]	2018	Application of a risk management tool focused on helping to small and medium enterprises implementing the best practices in software development projects	link.springer.com	44
[54]	2018	Risk Analysis and Management of Software V&V Activities in NPPs	link.springer.com	45
[55]	2019	Adaptation of open up in the scrum framework to improve compliance in scope, risk management and delivery times in software development projects	link.springer.com	46
[56]	2019	Towards risk-driven security requirements management in agile software development	link.springer.com	47
[57]	2021	A Scalable and Automated Machine Learning Framework to Support Risk Management	link.springer.com	48
[58]	2021	Requirement-oriented risk management for incremental software development	link.springer.com	49
[59]	2021	Open Chance and Risk Management Process Supported by a Software Tool for Improving Urban Security	link.springer.com	50
[60]	2016	A Multi-Disciplinary Software Suite for Uncertainty Quantification and Risk Management	researchgate.net	51
[61]	2017	Impact of Risk Management on Software Projects in Nigeria Using Linear Programming	researchgate.net	52
[62]	2019	Drinking Water Quality Risk Management. Risk Analysis of Nitrogen Groundwater Contamination Using Analytica Software	researchgate.net	53
[63]	2021	A Review on Some Pertinent Software Security Risk Management Frameworks	researchgate.net	54

A grouping of articles by database source can be seen in Figure 1.

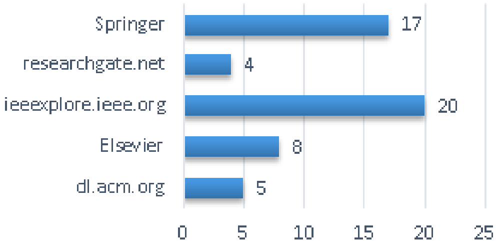


Figure 1 A grouping of articles by database source

After implementing the PICOC methodology for these papers, the results related to the research questions (RQ) presented in the previous section were obtained. Here are the answers to the five questions.

a. Does the paper discuss risk management?

Fifty-four articles can be accessed to carry out a study in the abstract, introduction, and discussion sections. These papers discuss risk management; the following is a mapping based on the year of publication, as shown in Table 5.

Table 5 Mapping based on the year of publication

	2016	2017	2018	2019	2020	2021	Total
Qty	12	8	11	11	5	7	54

b. Does the paper discuss software development projects?

Fifty-four papers or journals discuss risk management, and 20 discuss software development projects. With the help of the Mendeley Reference Manager application, these articles can be categorized into two parts, namely Journal Articles, and Conference Proceedings. Each contribution can be seen in Figure 2.

A total of 20 articles have been studied; each was coded (C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11, C12, and C13) for articles in the Conference Proceedings category. And the code (J1, J2, J3, J4, J5, J6, and J7) for articles in the Journal Article category, the bibliographic details can be seen in Table 7.

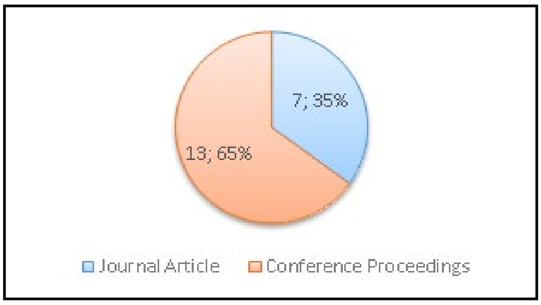


Figure 2 Articles of software development projects

c. What is the main focus of the research?

Table 6 shows the mapping of the primary research focuses from articles published between January 2016 to September 2021.

A total of 12 articles (C3, C5, C6, C10, C11, C13, J1, J2, J3, J4, J6, and J7) focused on Agile as the object of research. Agile methodology is an alternative to traditional linear sequential software development processes such as Waterfall. The term "Agile" in software development methodologies comes from the "Agile Manifesto," compiled in 2001. Among the Agile methods are Extreme programming, Test-driven development, Feature-driven development, and Scrum. In recent years, the software industry has shifted to adopting Agile practices that are responsive and flexible to change instead of traditional methods [64].

Three articles (C8, J5, and J6) specifically examine risk management in the software development process with Scrum. Scrum is based on empiricism and lean thinking. Empiricism asserts that knowledge comes from experience and makes decisions based on observation. Lean thinking reduces waste and focuses on what matters. Scrum uses an iterative and incremental approach to optimize predictability and control risk [65].

The Risk Management Tool includes the second most popular research focus. A total of 8 articles (C4, C5, C6, C13, J2, J3, J5, and J6) were reviewed, and even some articles carried the tools or applications used in risk management. The framework for risk management includes a widely discussed research focus, including eight articles (C5, C7, C8, C9, C11, J1, J4, and J7).

The risks involved in deployment have not gone unnoticed by researchers in the last five years. Five articles (C1, C3, C7, C10, and C12) focused on this issue. Each piece (C3 and C10) focuses on Quality Risks, and the last report (C2) focuses on People Risks. Visually, the trend of research focus is shown in Figure 3.

Table 6 Mapping of research focus

Method	Risk Focus				
	Deployment	People	Tools	Frame-work	Quality
Agile	C3 C10		C5 C6 C13 J2 J3 J6	C5 C11 J1 J4 J7	C3 C10
Scrum			J5 J6	C8	
Traditional	C1 C7 C12	C2	C4 C6	C7 C9	

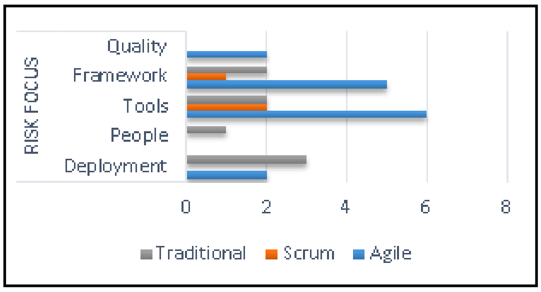


Figure 3 The trend of research focus

Table 7 Bibliographic information

Code	Author	Title	Year	Research Result
C1	B. Roy, R. Dasgupta, and N. Chaki	A study on software risk management strategies and mapping with SDLC	2016	Risk classification by SDLC phase
C2	S. Sharma and B. Ram	Causes of human errors in early risk assessment in software project management	2016	Strauss and Glasser's theoretical approach to human error
C3	K. Ghane	Quantitative planning and risk management of agile software development	2017	The concept for calculating risk value

Code	Author	Title	Year	Research Result
C4	A. Boranbayev, S. Boranbayev, A. Nurusheva, K. Yersakhanov, and Y. Seitkulov	A Software System for Risk Management of Information Systems	2018	Software to improve reliability and fault tolerance
C5	P. Gouthaman and S. Sankaranarayanan	Agile software risk management architecture for IoT-fog based systems	2018	Risk assessment software and framework to assist in identification and planning
C6	Y. M. García, M. Muñoz, J. Mejía, G. P. Gasca, and A. Mireles	Application of a risk management tool focused on helping to small and medium enterprises implementing the best practices in software development projects	2018	Case studies of the application of tools in risk management
C7	T. Hussain	Risk management in software engineering: What still needs to be done	2018	A framework that categorizes risks based on the relative importance and level of control of the project manager
C8	S. Chaouch, A. Mejri, and S. A. Ghannouchi	A framework for risk management in Scrum development process	2019	A framework involved in the deployment of the risk management process in Scrum
C9	W. S. Wan Husin, Y. Yahya, N. F. Mohd Azmi, N. N. Amir Sjarif, S. Chuprat, and A. Azmi	Risk management framework for distributed software team: A case study of telecommunication company	2019	Communication elements within the framework on DSD risk categories
C10	M. Hammad, I. Inayat, and M. Zahid	Risk management in agile software development: A survey	2019	Mitigation strategies used to minimize the impact of risk in the risk management process
C11	D. Ionita, C. van der Velden, H. J. K. Ikkink, E. Neven, M. Daneva, and M. Kuipers	Towards risk-driven security requirements management in agile software development	2019	A framework that can help Agile development teams consider security a priority in software risk
C12	C. M. Tae, P. D. Hung, and L. D. Huynh	Risk Management for Software Projects in Banking	2020	Analyze size, accuracy, time, cost, effort, knowledge, and experience to avoid the risk
C13	A. Puri and S. Sharma	Risk Management in Software Engineering Using Big Data	2020	Big data predictive analytics to make risk predictions in software projects
J1	S. V. Shrivastava and U. Rathod	A risk management framework for distributed agile projects	2017	A framework that categorizes risks in DAD (Distributed Agile Development) projects
J2	A. Aslam et al.	Decision Support System for Risk Assessment and Management Strategies in Distributed Software Development	2017	Tools that serve to make decisions for risk management in the software development process
J3	E. E. Odzaly, D. Greer, and D. Stewart	Agile risk management using software agents	2018	Tools used to support risk identification, assessment, and monitoring.
J4	V. Muntés-Mulero et al.	Agile risk management for multi-cloud software development	2019	The framework is generated from combining the previous risks that are used to mitigate the following risks
J5	A. S. Filippetto, R. Lima, and J. L. V. Barbosa	A risk prediction model for software project management based on similarity analysis of context histories	2021	Atropos model for measuring uncertainty in projects
J6	M. I. Lunesu, R. Tonelli, L. Marchesi, and M. Marchesi	Assessing the risk of software development in agile methodologies using simulation	2021	Model several key risk factors using the Agile development simulator
J7	M. Roy, N. Deb, A. Cortesi, R. Chaki, and N. Chaki	Requirement-oriented risk management for incremental software development	2021	A risk management framework for the ISD (Incremental Software Development) process that provides risk exposure estimates for projects

d. What is the result of the research?

The article (C1) produces a risk classification based on the phases in the SDLC (Systems Development Life Cycle). This allows researchers to apply various conceptual models or risk management frameworks and then analyze the occurrence of risk across all steps of the SDLC so that risk mitigation can be inventoried as quickly as possible [44]. However, the results of this study have not brought up a framework that can be integrated into all phases of the SDLC.

Article (C2) takes the theoretical approach of Strauss and Glaser to detect human errors in information security that can pose risks to the software development process [11]. However, this is only partial mitigation for the overall chances of a project.

In the article (C3), the researcher proposes a concept to calculate the risk value in the software development process using the Agile method based on input parameters with the desired target value limits and the appropriate level of confidence [26]. However, the concept being carried out does not cover the planning and mitigation strategies that may occur at a value that has exceeded the target limit.

The article (C4) describes the software that has been developed to manage risk in the information system process. It enables developers to identify, evaluate, and neutralize information and other automated systems risks. In addition, the developed system has several other advantages, such as the ability to identify risks at an early stage of development, the convenient interface, and time-saving [29]. However, the addition of tools in the software development process can increase the workload of developers, considering that agencies that are only used half-heartedly will create invalid output results.

Article (C5) proposes an architecture and risk assessment framework system to identify and plan risk management in the software development process using the Agile method. Especially software development in IoT, Fog, and Cloud-based systems [30]. However, the proposed framework does not include an analysis of risk parameters, so further research is needed to create a more effective framework.

The article (C6) presents the results of a case study of the application of tools in essential risk management in two companies. The results of the hypothesis indicate that the use of these tools is helpful for implementation in software engineering projects [53]. However, these results cannot be generalized to other projects in the software development process. Because environmental and ethnic, or cultural factors can affect the results obtained.

The article (C7) examines the risk management process: risk planning; risk identification; risk analysis; risk response; and monitoring and control. This article presents a framework that categorizes risks based on their relative importance and perceived level of control over the project manager. The framework is classified into four quadrants: customer mandate, scope and requirements; execution; and the environment [52]. However, risk analysis becomes very difficult or impractical for large projects because the research only focuses on quantitative aspects and ignores qualitative elements. This makes the framework that is carried out ineffective and can be applied to large projects.

Article (C8) proposes a model of the activities involved in spreading the risk management process in the Scrum framework. The model emerged based on the respondent's questionnaire. The aim is to improve the methodology that maps the risk management principles to increase project success [2]. However, the results need to be verified by testing them in software development scenarios in various Scrum organizations, both on medium and large-scale projects. This risk management framework also needs to be further developed for other Agile methodologies such as Extreme Programming (XP), Dynamic System Development Method, Kanban, and Feature Driven Development (FDD).

The article (C9) adds a communication element to the DSD (Distributed Software Development) risk category. Communication will help grow team members to become aware of the risk, facilitate everyone responsible for managing risk, and understand the basis for decisions made and the reasons behind specific treatments or actions chosen [20]. However, this addition does not cover the categories of other risks.

The article (C10) presents the results of a survey conducted on industry practitioners of software developers using the Agile method. The survey contains the mitigation strategies to minimize risk in the risk management process to various software development life cycle stages. According to an industry survey, scheduled risk and varying requirements are the most experienced by practitioners. Most of the risk mitigation strategies followed involve using tools to communicate with clients, tracking requirements and change requests implemented in the project, and reducing the number of software bugs [34]. The author considers these parameters can be used as material to create a framework for risk management for future research to get more optimal results.

Article (C11) presents a framework to help Agile development teams consider security priority in software risk. The framework was developed and tested on a single software developer in the Netherlands and only applied to mobile and web applications [56]. Therefore, this framework only covers one aspect of risk management, so it is necessary to develop a framework for other elements.

The article (C12) analyzes the size, accuracy, time, cost, effort, knowledge, and experience to avoid or overcome many risks in information system project management at the Bank [15]. However, it does not discuss planning and risk mitigation strategies. In addition, the scope of research is only in the banking sector, so it cannot be confirmed for other industrial sectors.

The article (C13) raised issues in risk management in software engineering using big data. Predictive big data analysis is used to predict risks experienced before in software projects and provide proposals for possible risks that will arise accordingly [39]. However, checking unstructured data will be inconvenient and requires special skills to avoid invalid analysis results.

Article (J1) proposes a framework for risk categories, 'Group Awareness', 'External Stakeholder Collaboration', and 'Software Development Lifecycle' on a DAD (Distributed Agile Development) project. However, the DAD team needs to adopt practices to reduce the impact of spatial distance between stakeholders. Apart from geographic dispersion, other properties, including work culture, enormous project scope, temporal distance, and language barriers, which impact the DAD project, should also be considered to control risk [19]. Therefore, further research is needed to improve this framework.

The article (J2) proposes tools that can help decision-makers during DSD (Distributed Software Development) risk management [27]. However, these proposed tools have not linked the various planning stages with identifying variations in DSS (Decision Support Systems) outputs at different project stages. It is still necessary to add features in distributed development to the risk assessment results.

An article (J3) describes the underlying risk management model in Agile risk tools where software agents support risk identification, assessment, and monitoring. Interaction between agents, agent compliance with defined rules, and how agents react to project environmental data changes. The results show that agents help detect risks and respond dynamically to changes in the project environment, thereby helping to minimize human effort in managing risks in software development projects with Agile methods [51]. However, tools that are not perfect can increase the development team's workload because they can produce inappropriate analysis results.

Article (J4) combines the information gathered from the joint work in the previous process to become a framework used to mitigate the risks that will arise in Agile software development projects [35]. However, the risks that have not appeared before have not been thoroughly analyzed, so there is still a need to improve this framework.

The article (J5) uses the Atropos model to measure the uncertainty in the project with a value that is close to the actual financial impact of the identified risks. Implementation of risk recommendations based on historical similarity analysis of the context by providing advice and considering the characteristics of each new project [22]. However, additional prototypes are needed to compile a complete project history, thus allowing more information to be generated to support more significant risk recommendations and improve analysis of similarity and accuracy of risk recommendations.

The article (J6) introduces a new approach to modeling several key risk factors: project duration, the number of problems applied, and key statistics of problem-solving time. Using an Agile development simulator, this approach includes modeling Agile processes, collecting data from tools used for project management, and performing Monte Carlo process simulations to gain insight into the time and effort expected to complete a project and its distribution. The model parameters that can pose a risk are the error in the estimated effort to be developed, variations in developer assignments for these features, and obstacles related to developer availability and work completion [42]. However, this model still needs improvement by conducting more evaluations on case studies. And scale the model from one team to multiple teams involved in one or more projects.

Article (J7) proposes a risk management framework for the ISD (Incremental Software Development) process that estimates risk exposure for a project. The framework offers appropriate risk reduction strategies and works with the risk assessment module [58]. However, this proposed framework does not yet link the various planning stages with identifying risks at different project stages. It is still necessary to add features in distributed development to the risk assessment results.

e. Who has researched the most in this field?

Researchers who have contributed the most to research on risk management in software development projects can be seen in Table 8.

Table 8 List of Authors

Full Name	Paper	Risk Management	Development Method	Score
Aakash Puri	C13	Tools	Agile	1
Abdulaziz S. Almazyad	J2	Tools	Agile	1
Abid Khan	J2	Tools	Agile	1
Adeel Anjum	J2	Tools	Agile	1
Adeel Aslam	J2	Tools	Agile	1
Agostino Cortesi	J7	Framework	Agile	1
Alexsandro Souza Filippetto	J5	Tools	Scrum	1
Amjad Rehman	J2	Tools	Agile	1
Antonia Mireles	C6	Tools and Framework	Traditional	1
Askar Boranbayev	C4	Tools	Traditional	1
Asma Mejri	C8	Framework	Scrum	1
Assel Nurusheva	C4	Tools	Traditional	1
Azri Azmi	C9	Framework	Traditional	1
Babu Ram	C2	Human error	Traditional	1
Balázs Somosköi	J4	Framework	Agile	1
Bibhash Roy	C1	Deployment	Traditional	1
Chung Min Tae	C12	Deployment	Traditional	1
Coco van der Velden	C11	Framework	Agile	1
Dan Ionita	C11	Framework	Agile	1

Full Name	Paper	Risk Management	Development Method	Score
Darryl Stewart	J3	Tools	Agile	1
Des Greer	J3	Tools	Agile	1
Edzreena Edza Odzaly	J3	Tools	Agile	1
Eelko Neven	C11	Framework	Agile	1
Eric Willeke	J4	Framework	Agile	1
Gloria Piedad Gasca	C6	Tools and Framework	Traditional	1
Henk Jan Klein Ikkink	C11	Framework	Agile	1
Irum Inayat	C10	Deployment and Quality	Agile	1
Jacek Dominiak	J4	Framework	Agile	1
Jezreel Mejía	C6	Tools and Framework	Traditional	1
Jorge Luis Victória Barbosa	J5	Tools	Scrum	1
Kamran Ghane	C3	Deployment and Quality	Agile	1
Kuanysh Yersakhanov	C4	Tools	Traditional	1
Le Dinh Huynh	C12	Deployment	Traditional	1
Lodovica Marchesi	J6	Tools	Scrum	1
Mandira Roy	J7	Framework	Agile	1
Maria Ilaria Lunesu	J6	Tools	Scrum	1
Maryam Zahid	C10	Deployment and Quality	Agile	1
Maya Daneva	C11	Framework	Agile	1
Michael Kuipers	C11	Framework	Agile	1
Michele Marchesi	J6	Tools	Scrum	1
Mirna Muñoz	C6	Tools and Framework	Traditional	1
Muhammad Hammad	C10	Deployment and Quality	Agile	1
Nabendu Chaki	C1, J7	Deployment, Framework	Traditional, Agile	2
Naveed Ahmad	J2	Tools	Agile	1
Nilam Nur Amir Sjarif	C9	Framework	Traditional	1
Novarun Deb	J7	Framework	Agile	1
Nurulhuda Firdaus Mohd Azmi	C9	Framework	Traditional	1
Oscar Ripolles	J4	Framework	Agile	1
P. Gouthaman	C5	Tools and Framework	Agile	1
Peter Matthews	J4	Framework	Agile	1
Phan Duy Hung	C12	Deployment	Traditional	1
Ranjan Dasgupta	C1	Deployment	Traditional	1
Rituparna Chaki	J7	Framework	Agile	1
Roberto Tonelli	J6	Tools	Scrum	1
Robson Lima	J5	Tools	Scrum	1
Seema Sharma	C2	Human error	Traditional	1
Seilkhan Boranbayev	C4	Tools	Traditional	1
Shilpi Sharma	C13	Tools	Agile	1
Smrati Gupta	J4	Framework	Agile	1

Full Name	Paper	Risk Management	Development Method	Score
Sonia Ayachi Ghannouchi	C8	Framework	Scrum	1
Suprika Vasudeva Shrivastava	J1	Framework	Agile	1
Suresh Sankaranarayanan	C5	Tools and Framework	Agile	1
Suriayati Chuprat	C9	Framework	Traditional	1
Syrine Chaouch	C8	Framework	Scrum	1
Tanzila Saba	J2	Tools	Agile	1
Tauqeer Hussain	C7	Deployment and Framework	Traditional	1
Urvashi Rathod	J1	Framework	Agile	1
Victor Muntés-Mulero	J4	Framework	Agile	1
Wan Suzila Wan Husin	C9	Framework	Traditional	1
Yazriwati Yahya	C9	Framework	Traditional	1
Yerzhan Seitkulov	C4	Tools	Traditional	1
Yolanda Meredith García	C6	Tools and Framework	Traditional	1

Only Nabendu Chaki has published two articles (C1 and J7), focusing on research on deployment and quality of risk management in software development projects.

4. Discussion

From January 2016 to September 2021, we see trends in risk management research in software development projects focusing on Agile development methods and risk management tools. Fourteen articles (C3, C4, C5, C7, C8, C9, C11, J1, J2, J3, J4, J5, J6, and J7) carried new frameworks, conceptual models, and software tools.

The current issue, many researchers are trying to integrate risk management explicitly in every process of the software development life cycle in the Agile methodology by creating a comprehensive risk management framework, considering that the Agile and Scrum methods do not have a specific process for risk management [66]. So there is a need to integrate risk management into it explicitly.

Several SLRs have been carried out by other researchers, with the following results:

[67] identified challenges in the context of Global Software Development (GSD) with Software Project Management (SPM) activities that include an integrative framework. The difference with this paper is that the researcher did not examine research on tools in risk management and only examined 15 articles.

[68] research on applying risk mitigation techniques in Agile GSD to increase time efficiency, acquire more resources, lower costs, and maintain a competitive advantage. However, it only focuses on 53 papers discussing Agile methods.

[69] looked for potential studies in 45 articles discussing the identification process in risk management and activation of risk management using the ISO 31000 standard, which differs from this paper which does not compare the use of the ISO standard.

5. Conclusion

There are seven articles (C5, C7, C8, C11, J1, J4, and J7) that are very interesting to be explored further. The framework proposed by these articles deserves to be tested in-depth.

This SLR will be used for further research to strengthen the theoretical basis, compare research results, and describe the shortcomings of previous research.

This literature research is not perfect because the authors have difficulty accessing several international journals fully. Other researchers can use different methods like SPIDER (sample, phenomenon of interest, design, evaluation, research type) to get more significant results.

References

[1] S. L. Fahrenkrog, D. Bolles, J. D. Blaine, and C. Steuer, "PMBOK®guide: an overview of the changes," *Project Management Institute, Newtown Square, US*, 2004.

[2] S. Chaouch, A. Mejri, and S. A. Ghannouchi, "A framework for risk management in Scrum development process," in *Procedia Computer Science*, 2019, vol. 164, pp. 187–192. doi: 10.1016/j.procs.2019.12.171.

[3] L. Sarigiannidis, P. D. Chatzoglou, and others, "Software development project risk management: A new conceptual framework," *Journal of Software Engineering and Applications*, vol. 4, no. 05, p. 293, 2011.

[4] S. Rizky and others, “Konsep dasar rekayasa perangkat lunak,” *Jakarta: Prestasi Pustaka*, 2011.

[5] T. Rudy, “Manajemen Proyek Sistem Informasi, bagaimana mengolah proyek sistem informasi secara efektif & efisien,” *Andi Offset: Yogyakarta*, 2016.

[6] D. Crnković and M. Vukomanović, “Comparison of Trends in Risk Management Theory and Practices Within the Construction Industry,” *Elektronički časopis građevinskog fakulteta Osijek*, no. December 2016, pp. 1–11, 2016, doi: 10.13167/2016.13.1.

[7] J. Partogi, “Manajemen Modern dengan Scrum,” *Yogyakarta: Penerbit Andi*, 2015.

[8] B. Verma, M. Dhanda, B. Verma, and M. Dhanda, “A review on risk management in software projects,” *International Journal*, vol. 2, pp. 499–503, 2016.

[9] Romi Satria Wahono, “A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks,” *Andi Offset*, vol. 1, no. 1, pp. 1–16, 2015, [Online]. Available: https://www.researchgate.net/publication/275945834_A_Systematic_Literature_Review_of_Software_Defect_Prediction_Research_Trends_Datasets_Methods_and_Frameworks

[10] B. Kitchenham and S. Charters, “Guidelines for performing systematic literature reviews in software engineering,” 2007.

[11] S. Sharma and B. Ram, “Causes of human errors in early risk assesment in software project management,” in *ACM International Conference Proceeding Series*, 2016, vol. 04-05-Marc, pp. 1–11. doi: 10.1145/2905055.2905069.

[12] R. Gandhi, M. Germonprez, and G. J. P. Link, “Open Data Standards for Open Source Software Risk Management Routines,” in *Proceedings of the 2018 ACM Conference on Supporting Groupwork*, Jan. 2018, pp. 219–229. doi: 10.1145/3148330.3148333.

[13] N. D. Linh, P. D. Hung, V. T. Diep, and T. D. Tung, “Risk Management in Projects Based on Open-Source Software,” in *Proceedings of the 2019 8th International Conference on Software and Computer Applications*, Feb. 2019, vol. Part F1479, pp. 178–183. doi: 10.1145/3316615.3316648.

[14] S. Santos, F. Carvalho, Y. Costa, D. Viana, and L. Rivero, “Risking: A game for teaching risk management in software projects,” in *Proceedings of the XVIII Brazilian Symposium on Software Quality*, Oct. 2019, pp. 188–197. doi: 10.1145/3364641.3364662.

[15] C. M. Tae, P. D. Hung, and L. D. Huynh, “Risk Management for Software Projects in Banking,” in *PervasiveHealth: Pervasive Computing Technologies for Healthcare*, Feb. 2020, pp. 65–69. doi: 10.1145/3387263.3387268.

[16] P. Kumar, S. Gupta, M. Agarwal, and U. Singh, “Categorization and standardization of accidental risk-criticality levels of human error to develop risk and safety management policy,” *Safety Science*, vol. 85, pp. 88–98, Jun. 2016, doi: 10.1016/j.ssci.2016.01.007.

[17] E. Han, A. V. M. Ines, and W. E. Baethgen, “Climate-Agriculture-Modeling and Decision Tool (CAMDT): A software framework for climate risk management in agriculture,” *Environmental Modelling & Software*, vol. 95, pp. 102–114, Sep. 2017, doi: 10.1016/j.envsoft.2017.06.024.

[18] C. F. Oduoza, O. Odimabo, and A. Tamparapoulos, “Framework for Risk Management Software System for SMEs in the Engineering Construction Sector,” *Procedia Manufacturing*, vol. 11, no. June, pp. 1231–1238, 2017, doi: 10.1016/j.promfg.2017.07.249.

[19] S. V. Shrivastava and U. Rathod, “A risk management framework for distributed agile projects,” *Information and Software Technology*, vol. 85, pp. 1–15, 2017, doi: 10.1016/j.infsof.2016.12.005.

[20] W. S. Wan Husin, Y. Yahya, N. F. Mohd Azmi, N. N. Amir Sjarif, S. Chuprat, and A. Azmi, “Risk management framework for distributed software team: A case study of telecommunication company,” in *Procedia Computer Science*, 2019, vol. 161, pp. 178–186. doi: 10.1016/j.procs.2019.11.113.

[21] V. Vujović *et al.*, “Project planning and risk management as a success factor for IT projects in agricultural schools in Serbia,” *Technology in Society*, vol. 63, no. August, p. 101371, Nov. 2020, doi: 10.1016/j.techsoc.2020.101371.

[22] A. S. Filippetto, R. Lima, and J. L. V. Barbosa, “A risk prediction model for software project management based on similarity analysis of context histories,” *Information and Software Technology*, vol. 131, Mar. 2021, doi: 10.1016/j.infsof.2020.106497.

[23] U. I. Janjua, J. Jaafar, and F. W. Lai, “Expert’s opinions on software project effective risk management,” in *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, Aug. 2016, pp. 471–476. doi: 10.1109/ICCOINS.2016.7783261.

[24] T. Lueddemann, S. Sahin, J. Pfeiffer, and T. C. Lueth, “Experimental evaluation of a novel ISO 14971 risk management software for medical devices,” in *2016 IEEE/SICE International*

Symposium on System Integration (SII), Dec. 2016, pp. 162–167. doi: 10.1109/SII.2016.7843992.

[25] Y. Hsu, M.-F. Hsu, and S.-J. Lin, "Corporate risk estimation by combining machine learning technique and risk measure," in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Jun. 2016, pp. 1–4. doi: 10.1109/ICIS.2016.7550763.

[26] K. Ghane, "Quantitative planning and risk management of agile software development," in *2017 IEEE Technology and Engineering Management Society Conference, TEMSCON 2017*, Jun. 2017, pp. 109–112. doi: 10.1109/TEMSCON.2017.7998362.

[27] A. Aslam *et al.*, "Decision Support System for Risk Assessment and Management Strategies in Distributed Software Development," *IEEE Access*, vol. 5, pp. 20349–20373, Oct. 2017, doi: 10.1109/ACCESS.2017.2757605.

[28] M. Pasha, G. Qaiser, and U. Pasha, "A Critical Analysis of Software Risk Management Techniques in Large Scale Systems," *IEEE Access*, vol. 6, no. c, pp. 12412–12424, 2018, doi: 10.1109/ACCESS.2018.2805862.

[29] A. Boranbayev, S. Boranbayev, A. Nurusheva, K. Yersakhanov, and Y. Seitkulov, "A Software System for Risk Management of Information Systems∗," in *IEEE 12th International Conference on Application of Information and Communication Technologies, AICT 2018 - Proceedings*, Oct. 2018, pp. 1–6. doi: 10.1109/ICAICT.2018.8747045.

[30] P. Gouthaman and S. Sankaranarayanan, "Agile software risk management architecture for IoT-fog based systems," in *Proceedings of the International Conference on Smart Systems and Inventive Technology, ICSSIT 2018*, Dec. 2018, pp. 48–51. doi: 10.1109/ICSSIT.2018.8748457.

[31] O. E. Lieh and Y. Irawan, "Exploring Experiential Learning Model and Risk Management Process for an Undergraduate Software Architecture Course," in *2018 IEEE Frontiers in Education Conference (FIE)*, Oct. 2018, vol. 2018-Octob, pp. 1–9. doi: 10.1109/FIE.2018.8659200.

[32] Y.-T. Chen, "Modeling Information Security Threats for Smart Grid Applications by Using Software Engineering and Risk Management," in *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, Aug. 2018, pp. 128–132. doi: 10.1109/SEGE.2018.8499431.

[33] A. Senkov, "Intelligent Software Platform and End-Point Software for Risk Management," in *2018 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, Oct. 2018, no. 16, pp. 1–5. doi: 10.1109/FarEastCon.2018.8602702.

[34] M. Hammad, I. Inayat, and M. Zahid, "Risk management in agile software development: A survey," in *Proceedings - 2019 International Conference on Frontiers of Information Technology, FIT 2019*, Dec. 2019, pp. 162–166. doi: 10.1109/FIT47737.2019.00039.

[35] V. Muntés-Mulero *et al.*, "Agile risk management for multi-cloud software development," *IET Software*, vol. 13, no. 3, pp. 172–181, Jun. 2019, doi: 10.1049/iet-sen.2018.5295.

[36] V. G. Psoyants, A. I. Taganov, A. N. Kolesenkov, and I. v. Bodrova, "Risk Management Technology of Software Project Sustainability in Fuzzy Conditions," in *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, Jun. 2019, no. June, pp. 1–4. doi: 10.1109/MECO.2019.8760176.

[37] V. Machado, P. Afonso, and H. Costa, "Risk Catalogs in Software Project Management," in *2019 XLV Latin American Computing Conference (CLEI)*, Sep. 2019, vol. 2019-Janua, pp. 1–10. doi: 10.1109/CLEI47609.2019.9089044.

[38] F. Wiesweg, A. Vogelsang, and D. Mendez, "Data-driven Risk Management for Requirements Engineering: An Automated Approach based on Bayesian Networks," *Proceedings of the IEEE International Conference on Requirements Engineering*, vol. 2020-Augus, pp. 125–135, 2020, doi: 10.1109/RE48521.2020.00024.

[39] A. Puri and S. Sharma, "Risk Management in Software Engineering Using Big Data," in *Proceedings of International Conference on Intelligent Engineering and Management, ICIEM 2020*, Jun. 2020, pp. 63–68. doi: 10.1109/ICIEM48762.2020.9160170.

[40] E. Khanna, R. Popli, and N. Chauhan, "Artificial Intelligence based Risk Management Framework for Distributed Agile Software Development," in *2021 8th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2021, pp. 657–660.

[41] B. Tenbergen and N. R. Mead, "Adapting a Software Acquisition Curriculum to Instruct Supply Chain Risk Management in a Project-Based Software Development Course," in *2021 Third International Workshop on Software Engineering Education for the Next Generation (SEENG)*, 2021, pp. 36–40.

[42] M. I. Lunesu, R. Tonelli, L. Marchesi, and M. Marchesi, "Assessing the Risk of Software Development in Agile Methodologies Using Simulation," *IEEE Access*, vol. 9, pp. 134240–134258, 2021, doi: 10.1109/ACCESS.2021.3115941.

[43] S. M. Avdoshin and E. Y. Pesotskaya, "Software

Risk Management: Using the Automated Tools," in *CEUR Workshop Proceedings*, vol. 963, 2016, pp. 85–97. doi: 10.1007/978-3-319-23929-3_8.

[44] B. Roy, R. Dasgupta, and N. Chaki, "A Study on Software Risk Management Strategies and Mapping with SDLC," in *Advances in Intelligent Systems and Computing*, vol. 396, Springer Verlag, 2016, pp. 121–138. doi: 10.1007/978-81-322-2653-6_9.

[45] M. Felderer, F. Auer, and J. Bergsmann, "Risk Management During Software Development: Results of a Survey in Software Houses from Germany, Austria and Switzerland," vol. 10224, J. Großmann, M. Felderer, and F. Seehusen, Eds. Cham: Springer International Publishing, 2017, pp. 143–155. doi: 10.1007/978-3-319-57858-3_11.

[46] A. Stavert-Dobson, "Software Testing in Clinical Risk Management," 2016, pp. 233–247. doi: 10.1007/978-3-319-26612-1_16.

[47] D. Książkiewicz, "Risk Factor Classification GEMIO in the Planning Phase of Logistic Project Management," M. Bąk, Ed. Cham: Springer International Publishing, 2016, pp. 211–219. doi: 10.1007/978-3-319-26848-4_19.

[48] Y.-S. Chen, C.-K. Lin, and H.-M. Chuang, "Improving Project Risk Management of Cloud CRM Using DANP Approach," in *Lecture Notes in Electrical Engineering*, vol. 375, 2016, pp. 1023–1031. doi: 10.1007/978-981-10-0539-8_100.

[49] V. Boyko, N. Rudnichenko, S. Kramskoy, Y. Hrechukha, and N. Shibaeva, "Concept Implementation of Decision Support Software for the Risk Management of Complex Technical System," in *Advances in Intelligent Systems and Computing*, vol. 512, 2017, pp. 255–269. doi: 10.1007/978-3-319-45991-2_17.

[50] K. A. Demir, "3PR Framework for Software Project Management: People, Process, Product, and Risk," 2017, pp. 143–170. doi: 10.1007/978-3-319-54325-3_7.

[51] E. E. Odzaly, D. Greer, and D. Stewart, "Agile risk management using software agents," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 3, pp. 823–841, Jun. 2018, doi: 10.1007/s12652-017-0488-2.

[52] T. Hussain, "Risk management in software engineering: What still needs to be done," in *Advances in Intelligent Systems and Computing*, 2019, vol. 857, pp. 515–526. doi: 10.1007/978-3-030-01177-2_37.

[53] Y. M. García, M. Muñoz, J. Mejía, G. P. Gasca, and A. Mireles, "Application of a risk management tool focused on helping to small and medium enterprises implementing the best practices in software development projects," in *Advances iGarcía, Y. M., Muñoz, M., Mejía, J., Gasca, G. P., & Mireles, A. (2018). Application of a risk management tool focused on helping to small and medium enterprises implementing the best practices in software development projects. Advances in Intel*, 2018, vol. 746, pp. 429–440. doi: 10.1007/978-3-319-77712-2_41.

[54] P.-F. Gu, J.-Z. Tang, W.-H. Chen, and others, "Risk Analysis and Management of Software V&V Activities in NPPs," in *International Symposium on Software Reliability, Industrial Safety, Cyber Security and Physical Protection for Nuclear Power Plant*, 2018, pp. 123–128.

[55] O. L. Loaiza and J. M. de León, "Adaptation of open up in the scrum framework to improve compliance in scope, risk management and delivery times in software development projects," in *Proceedings of the Computational Methods in Systems and Software*, 2019, pp. 404–418.

[56] D. Ionita, C. van der Velden, H. J. K. Ikkink, E. Neven, M. Daneva, and M. Kuipers, "Towards risk-driven security requirements management in agile software development," in *Lecture Notes in Business Information Processing*, 2019, vol. 350, pp. 133–144. doi: 10.1007/978-3-030-21297-1_12.

[57] L. Ferreira, A. Pilastri, C. Martins, P. Santos, and P. Cortez, "A Scalable and Automated Machine Learning Framework to Support Risk Management," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12613 LNAI, 2021, pp. 291–307. doi: 10.1007/978-3-030-71158-0_14.

[58] M. Roy, N. Deb, A. Cortesi, R. Chaki, and N. Chaki, "Requirement-oriented risk management for incremental software development," *Innovations in Systems and Software Engineering*, vol. 17, no. 3, pp. 187–204, Sep. 2021, doi: 10.1007/s11334-021-00406-6.

[59] J. Finger, K. Ross, I. Häring, E.-M. Restayn, and U. Siebold, "Open Chance and Risk Management Process Supported by a Software Tool for Improving Urban Security," *European Journal for Security Research*, vol. 6, no. 1, pp. 39–71, Apr. 2021, doi: 10.1007/s41125-021-00072-6.

[60] E. Patelli, *A Multi-Disciplinary Software Suite for Uncertainty Quantification and Risk Management*, no. November. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-11259-6.

[61] A. K. Chinemeze and B. C. Mbam, "Impact of Risk Managementon Software Projectsin Nigeria Using Linear Programming," no. 7, pp. 142–147, 2019, [Online]. Available: https://

www.researchgate.net/profile/Kyrian-Adimora-2/publication/347937527_U0807186192/links/5fe8dfd9299bf14088503489/U0807186192.pdf

[62] A. Iordache and A. Woinaroschy, “Drinking Water Quality Risk Management. Risk Analysis of Nitrogen Groundwater Contamination Using Analytica Software,” *Revista de Chimie*, vol. 70, no. 11, pp. 3971–3976, Dec. 2019, doi: 10.37358/RC.19.11.7684.

[63] W. Khan, “A Review on Some Pertinent Software Security Risk Management Frameworks,” no. September 2020, pp. 5–10, 2021.

[64] J. Nyfjord, “Towards integrating agile development and risk management,” Institutionen för data-och systemvetenskap (tills m KTH), 2008.

[65] Schwaber Ken and Sutherland Jeff, “Panduan Definitif untuk Scrum: Aturan Permainan,” *Scrum.Org*, no. November, pp. 1–17, 2020.

[66] A. Moran, “Agile risk management,” in *Agile Risk Management*, Springer, 2014, pp. 33–60.

[67] M. el Bajta and A. Idri, “Identifying Risks of Software Project Management in Global Software Development: An Integrative Framework,” in *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*, Sep. 2020, pp. 1–5. doi: 10.1145/3419604.3419780.

[68] M. A. Rafeek, A. F. Arbain, and E. Sudarmilah, “Risk mitigation techniques in agile development processes,” *International Journal of Supply Chain Management*, vol. 8, no. 2, pp. 1123–1129, 2019.

[69] J. Masso, F. J. Pino, C. Pardo, F. García, and M. Piattini, “Risk management in the software life cycle: A systematic literature review,” *Computer Standards and Interfaces*, vol. 71. Elsevier B.V., Aug. 01, 2020. doi: 10.1016/j.csi.2020.103431.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Backtracking and k-Nearest Neighbour for Non-Player Character to Balance Opponent in a Turn-Based Role Playing Game of Anagram

Yosa Aditya Prakosa*, Alfa Faridh Suni

Departement of Electrical Engineering, Faculty of Engineering, Universitas Negeri Semarang

*yosaadityap@students.unnes.ac.id

Abstract-Anagram is a turn-based role-playing game where two players construct words by arranging given letters. A significant aspect of playing a game is the challenge. A good challenge comes from an opponent with a close ability. In a two-player game like Anagram, the second player can be a nonhuman player called Non-Playable Character (NPC). A balanced game is more engaging. Therefore, it is imperative to insert artificial intelligence (AI) into an NPC to make it possess a balance ability. This study investigates the AI algorithm that is the most appropriate to make a balance NPC for Anagram games. We tested three scenarios: Descending AI, Random AI, and AI with k-Nearest Neighbour (k-NN). Descending AI gets an Anagram solution by selecting a word with the highest score from all possible answers. Random AI picks a word randomly from the possible answers, while AI with k-NN chooses a word closest to one of the human players. The results show that Descending AI is the best algorithm to make the strongest NPC, which always gets the highest score, followed by Random AI and AI with k-NN. However, AI with the k-NN algorithm makes the constructed NPC has the highest number of turns at an average of 18, while Descending AI gets 14 turns and Random AI has 15 turns. Looking at the remaining lives at the end of the game, AI with k-NN makes the NPC has 25 lives left, while Descending AI has 59 lives, and Random AI has 48 lives. Less remaining lives suggest that NPC containing AI with the k-NN algorithm matches closer to the human player and therefore is more suitable for Anagram NPC.

Keywords: anagram, artificial intelligence, game, non-player character

Article info: *submitted December 27, 2021, revised May 27, 2022, accepted June 11, 2022*

1. Introduction

One of the rapidly growing computer software technologies is gaming. There are many types of games that have been developed, one of which is word puzzle games or anagrams. An anagram is a game that forms a word by rearranging the letters of another word [1]. The main purpose of anagrams is to form correct words according to the spelling system of a language [2]. One of the innovations that have been developed in anagram games is the gameplay, where players must be able to compose anagram and sub-anagram words to attack enemies [3]. In the research by Kuswardyan et al. [3], anagrams are used as a battle system in turn-based role-playing games (RPGs). Turn-based RPG is a combat system in which attacks from players or enemies have been carried out alternately [4], waiting for input from the player [5]. In turn-based RPGs, the enemy can be a Non-Player Character (NPC) that performs activities automatically [6]. NPCs that have artificial intelligence (AI) can increase user engagement in playing games [6]. Artificial intelligence is a technology that can make decisions by analyzing and using the data available in the system [7].

In the anagram game by [3], NPCs don't have artificial intelligence yet. NPCs that do not have AI will be easily defeated by players, making the game less challenging [6]. On the other hand, NPCs who have an invincible AI will make players desperate so they are not interested in playing the game [8]. Therefore, intelligent AI is needed for NPCs to be able to keep up with players [9]. Several methods can be used to provide intelligence to NPCs, for example, fuzzy [5] and rule base [7]. The rule base method applies the rules according to the situation and actions [10]. An example of applying the rule base is to divide the behavior of NPCs into 3 types, namely descending AI, random AI, and k-NN AI. Descending AI answers the anagram with the highest value word. Random AI answers with a random word. Meanwhile, k-NN AI answers with words calculated by the k-nearest neighbor (k-NN) classification algorithm. The k-NN algorithm has the opportunity to be an option because it is an adaptive algorithm and can adapt to the players' answers [11]. The k-NN algorithm has better accuracy than the nave Bayes method.

This paper discusses the results of research on the most suitable rule base to be applied to anagram games with the turn-based role-playing game (RPG) genre. The match in question is the ability of the NPC to provide a balanced resistance to the player.

There have been many studies related to the addition of intelligence to NPCs. Research [3] applies a backtracking algorithm to search for anagrams. They proved that the backtracking algorithm was able to find anagram words well.

Susanto et al. compared 2 classification methods that have high accuracy, namely k-NN and Naïve Bayes [11]. They stated that the k-NN method has higher accuracy than Naïve Bayes. The accuracy of k -NN is at 93.17% while the accuracy of Naïve Bayes is at the level of 78.38%. Higher accuracy is expected to make NPCs more adaptive and dynamic so that they become a balanced opponent for players.

2. Methods

a. Non-Player Character (NPC)

A non-Player Character is an entity in the game that is controlled automatically by a computer, not by humans [12]. NPCs can be friends, foes, or neutrals [13]. NPCs are expected to behave intelligently like humans. He can respond to answers according to the actions of the original player. Intelligent NPCs can be obtained by adding artificial intelligence (AI) to characters [14]. The use of AI on NPCs is done by giving certain algorithms according to the expected intelligent behavior [15]. In anagram games, AI is made in such a way as to create intelligent NPCs in choosing anagram words, so that players feel challenged in playing the game.

b. Implementation of AI on NPCs

Artificial Intelligence in anagram games is applied to be able to search and choose good anagram words so that they are not easily defeated by players. The flow of AI implementation on NPCs can be seen in Figure 1.

c. Word scramble

The first step when the game starts is word shuffling. The scrambled word is a combination of all letters with vowels. It is intended that in every word scramble there are vowels in it. The number of anagram letters used in this study was 6 letters.

d. Composing and matching words with Backtracking

The second step after the words have been scrambled is to generate or arrange words. At this stage, a backtracking algorithm is applied (see Figure 2). If the word is not available it will change to another word (backtrack). Words are compiled from the existing letters and then searched in a database of available dictionaries. The words available in the dictionary are added to the list to calculate the score.

c. Scoring

Each word in the list is scored by converting each letter used to a number. Each letter has a different score (see the scoring guide in Figure 3). The score is between 1 and 10. Frequently used characters are low and rarely used are high. The vowels and letters l, n, r, s, and t have a value of 1, while the letters q and z are worth 10. The results of the score calculation for each word in the list are stored in the score list.

d. Word Selection

Based on the list of scores formed, the process of selecting words is carried out. This study tested 3-word selection algorithms, namely Descending AI, Random AI, and k-NN.

1) Descending AI

   The Descending AI method selects the word from the list that has the highest score. The score list is sorted in descending order and then takes the word that is in the topmost position.

2) Random AI

   The Random AI method selects words from the list randomly. Although it is not necessary, in this research, the process of calculating the score and the process of sorting the data is still carried out. Then the word is selected using random.next() command to ensure a random word selection. The word selected may have the highest score and may have the lowest score or in between.

3) k-NN

   The application of the k-NN method requires observations to determine the correct value of k. The value of k shows the number of neighbors that become the benchmark for selecting groups for new data. The new data is placed in groups based on the distance between the data and other data already in the group. The distance calculation is done by equation (1) which is none other than Euclidean distance [6].

$$d_{ij} = \sqrt{\sum_{k=1}^{n}\left(a_{ik} - a_{jk}\right)^2} \qquad (1)$$

In equation (1), n is the number of attributes and is the distance between data. The smaller the distance value, the more similar the two data. The data that is calculated for the proximity value is the player's answer score and the available score. The player's answer is calculated for its proximity to all available potential answers, then the answer that has the smallest closeness to the player's answer is selected.

e. Assault

Assault is the process of reducing the live score of players and NPCs. Each opponent is given a live score of 100 at the start of the game. If one of them manages to make a word with a score of 15 for example, then the live score of the enemy is reduced by 15 points, as a form of damage to the opponent. The higher the player's score when answering the anagram, the more damage the opponent takes.

Players get the opportunity to answer anagrams many times in a turn. Each time an answer is given, the opponent takes damage and the live score is reduced. The game ends when the live score of the player or NPC is exhausted (less than or equal to zero).

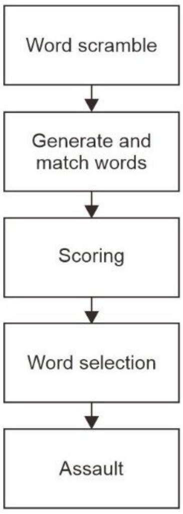


Figure 1. Implementation of AI on NPCs

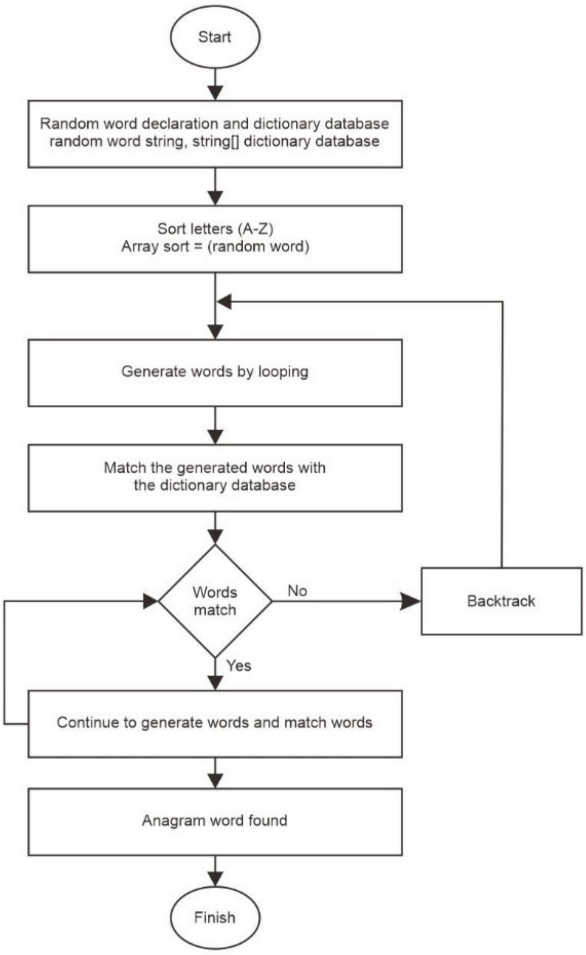

Start
Random word declaration and dictionary database random word string, string[] dictionary database
Sort letters (A-Z) Array sort = (random word)
Generate words by looping
Match the generated words with the dictionary database
Words match
No
Backtrack
Yes
Continue to generate words and match words
Anagram word found
Finish

Figure 2. Flowchart generates and matches words

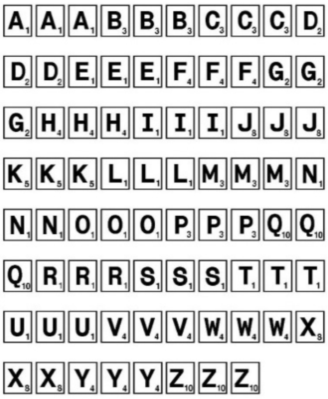

A_1 A_1 A_1 B_3 B_3 B_3 C_3 C_3 C_3 D_2
D_2 D_2 E_1 E_1 E_1 F_4 F_4 F_4 G_2 G_2
G_2 H_4 H_4 H_4 I_1 I_1 I_1 J_8 J_8 J_8
K_5 K_5 K_5 L_1 L_1 L_1 M_3 M_3 M_3 N_1
N_1 N_1 O_1 O_1 O_1 P_3 P_3 P_3 Q_{10} Q_{10}
Q_{10} R_1 R_1 R_1 S_1 S_1 S_1 T_1 T_1 T_1
U_1 U_1 U_1 V_4 V_4 V_4 W_4 W_4 W_4 X_8
X_8 X_8 Y_4 Y_4 Y_4 Z_{10} Z_{10} Z_{10}

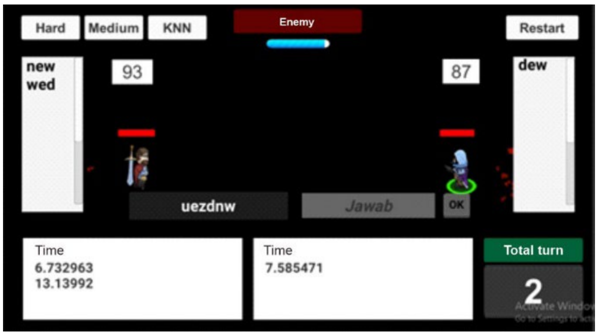
Figure 3. Anagram letter conversion guide

3. Result and Discussion

This section describes the results of game simulations, game testing, and performance comparisons of Descending AI, Random AI, and AI k-NN algorithms.

a. Game simulation

For this research, we designed the game and developed it using Unity 3D. Figure 4 shows a screenshot of the running game, where human players (on the left) are playing against NPCs (on the right). On the display, various display boxes show the status of the game being played. At the bottom is a time display that shows how long one anagram step was answered. The left and right sections show the words that have been answered by players and NPCs. A bit in the middle of the live performance of the two players. At the bottom right there is a display of the number of turns that have been played.



Time
6.732963
13.13992

Time
7.585471

Total turn
2

Figure 4. Display when the game is played

b. Game testing

Data was collected by asking 10 to play the game. The ten people consisted of 5 ordinary gamers and 5 teachers who had good English skills. Each person is asked to play the game 3 times, each against an NPC controlled by Descending AI, Random AI, and AI k-NN algorithms.

c. Game with Descending AI

The results of the game simulation between players and NPCs controlled by the Descending AI algorithm are shown in table 1. The table shows the time it takes for the NPC to answer the word, the number of turns required in the game, and the remaining live belonging to the NPC.

Table 1. NPC Descending AI game data measurement

Player	Answering Time	Total Turn	Life Remaining
Gamer 1	5.2 s	15 *turn*	67
Gamer 2	6.1 s	14 *turn*	73
Gamer 3	5.8 s	13 *turn*	65
Gamer 4	5 s	20 *turn*	66
Gamer 5	6.6 s	13 *turn*	66
Teacher 1	6.4 s	13 *turn*	49
Teacher 2	7.6 s	13 *turn*	52
Teacher 3	7.3 s	18 *turn*	53
Teacher 4	6.8 s	10 *turn*	48
Teacher 5	5.5 s	15 *turn*	56

Based on the data presented in table 1, the average length of the answer is calculated, the average number of turns, and the average remaining live for NPCs. The calculation results are presented in table 2.

Table 2. Average NPC Descending AI game data

Player	Average Answering Time	Average Total Turns	Average NPC Life Remaining
Gamer	5.74 s	15 turn	67
Teacher	6.72 s	14 turn	52

Table 2 shows that NPCs need more time when playing against teachers, which on average takes 6.72 seconds compared to 5.74 seconds when playing against ordinary gamers. NPCs get fewer turns, which is 14 turns against teachers compared to 15 turns against regular gamers. Furthermore, the remaining live NPCs when playing against teachers are a smaller number than the remaining when against ordinary gamers.

d. Game with Random AI

The results of the game simulation between human players against NPCs who work using the Random AI algorithm are shown in table 3. Then the average data is presented in table 4.

Table 3. Random AI NPC game data measurement

Player	Answering Time	Total Turn	Life Remaining
Gamer 1	6.5 s	14 *turn*	57
Gamer 2	7.4 s	18 *turn*	43
Gamer 3	8.3 s	13 *turn*	49
Gamer 4	7.4 s	14 *turn*	65
Gamer 5	6.6 s	15 *turn*	76
Teacher 1	6.4 s	16 *turn*	29
Teacher 2	6.3 s	19 *turn*	19
Teacher 3	6.9 s	14 *turn*	47
Teacher 4	6.8 s	14 *turn*	58
Teacher 5	7.3 s	17 *turn*	43

Table 4. Random AI NPC game data average

Player	Average Answering Time	Average Total Turns	Average NPC Life Remaining
Gamer	7.24 s	15 *turn*	58
Teacher	6.74 s	16 *turn*	39

NPCs with Random AI algorithms take longer to answer when facing regular gamers. The number of turns obtained by NPCs is 16 turns when facing teachers, which is more than the turns obtained when facing ordinary gamers. Furthermore, the life that NPCs have when the game is over is 39 when facing teachers, which is smaller than the rest when facing ordinary gamers.

e. Games with k-NN

Game simulations between human players and NPCs run with the k-NN algorithm are shown in table 5. Game data shows the average time to answer words, the number of turns, and the remaining lives of the NPCs at the end of the game. On two occasions playing against the teacher, it was seen that the remaining live was 0 which meant that the NPC had lost.

Table 5. NPC game data measurement with k-NN

Player	Answering Time	Total Turn	Life Remaining
Gamer 1	7.4 s	20 *turn*	54
Gamer 2	5.8 s	21 *turn*	31
Gamer 3	6.6 s	19 *turn*	28
Gamer 4	7.1 s	15 *turn*	33
Gamer 5	5.3 s	20 *turn*	37
Teacher 1	6.5 s	15 *turn*	0
Teacher 2	6.2 s	22 *turn*	22
Teacher 3	6.9 s	19 *turn*	16
Teacher 4	5.2 s	17 *turn*	0
Teacher 5	6.7 s	17 *turn*	26

Furthermore, the data in table 5 are averaged and presented in table 6. The calculation results show that the teacher as a player is a formidable opponent for the NPC. In general, NPCs need a little more time to respond when playing against teachers. The number of turns obtained is less when facing the teacher. Even the live remaining NPC is smaller when playing against the teacher.

Table 6. Average NPC game data with k-NN

Player	Average Answering Time	Average Total Turns	Average NPC Life Remaining
Gamer	6.44 s	19 *turn*	37
Teacher	6.3 s	18 *turn*	13

f. NPC performance for various algorithms

The data in table 2, table 4, and table 6 become the basis for comparing the performance of NPCs when controlled by various algorithms. The three tables present the average NPC game data in the form of average answering time, the average number of turns, and the average remaining live when NPCs are controlled by Descending AI, Random AI, and AI with k-NN algorithms.

Table 7. Comparison of Answering Time

Player	NPC Answering Time		
	Descending AI	Random AI	AI k-NN
Gamer	5.74 s	7.24 s	6.44 s
Teacher	6.72 s	6.74 s	6.3 s
Average	6.23 s	6.99 s	6.37 s

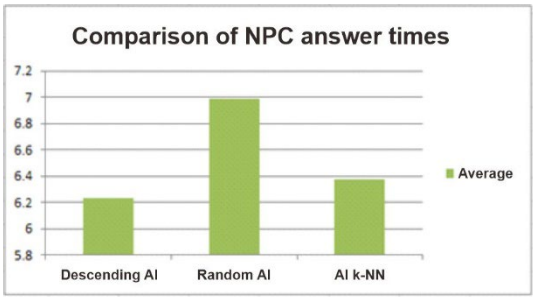


Figure 5. Comparison Graph of Answering Time Descending AI, Random AI, and AI k-NN

Table 7 shows the average data for answering NPCs when playing against gamers and teachers. The average answer time is presented visually in Figure 5. It can be seen that the fastest answering time is given by the NPC when controlled by the Descending AI algorithm, which is 6.2 seconds and the longest time is obtained when controlled by the Random AI algorithm, which is 6.99 seconds. The difference in the speed of answering is not significant enough when viewed from the variation in the data presented in Table 1 and Table 3.

Table 8. Comparison of Total Turn

Player	Total Turn		
	Descending AI	Random AI	AI k-NN
Gamer	15 *turn*	15 *turn*	19 *turn*
Teacher	14 *turn*	16 *turn*	18 *turn*
Average	14 *turn*	15 *turn*	18 *turn*

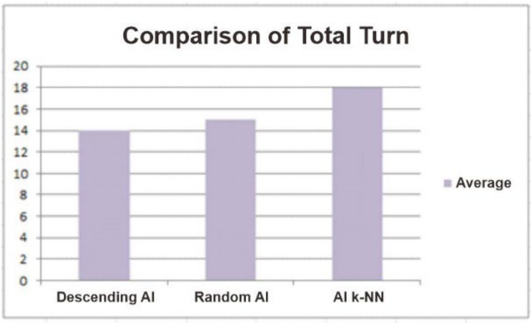


Figure 6. Comparison Graph of Descending AI Turns, Random AI, and AI k-NN

Table 8 presents data on the average number of turns that NPCs get when playing against gamers and teachers. The average number of turns is then presented in Figure 6. It can be seen that the highest number of turns is obtained by NPCs when controlled using the k-NN algorithm, which is 18 turns. While the least number of turns is obtained when the NPC is controlled using the Descending AI algorithm, which is 14 turns. This difference is significant when viewed from the variation of the data in table 1 and table 5.

Table 9. NPC Life Remaining Comparison

Player	NPC Life Remaining		
	Descending AI	Random AI	AI k-NN
Gamer	67	58	37
Teacher	52	39	13
Average	59	48	25

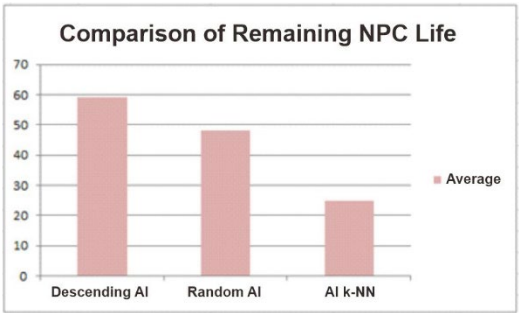


Figure 7. Comparison Graph of NPC Descending AI, Random AI, and AI k-NN

Table 12 shows the average remaining live that NPCs have at the end of the game when facing gamers and teachers. The average game data against the two types of players is presented graphically in Figure 7. It can be seen that the NPCs with Descending AI dominated the game and only lost a few lives with the remainder at 59. The NPCs with AI k-NN had an average remaining live of Rp. 25 which shows that NPCs can have a chance of winning but not very much. The data in table 7 confirms that NPCs with AI k-NN lost after losing all lives.

Based on the test data, it shows that NPCs with Descending AI cannot be defeated and tend to dominate the game. The game ends quickly, with an average of 14 turns with an average remaining live of 59. Each time answering an NPC with Descending AI only takes an average of 6.2 seconds.

The test results show that NPCs with Random AI still dominate the game relatively. The game ends in 15 turns with an average remaining live of 48. Test data shows this NPC has never been beaten even though it took the longest to come up with an answer. NPCs with k-NN do not dominate the game even though statistically still win more games. Games against NPCs with k-NN averaged over 18 turns with an average remaining live of 25.

The ideal NPC can keep up with players [16]. The NPCs worth implementing aren't the ones that dominate the game and can't be beaten. A good NPC is not an easy one to beat in every game. Therefore, NPC with k-NN in the Anagram game is the best choice among the AI algorithms tested in this study.

4. Conclusion

Results and discussions show that the NPCs with Descending AI dominate the play. The NPC ends the game in an average of 14 turns and saves 59 remaining lives. The less dominant NPCs are those controlled by k-NN. The latter finishes the game in 18 turns with 25 remaining lives. NPCs with k-NN can be defeated by human players, while NPCs with Descending AI and Random AI are unbeatable. Therefore, k-NN is the best choice of the three algorithms tested in this study as controller of the Anagram NPC.

Reference

[1] M. A. Rahman, "The Effectiveness of Anagram on Students ' Vocabulary Size," *J. IAIN Palangkaraya*, no. December, pp. 129–139, 2016.

[2] C. T. Panagiotakopoulos and M. E. Sarris, "'Playing with words': Effects of an anagram solving game-like application for primary education students," *Int. Educ. Stud.*, vol. 6, no. 2, pp. 110–126, 2013, doi: 10.5539/ies.v6n2p110.

[3] I. Kuswardayan, R. Rahman, and N. Suciati, "Design and Implementation of Random Word Generator using Backtracking Algorithm for Gameplay in Ambrosia Game," *Int. J. Comput. Appl.*, vol. 158, no. 6, pp. 27–30, 2017, doi: 10.5120/ijca2017912822.

[4] R. D. Pertiwi, "Planning of Making the Introducing Edifice Game," no. December, pp. 12–13, 2014.

[5] D. Ratanajaya and H. A. Wibawa, "Implementasi Kecerdasan Buatan dalam Menentukan Aksi Karakter pada Game RPG dengan Logika Fuzzy Tsukamoto," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 4, no. 2, p. 82, 2018, doi: 10.23917/khif.v4i2.6744.

[6] M. I. A. Putera and D. H. Murti, "Peningkatan Kecerdasan Computer Player Pada Game Pertarungan Berbasis K-Nearest Neighbor Berbobot," *JUTI J. Ilm. Teknol. Inf.*, vol. 16, no. 1, p. 90, 2018, doi: 10.12962/j24068535.v16i1.a710.

[7] M. Abdi, D. Herumurti, and I. Kuswardayan, "Analisis Perbandingan Kecerdasan Buatan pada Computer Player dalam Mengambil Keputusan

pada Game Battle RPG," *JUTI J. Ilm. Teknol. Inf.*, vol. 15, no. 2, p. 226, 2017, doi: 10.12962/j24068535.v15i2.a671.

[8] V. Vorachart and H. Takagi, "Evolving fuzzy logic rule-based game player model for game development," *Int. J. Innov. Comput. Inf. Control*, vol. 13, no. 6, pp. 1941–1951, 2017, doi: 10.24507/ijicic.13.06.1941.

[9] G. Grund, P. M. Nilsson, M. Larsson, O. Olsson, T. Foughman, and V. Gustafsson, "Realistic NPCs in Video Games Using Different AI Approaches," no. June, 2016.

[10] M. Gwon, E. Goh, C. Sohn, and A. R. B. Ai, "The VR Trip Simulator with Multi Networking of Rule-based Model," vol. 13, no. 22, pp. 15754–15757, 2018.

[11] E. bagus Susanto, M. K. Triawan adi cahyanto, and pd M. Reni umilasari S, "Analisis Perbandingan Algoritma Naive Bayes Dan k-NN Untuk Klasifikasi Multi Dataset," no. 1410651097, 2019.

[12] H. Warpefelt, *The Non-Player Character: Exploring the believability of NPC presentation and behavior*, no. 16. 2016.

[13] R. D. Agustin, "Komponen Konsep dan Desain Game," *J. Ilm. Teknol. Inf. Terap.*, vol. III, no. 2, 2017.

[14] E. Siswanto and A. F. Suni, "Aksi Penyerangan Non-Player Character (NPC) Menggunakan Metode Naïve Bayes Pada Shooter Game," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 6, 2021, doi: 10.25126/jtiik.202183804.

[15] F. B. Adi, M. Hariadi, and I. K. E. Purnama, "Simulasi Perilaku Tempur Pada Sekumpulan NPC Berbasis Boid," 2016.

[16] M. Kopel and T. Hajas, "Implementing AI for Non-player Characters in 3D Video Games," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10751 LNAI, no. January 2018, pp. 610–619, 2018, doi: 10.1007/978-3-319-75417-8_57.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Object Detection to Identify Shapes of Swallow Nests Using a Deep Learning Algorithm

Denny Indrajaya [1]*, Adi Setiawan[1], Djoko Hartanto[2], Hariyanto[2]

[1]Department of Mathematics and Data Science, Faculty of Science and Mathematics

[1]Universitas Kristen Satya Wacana

Salatiga

[2]PT Waleta Asia Jaya

Salatiga

*662018003@student.uksw.edu

Abstract-Object detection is basic research in the field of computer vision to detect objects in an image or video. the TensorFlow framework is a widely adopted framework to create object detection programs and models. In this study, an object detection program and model are designed to detect the shape of a swallow's nest which consists of three classes, namely oval, angular, and bowl. The purpose model creation is to find out the likeliness of the swallow's nest to the three classes for the swallow's nest sorting machine. The adopted architecture in the modeling is the MobileNet V2 FPNLite SSD since the model obtained from this architecture results in a good speed in detecting objects. Based on the evaluation results that has been carried out, the model can detect the shape of the swallow's nest which is divided into 3 classes, but in some cases swallow's nest are detected into two classes. This issues can still be handled by adjustmenting several parameterss to the object detection program. Results shows that the obtained mAP value of 61.91%, indicating the model can detect the shape of a swallow's nest moderately.

Keywords: object detection, swallow's nest, SSD MobileNet V2 FPNLite, classification, deep learning

Article info: *submitted December 1, 2021, revised May 10, 2022, accepted June 11, 2022*

1. Introduction

Swallow's nest is a nest formed from swallow's saliva which can be consumed by humans. The nest has many benefial properties and it tastes delicious [1]. Before the swallow's nest is consumed, it needs to be processed, in which the condition of the swallow's nest before being processed varies according to various things, such as the color, shape, and intensity of the feathers. With these various conditions, of course, the treatment during processing given for each nest condition is different. Therefore, the first step that needs to be done in processing swiftlet nests is the nest sorting process. Currently, a company i.e., PT Waleta Asia Jaya processes swallow nests by sorting the swallow nests based on color, shape, and feather intensity manually by humans power, thus it slows the production lane. In the development of technology, various tasks can be done with the help of machines to improve company performance in various aspects, two of which are speed and automation. In connection with the process of sorting swallow's nests, a sorting machine can be designed that has the human-like ability to sort swallow's nests. To make machines able to do the task, it is necessary to adopt the theory of computer vision and embed it to the machines.

Computer vision is a branch of computer science that involves image processing and pattern recognition to understand an image or object in images and videos [2], [3]. In computer vision, object detection science uses deep learning algorithm to do complex things such as tracking an object, detecting events, and analyzing behavior [4]. One of the deep learning architectures for creating object detection models with a learning process using data in the form of digital images is SSD MobileNet.

SSD MobileNet is an SSD architecture (Single Shot Multibox Detector) with the MobileNet extractor feature, which architecture can work with little computation, so it is fit to run in real-time [5]. SSD itself is an object detection architecture that has high accuracy and is fast [6], while MobileNet is a lightweight feature extractor [7]. In the research of M. F. Supriadi, E. Rachmawati, and A. Arifianto, the mean Average Precision (mAP) value of the SSD MobileNet V2 architecture is better than the SSD MobileNet V1 in detecting 20 types of objects in the house [8]. In addition, some studies develop the MobileNet SSD architecture using the Feature Pyramid Network (FPN) module to improve detection accuracy [9]. Research on bird nest detection has been carried out several times, but in these studies, swallow nests were not used and bird nests were not divided into classes [10]–[12].

Based on the description, in this study, an object detection model will be created that can be used for the development of a swallow's nest sorting machine in a swallow's nest company. The model made focuses on detecting swallow nests which have 3 classes based on shape, namely swallow nests with oval, angular, and bowl shapes. The model for shape detection is made because the shape of the swallow's nest affects the nest processing process after sorting and the shape of the finished goods obtained, which will also affect the selling price. The model in this study was made using the python programming language, TensorFlow framework, and the SSD MobileNet V2 FPNLite architecture because this architecture it can make the detection process fast and light, which is suitable for use on sorting machines that demand speed in the sorting process.

2. Methods

In carrying out this research, a training process was carried out using data in the form of digital images. The research process consists of several steps, including:

a. Research Implementation Planning

Planning is the first stage carried out in this research, including identifying the problems faced, literature study, and analysis to solve these problems.

b. Data retrieval

Making object detection models requires data from digital images. The object detection model is designed to detect swallow nests and distinguish it shapes to bowl, oval, or angular. The shapes of the swallow's nest can be seen from the part of the nest attached to the wall before the nest is harvested. The bowl shape is a nest that has a flat surface on the part that is attached to the wall while the oval shape has a surface that is not flat, but there is a hollow that forms an angle of less than 90° and not close to 90°. For the angular shape, the surface attached to the wall when viewed in 2 dimensions forms an angle of about 90°. Illustrations of shapes for each class can be seen in Figure 1.

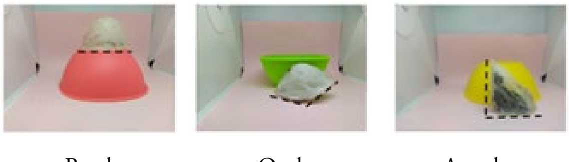


Figure 1. Swallow's nest shape illustration

Detection of objects is done on 2-dimensional images. For this reason, in taking digital pictures using a smartphone camera, it is necessary to pay attention to the shape of a swallow's nest when viewed from the camera's point of view. In addition, giving the background color to the image is also considered in this study. The sample data used in the study is shown in Figure 2.

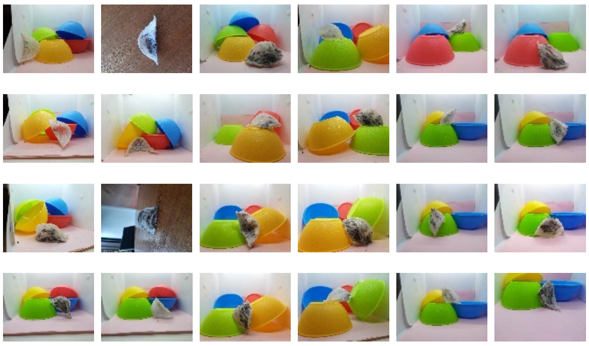
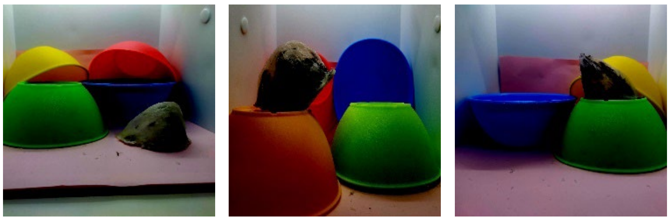


Figure 2. Sample images used in research

c. Data processing

Prior to data processing for model creation, image sorting based on its clarity is conducted at first. From this step, we obtain images with various sizes, where the length and width of the images obtained are 4608×3456, 4000×3000, and 3264×2448 (pixels) with a horizontal resolution of 72 dpi, and a vertical resolution of 72 dpi and the proportion of each swallow's nest in the figure is 5% to 20%. The next process is to divide the image into 2 parts, namely 80% as a training dataset and 20% as a testing dataset. The number of images used is 360 images with 1 nest object in each image, which means there are 288 images in the training dataset and 72 images in the testing dataset. In this study, swallow nests were grouped into 3 classes, where the data used for each class were 96 images on the training dataset and 24 images on the testing dataset.

In making the model, the class labeling process is carried out using the LabelImg software with the output in the form of a file with *.xml format for each image. The results of the class labeling process are then combined into a *.csv file format. There are 2 files with *.csv format created, namely files for dataset training and dataset testing. Then the two files are converted back into files with the *.record format. In addition, a file with the *.pbtxt format is also created which contains a list of the classes used. In the training process, the image data is also reprocessed into a size of 640×640, which is shown in Figure 3.



Bowl Oval Angular

Figure 3. Sample training pictures

d. Object Detection Modeling

Modeling is done using the python programming language and utilizing TensorFlow which is one of the deep learning frameworks for making object detection [13].

In this study, the SSD MobileNet V2 FPNLite architecture is used, where the SSD (Single Shot Multibox Detector) plays a role in detecting by creating bounding boxes to create image localization and determine object positions [14]. Based on [15], the determination of regional candidate boxes on the SSD architecture is used by formula (1).

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1), \quad k \in [1, m] \tag{1}$$

Note:

m = many layers,

s_{min} = lowest feature map scale,

s_{max}= highest feature map scale.

The regional width of the candidate box is calculated using the formula (2).

$$w_k^a = s_k\sqrt{a_r} \tag{2}$$

The regional height of the candidate box is calculated using formula (3).

$$\left(\frac{(i+0{,}5)}{w_{fk}}, \frac{(j+0{,}5)}{h_{fk}}\right), \ j \in [0, h_{fk}), i \in [0, w_{fk}), \tag{3}$$

Special a_r=1 additional scale required $s_k' = \sqrt{s_k s_{k+1}}$.

The coordinate center for each regional candidate box is

$$\left(\frac{(i+0{,}5)}{w_{fk}}, \frac{(j+0{,}5)}{h_{fk}}\right), \ j \in [0, h_{fk}), i \in [0, w_{fk}),$$

Note:

W_{fk}= width of feature map k,

H_{fk} = height of feature map k.

The SSD architecture itself uses VGG as a feature extractor. The SSD architecture with the VGG-16 feature extractor is shown in Figure 4. However, on the MobileNet V2 FPNLite SSD, the VGG-16 feature extractor is changed to MobileNet V2 FPNLite.

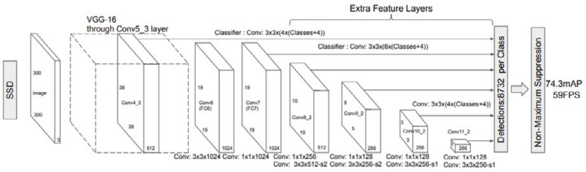


Figure 4. SSD architecture with VGG-16 feature extractor [16]

As explained in the previous paragraph, in the MobileNet V2 FPNLite SSD architecture, MobileNet V2 FPNLite is a feature extractor in extracting features on the image which will later be used on the SSD architecture for detecting objects in the image and their classification [9], [17], [18]. FPN itself is an architecture to produce pyramidal features in object detection, whereas FPNLite is a development of FPN which can produce models with lighter detection capabilities when run [19], [20].

The architecture of MobileNet is shown in Figure 5. The MobileNet structure uses Batch Normalization and the activation function of Rectified Liner Unit (ReLU) for depthwise convolution and pointwise convolution. Formula (4) is the ReLU6 activation function used in the MobileNet V2 structure [21], namely

$$y=\min(\max(z,0),6) \quad (4)$$

where z is the value for each pixel in the feature map. The MobileNet V2 architecture begins with a convolution layer of 32 filters, followed by 19 residual bottleneck layers as shown in Table 1.

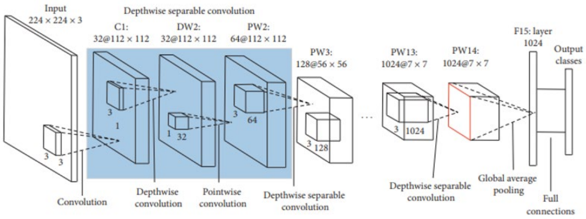


Figure 5. The architecture of MobileNet [22]

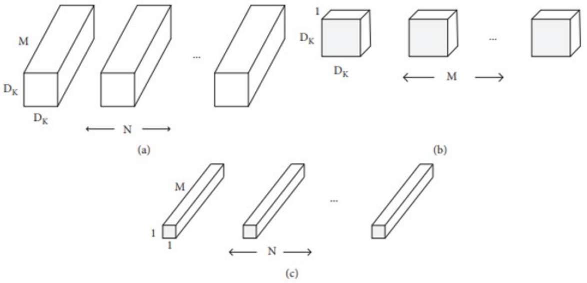


Figure 6. Standard convolutional filters and depthwise separable filters. (a) Standard convolutional filters, (b) depthwise convolutional filters, and (c) point convolutional filters [22]

Table 1. MobileNet V2 Architecture [21]

Input	Operator	*t*	*c*	*n*	*s*
224^2×3	conv2d	-	32	1	2
112^2×32	bottleneck	1	16	1	1
112^2×16	bottleneck	6	24	2	2
56^2×24	bottleneck	6	32	3	2
28^2×32	bottleneck	6	64	4	2
14^2×64	bottleneck	6	96	3	1
14^2×96	bottleneck	6	160	3	2
7^2×160	bottleneck	6	320	1	1
7^2×320	conv2d 1x1	-	1280	1	1
7^2×1280	avgpool 7x7	-	-	1	-
1×1×1280	conv2d 1x1	-		-	

In Table 1 each row describes a sequence of 1 or more identical layers (modulo stride) which are repeated *n* times. All layers in the same order have the same number of output channels c. The first layer of each sequence has stride s and the others have stride 1. All spatial convolutions use a kernel of size 3 × 3. The expansion factor t is always applied to the input size as shown in Table 2 [21].

Table 2. Bottleneck residue block transformation from channel k to k' with stride s and expansion factor t [21]

Input	Operator	Output
$h \times w \times k$	1x1 conv2d, ReLU6	$h \times w \times (tk)$
$h \times w \times tk$	3x3 dwise , ReLU6	$\frac{h}{s} \times \frac{w}{s} \times (tk)$
$\frac{h}{s} \times \frac{w}{s} \times tk$	linear 1x1 conv2d	$\frac{h}{s} \times \frac{w}{s} \times k'$

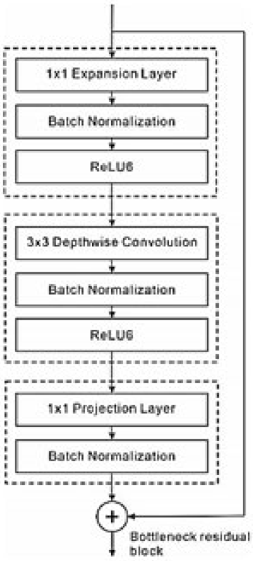


Figure 7. MobileNet V2 convolutional blocks [23]

e. Evaluation

The evaluation of the model is conducted using the object detection program. In the evaluation, the mean Average Precision (mAP) value is used by utilizing the confusion matrix. To obtain the mAP value, data from the test results and data from the class labeling process are used. The data from the test results are composed from the results of the classification, confidence score, and corner points in the predicted bounding box. The data used from the class labeling process is the result of the classification and the corner points in the ground-truth bounding box.

The Confusion Matrix is a table used to measure the performance of algorithms or classification models [24]. The values in the confusion matrix used to measure the performance of an algorithm are True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). Here is the form of the confusion matrix.

Table 3. Confusion matrix

	Positive Prediction	**Negative Prediction**
Positive Actual	*True Positive* (TP)	*False Negative* (FN)
Negative Actual	*False Positive* (FP)	*True Negative* (TN)

In using the confusion matrix in object detection, the Intersection over Union (IoU) threshold value will be determined first. IoU is the ratio between the intersection and the union of the ground-truth bounding box (Bgt) and the predicted bounding box (Bp). Formula (5) shows the calculation of the IoU value, namely

$$\text{IoU} = \frac{area\ (B_p \cap B_{gt})}{area\ (B_p \cup B_{gt})} \qquad (5)$$

The IoU threshold value is used to determine whether the detected object is true or false. For example, if the IoU threshold of 0.5 is selected, then:

- True Positive (TP): the model detects the object correctly and the IoU ≥ 0.5.
- False Positive (FP): the model detects the object correctly but the IoU value < 0.5, or the model detects the background as an object when it shouldn't be an object.
- False Negative (FN): the model failed to detect the object.
- True Negative (TN): the model does not detect the background or other objects.

The number of regional candidate boxes contained in the background or other objects that are not detected is very large. Therefore, TN cannot be used. Based on the values obtained from the confusion matrix, the precision and recall values can be searched with formulas (6) and (7), namely

$$Precision = \frac{\text{TP}}{\text{TP+FP}} (6)$$

$$Recall = \frac{\text{TP}}{\text{TP+FN}} (7)$$

Precision shows the model's ability to detect objects correctly, and recall shows the model's ability to detect objects in an image [25].

In measuring the performance of the object detection model, mAP can be used which is indicated by the formula (8) [26].

$$\text{mAP} = \frac{\sum_{i=1}^{k} \text{AP}_i}{k} \qquad (8)$$

note, k = number of classes used.

Based on the mAP formula shown in formula (8), it is necessary to find the value of the Average Precision (AP) for each class first with formula (9).

$$\text{AP} = \int_0^1 p(r)dr \qquad (9)$$

AP can be defined as the area on the Interpolated Precision Recall curve, which is to find the AP value by approximating the formula (10).

$$\text{AP} = \sum_{i=0}^{n-1} (r_{i+1} - r_i) P_{interp}(r_{i+1}) \qquad (10)$$

with

$$P_{interp}(r_{i+1}) = \max_{r' \geq r_{i+1}} P(r') (11)$$

To use formula (10) a table can be designed as shown in Table 4, where the table is the result of model testing to detect 5 objects. The following are the table cration steps:

1. Write down the detection results in the table, including the ID of the detection result which shows the predicted bounding box, then the IoU value and confidence score.
2. Sort the detection results in the table based on the confidence score from the largest to the smallest.
3. Fill in the data in the TP and FP columns based on the detection results and the IoU value.
4. Fill in the data in the Cumulative TP and Cumulative FP columns. In filling in the data in these columns, the data in the TP and FP columns are used in the row to be filled in and the previous rows.
5. Fill in the TP+FP column to calculate the value in the Precision column.
6. Fill in the TP+FN column to calculate the value in the Recall column. In this case, the object used in the test is 5 objects so the value of TP+FN is always 5.
7. Fill in the Precision and Recall fields. Using data in the Cumulative TP, Cumulative FP, TA+FP, and TP+FN columns.

Table 4. Table to calculate AP

Detection result ID	Confidence score	IoU	TP	FP	Cumulative TP	Cumulative FP	TA+FP	TP+FN	Precision	Recall
A	92	92	1	0	1	0	1	5	1	0.2
B	83	73	1	0	2	0	2	5	1	0.4
F	74	21	1	0	3	0	3	5	1	0.6
E	72	52	0	1	3	1	4	5	0.75	0.6
C	71	23	1	0	4	1	5	5	0.8	0.8
D	66	88	0	1	4	2	6	5	0.67	0.8

By using Table 4, especially the values in the Precision and Recall columns, a graph is made as shown in Figure 8. In this case, the recall values (r⟧_i) and precision (p(r⟧_i)) are implemented in the graph and then connected by orange lines. By using formula (11) the points P_interp (r_(i+1)) are also connected by a green line. Furthermore, the AP value is obtained by calculating the area under the green curve.

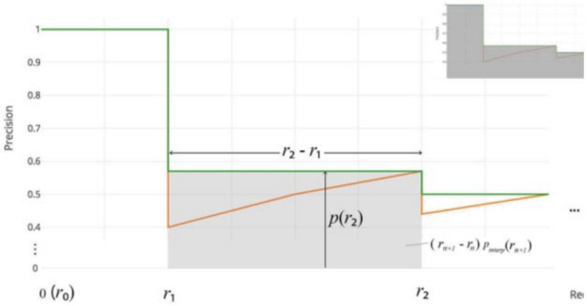


Figure 8. Precision-Recall Curve [27]

3. Result

The object detection model that has been created is then evaluated by testing using a program designed with the python programming language. The sample of the test results is shown in Figure 9.

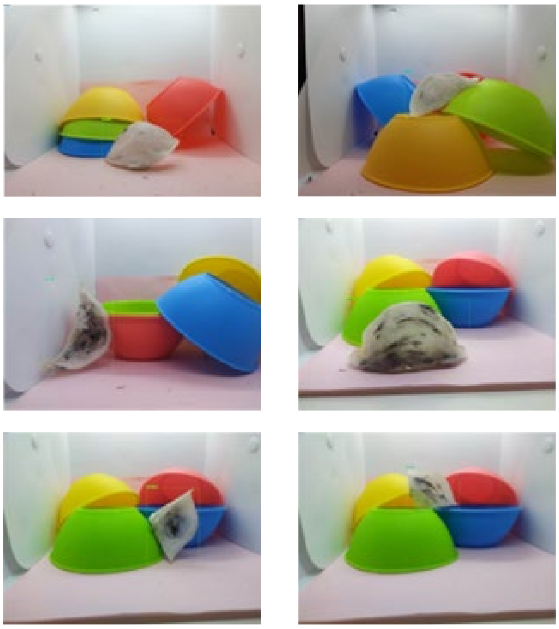


Figure 9. Sample object detection model test results

The test was carried out using 72 images which were divided into 3 parts, namely 24 images for each class. Where in each image there is only 1 bird's nest object, the results of the tests on the 72 images are shown in Figure 10.

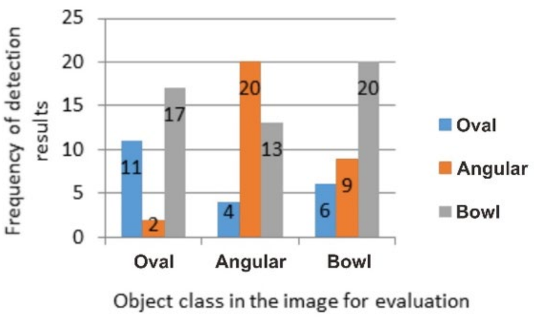


Figure 10. Graph of object detection model test results

Using the test results shown in Figure 10, an evaluation was carried out by finding the mean Average Precision (mAP). To get the mAP value, the Average Precision (AP) value for each class is required. In this case, the designed object detection has 3 classes. Therefore, the AP value of the three classes is sought first by using the Precision Recall curve.

In making the Precision Recall curve, an IoU threshold value is required, and the IoU threshold value used in this evaluation is 0.5. The following is the Precision Recall curve for each class obtained from the detection results shown in Figure 10.

Oval class

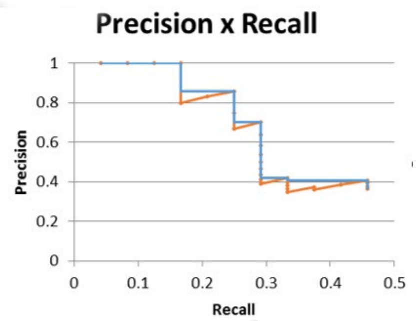


Figure 11. Precision Recall curve for oval class

Angular class

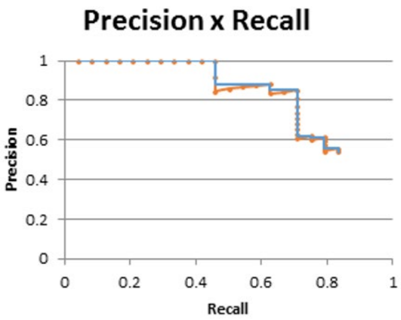


Figure 12. Precision Recall curve for angular class

Bowl class

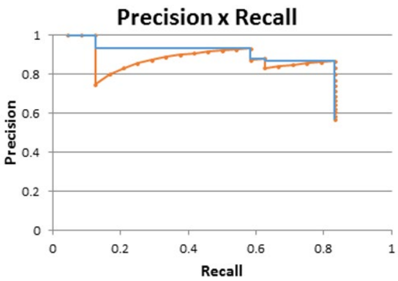


Figure 13. The precision-Recall curve for bowl class

Based on the Precision-Recall curve shown in Figures 11, 12, and 13, the AP value of swallow nest detection with oval class is 0.3357 or 33.57%, angular class is 0.7508 or 75.08%, and bowl class is 0.7707 or 77.07%. The AP values for each class are then averaged so that the mAP value is 0.6191 or 61.91%.

4. Discussion

According to the obtained results, the mAP value is not so large, which is 61.91%. This is due to the relatively small AP value of the oval class when compared to the AP value of the angular and bowl class. In addition, the amount of data used and other parameters for the training process such as the value of the training step and also affect the obtained mAP value.

If we take a look deper, the detection results from the experiments that have been carried out, the obtained results of the detection of a few oval class and many oval class nests detected as bowls. Likewise, many angular class nests are detected as bowls, but many angular classes are detected correctly. While the detection results for the correct bowl class are the same as the angular class, the number of detection results in the overall image with the angular and bowl class nest object is different. In addition, the correct detection results for the corner and bowl class objects are the same, but the AP values obtained are different. The difference is caused by the number of detection results in the entire image and the IoU threshold value that has been determined. These things make the AP value for the small oval class and the AP value for the angular class almost the same as the bowling class. Of the three classes, the highest AP value was obtained by the bowling class. Referring to the Table 5 which presents a summary of the results of the detection of the shape of the swallow's nest.

Table 5. Summary of swallow nest detection results

Class	Evaluation Data		Number of Detection Results			The number of total object detection results	AP (%)
	Number of pictures	Number of nest objects	Oval	Angular	Bowl		
Oval	24	24	11	2	17	30	33.57
Angular	24	24	4	20	13	37	75.08
Bowl	24	24	6	9	20	35	77.07

In Table 5 it can be seen that the detection results obtained for each class are greater than the number of swallow nest objects that should be. In testing the ability to detect an oval-shaped swallow's nest, 24 images were used with 24 objects of an oval-shaped swallow's nest in the entire image, but the results of detection of a swallow's nest obtained in these 24 images were bounding boxes of 30 consisting of 11 detection results of oval class, 2 corner classes, and 17 bowl classes. This is caused by the detection results that appear more than 1 on an object.

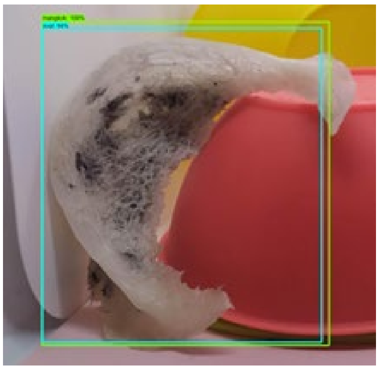


Figure 14. Two detection results on an object

The appearance of 2 detection results in 1 swallow's nest object shown in Figure 14 is caused by the shape of the swallow's nest which is almost similar when viewed in 2 dimensions. However, this can be handled by adjustments to the program made, such as limiting the number of objects that can be detected in 1 frame or 1 image and determining the minimum value for the confidence score. has been determined does not appear in the image. As is the case in Figure 14, if the minimum value of the 95% confidence score is determined, the results of the "oval" detection with a confidence score of 94% will not appear. That way, if the model is applied to a swallow's nest sorting machine, there will be no detection results of more than one result on 1 swallow's nest object.

Research related to bird nest detection has also been carried out by J. Li, D. Yan, K. Luan, Z. Li, and H. Liang [12]. In this study, several architectures were used to detect bird nests, namely Faster RCNN, Faster RCNN with Focal Loss, Cascade RCNN, and ROI Mining Faster RCNN with mAP values of 78.29%, 78.99%, 79.19%, and 82.51%, respectively. Based on this research, it can be seen that the object detection model can detect bird's nests with a fairly good performance. In this study, the development of detected nest objects, namely swiftlet nests, was divided into 3 classes based on their shape. From the results of the research that has been carried out, it is found that the object detection model that is made can detect the shape of a swallow's nest which is divided into 3 classes with an mAP value of 61.91%. The mAP value obtained is not so high when compared to the mAP value obtained from the bird's nest detection model in the study of J. Li, D. Yan, K. Luan, Z. Li, and H. Liang.

As previously shown, the mAP value obtained in this study is influenced by several things, such as the number of images used in the training process and the emergence of more detection results than the number of swallow nests in all images. In addition, the obtained mAP is also influenced by the image of a swallow's nest whose shape is difficult to classify as shown in Figure 15 (oval-shaped nest). The swallow's nest in the picture is difficult to classify because the part of the nest attached to the wall is not visible. Therefore, in classifying it, it is necessary to pay attention to certain parts of the nest, the parts in question are indicated by circles in Figure 15. After paying attention to these parts and a sufficient understanding of the shapes of the swallow's nest, the nest can be said to tend to enter the oval class. An example of the analysis in classifying the nest cannot be carried out by the object detection model that has been made because the model classifies the nest based on the color in the image that is converted to numeric and the computational process is based on the training results, so in this case, the probability of an error occurring in the model for classifying the nest will become large and has an impact on the mAP value.

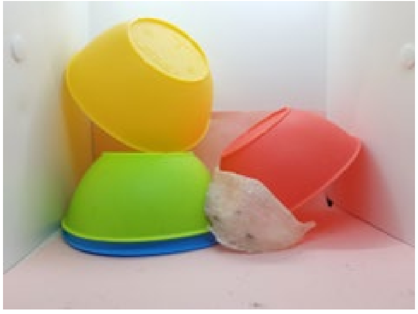


Figure 15. Swallow's nest is difficult to classify when viewed in 2 dimensions

Although in this study the mAP value obtained was not high, some of the detection results obtained showed good results. This can be seen from the size of the bounding box which corresponds to the size of the nest in the picture. In addition, it is also supported by the correct classification results and a high confidence score as shown in Figure 16. The results of incorrect detection can also be seen in Figure 17.

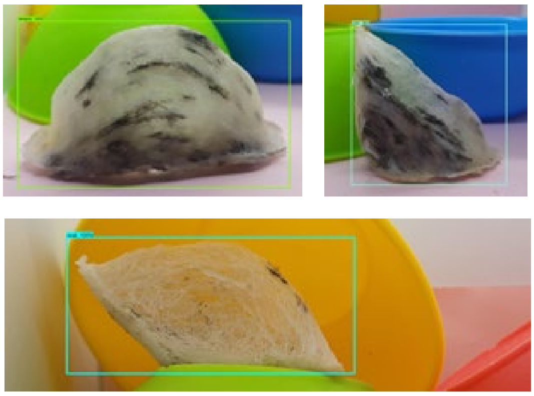


Figure 16. Good detection results

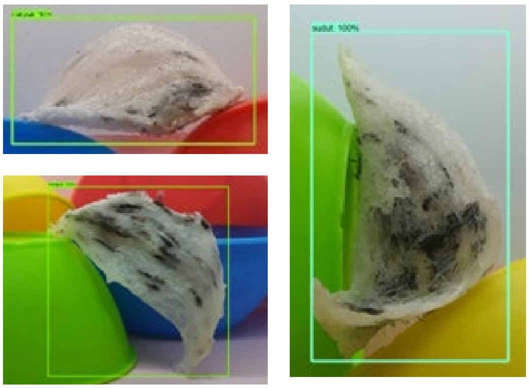


Figure 17. Incorrect detection result

In making the model, a training process is carried out, which if retraining is carried out with the same data, different results can be obtained. For this reason, the mAP value obtained in this study is still possible to be increased using the same data, by modifying the MobileNet V2 FPNLite SSD architecture and the values of the parameters used in the training process. Alternatives to using other architectures while taking into account the purpose of modeling can also be considered. In addition, it is also necessary to design an object detection model that is useful as validation when detecting swallow's nests. In this case the model will work by detecting the position of the swallow's nest from the camera's point of view. If the visible position of the swallow's nest is detected as a valid position, the results of the detection of the shape of the swallow's nest can be accepted. Valid in this case means that the shape of the swallow's nest when viewed from the camera's point of view can be determined.

5. Cunclusion

From the results of the research that has been done, the object detection model created using the SSD MobileNet V2 FPNLite architecture can detect the shape of a swallow's nest which has 3 classes, namely bowl, oval, and angular. In addition, based on the evaluation that has been carried out using 72 swallow nest images, an mAP value of 61.91% is obtained which shows the model's performance in detecting the shape of a swallow's nest which is divided into 3 classes.

Further research that can be done is to increase the mAP value by making a swallow nest shape detection model using another architecture or modifying the SSD MobileNet V2 FPNLite architecture by paying attention to the values of the parameters used in the training process. In addition, an object detection model can be made to detect the position of the swallow's nest when viewed from the camera's point of view, to reduce the error rate in the detection results if the model is to be used on a sorting machine.

6. Acknowledgment

Thank you to PT Waleta Asia Jaya for allowing us to conduct research using the company's swiftlet nests, as well as providing facilities for conducting research, such as computers used to create programs and object detection, models.

7. Reference

[1] L. Elfita, "Analysis on Protein Profile and Amino acid of Edible Bird's Nest (Collocalia fuchiphaga) from Painan," *Jurnal Sains Farmasi & Klinis*, vol. 1, no. 1, pp. 27–37, 2014.

[2] V. Wiley and T. Lucas, "Computer Vision and Image Processing: A Paper Review," *International Journal of Artificial Intelligence Research*, vol. 2, no. 1, pp. 28–36, 2018.

[3] S. R. U. . Dompeipen, Tresya Anjali Sompie and M. E. I. Najoan, "Computer Vision Implementation for Detection and Counting the Number of Humans," *Jurnal Teknik Informatika*, vol. 16, no. 1, pp. 65–76, 2021.

[4] J. Deng, X. Xuan, W. Wang, Z. Li, H. Yao, and Z. Wang, "A Review of Research on Object Detection based on Deep Learning," *Journal of Physics: Conference Series*, vol. 1684, 2020.

[5] A. N. A. Thohari and R. Adhitama, "Real-Time Object Detection for Wayang Punakawan Identification Using Deep Learning," *Jurnal Infotel*, vol. 11, no. 4, pp. 127–132, 2019.

[6] J. Huang *et al.*, "Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3296–3305.

[7] M. N. Rizal, D. T. Nugrahadi, R. A. Nugroho, M. R. Faisal, and F. Abadi, "Implementasi SSD_ Resnet50_V1 untuk Penghitung Kendaraan," *Kumpulan Jurnal Ilmu Komputer (KLIK)*, vol. 8, no. 2, pp. 106–115, 2021.

[8] M. F. Supriadi, E. Rachmawati, and A. Arifianto, "Pembangunan Aplikasi Mobile Pengenalan Objek untuk Pendidikan Anak Usia Dini," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 8, no. 2, pp. 357–364, 2021.

[9] Y. C. Chiu, C. Y. Tsai, M. Da Ruan, G. Y. Shen, and T. T. Lee, "Mobilenet-SSDv2: An Improved Object Detection Model for Embedded Systems," in *2020 International Conference on System Science and Engineering (ICSSE)*, 2020, pp. 1–5.

[10] X. Wu, P. Yuan, Q. Peng, C. W. Ngo, and J. Y. He, "Detection of Bird Nests in Overhead Catenary System Images for High-Speed Rail," *Pattern Recognition*, vol. 51, pp. 242–254, 2016.

[11] F. Li *et al.*, "An Automatic Detection Method of Bird's Nest on Transmission Line Tower Based on Faster_RCNN," *IEEE Access*, vol. 8, pp. 164214–164221, 2020.

[12] J. Li, D. Yan, K. Luan, Z. Li, and H. Liang, "Deep Learning-Based Bird's Nest Detection on Transmission Lines Using UAV Imagery," *Applied Sciences*, vol. 10, no. 18, p. 6147, 2020.

[13] D. J. P. Manajang, S. R. U. A. Sompie, and A. Jacobus, "Implementasi Framework Tensorflow Object Detection dalam Mengklasifikasi Jenis Kendaraan Bermotor," *Jurnal Teknik Informatika*, vol. 15, no. 3, pp. 171–178, 2020.

[14] P. R. Aningtiyas, A. Sumin, and S. Wirawan, "Pembuatan Aplikasi Deteksi Objek Menggunakan TensorFlow Object Detection API dengan Memanfaatkan SSD MobileNet V2 Sebagai Model Pra - Terlatih," *Jurnal Ilmiah Komputasi*, vol. 19, no. 3, pp. 421–430, 2020.

[15] K. Hu, F. Lu, M. Lu, Z. Deng, and Y. Liu, "A Marine Object Detection Algorithm Based on SSD and Feature Enhancement," *Complexity*, vol. 2020, 2020.

[16] W. Liu *et al.*, "SSD: Single Shot Multibox Detector," in *Computer Vision – ECCV 2016*, Springer International Publishing, 2016, pp. 21–37.

[17] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," 2017, pp. 2117–2125.

[18] I. B. Pakpahan and I. C. Dewi, "Pendeteksian Lubang Pada Jalanan Menggunakan Metode SSD-MobileNet," *Indonesian Journal of Electronics and Instrumentation Systems (IJEIS)*, vol. 11, no. 2, pp. 213–222, 2021.

[19] G. Ghiasi, T. Y. Lin, and Q. V. Le, "NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern*

Recognition (CVPR), 2019, pp. 7029–7038.

[20] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the Performance of One-Stage and Two-Stage Object Detectors in Autonomous Vehicles using Camera Data," *Remote Sensing*, vol. 13, no. 1, 2021.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[22] W. Wang, Y. Li, T. Zou, X. Wang, J. You, and Y. Luo, "A Novel Image Classification Approach via Dense-Mobilenet Models," *Mobile Information Systems*, vol. 2020, pp. 1–8, 2020.

[23] W. Rahmaniar and A. Hernawan, "Real-Time Human Detection using Deep Learning on Embedded Platforms: A Review," *Journal of Robotics and Control (JRC)*, vol. 2, no. 6, pp. 462–468, 2021.

[24] M. R. Faisal and D. T. Nugrahedi, *Belajar Data Science: Klasifikasi dengan Bahasa Pemrograman R.* Banjarbaru: Scripta Cendekia, 2019.

[25] R. Padilla, S. L. Netto, and E. A. B. Da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2020, pp. 237–242.

[26] G.-S. Peng, "Performance and Accuracy Analysis in Object Detection," California State University at San Marcos, 2019.

[27] J. Hui, "mAP (mean Average Precision) for Object Detection," *jonathan-hui.medium.com*, 2018. https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Application of Low Back Pain Myogenic Therapy Based on Multimedia

Alfian Gema Negara[1], Niken Sylvia Puspitasari[2], Izzati Muhimmah[3], Muhamad Anwar Aditya[4]
[1,3]Islamic University of Indonesia
Yogyakarta
[2,4]University of Darusssalam Gontor
Ponorogo
*nikensylviap@unida.gontor.ac.id

Abstract-Low back pain restricts activity and causes work absenteeism. Cases of low back pain are common worldwide. This paper presents the design of multimedia-based low back pain myogenic therapy aids. Data collection involves observation and interviews with medical rehabilitation specialists and physiotherapists. The collected data is represented using a production ruler in the form of if - then. Rule-based reasoning can be used as an expert system knowledge base in cases of myogenic low back pain. Forward chaining can be used as an inference engine for similar cases because the reasoning starts from the facts section before reaching the hypothesis. Design of this assistive device model is expected to provide information regarding the choice of therapy for low back pain myogenic by patients, independently at home or with the help of close family. Application design is multimedia-based to make it easier for users to look at examples visually. The expert system application is well accepted by users. Ten physiotherapists and one doctor consider the application performance good because it attains an acceptable value of 0.80 or 80%. The physiotherapists suggest that this assistive device model will likely increase the intensity of therapy because it can be carried out by the patient's family independently.

Keywords: multimedia application, low back pain, myogenic, design, therapy

Article info: *submitted September 1, 2021, revised January 26, 2022, accepted February 25, 2022*

1. Introduction

Low back pain (LBP) is an activity restrictions and work absenteeism. It is a health problem that very common worldwide [1], [2]. Low back pain does not cause death, but causes individuals becoming unproductive [3]. Since the mid-1990s, the incidence of back pain in the UK has increased by 12.7% and outpatient visits for back pain are then five times greater [4]. In the United States, 80% of the population has complained of low back pain. This complaint is the second most common after headache. In fact, low back pain is the third most common cause of disability in the United States [5].

Low back pain myogenic is an unpleasant sensory and emotional experience in the area between the 12th thoracic vertebra to the lower part of the hip or anal canal [6], [7]. It may result in potential damage or tissue damage, such as vascular dermis, fascia, muscle, tendon, cartilage, bone, ligament, intra-articular meniscus, bursa [8].

Back pain disorder in this research is a part of post-pain rehabilitation**.** Before using the tool, developed in this research, the patient and the patient's family are advised to consult with a medical rehabilitation specialist, to ensure the patient's condition. This paper provides some examples of motion simulation for healing**.** Medical record data processing will be used in the input and output processes of the multimedia application. Rule based reasoning is used to process the input to produce an output in the form of the right therapeutic solution.

An expert system is a system that seeks to adopt human knowledge to computers, so that computers can solve problems as usually done by experts. The structure of the expert system is divided into two environment, the development environment and consulting environment [9]. Decision support system applications (Expert Systems) use data, provide an easy user interface (user friendly), and can incorporate the thoughts of decision makers. The user interface is one of the important supporting components in building a decision support system or expert system.

The expected objectives of this research are as follows:

1) Educate the public about low back pain myogenic
2) Provide information related to the choice of therapy for low back pain myogenic by patients and families

Application of multimedia-based aids produced in this research

1) Can be used as a means of information for the community.

2) Early detection of low back pain disorders in general.
3) In order to avoid more severe back problems.
4) Helping low back pain patients in general to do independent therapy at home or with the help of close family.

a. Low Back Pain Therapy

Back Exercise is an exercise that is used to restore the strength, endurance and flexibility of the back muscles [5]. Its purpose is to reduce body pressure on the facets and stretch the lumbar region muscles and correct body malformations. The exercise program includes everything about the dose of exercise, the frequency of exercise, the time of exercise, and other training principles [10], [11]. This training program is structured systematically, measurably, and adapted to the training objectives needed. Physical exercise requires a relatively long time to get optimal results. The results of physical exercise are not something that can be obtained instantly, in one or two week.

b. Multimedia Applications in the World of Health

Multimedia applications that are considered suitable for hospital needs are applications that can respond and provide information needed by patients, besides that patients can also choose the type of information they want, in this case that includes the above criteria is a CD interactive, as a medium for delivering information with an informative and attractive appearance. The provision of the information center provided must be really considered because the ease of obtaining information will provide understanding [12].

2. Methods

This Research used R&D method. Data was collected through observation, interview with medical rehabilitation specialists and physiotherapists. The result of collecting data used to produce the rule in the form of **if-then**. A decision table was created then converted into production rules [5].

The mechanism of the forward chaining system begins by entering a set of known facts into working memory, then matching these facts with the IF part of the IFTHEN rules. If there are facts that match the IF part, then the rule is executed. When a rule is executed, a new fact (the THEN part). Each time a match, starting from the top rule. Each rule can only be executed once. The matching process stops when there are no more rules that can be executed or have reached the goal or there are no more rules whose premise matches the known facts. The form of representation of Rule Based Reasoning is used because it has a certain amount of expert knowledge on a particular problem and the expert can solve the problem systematically and sequentially. A rule-based representation that has an **if** condition/premise then action/conclusion pattern in an expert table will provide benefits in various aspects, including easy modification, whether changing data, adding data or deleting data. In this case, **"if"** can be represented as symptoms felt by the patient and "**then**" in the form of solutions achieved.

The system design method used is based on Rule-based Reasoning. By using the step of modifying the source of knowledge from experts so that it can be implemented into a rule-based design. The knowledge is then grouped based on the diagnosis step, then tabulated and given a special code to make it easier to carry out the process of forming rules. The code will represent the implementation process. From the results of medical record data processing, knowledge can be obtained, from this knowledge, for the case of diagnosing muscle strength in this research, a decision tree can be used as can be seen in Figure1. The description of the decision tree is presented in Table 1.

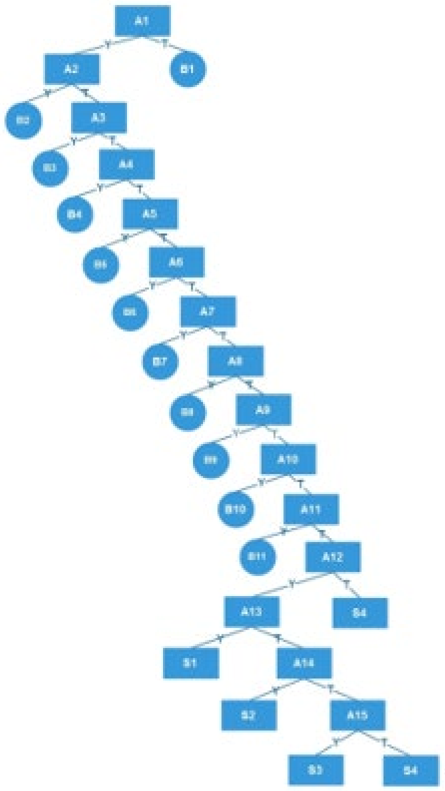


Figure 1. Decision Tree

Table 1. Decision Tree Description

No	Code	Information
1	A1	Does the patient have movement disorders?
2	B1	normal patient
3	A2	Does the patient have a fever above 38°C?
4	B2	Cannot be treated with this assistive model, because the patient has symptoms of high fever which may have more serious consequences. (consult doctor)
5	A3	Does the patient have difficulty urinating?
6	B3	Cannot be treated with this assistive model, because the patient has symptoms of difficulty urinating, which may have more serious consequences. (consult doctor)

No	Code	Information
7	A4	Does the patient experience symptoms of weight loss without knowing the cause?
8	B4	Cannot be treated with this assistive model, because there are symptoms of losing weight without knowing the cause. (consult doctor)
9	A5	Does the patient have back pain that doesn't go away after you lie down?
10	B5	Cannot be treated with this assistive model, because the patient experiences back pain symptoms that do not subside after the patient lies down. (consult doctor)
11	A6	Does the patient have chest pain?
12	B6	Cannot be treated with this assistive model, because the patient experiences chest pain. (consult doctor)
13	A7	Does the patient have pain in one or both legs, especially if the pain radiates down to the knee?
14	B7	Cannot be treated with this assistive device model, because the patient has symptoms of pain in one or both legs, especially if the pain radiates below the knee. (consult doctor)
15	A8	Does the patient have pain that gets worse at night?
16	B8	Cannot be treated with this assistive model, because the patient has pain symptoms that get worse at night. (consult doctor)
17	A9	Is the patient unable to hold urination and defecation?
18	B9	Cannot be treated with this assistive device model, because in the sand there are symptoms of not being able to hold urination and defecation. (consult doctor)
19	A10	Does the patient experience numbness in the genital area, buttocks, or back?
20	B10	Cannot be treated with this assistive model, because the patient has several symptoms of numbness in the genital area, buttocks, or back of the body. (consult doctor)
21	A11	Does the patient have swelling and redness on the back?
22	B11	Cannot be treated with this assistive model, because the patient has symptoms of swelling and redness on the back. (consult doctor)
23	A12	Does the patient have posture problems?
24	S1	There is a possibility that the patient's low back pain includes mild low back pain, one of the causes is due to kyphosis posture disorders, so the patient can heal with special movement therapy for kyphosis sufferers.
25	A13	Does the patient have kyphosis?
26	S2	There is a possibility that the patient's low back pain includes mild low back pain, one of the causes is due to a lordosis posture disorder, so the patient can heal with special movement therapy for lordosis sufferers.
27	A14	Does the patient have a lordotic disorder?
28	S3	There is a possibility that the patient's low back pain includes mild low back pain, one of the causes is due to scoliosis posture disorders, so the patient can heal with special movement therapy for scoliosis sufferers.
29	A15	Does the patient have scoliosis?
30	S4	There is a possibility that the patient's low back pain includes mild low back pain and the patient also does not have an abnormal posture so that the patient can heal with therapeutic movements.

a. Interface Implementation

In this application, the "intro" page, Figure 2, is the opening page before entering the "home" page, Figure 3, which contains the title "Dangerous Low Back Pain Therapy Application Multimedia-Based" then there is a "Start" button to start using the application

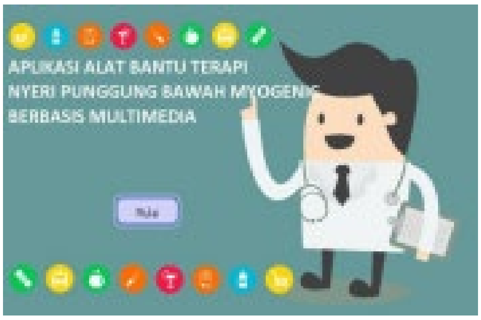


Figure 2. **Intro menu**

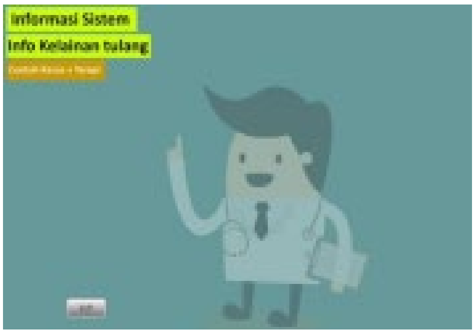
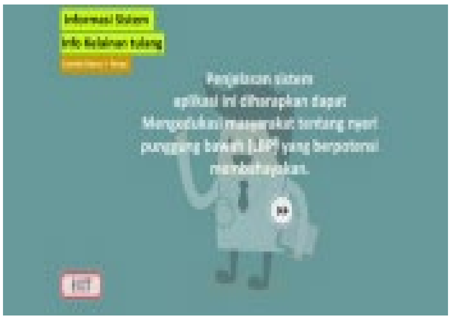


Figure 3. **Home menu**

The "home" page consists of three main menus, namely "system information", "bone abnormalities info", and "case & therapy examples". The "system information" button is used to move to the "system description" page. The "info bone disorder" page is used to move to the "muscle strength information" page which contains 3 kinds of pictures of bone disorders and their descriptions. The "case & therapy example" button is the main module of the expert system created in this study.

b. Information Menu

The "information" page, Figure 4, will appear when the user selects the "system information" menu on the "home" page. This menu provides information of the purpose of this application.

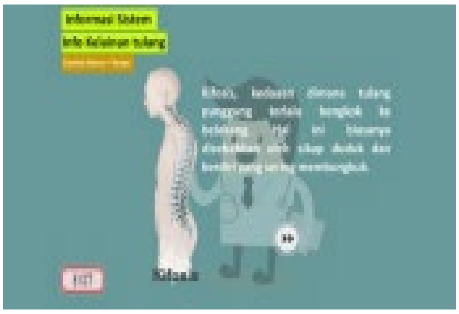


Penjelasan sistem
aplikasi ini diharapkan dapat
Mengedukasi masyarakat tentang nyeri
punggung bawah (LBP) yang berpotensi
membahayakan.

Figure 4. Information menu

c. Menu of the Bone Abnormalities Info

The "Info Bone Disorders" page will appear when the user selects the menu "Info Bone Abnormalities" on the "home" page. This menu provides the information of bone disorders, namely Kyphosis, Lordosis and Ecoliosis along with their descriptions. The interface for the "Bone Abnormalities information" page can be seen in Figure 5.



Kifosis, kelainan dimana tulang punggung terlalu bengkok ke belakang. Hal ini biasanya disebabkan oleh sikap duduk dan berdiri yang sering membungkuk.

Figure 5. Bone disorder information page

d. Expert System Testing The expert

The application is expected to produce output in accordance with the decision tree. Testing the expert system was performed by testing all existing structured questions. The structured questions include:

1) Does the patient have limb disorders?
   If yes then proceed to the next structured question (valid)
   If no then the patient is declared normal (valid)
2) Does the patient have a fever above 38°C?
   If yes, then it cannot be treated with this assistive model, because the patient has symptoms of high fever which may have more serious consequences. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
3) Does the patient have difficulty urinating?
   If yes, then it cannot be treated with this assistive device model, because the patient has symptoms of difficulty urinating, which may have more serious consequences. consult a doctor (valid) If not then proceed to the next structured question (valid)
4) Does the patient experience weight loss without knowing the cause?
   If yes, then it cannot be treated with this assistive model, because there are symptoms of losing weight without knowing the cause. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
5) Does the patient have back pain that does not subside after you lie down?
   If yes, then it cannot be treated with this assistive model, because the patient experiences back pain symptoms that do not subside after the patient lies down. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
6) Does the patient experience chest pain?
   If yes, then it cannot be treated with this assistive model, because the patient experiences chest pain. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
7) Does the patient experience pain in one or both legs, especially if the pain radiates down the knee?
   If yes, then it cannot be treated with this assistive model, because the patient has symptoms of pain in one or both legs, especially if the pain radiates below the knee. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
8) Does the patient experience pain that gets worse at night?
   If yes, then it cannot be treated with this assistive model, because the patient has pain symptoms that get worse at night. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
9) Does the patient experience inability to hold urine and bowel movements?
   If yes, then it cannot be treated with this assistive device model, because in the sand there are symptoms of not being able to hold urination and defecation. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
10) Does the patient experience numbness in the genital area, buttocks, or back of the body?
   If yes, then it cannot be treated with this assistive model, because the patient has several symptoms of numbness in the genital area, buttocks, or back of the body. consult a doctor (valid)
   If not then proceed to the next structured question (valid)
11) Does the patient have swelling and redness on the back?
   If yes, then it cannot be treated with this assistive model, because the patient has symptoms of swelling and redness on the back. consult a doctor (valid)
   If not then proceed to the next structured question (valid)

12) Does the patient have posture problems?
    If yes, then the question will arise which posture disorder is the patient suffering from?
    If you select the kyphosis image, a summary of the diagnosis information appears showing kyphosis, lordosis, and scoliosis posture disorders resulting in LBP, one of the causes is a kyphosis patient's posture disorder and can be cured with physiotherapy particular patient kyphosis(valid)
    If you select an image lordosis appears the summary information for diagnosis showed impaired posture kyphosis, lordosis and scoliosis resulting LBP, one possible cause is due to interference with the posture of the patient's body lordosis and could do healing with physiotherapy particular patient lordosis(valid)
    If you select the scoliosis image, a summary of the diagnosis information appears showing that scoliosis posture disorders result in LBP, one of the causes is due to a scoliosis patient's posture disorder and can be cured with special physiotherapy for scoliosis sufferers (valid) If not, a summary information appears diagnosticshowing mild LBP (valid)

The application was evaluated through Focus Group Discussion (FGD). FGD is a systematic effort in collecting data and information [13], modeling for performance measurement of computing power and system performance factors. The performance factor is measured by the user depending on the system and the complexity of the interface. The model is divided into three parts, namely Functionality, Flexibility, and Productivity [14]. FGD is done by gathering several users who have the ability to diagnose muscle strength and therapeutic solutions in cases oflow back pain myogenic. Then the user is given an explanation about the system created and also given the opportunity to use the system. Then users are asked to tell their experiences using the system and give an assessment of the system they have used. The assessed components of the system assessment are functionality, flexibility, and productivity. The functionality component consists of reliability, consistency, quality and material modeling. Flexibility components consist of display, language, creativity, convenience, grouping of actions. While the productivity component consists of information, material suitability, level of truth, completeness, and effectiveness. In addition to User testing which consists of 14 sub-indicators, there is an additional assessment referring to the Usability theory according to ISO 9126 consisting of 4 sub-indicators (Understandibility, Learnability, Operability, Attractiveness). There were ten FGD respondents consisting of physiotherapists who acted as users. Physiotherapists have been invited to try the application when the discussion forum has not started, in that way it is hoped that users can provide responses to the expert system application that is being developed. Qualitative data in the questionnaire is converted into quantitative data by assigning a value to each answer. The value given is the value for positive statements with the rules STS (strongly disagree) = 1, TS (disagree) = 2, N (neutral) = 3, S (agree) = 4, and SS (strongly agree) = 5.

3. Result

The participants of the FGD are physiotherapists as users. The FGD produced the assessment of the performance of the application. Table 2 presents the results of FGD assessment.

Table 2. Results of FGD assessment

No	Indicator	Sub indicator	SD	D	N	A	SA	Total
1	Application flexibility	Appearance			2	9		42
2		Language			1	10		43
3		Creativity			3	4	4	45
4		Ease of use			2	6	3	45
5		Action grouping			2	7	2	44
6	Application functionality	Reliability			2	6	3	45
7		Consistency			2	7	2	44
8		Material model			3	6	2	43
9		Quality			2	5	4	46
10	Application productivity	Suitability of material			3	7	1	42
11		Information delivery			2	6	3	45
12		Truthfulness			3	6	2	43
13		Information completeness			3	7	1	42
14		Effectiveness			1	7	3	46
		Usability compliance to ISO standard						
1	ISO compliance usability	Understand-ability			1	10		43
2		Learnability			1	10		43
3		Operability			1	8	2	45
4		Attractiveness			1	7	3	46

The total average user of acceptance of the system is calculated using the formula (1).

$$P = \frac{\sum weight \times value\ of\ respondent}{5 \times \sum respondent}$$

$$\acute{P} = \frac{\sum P}{\sum Indicator} \qquad (1)$$

where:

P	=	value of each indicator
$\acute{P}$	=	Average revenue
$\sum$	=	Total Value
5 $\sum$respondent	=	55
$\sum$ Indicator	=	Total 18 indicators

Our results shows that

$$\acute{P} = \frac{14.40}{18} = 0.80$$

which suggest that the average acceptance is 80%.

The hypothesis given in this research is "The performance of the expert system is acceptable to the user" with an acceptance value of 80%. Because the average total value of user acceptance is greater than the total value of acceptance, from the calculated results of 0.80, it can be concluded that the application can be accepted by the user (physiotherapist). According to 10 physiotherapists and 1 doctor the performance of the application is good.

Indicators of Improvement.

Summary of the results of the discussion through the FGD of 10 Physiotherapists and 1 Doctor, can be divided into two points about the application and about how the application will be applied to patients.

In term of application, some physiotherapists respond to slower movements, but there are other physiotherapists who respond to slow movements according to the patient's condition, so examples of movements in this model of assistive devices do not have to be slowed down (very slow). Especially For Muscle Strength 4 which carries the load, it is better to slow down a little. The application should be added with sound to explain each movement, such as what kind of sitting position and what kind of movement, but there are other physiotherapists who argue that medical language in cases of decreased muscle strength is difficult to translate into layman's language so that with examples of movements appropriate and clear is sufficient.

In order to help patients performing self-therapy, Physiotherapists request that the results of the compilation of the .swf model of this tool to be copied in the form of a CD (compact disk) which will be tested on their patients. The webbase version should be provided so that patients can use this model of assistive devices via online.

4. Discussion

The result of research show that the application using forward chaining can help the user to detect low back pain. This is in accordance with the research conducted by Husin on an expert system for detecting diseases based on complaints of urination using forward chaining. He revealed that the application is very helpful for user because it can be useful for diagnosing types of diseases based on complaints of urination, how to treat it, explanations the disease, and is also equipped with the cause of the disease experienced by patient based on the various his symptoms experiences. Forward Chaining can be used as an inference engine in cases of myogenic low back pain, because the reasoning used in this research must start from the facts section first, to reach the hypothesis. By using an expert system in application, it can substitute the ability of experts in health problems based on complaints of urination, furthermore patients can use them to solve the problems they experience independently without having to visit or present an expert directly so that he can conclude a diagnosis, Causes, treatment and prevention his disease. The important one is that this system is built based on the knowledge of experts who are mastering of a disease so that it can be trusted for certainty of diagnosis, causes, as well as treatment and prevention that must be done [15]. A similar research was also conducted by Ariyawan [16]. In her report she declares that with the application of an expert system the user can analyze the disease, and can find out the treatment quickly and precisely without inviting experts, can gain the time and costs. With this system, user may consult with an expert of the disease.

We can conclude that expert systems help humans to solve problems quickly without an expert to diagnose, provide therapy and advice on treatment and prevention of a disease, besides this expert system is not expensive, save money and time

5. Conclusion

Rule-Based Reasoning can be used as an expert system knowledge Base, and Forward Chaining as an inference engine, for application of myogenic low back pain therapy. The expert system application for myogenic low back pain can be accepted by the user. This can be seen from the acceptance value which results that "the ten physiotherapists and one doctor consider the application performance to be good" with an acceptance value of 0.80 or 80%. According to the physiotherapist, this assistive device model is likely to increase the intensity of therapy because it can be carried out by the patient's family independently.

6. Acknowledgement

The authors would like to thanks IQT Unida for supporting this research.

Reference

[1] J. Hartvigsen *et al.*, "What low back pain is and why we need to pay attention," *Lancet*, vol. 391,

no. 10137, pp. 2356–2367, 2018.

[2] S. T. Illes, “Low back pain: when and what to do,” *Orv. Hetil.*, vol. 156, no. 33, pp. 1315–1320, 2015.

[3] I. Urits *et al.*, “Low back pain, a comprehensive review: pathophysiology, diagnosis, and treatment,” *Curr. Pain Headache Rep.*, vol. 23, no. 3, pp. 1–10, 2019.

[4] B. Eleanor and A. Graham, *A simple guide to back pain*. Jakarta: Erlangga, 2007.

[5] L. Williams and Wilkins, *Pain Management Made Incredibly Easy*, 1st ed. Lippincott Williams & Wilkins, 2003.

[6] D. Wardianti and W. Wahyuni, “Physiotherapy Management of William Flexion Exercise for Pain Reduction in Low Back Pain Myogenic: Case Study,” in *Academic Physiotherapy Conference Proceeding*, 2021.

[7] M. Rabey *et al.*, “STarT Back Tool risk stratification is associated with changes in movement profile and sensory discrimination in low back pain: A study of 290 patients,” *Eur. J. Pain*, vol. 23, no. 4, pp. 823–834, 2019.

[8] M. J Paliyama, “Perbandingan Efek Terapi Arus Interferensi Dengan Tens Dalam Pengurangan Nyeri Pada Penderita Nyeri Punggung Bawah Musulosekletal,” Diponegoro, 2004.

[9] S. Kusuma Dewi, *Artificial Intelligence (Teknik Dan Aplikasinya)*, Ke-1. Yogyakarta: Graha Ilmu, 2003.

[10] P. J. Owen *et al.*, “Which specific modes of exercise training are most effective for treating low back pain? Network meta-analysis,” *Br. J. Sports Med.*, vol. 54, no. 21, pp. 1279–1287, 2020.

[11] J. A. Hayden *et al.*, “Some types of exercise are more effective than others in people with chronic low back pain: a network meta-analysis,” *J. Physiother.*, vol. 67, no. 4, pp. 252–262, 2021.

[12] Suyami, “Interaktif Media Pembelajaran Untuk Perawatan Bayi Berbasis Multimedia Di Rumah Sakit,” Universitas Indonesia, 2012.

[13] Irwanto, *Focused group discussion (FGD) : sebuah pengantar praktis*. Jakarta: Yayasan Obor Indonesia, 2006.

[14] C. B. Macknight and S. Balagopalan, “An Evaluation Tool for Measuring Authoring System Performance,” *Commun. ACM*, vol. 32, no. 10, 1989.

[15] A. Husin, M. P. Faren, and U. I. Indragiri, “Sistem Pakar Pendeteksi Penyakit Berdaasarkan Keluhan Buang Air Kecil Menggunakan Metode Forward Chaining,” *J. IPTEK Terap.*, vol. 12, no. 4, 2018.

[16] M. Dwi Ariyawan, “Aplikasi Sistem Pakar Diagnosa Penyakit Umum Pada Manusia Berbasis Web,” *J. Ilmu Komput. Udayana*, vol. 7, no. 2, 2018.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Recommendation System to Propose Final Project Supervisor using Cosine Similarity Matrix

Zulfa Fajrul Falah[1], Fajar Suryawan*[2]
[1]Department of Informatics
[2] Department of Electrical Engineering
Universitas Muhammadiyah Surakarta
Central Java 57162, Indonesia
*Fajar.Suryawan@ums.ac.id

Abstract-The selection of a supervisor is an important thing and one of the determinants of whether or not a student's final project research is successful. At the location of this research, students select a supervisor by considering his academic records and recommendations from classmates or seniors. Words of mouth dominate their motivation, and many students do not have a basis for their choice. Selection of the best-fit supervisor significantly impacts a student's progression. Students will be more enthusiastic about doing the final project and may get facilitation in their research because the topics of the student projects match the supervisor's interests and ongoing work. This study aims to make a recommendation system that suggests a supervisor for a student. The student fills in the title, abstract, and keywords of his proposal. The system gives suggestions to prospective supervisors by calculating the similarity of the data with titles, abstracts, and keywords of published articles found in Google Scholar. The recommendation system uses the content-based filtering method to produce a list of recommendations. The cosine similarity algorithm calculates how similar the topic proposed by students is to the lecturers' interests. In building a website-based recommendation system, the authors use Django web framework as the backend and ReactJs as the frontend. The application succeeds in suggesting final project supervisors that match lecturers' interests and expertise with students' proposals.

Keywords: cosine similarity, recommendation system, web scraping, content-based filtering

Article info: *submitted November 9, 2021, revised May 4, 2022, accepted September 16, 2022*

1. Introduction

A final project is a common requirement for students to graduate from a university. This final project is typically the ultimate scientific work of a student in completing his or her undergraduate education study period. In the final project, students are accompanied by a supervisor. It is the supervising lecturer who will become a partner in collaboration between students and lecturers in carrying out the research that has been submitted. The supervising lecturer also functions as a facilitator for students if students experience difficulties or doubts in the research process. The supervising lecturer must also master the field that is in accordance with the topic taken by the student so that the research results are maximized, therefore the role and suitability of the research field of a supervisor is vitally important.

In the department where this research takes place, the selection of supervisors is mostly done manually and independently by students. Students choose their supervisors directly when the study planning phase takes place. The selection of supervisors is roughly based on personal knowledge related to the specialization of lecturers and is also based on research carried out by students themselves with minimal data sources, some of which are sourced from classmates or from seniors who have graduated. There are even students who choose a supervisor without a special reason regardless of whether the lecturer matches their interests or the topic they are going to propose.

From the problems above, the authors see the urgency of building a recommendation system to help students determine supervisors. A recommendation system serves to sort through large amounts of data to identify user interests and make it easier to find information and form decisions [1]. A recommendation system is also an information filtering system used to predict the rating or preference that will be given to a user on an item such as music, books, movies, and documents. The recommendation system model can be built from the characteristics of an item (content-based filtering) or with a user environment approach (collaborative filtering approaches)[2] which will also be used in this study.

Content-based filtering works using an item's feature to recommend other items that are similar to what the user likes. It is also one of the most successful recommendation techniques, which is based on correlation between content. It uses item information represented as attributes to calculate similarity between items [3]. However, this strategy also has weaknesses, one of which is that this method cannot produce appropriate recommendations if the content analyzed for an item does not contain information suitable for categorization [4]. There are several methods that can be used to calculate similarity between content such as Euclidean distance, cosine similarity and Manhattan distance. Research conducted by Fathin in 2019 which compared the results of calculations between cosine similarity and Euclidean distance showed similarity values and had the same level of accuracy [16]. In this study, the method used to compare the similarity between content using cosine similarity.

There have been several studies conducted on the topic of selecting a final project supervisor. One of them is done by Asrul in 2018 who used the Analytical Hierarchy Process (AHP) method to build a decision support system for selecting supervisors. In the AHP method, the weighting of the criteria is carried out by experts, which is very subjective because the scoring of the criteria depends on each expert [17]. There is another research conducted at the Department of Computer Science/Informatics, Faculty of Science and Mathematics, Diponegoro University regarding the supervisory lecturer selection system, in this study using the Vector Space Model (VSM) as a method of matching the strings of student research titles and research that has been carried out by lecturers [18].

This study uses data from lecturers' publications, the abstracts of which have been published on the Google Scholar page. Data retrieval is done by scraping each lecturer's Google Scholar profile page. This data will be the knowledge base in building a recommendation system model which will then produce information in determining the appropriate supervisor for students. Data taken from Google Scholar includes titles, abstracts and keywords from lecturer publications. To the best of the authors' knowledge, this research is the first research in Indonesia that uses data from Google Scholar to build a recommendation system.

The purpose of this recommendation system is to produce a list of final project supervisors that are in accordance with the topics proposed by students, and can make it easier for students to choose suitable supervisors based on the proposed topic.

2. Methods

In conducting this research, we first collect publication data from Google Scholar, followed by the initial preparation of the data. The data that is ready is then fed to the main algorithm. A website was built to embed the recommendation system, which is the main interface between the system and the user. The details are as follows.

a. Data Collection

The data was taken by doing web scraping on the Google Scholar profile page for each prospective supervisor.

Web scraping is the process of retrieving a semi-structured document from the internet, which is generally in the form of web pages in a markup language such as HTML or XHTML, then analyzing the document to retrieve certain data that is used for several purposes. In this study, the author uses Python to do web scraping with the help of third party libraries such as Selenium, BeautifulSoup, Requests, and CSV, which then the data from the web scraping is saved into a CSV file for processing to the next stage. This data set has 603 rows and 4 attributes as shown in Table 1.

Table 1. Description of the supervisor's research data attributes

Number	Attribute	Information
1	Name	Supervisor's name
2	Title	Research title
3	Abstract	Research abstract
4	Keyword	Research keywords

b. Data Preprocessing

Data preprocessing is one of the main stages in the knowledge discovery process, although this stage is not as popular as other stages such as data mining, data preprocessing actually involves more time and effort in the entire data analysis process [5]. Data from web scraping is generally in the form of raw data, such as there are still HTML elements that are accidentally taken. There are also non-alphanumeric characters and incomplete rows. The presence of data inconsistencies and noise contained in the dataset can affect the performance of the machine [6]. Data preprocessing also serves to improve the data format and to clean interference and noise from the raw data [7]. The following are some of the data preprocessing steps performed here.

Punctuation Removal is the process of removing characters that are not included in the letters of the alphabet because of other characters such as punctuation marks and non-alphanumeric characters (except spaces) such as !"#$%&'()*+,-./:;<=>? @[\]^_`{|}~ can affect the accuracy of the analysis. In this study the author uses python to clean punctuation and non-alphanumeric characters.

Case Folding, to change all uppercase letters in the document to lowercase letters.

Tokenization, which is shown in Table 2, is a process to divide or break texts in the form of sentences, paragraphs, or documents into tokens. In linguistics, token is the smallest unit in a text. This token will help to understand the context and for the development of the NLP model. Tokenization will also be useful for interpreting the meaning of the text by analyzing the order of words.

Table 2. Tokenization Process

Original Text	After Tokenization
'rancang bangun aplikasi pembelajaran hadist'	'rancang', 'bangun', 'aplikasi', 'pembelajaran', 'hadist'

Stop-words removal, which is the process of taking only important words. In general, in a text there are words that commonly appear such as prepositions, conjunctions, pronouns, and others. These words do not provide much information in the text. Removing less informative words from the text can give the engine more focus to process only the words that are important. In other words, removing less informative words will not have a negative impact on the vector, and will be more efficient from the processing side.

Word vectorization is a methodology in NLP to map words or phrases from vocabulary into a vector of real numbers and to determine word predictions or word/semantic similarity. By doing word vectorization on the supervisor's research text, the machine can process it as a vector of real numbers no longer as a collection of words. To represent a document, it is necessary to convert it into a vector form, so that it can be processed by machines [8]. The use of Term Frequency (TF) and Inverse Document Frequency (IDF) schemes has proven to be a powerful algorithm in processing text data or other purposes [9]. TF-IDF uses word frequency and document frequency to produce weighted words that are used to represent documents [8]. Terms of word frequency or document frequency in the TF-IDF approach are usually used to weigh each word in a text document according to its uniqueness [10].

c. Data Processing: Cosine similarity

Recommendation systems have several algorithms such as content-based filtering, collaborative filtering and a combination of the two [1], [11]. In this study, the author uses a content-based filtering algorithm as a method to determine the results of recommendations from supervisors. The content-base used is the text of titles, abstracts, and lecturers' research keywords. Recommendation systems using this technique have similarities with other techniques in terms of item descriptions, user profiles, and techniques for comparing profiles with items to identify the most suitable recommendation results for users [2]. One of the methods used to measure the closeness between texts is the cosine similarity method, which will be used here.

Cosine Similarity is a method used to measure the similarity between two text documents which are considered as vectors [12]. Cosine similarity is also a matrix that is widely implemented in information retrieval and is often applied in comparing the similarity of two texts (sentences, paragraphs or entire documents), the similarity between two documents is obtained by calculating the cosine value of the vectors between documents [13]. In this study, the method to calculate the similarity between the lecturer's research and the topic that will be proposed by students is by comparing the similarity of the title, abstract, keywords of potential supervisors' research with those of the student-submitted proposals. The value of cosine similarity between vectors can be calculated by the following equation:

$$cos\ a = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} \qquad (1)$$

where:

A = Vector A, is the lecturer's research vector

B = Vector B, is a vector of topics proposed by students

A • B = dot product between vector A and vector B

|A| = vector length A

|B| = vector length B

|A||B| = cross product between |A| and |B|

Further explanation and illustration of this cosine similarity calculation can be seen in sections 3.A and 3.B.

A. Implementation and Interface

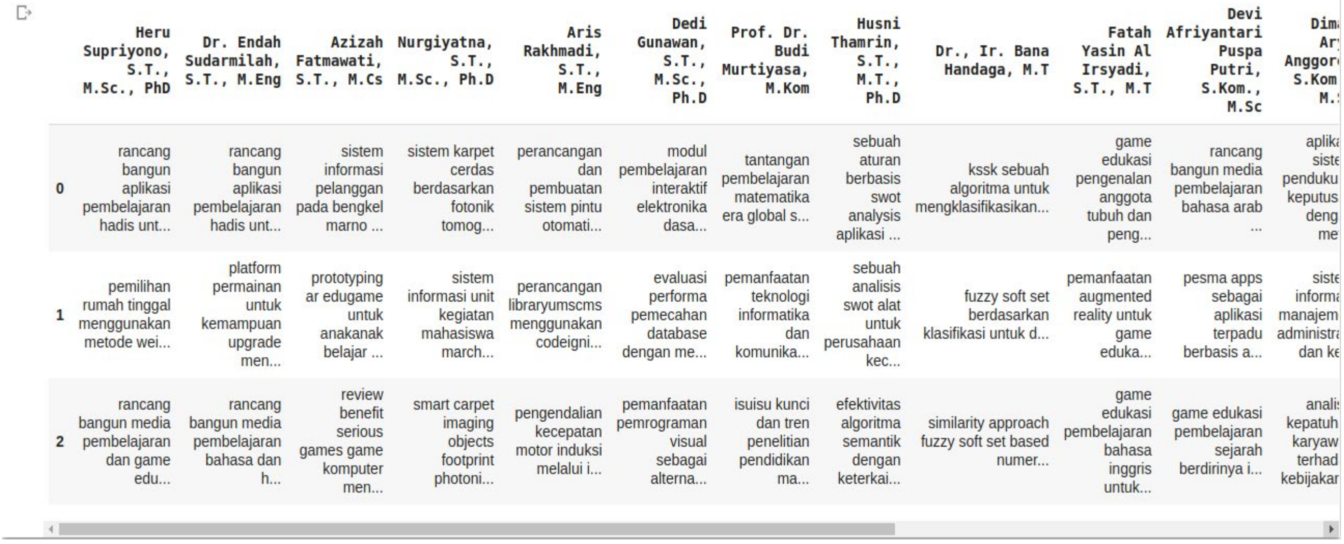
A website is created as an interface between the recommendation algorithm and users, namely students. The system is built using the Python programming language, and the interface is built using Django, a web framework built on top of Python. Here Django is used as the back-end of the web system, which is in charge of providing the data needed by the user. Testing has also been carried out on this information system [15]. A more detailed description can be found in the next section.

3. Development Process and Results

This section will detail the stages of development carried out and their results. The first is preprocessing, where the data is cleaned and formed into tokens, then the clean data is fed to the recommendation algorithm. The next stage is the implementation of this recommendation system into a web framework.

a. Preprocessing Stage

A total of 603 web scraping datasets from the Google Scholar page have 4 attributes. Then the dataset will be reduced to 2 attributes as shown in Figure 1, with the aim of simplifying the next preprocessing process. Before being used as a vector and measuring its proximity, the data is first cleaned of noise. Furthermore, the data will go through the tokenization process. Data in the form of long sentences will be broken down into words or into a token. Then after the data becomes a token, the data will enter the stopword process. Words that appear frequently in the document, and those words are listed in the stoplist, will be removed. For example the words 'at', 'and', 'to'.



	Heru Supriyono, S.T., M.Sc., PhD	Dr. Endah Sudarmilah, S.T., M.Eng	Azizah Fatmawati, S.T., M.Cs	Nurgiyatna, S.T., M.Sc., Ph.D	Aris Rakhmadi, S.T., M.Eng	Dedi Gunawan, S.T., M.Sc., Ph.D	Prof. Dr. Budi Murtiyasa, M.Kom	Husni Thamrin, S.T., M.T., Ph.D	Dr., Ir. Bana Handaga, M.T	Fatah Yasin Al Irsyadi, S.T., M.T	Devi Afriyantari Puspa Putri, S.Kom., M.Sc	Dim Ar Anggor S.Kom M.:
0	rancang bangun aplikasi pembelajaran hadis unt...	rancang bangun aplikasi pembelajaran hadis unt...	sistem informasi pelanggan pada bengkel marno ...	sistem karpet cerdas berdasarkan fotonik tomog...	perancangan dan pembuatan sistem pintu otomati...	modul pembelajaran interaktif elektronika dasa...	tantangan pembelajaran matematika era global s...	sebuah aturan berbasis swot analysis aplikasi ...	kssk sebuah algoritma untuk mengklasifikasikan...	game edukasi pengenalan anggota tubuh dan peng...	rancang bangun media pembelajaran bahasa arab ...	aplik siste penduku keputus deng me
1	pemilihan rumah tinggal menggunakan metode wei...	platform permainan untuk kemampuan upgrade men...	prototyping ar edugame untuk anakanak belajar ...	sistem informasi unit kegiatan mahasiswa march...	perancangan libraryumscms menggunakan codeigni...	evaluasi performa pemecahan database dengan me...	pemanfaatan teknologi informatika dan komunika...	sebuah analisis swot alat untuk perusahaan kec...	fuzzy soft set berdasarkan klasifikasi untuk d...	pemanfaatan augmented reality untuk game eduka...	pesma apps sebagai aplikasi terpadu berbasis a...	siste informa manajem administra dan ke
2	rancang bangun media pembelajaran dan game edu...	rancang bangun media pembelajaran bahasa dan h...	review benefit serious games game komputer men...	smart carpet imaging objects footprint photoni...	pengendalian kecepatan motor induksi melalui i...	pemanfaatan pemrograman visual sebagai alterna...	isuisu kunci dan tren penelitian pendidikan ma...	efektivitas algoritma semantik dengan keterkai...	similarity approach fuzzy soft set based numer...	game edukasi pembelajaran bahasa inggris untuk...	game edukasi pembelajaran sejarah berdirinya i...	anali kepatuh karyaw terhad kebijakar

Figure 1. Dataset before and after preprocessing stage

After the data undergoes several processes until the data becomes a token, the next step is the data will be converted into a vector using the TF-IDF method with an n-gram range of 1-2 words. In linguistics and computational probability, an n-gram is a contiguous sequence of n items from a text. The N-gram will give the probability of the next word that can help in understanding the meaning of a text. The essence of this method is to calculate the TF and IDF values of each keyword against each document. The TF-IDF value can be calculated using the equation:

$$w_{i,j} = \ tf_{i,j} \times ln\ ln\ \left(\frac{N+1}{df_i+1}\right) + 1 \tag{2}$$

Note:

$tf_{i,j}$ = Many *i*-words on document *j*

N = Total documents

df_i = Many documents contain the word *i*

There are 2 abstracts as follows:

d1 = "Design and build a Hadith learning application"

d2 = "child-based learning application"

Suppose we want to calculate the weight of the word "child" in abstract d2, because the word "child" appears once in abstract d2, the calculation of the weight of the word "child" becomes:

$$w_{anak,d2} = ln\ ln\ \left(\frac{N+1}{df_i+1}\right) + 1$$

$$= 1 \times \left[ln\ ln\ \left(\frac{2+1}{1+1}\right) + 1\right]$$

$$= 1 \times 1.40$$

$$= 1.40 \tag{3}$$

After all the words are weighted, the results are as shown in table 3, so that the abstract vector d1 is [0.0, 0.0, 1.0, 1.0, 1.4, 1.4, 0.0, 1.4, 1.0, 0.0, 1.4, 1.4, 1.4, 0.0, 0.0] and the abstract vector d2 is [1.4, 1.4, 1.0, 1.0, 0.0, 0.0, 1.4, 0.0, 1.0, 1.4, 0.0, 0.0, 0.0, 1.4, 1.4]. After converting abstract d1 and abstract d2 into vectors, the next step is to measure the closeness between vectors which is discussed in more detail in subsection 3.B.

Table 3. TF-IDF calculation results

Term	tf		df	tf-idf $tf_{i,j} \times ln(\frac{N+1}{df_i+1}) + 1$	
	d1	d2		d1	d2
anak	0	1	1	0.00	1.40
anak usia	0	1	1	0.00	1.40
aplikasi	1	1	2	1.0	1.0
aplikasi pembelajaran	1	1	2	1.0	1.0
bangun	1	0	1	1.40	0.00
bangun aplikasi	1	0	1	1.40	0.00
berbasis	0	1	1	0.00	1.40
hadist	1	0	1	1.40	0.00
pembelajaran	1	1	2	1.0	1.0
pembelajaran anak	0	1	1	0.00	1.40
pembelajaran hadist	1	0	1	1.40	0.00
rancang	1	0	1	1.40	0.00
rancang bangun	1	0	1	1.40	0.00
usia	0	1	1	0.00	1.40
usia berbasis	0	1	1	0.00	1.40

b. Model Recommendation System

Data that has become a vector will be measured for its proximity to the input vector of the title, abstract and keywords of the students. In this study, the measurement of proximity between vectors is calculated using the cosine similarity method, which is the method used to measure the similarity between vectors. In the previous stage, the vector values for each lecturer's research have been represented by vector A (taken from column d1 in Table 3) and the value of each student input will be represented by vector B (taken from column d2 in Table 3).

vector A = [0.0, 0.0, 1.0, 1.0, 1.4, 1.4, 0.0, 1.4, 1.0, 0.0, 1.4, 1.4, 1.4, 0.0, 0.0]

vector B = [1.4, 1.4, 1.0, 1.0, 0.0, 0.0, 1.4, 0.0, 1.0, 1.4, 0.0, 0.0, 0.0, 1.4, 1.4]

The two vectors are processed with the cosine similarity equation:

$$\cos a = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad (4)$$

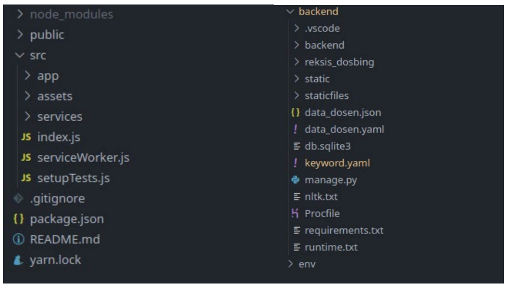
$$\begin{aligned} &= (0 + 0 + 1 + 1 + 0 + 0 + 0 + 0 + 1 + 0 + 0 \\ &\quad + 0 + 0 + 0 \\ &\quad + 0)/(\sqrt{(14.75)} \times \sqrt{(14.75)}) \\ &= \frac{3}{14.75} \\ &= 0.20 \end{aligned}$$

The measurement results of these vectors will be sorted based on their cosine similarity values. If the cosine similarity value is close to 1, then the vector has a tendency to be similar to the student input vector, and if the cosine similarity value is close to 0, then the vector has a tendency not to be similar to the student input vector. After getting the sorting results, the next step is to wrap all the cosine similarity calculation processes and sorting processes into a python class model which will later be installed into the website system.

c. System Implementation

The website system is designed as an interface for students, as well as being used as an implementation of a recommendation system to select project supervisors. The flow of the system runs in one direction starting from students entering the title, abstract and keywords. Then the system will perform computations to produce suitable recommendation lecturers based on input from students.

The website system is built using Django as the back-end and ReactJs as the front-end. Inside Django there is a project directory structure which as shown in Figure 2, each directory has its own function, the backend directory as the base project which contains the config and settings of the Django apps, then in the reksis_back-end directory functions as apps that will handle requests- requests and data processing from the front-end. Likewise in ReactJs there is also a directory structure that has its own function, the node_modules directory serves as a place to store packages needed by ReactJs such as bootstrap, multiple select, redux, etc. Then the public directory is used to store assets such as images, icons, and html files, the last is the src directory, in this directory there are javascript files that are useful for handling the components needed in making the user interface.



```
> node_modules                  ∨ backend
> public                          > .vscode
∨ src                             > backend
  > app                           > reksis_dosbing
  > assets                        > static
  > services                      > staticfiles
  JS index.js                     {} data_dosen.json
  JS serviceWorker.js             ! data_dosen.yaml
  JS setupTests.js                ☰ db.sqlite3
◆ .gitignore                      ! keyword.yaml
{} package.json                   manage.py
ⓘ README.md                       ☰ nltk.txt
yarn.lock                         Procfile
                                  ☰ requirements.txt
                                  ☰ runtime.txt
                                > env
```

Figure 2. ReactJs and Django directory structure

In Tables 4 and 5 there are 6 endpoints consisting of 2 POST methods and 4 GET methods, each endpoint has its own function, the rest-auth/google endpoint serves as a path to use OAuth2 google authentication to enter the website. Then the api/keyword endpoint serves to provide keywords that will be used by students, there are 2466 keywords that come from the research of the supervisor on the Google Scholar page. Furthermore, the api/rexis endpoint serves as a pathway to process data input from students, which will then be forwarded to the recommendation model in the back-end system and will be returned with the recommendation results.

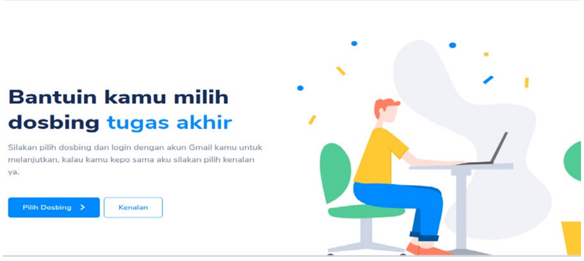
Table 4. List of endpoints on back-end

Method	Endpoint	Information
GET	api/keyword	Provide keyword data
GET	api/dosen	Provide supervisor lecturer data
POST	api/reksis	Perform data processing and provide recommendation data
POST	api/auth/google	Sign in with OAuth2 Google
GET	api/auth/logout	Log out of the system

Table 5. List of routes on the front-end

Method	Route	Information
GET	/	Show main page
GET	/reksis	Filling in data by the user and displaying recommendation results
GET	/dosbing	Displays a list of supervisors
GET	/about	Showing the about page

Figure 3 is the main page when the website is accessed by students. This page is accessed using route/, on this page there will be two buttons, the "select dosbing" button and the "acquaintance" button. the "select dosbing" button will then be redirected to google's OAuth2 system, to authenticate.

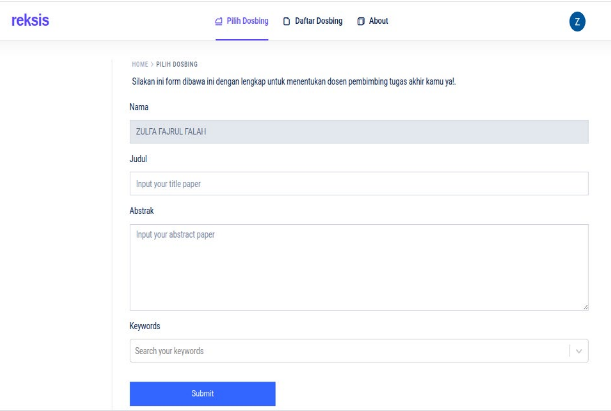


Bantuin kamu milih dosbing tugas akhir

Silakan pilih dosbing dan login dengan akun Gmail kamu untuk melanjutkan, kalau kamu kepo sama aku silakan pilih kenalan ya.

Figure 3. Website main page

The route/resis function is to handle when students successfully authenticate, as well as handle students in filling out the title, abstract, research keyword forms as shown in Figure 4. In this route students will also get results from the recommendation system. The results of this recommendation system can be seen in Figure 5.



HOME > PILIH DOSBING

Silakan ini form dibawa ini dengan lengkap untuk menentukan dosen pembimbing tugas akhir kamu ya!.

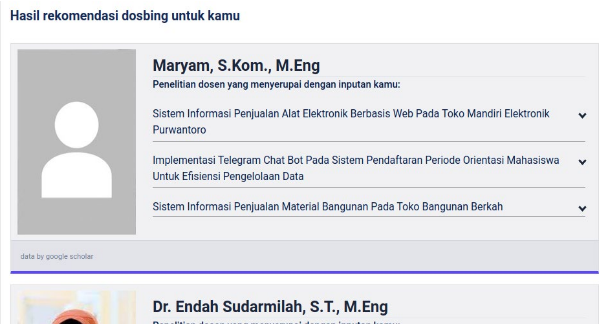
Nama: ZULFA FAJRUL FALAH

Judul: Input your title paper

Abstrak: Input your abstract paper

Keywords: Search your keywords

Figure 4. Input page title, abstract and keywords


Hasil rekomendasi dosbing untuk kamu

Maryam, S.Kom., M.Eng

Penelitian dosen yang menyerupai dengan inputan kamu:

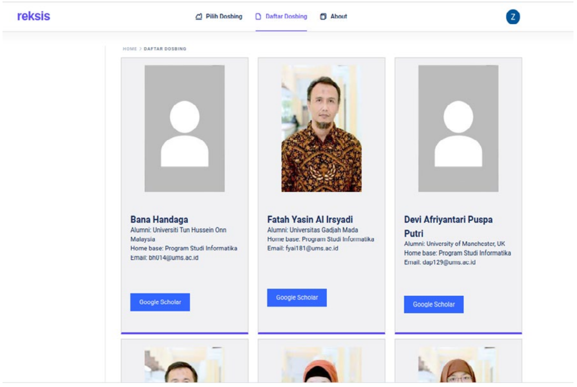
- Sistem Informasi Penjualan Alat Elektronik Berbasis Web Pada Toko Mandiri Elektronik Purwantoro
- Implementasi Telegram Chat Bot Pada Sistem Pendaftaran Periode Orientasi Mahasiswa Untuk Efisiensi Pengelolaan Data
- Sistem Informasi Penjualan Material Bangunan Pada Toko Bangunan Berkah

data by google scholar

Dr. Endah Sudarmilah, S.T., M.Eng

Figure 5. Results page of the supervisor's recommendation

Route/dosbing which is shown in Figure 6, serves to display a list of available final project supervisors, then in Figure 7 is the route / about which serves to display writings about the data used by the recommendation system, including the data sources and at a glance how the recommendation system works.



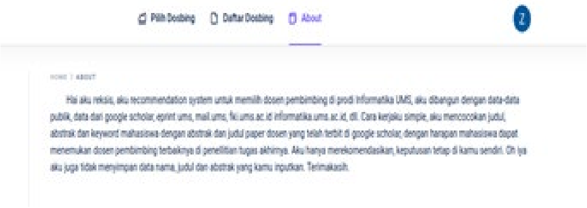
HOME > DAFTAR DOSBING

Bana Handaga
Alumni: Universiti Tun Hussein Onn Malaysia
Home base: Program Studi Informatika
Email: bh014@ums.ac.id

Fatah Yasin Al Irsyadi
Alumni: Universitas Gadjah Mada
Home base: Program Studi Informatika
Email: fyai181@ums.ac.id

Devi Afriyantari Puspa Putri
Alumni: University of Manchester, UK
Home base: Program Studi Informatika
Email: dap129@ums.ac.id

Figure 6. Final project supervisor list page



HOME > ABOUT

Hai aku reksis, aku recommendation system untuk memilih dosen pembimbing di prodi Informatika UMS, aku dibangun dengan data-data publik, data dari google scholar, eprint ums, mail.ums, fki.ums.ac.id informatika.ums.ac.id, dll. Cara kerjaku simple, aku mencocokan judul, abstrak dan keyword mahasiswa dengan abstrak dan judul paper dosen yang telah terbit di google scholar, dengan harapan mahasiswa dapat menemukan dosen pembimbing terbaiknya di penelitian tugas akhirnya. Aku hanya merekomendasikan, keputusan tetap di kamu sendiri. Oh iya aku juga tidak menyimpan data nama, judul dan abstrak yang kamu inputkan. Terimakasih.

Figure 7. Page about website

The system testing stage is the last stage that focuses on the final result and the features contained in the system. Table 6 shows the system testing with a black box which is a test where the system is directly faced with the user to interact and the system is able to respond properly and as planned.

Table 6. Black Box Testing

Function	Input	Output	Status
Main page	Access the website	Show main page	Valid
Login page	Enter Gmail and Password	Displays the page for filling out the recommendation system form	Valid
Recommendation page	Input data such as title, abstract and keywords	displaying the results of the supervisor's recommendation	Valid
Supervisor list page	Access the supervisor list page	Displays a list of supervisors	Valid

4. Conclusion

There have been several previous studies with the same topic as decision support systems using the Analytical Hierarchy Process (AHP) method in which the method is subjective depending on the expert in weighting the predetermined criteria [17]. Then there is also research conducted at the Department of Computer Science/Informatics, Faculty of Science and Mathematics, Diponegoro University with the Vector Space Model (VSM) method which is used to compare the strings of research titles between lecturers and students to build a recommendation system [18]. While the recommendation system built in this study uses the basis of comparison between title strings, abstracts, keywords from lecturer research and topics to be proposed by students with data sources from Google Scholar.

Data from web scraping from Google Scholar is still raw and rather polluted data, it still has to go through various processes before the data can really be used. Then the functionality of the recommendation system in general functions as planned. In the UMS Informatics study program, there are lecturers who have a tendency to have interests and expertise in the field of networking and this recommendation system will also recommend the lecturer if given input on topics about networking. Thus, it can be concluded that this research is in accordance with the objectives.

Recommendation systems utilizing content-based filtering method will depend heavily on the content in each item. More contents in the item, generally leads to better recommendation results. On the other hand, having more content will affect the execution time: the greater the content, the greater the time required by the system to perform calculations.

Content-based filtering also has weaknesses, because this method is very dependent on the content of the item. It is possible that this method cannot produce appropriate recommendations if the content analyzed for an item does not contain information suitable for categorization, or the item does not have enough content to categorize.

Reference

[1] P. Nagarnaik and A. Thomas, "Survey on recommendation system methods," in *2nd International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, India, Feb. 2015, pp. 1603–1608.

[2] L. Sharma and A. Gera, "A survey of recommendation system: research challenges," *International Journal of Engineering Trends and Technology.*, vol. 4 no. 5, pp. 1989–1992, 2013.

[3] J. Son and S. B. Kim, "Content-based filtering for recommendation systems using multi-attribute networks," *Expert Syst. Appl.*, vol. 89, pp. 404–412, 2017.

[4] S. Debnath, N. Ganguly, and P. Mitra, "Feature weighting in content based recommendation system using social network analysis," in *Proceeding of the 17th international conference on World Wide Web*, Beijing, China, 2008, pp. 1041–1042.

[5] D. Pyle, *Data Preparation for Data Mining*. Morgan Kaufmann, 1999.

[6] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep convolution neural networks for Twitter sentiment analysis," *IEEE Access*, vol. 6, pp. 23253–23260, 2018.

[7] D. Gunawan, "Evaluasi performa pemecahan database dengan metode klasifikasi pada data preprocessing data mining," *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 1. pp 10 – 13, 2016.

[8] R. K. Roul, J. K. Sahoo, and K. Arora, "Modified TF-IDF term weighting strategies for text categorization," in *14th IEEE India Council International Conference (INDICON)*, 2017.

[9] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for IDF," *Journal of Documentation*, vol. 60 no. 5, pp. 503–520, 2004.

[10] Z. Yun-Tao, G. Ling, and W. Yong-Cheng, "An improved TF-IDF approach for text classification." *Journal of Zheijang University Science – A*, vol. 6, no. 1, pp. 49–55, 2005.

[11] M. Nilashi, K. Bagherifard, O. Ibrahim, H. Alizadeh, L. A. Nojeem, and N. Roozegar, "Collaborative filtering recommender systems," *Research Journal of Applied Science, Engineering, and Technology*, vol. 5, no. 16, pp. 4168–4182, 2013.

[12] R. Samuel, R. Natan, and U. Syafiqoh, "Penerapan cosine similarity dan K-Nearest Neighbor (K-NN) pada klasifikasi dan pencarian buku," *Journal of Big Data Analytic and Artificial Intelligence*, vol. 4, no. 1, pp. 9 – 14, 2018.

[13] F. Rahutomo, T. Kitasuka, and M. Aritsugi, "Semantic cosine similarity". *The 7th International Student Conference on Advanced Science and Technology (ICAST)*, vol. 4, no. 1, 2012.

[14] I. Sommerville, *Software engineering, 9th ed.* Pearson Education, 2011.

[15] Mohd. Ehmer Khan *et al.*, "Different approaches to black box testing technique for finding errors," *International Journal of Software Engineering and Applications*, vol. 2, no. 4, pp. 31–40, 2011.

[16] F. Mubarak, "Perbandingan cosine similarity dan euclidean distance pada sistem rekomendasi film menggunakan metode item based multi criteria collaborative filtering," Bachelor's thesis, Universitas Sebelas Maret, 2019.

[17] A. Abdullah and M. W. Pangestika, "Perancangan sistem pendukung keputusan dalam pemilihan dosen pembimbing skripsi berdasarkan minat mahasiswa dengan metode AHP (analytical hierarchy process) di Universitas Muhammadiyah Pontianak," *J. Edukasi Dan Penelit. Inform.*, vol. 4, no. 2, pp. 184–191, 2018

[18] N. Amalina, & S. Sutikno "Sistem rekomendasi dosen pembimbing tugas akhir berbasis text mining menggunakan vector space model", Bachelor's thesis, Universitas Diponegoro, 2017.

khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Android-Based Short Message Service Filtering using Long Short-Term Memory Classification Model

M. Laylul Mustagfirin [1], Giri Wahyu Wiriasto [2*], I Made Budi Suksmadana [3], Indira Puteri Kinasih [4]

[1,2,3]Departement of Electrical Engineering
University of Mataram
Mataram
[4]Departement of Mathematics Education
Universitas Islam Negeri Mataram
Mataram
*giriwahyuwiriasto@unram.ac.id

Abstract-Short Message Service (SMS) is a technology for sending messages in text format between two mobile phones that support such a facility. Despite the emergence of many mobile text messaging applications, SMS still finds its use in communication among people and broadcasting messages by governments and mobile providers. SMS users often receive messages from parties, particularly for marketing and business purposes, advertisements, or elements of fraud. Many of those messages are irrelevant and fraudulent spam. This research aims at developing android-based applications that enable the filtering of SMS in Bahasa Indonesia. We investigate 1469 SMS text data and classify them into three categories: Normal, Fraudulent, and Advertisement. The classification or filtering method is the long short-term memory (LSTM) model from TensorFlow. The LSTM model is suitable because it has cell states in the architecture that are useful for storing previous information. The feature is applicable for use on sequential data such as SMS texts because every word in the texts constructs a sequential form to complete a sentence. The observation results show that the classification accuracy level is 95%. This model is then integrated into an Android-based mobile application to execute a real-time classification.

Keywords: text filtering, recurrent neural network, long short term memory

Article info: *submitted April 4, 2022, revised June 11, 2022, accepted September 23, 2022*

1. Introduction

Short message service (SMS) is a communication service in text format that has been used by humans in the last few decades and has become an embedded feature on every cellphone, be it a featured phone or smartphone. Since it is a service that has advantages such as low cost and eases to use, this service is also used by certain parties to send an unwanted text message, namely, spam message [1,2]. Spam is a type of message that is sent arbitrarily with various purposes such as promotions/advertising, borrowing money, announcements of sweepstakes, and such so that they are disturbing to mobile phone users [3], [4]. Spam message itself has been found in many countries including Indonesia. In 2019, Indonesia was included in the top 20 countries with the highest number of spam text messages in the world with an average mobile phone user in Indonesia receiving 46 spam messages every month [5].

There are several ways that have been done by the government, operators, and researchers to overcome spam message attacks. It can be prevented by means of operators filtering all text messages sent through SMSC (short message service center) or by installing a system that can detect spam text when it is sent to end-users [2]. Handling with the system installed at the end-user can then be done by filtering based on the content of the message received [6]. This method involves a classification method that is part of machine learning applications such as those that have been applied to spam e-mail filtering [3].

Machine learning methods have been used in several previous studies to classify spam text in Indonesian. Setifani et al. [7] compared the naive Bayes algorithm, SVM and decision tree in classifying text messages into three different classes. Herwanto et al. [8] classified Indonesian spam text messages using the multinomial naive Bayes algorithm and produced an F1-score of 0.93. The multiclass classification of the text message was carried out by Theodorus et al. [9] by comparing the performance of several different models with an average accuracy of 94% obtained. The research was also conducted using a deep learning architecture conducted by Tandra et al [10] which compared the Multinomial Naïve Bayes capability with the Bi-Directional LSTM algorithm. All of these studies were carried out up to testing how the model's performance in classifying text messages according to their type.

The focus of this paper is on implementing the trained model to classify Indonesian text or '*Bahasa*' messages into three categories that are listed as the most popular text message type in Indonesia. It is in line with the study proposed by Sethi and Bhootna [11] regarding spam text filtering applications on android devices using Bayesian algorithms. Uysal et al. [12] did the same thing by conducting research on spam and non-spam text message filtering applications using the Bayesian method. Then there is also a study to create a mobile-based system for filtering spam text in English – India by Yadav et al using the SVM method [13].

Based on the success of previous studies in testing various methods to overcome spam text, we propose to make a model using the Long short-term memory (LSTM) method [14] and implement it in the form of an Android-based application. The LSTM method is a model designed to handle sequential data that depends on the ordered data such as word sequences with various lengths and is able to capture long-term dependencies of sentences on the data [15]. In this study, the authors classify Indonesian-language *('bahasa')* short messages service into three classes, namely normal/personal text, fraudulent text, and promotional text. The division into three classes is because the SMS received by cellphone users in Indonesia are received from known people (personal SMS), sent by unknown parties with the aim of deceiving the recipient, and SMS containing promotions or advertisements from third parties in the form of companies [7,8]. This research was conducted to see how the performance of the model when put together into a complete system that can classify incoming SMS in real-time and not just to test how effective and accurate the model is. With this research, it is hoped that the resulting system can be a reference for smartphone users. in managing incoming SMS so that no more users are exposed to fraud.

This paper consists of four parts. The first part explains the background and initial idea of this research as a developmental idea from the previous studies. The second part is the part that explains how the research is carried out or the research methods used. The third section describes the results of the research and a discussion based on the research results. Finally, the fourth section contains the conclusions obtained from the research that has been carried out.

2. Methods

In the previous studies, many attempts to classify short text messages using artificial intelligence to deal with spam have been carried out. In this research, the development of a system that has been combined with an artificial intelligence model and installed on the end-user's device is carried out. Figure 1 shows a general illustration of how this mobile application works where the authors in this study focus on the part that is inside the box with the dotted black line.

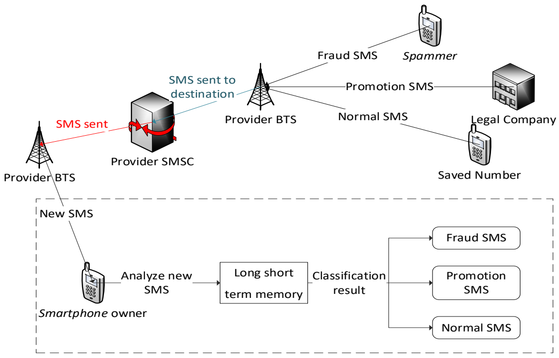


Figure 1. Implementation System

a. Data Collection

The text dataset used in this study was obtained in two ways, namely collection with the help of respondents who filled out a questionnaire form using the google form media and the second was obtained from the Kaggle website. The total data obtained are 1469 sentences of data text. The labeling of the data obtained from the questionnaire was carried out by the researcher based on the closeness of the meaning of the Indonesian text messages. The dataset is grouped into three classes, namely normal, fraudulent, and advertisement text messages with a total of 1469 data. The data is divided into training data (75%) and test data (25%). The division of each class in the dataset can be seen in Table 1.

Table 1. Text Message amount per category

Category	amount
Normal	537
Fraud	334
Advertisement	598
Total	1469

In table 2 we present 15 unprocessed data which are divided into three different classes.

Table 2. Example of Text Message Data Sets

Text Message (in bahasa)	Translated Mesage
NASABAH Yth! Anda diminta ke Kantor CABANG untuk monitor ulang REKENING TABUNGAN yg ERROR SISTEM. Sebelumnya harap telp dl pak WAWAN 082317744737	Dear CUSTOMERS! You are asked to go to the BRANCH office to re-monitor the SAVING ACCOUNT because SYSTEM ERROR. Before that, please call Mr. WAWAN 082317744737

Text Message (in bahasa)	Translated Mesage
INFO RESMI PT.TRI CARE PIN Pemenang (8jf2177) anda m-dptkan HADIAH I unit TOYOTA YARIS U/Info cek pin klik: *www.tricare2015.blogspot.com*	OFFICIAL INFO PT. TRI CARE PIN Winner (8jf2177) you get GIFT I unit TOYOTA YARIS U/Info check pin click: *www.tricare2015.blogspot.com*
Anda M'dptkan $ubs!d! Dri Pert4min4 Rp.189 jt Pin (717747) !nfo Wh4ts4pp:085243235227 atau Surat Keputusan dari Tri Care Indonesia No.XV/2015 Pin_Pemenang ANDA : 67ytg44 mendapatkan hadiah cek tunai Rp 45 jt. U/INFO kunjungi: *www.id.tri.webnode.com.*	You get allowance from Pertamina Rp. 189 million Pin (717747) Info Whatsapp: 085243235227 or Decision Letter from Tri Care Indonesia No.XV/2015 YOUR Winner_ Pin : 67ytg44 get cash reward Rp 45 million U/INFO visit: *www.id.tri.webnode.com.*
ASS..YTH BPK/IBU BTUH BIAYA TMBHAN UNTUK MDAL USAHA DLL U/INFO/SILAHKAN CHAT WA : 085387120337	ASS.. DEAR MR/MRS NEED ADDITIONAL COSTS FOR BUSSINESS CAPITAL ETC U/INFO/PLEASE CHAT WA : 085387120337
Yuk Ikuti akun dakwah, caranya: ketik IKUT [spasi] AkuCintaIslam kirim ke 082110001021.	Let's Follow da'wah account, step: type IKUT [space] AkuCintaIslam send to 082110001021.
Bebas Pulsa! Ambil bonusmu di *600# (GRATIS). Dptkan gratis nelpon atau internetas atau promo lainnya sesuai hobimu!	Free of charge! Take your bonus at *600# (FREE). Get free calls or internet or other promos according to your hoby!
Ayo klik tsel.me/maxendeal30gb utk kuota MAXstream 30GB hny 40rb dan tonton FA CUP: Leicester City VS MU atau Liga Serie A: Roma VS Napoli di MAXstream	Come click tsel.me/maxendeal30gb for quota MAXstream 30GB just rp. 40k dan watch FA CUP: Leicester City VS MU or Serie A league: Roma VS Napoli on MAXstream
Bebas online seharian dgn Unlimited Internet hanya Rp105rb/bln! Daftar XL PRIORITAS dgn chat ke wa.me/62818800055/?text=gabung. S&K berlaku.	Free online all day with Unlimited Internet only Rp 105k/month! Register XL PRIORITAS by chat to wa.me/62818800055/?text=join. T&C apply.
Yuk tetap gunakan Flash Volume Ultima utk update informasi Anda, kuota 60MB/7hr mulai Rp7rb di *100*431#. Tarif&lokasi cek di tsel.me/FL	Let's keep use Flash Volume Ultima to update your information, quota 60MB/7 day start from Rp 7k at *100*431#. Check rates & locations at tsel.me/FL
Nikmati nelpon dan SMS UNLIMITED ke sesama Indosat Ooredoo internetan sepuasnya 3 hari dengan paket Ramadhan Unlimited. ketik *123*88#	Enjoy UNLIMITED calls and SMS to other Indosat Ooredoo fellow internet all you want for 3 days with the Ramadhan Unlimited package. Type *123*88#
Aku senin udah ke tempat kerja. Minggu2 depan aku gaktau bisa/ngga :(	I'm at work Monday. I don't know I can or can't next week
Maaf kaprodi kita itu siapa yah hehe	Sorry, who is the head of our study program?
Pada ga aktif si tombol algoritma2 nya teh ga ngerti	The algorithm buttons not active, I don't understand
Gara gara batman. Pdhl udh diulang	Because of batman. Even though it's been repeated
Mohon maaf bang, kemarin untuk kelompok saya diberitahu kirim lewat chat, kami juga ndak tau alamat email abang	Sorry, yesteday my group was told to send via chat, we also don't know your email address

b. Data Classification

The data that has been obtained is then classified into three parts, namely training data, data validation and data testing. Data distribution was carried out using a training data ratio of 60%, data validation of 15%, and test data of 25% where data validation was used after each epoch was completed to see whether the model was overfitting or not.

c. Pre-Processing

In the pre-processing stage, the stages are carried out to generalize the format of the text. Pre-processing of text message sentences is conducted through four stages, namely punctuation removal, case folding, stopwords removal and tokenization. Then the text data will be converted into a number representation. Table 3 shows the data that has gone through the pre-processing stage of each class.

Table 3. Pre-processing Stage Data Example

SMS Text (in bahasa)	Class
Ass yth bpk ibu btuh biaya tmbhan untuk mdal usaha dll u info silahkan chat wa 085387120337	fraud
bebas pulsa ambil bonusmu di 600 gratis dptkan gratis nelpon atau internetas atau promo lainnya sesuai hobimu	advertisement
aku senin udah ke tempat kerja minggu2 depan aku gaktau bisa ngga	Normal

d. Word Embedding

Word embedding is a method used to represent words in a vector form consisting of real numbers. The vectors can then be plotted to see where the words are and words that have similarities will have a close position when plotted. Word embedding itself has a network similar to an ordinary neural network and is often used as input in deep learning models for solving NLP problems [3]. There are also word embedding models that have been trained previously and can be directly used, such as the Glove model [16] and the Word2Vec model [15]. Figure 2 shows examples of words that are plotted in a two-dimensional graph after their representation is generated using word embedding. It can be seen that the word "cat" has a close distance from the word "dog" because it has a relationship that is both including animals. While the word "computer" has a long distance from the word "tomatoes" because it has a much different contextual nature, namely "computer" is a tool while "tomatoes" is a vegetable. In this study, the author chose to train the word embedding model himself because the dataset used was in Bahasa.

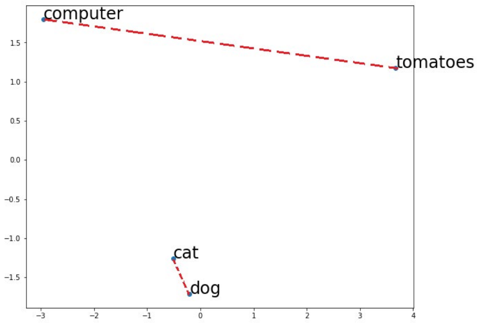


Figure 2. Words Plotting as the result of word embedding

In this study, the maximum number of words contained in each data is 17 words. The embedding layer used has an output dimension of 17x128. The output matrix of the embedding layer has 17 rows where each row represents the 17 words. Figure 3 shows an example of the embedding layer output.

1st word → [-2.72e-02 -2.46e-02 -2.285e-02 3.298e-02 -3.102e-02 3.677e-03 … -3.9e-02]

2nd word → [2.124e-02 -2.736e-02 4.604e-02 9.968e-03 1.386e-02 1.615e-02 … -2.4e-02]

3rd word → [2.519e-02 -3.179e-02 2.636e-02 -1.705e-02 -2.655e-02 3.463e-04 … 1.2e-02]

⋮ [⋮ ⋮ ⋮ ⋮ ⋮ ⋮ ⋮]

15th word →[2.210e-02 1.464e-02 1.673e-02 -4.484e-02 3.380e-02 4.852e-02 … 1.92e-02]

16th word →[-4.54e-02 3.226e-02 1.951e-02 -9.576e-03 -3.482e-02 3.880e-02 … 1.06e-02]

17th word →[1.00e-02 3.647e-02 8.621e-03 -4.725e-02 -4.777e-02 -3.587e-04 … 2.77e-02]

Figure 3. The output matrix of the embedding layer

e. Model Implementation

The model implementation process is carried out using the Tensorflow library provided by Google Collaboratory [17]. Google Collaboratory is a tool built on Jupyter Notebook. Jupyter Notebook is a tool that runs on a browser and has integration in interpreted languages such as python complete with libraries for data processing [18]. Google collaboratory or google collab is a product of the Google team with the aim of simplifying work related to machine learning, data analysis, and education based on the Jupyter Notebook [19,20]. Figure 4 shows the network architecture used in this study.

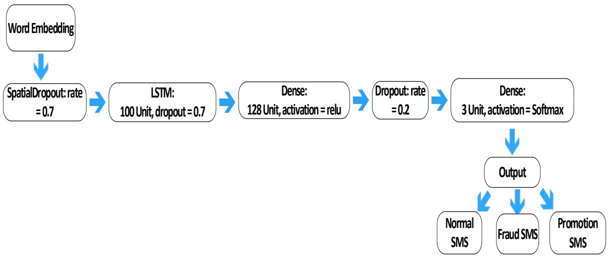

Word Embedding
SpatialDropout: rate = 0.7
LSTM: 100 Unit, dropout = 0.7
Dense: 128 Unit, activation = relu
Dropout: rate = 0.2
Dense: 3 Unit, activation = Softmax
Output
Normal SMS
Fraud SMS
Promotion SMS

Figure 4. Proposed Model Design

f. Dropout and Spatial Dropout

Dropout is one way to regularize the model so as not to overfit by not randomly involving nodes in the training process so that they do not depend on each other [21]. However, in certain cases, a variant of dropout is used, namely spatial dropout. Spatial dropout is a dropout method that does not include all feature mappings compared to doing a dropout on randomly selected nodes. This is done because the activation of feature mapping has a strong correlation such as in image data or text data so that the ordinary dropout method does not have much effect [22].

g. Long Short-Term Memory

Long short-term memory is a modified version of the Recurrent Neural Network (RNN) model. The RNN architecture itself has a long-term dependency problem that causes the model to be unable to process sequential data that is too long/has a large time step difference [23]. The LSTM network consists of repeating units where each unit consists of parts called gates that determine the addition or subtraction of information from the data, consisting of input gate (), forget gate () and output gate () as shown on figure 5 and figure 6. The following is a transition function that exists in the LSTM sections [24]:

$$
\begin{aligned}
i_t &= \sigma(W_i \cdot [h_{t-1} + b_i]) \\
f_t &= \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big) \\
q_t &= \tanh\big(W_q \cdot [h_{t-1}, x_t] + b_q\big) \\
o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\
c_t &= f_t \odot c_{t-1} + i_t \odot q_t \\
h_t &= o_t \odot tanh(c_t)
\end{aligned}
$$

Where is a sigmoid function, and is the weight and bias of each gate which will continue to be updated during the training process, is a vector in the cell state section, is a hidden state in the previous unit and is an operator for element-wise multiplication [1,25].

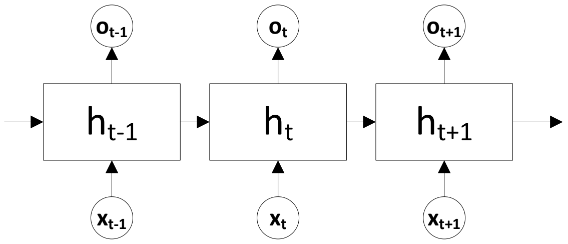


Figure 5. LSTM Network Structure

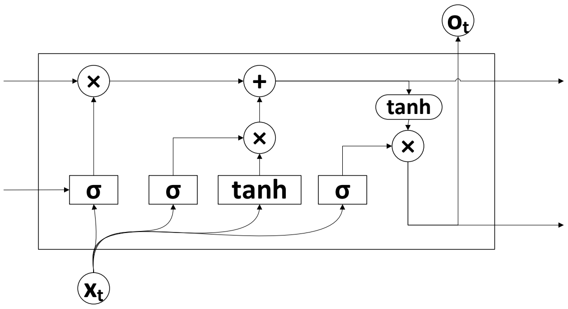


Figure 6. LSTM single unit architecture on the network

h. Dense

Dense is a layer that connects all neurons directly to each other with the next layer and can often also be referred to as a fully connected layer. In the model proposed by the authors, a dense layer is used in the final two layers, a hidden layer with 128 neurons and a Rectified Linear Unit (ReLU) activation function. The ReLu function returns if it is positive and 0 otherwise [26].

$$f(x) = max\,(0, x) \tag{7}$$

The reason for using a dense layer as a hidden layer is because it can improve the performance of the model in classification [27]. Dense layer is also used as the last layer or layer that gives the output of model predictions. Because in this study a multiclass classification was used, three neurons were used and the softmax activation function was employed [28].

$$softmax(x) = \frac{\exp\,(x_i)}{\sum_{j=1}^{n} \exp\,(x_j)} \tag{8}$$

i. Model Evaluation

To determine the performance of the model, standard metrics such as confusion matrix, accuracy, precision, recall, false-positive rate, F1-score, Receiver Operating Characteristics (ROC) Curve and Area Under The Curve (AUC) are used. These size indications are based on the method designed by Kohavi and Provost [29]. Then from the confusion matrix, the other metrics mentioned above can be calculated [30].

j. Designing Phase

As part of the implementation phase, the author develops a smartphone application for devices with Android OS. This application development was carried out using Android Studio with the Kotlin programming language on a Personal computer running Windows 10 with an Intel Core 13-9100f processor, 8 GB of RAM and an NVIDIA 1050TI graphics card. For testing during the development process, an emulator device provided directly by Android Studio was used, namely the Google Pixel 4 XL and the OPPO A92 physical device.

After the LSTM model has been trained and tested until it reaches the desired results, the model is exported using the TensorFlow Lite library. TensorFlow Lite is a tool developed by the TensorFlow team so that machine learning models can run on devices that have specifications below ordinary computers/PCs such as mobile, embedded and IoT devices [31]. The exported model is then combined with the application so that when a new SMS comes in, it will be preprocessed then the model will predict the class of the new SMS and finally group it together with other SMS with the same class. The workings of the system are illustrated using a flow chart which can be seen in figure 7.

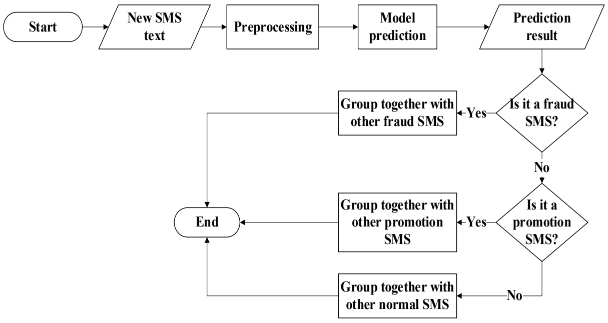


Figure 7. Flowchart of how the app works

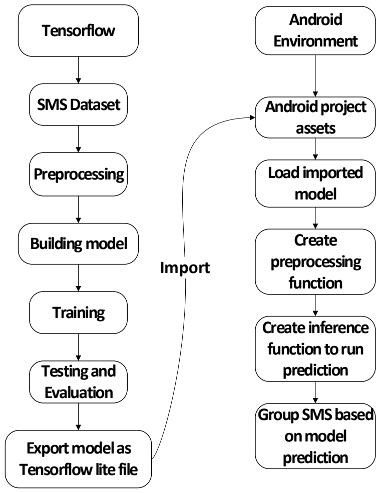


Figure 8. The process of integrating the model into the Android system

Figure 8 shows the steps involved when a model is added to an Android application. After the dataset is obtained, preprocessing is carried out, then the model is built according to the proposed initial design. The model is then trained, tested and evaluated until a satisfactory performance is found. To make the model usable on other devices such as Android, TensorFlow provides a library to convert the trained model into TensorFlow lite format. The TensorFlow lite file is imported into the android project directory.

In the Android application design process, the next step is to create a preprocessing function that is the same as the preprocessing function used when the model is trained to have an accurate output. The incoming SMS text then goes through the preprocessing stage and the model will make predictions on the data. The prediction results of the model determine which category the SMS falls into.

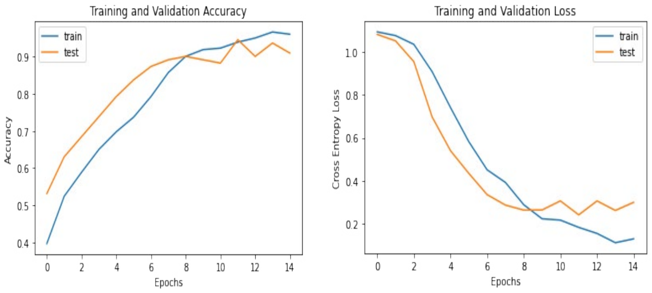
3. Results

a. Model Evaluation

The model evaluation process is carried out to see how well the model performs after going through the training process. Figure 6 shows a graph of the comparison of accuracy and loss to the increase in the number of epochs on the training data and validation data.

It can be seen in figure 9(a) that there is an increase in the value of accuracy as the number of epochs increases and in figure 9(b) there is a decrease in the value of loss/error when the number of epochs increases so that a convergent graph is obtained. Then testing using 20% of the data that has never been seen by the previous model. The results of model testing in the form of a confusion matrix can be seen in figure 10.

Based on the confusion matrix, table 4 shows the results of the metrics used to test the model's performance. The model obtains an accuracy level of 0.951, a precision value of 0.936, a recall value or true positive rate of 0.946, a false positive rate of 0.026, an F1-score as the average between precision and recall of 0.941 and the last ROC-AUC value of 0.959. In figure 11 we present the ROC curve of the LSTM model that has been trained.



(a) (b)

Figure 9. Model Trained Result

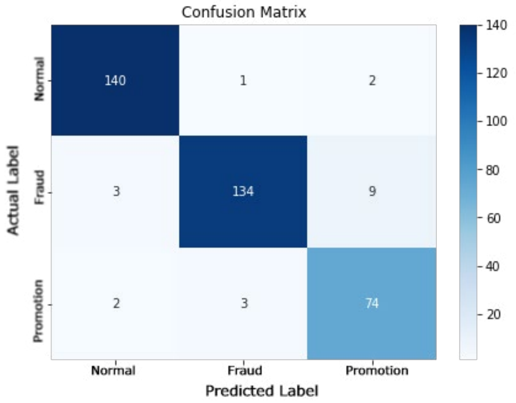


Figure 10. Confusion matrix of tested model

Table 4. Metrics Value of Model Evaluation Result

Metrik	Result
Accuracy	0.951
Precision	0.936
Recall/TPR	0.946
FPR	0.026
F1-Score	0.941
ROC-AUC	0.959

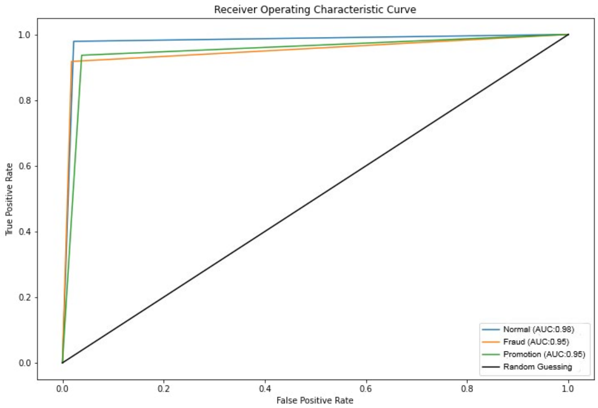


Figure 11. ROC-AUC model

Then in table 5 we compare the results of the LSTM model that we have trained with other machine learning (ML) models that have been carried out by Herwanto et al [8] and Setifani et al [7] using the same dataset. The LSTM model appears to have the highest value for each of the test metrics used in all models.

Table 5. Comparison of results from several classification models

Model	Accuracy	Precision	Recall	F1-Score
Naïve Bayes[8]	0.94	0.92	0.93	0.93
Decision Tree[8]	0.87	0.87	0.84	0.85
SVM[8]	0.93	0.93	0.92	0.92
MNB[7]	0.94	0.93	0.92	0.93
Random Forest[7]	0.93	0.92	0.93	0.92
LSTM	**0.95**	**0.93**	**0.94**	**0.94**

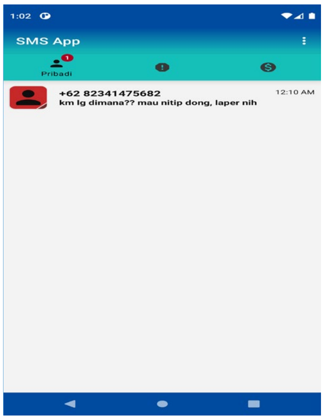
Based on table 5. it can be seen that the LSTM model proposed in this study has the best performance compared to other models. The highest accuracy is 0.95 obtained by the LSTM model, followed by Naïve Bayes, SVM, Random Forest and Decision Tree. For precision metrics, the LSTM model obtained the same value with 0.93 for the SVM and MNB models, followed by Naïve Bayes, Random Forest and Decision Tree. Then for recall, the LSTM model has the highest value obtained 0.94 followed by Naïve Bayes, Random Forest, SVM, MNB and Decision Tree. Meanwhile, the F1-score of the LSTM model also obtained the highest score of 0.94, followed by Naïve Bayes, MNB, SVM, and finally Decision Tree.
Based on the test results, it can be seen that the LSTM model has the best performance compared to other models.

b. Mobile Application Interface

This application is a direct implementation of the LSTM model that has been trained so that it can classify Indonesian-language text messages directly when a new text is received. The main part of the application is divided into three parts, namely a page that displays a list of normal, fraudulent, and also advertising text message. Navigating between pages can be done by swiping

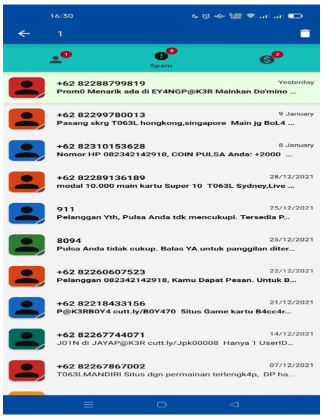
the screen to the left or right. To view the message content, you can do this by tapping the desired message so that the message opens and the view will change to a page containing the conversation between the recipient and the sender of the message. The application screen display can be seen in figure 12.

(a)



+62 82341475682 12:10 AM
km lg dimana?? mau nitip dong, laper nih

(b)



+62 82288799819 Yesterday
Prom0 Menarik ada di EY4NGP@K3R Mainkan Do'mino ...

+62 82299780013 9 January
Pasang skrg T063L hongkong,singapore Main jg BoL4 ...

+62 82310153628 8 January
Nomor HP 082342142918, COIN PULSA Anda: +2000 ...

+62 82289136189 28/12/2021
modal 10.000 main kartu Super 10 T063L Sydney,Live ...

911 25/12/2021
Pelanggan Yth, Pulsa Anda tdk mencukupi. Tersedia P...

8094 25/12/2021
Pulsa Anda tidak cukup. Balas YA untuk panggilan diter...

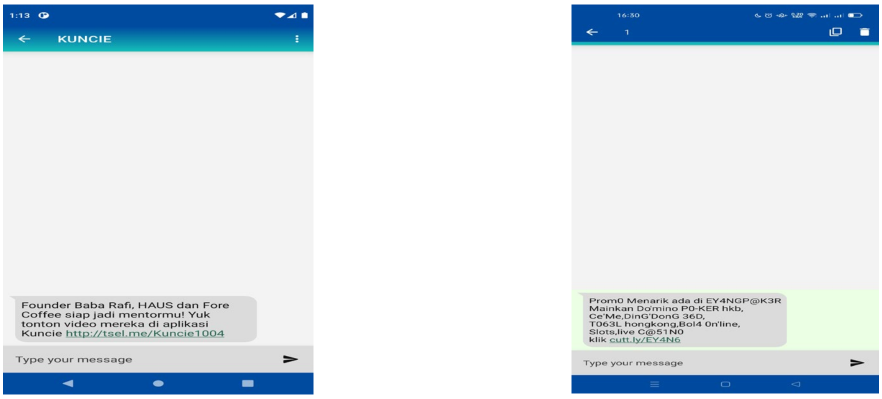
+62 82260607523 22/12/2021
Pelanggan 082342142918, Kamu Dapat Pesan. Untuk B...

+62 82218433156 21/12/2021
P@K3RB0Y4 cutt.ly/B0Y470 5itus Game kartu B4cc4r...

+62 82267744071 14/12/2021
J01N di JAYAP@K3R cutt.ly/Jpk00008 Hanya 1 UserID...

+62 82267867002 07/12/2021
T063LMANDIRI Situs dgn permainan terlengk4p, DP ha...

(c)



Founder Baba Rafi, HAUS dan Fore Coffee siap jadi mentormu! Yuk tonton video mereka di aplikasi Kuncie http://tsel.me/Kuncie1004

(d)

Prom0 Menarik ada di EY4NGP@K3R Mainkan Do'mino P0-KER hkb, Ce'Me,DinG'DonG 36D, T063L hongkong,Bol4 0n'line, Slots,live C@51N0 klik cutt.ly/EY4N6

(e)

Figure 12. Design of android Apps Interface

Figure 12(a) shows the appearance of the application icon on the smartphone screen when the application is first opened. Figure 12(b) is a classified text message list display screen with the initial display screen on the normal text message list page. Figure 12(c) is an additional menu display screen when a message is selected, the operation that can be performed is to delete the selected message. Figure 12 (d) is a screen that displays the contents of the message that is opened and contains a column to type and a button to send a message. Figure 12(e) is an additional menu display screen when an item in a conversation is selected, the operations that can be performed are to copy or delete the selected item.

4. Conclusion

In this study, the authors propose an LSTM model combined with an Android-based mobile application to filter Indonesian text messages according to their type. This research produces a model that performs well and applies to an android application. The evaluation suggests that the level of accuracy possessed by the LSTM model when trained using the Indonesian text message dataset was 0.951. The results obtained in previous studies using other methods have shown high accuracy. However, the studies limited themselves to examining the effectiveness of methods without implementation as an application. The application can directly classify incoming text messages in real time and group them into the specified message list.

References

[1] X. Liu, H. Lu, and A. Nayak, "A Spam Transformer Model for SMS Spam Detection," *IEEE Access*, vol. 9, pp. 80253–80263, 2021, doi: 10.1109/ACCESS.2021.3081479.

[2] M. T. Nuruzzaman, C. Lee, and D. Choi, "Independent and personal SMS spam filtering," in *Proceedings - 11th IEEE International Conference on Computer and Information Technology, CIT 2011*, 2011, pp. 429–435. doi: 10.1109/CIT.2011.23.

[3] A. Ghourabi, M. A. Mahmood, and Q. M. Alzubi, "A hybrid CNN-LSTM model for SMS spam detection in arabic and english messages," *Future Internet*, vol. 12, no. 9, Sep. 2020, doi: 10.3390/FI12090156.

[4] C. Khemapatapan, "Thai-English spam SMS filtering," in *2010 16th Asia-Pacific Conference on Communications, APCC 2010*, 2010, pp. 226–230. doi: 10.1109/APCC.2010.5679770.

[5] "Truecaller Insights: Top 20 Countries Affected by Spam Calls & SMS in 2019 - Truecaller Blog." https://truecaller.blog/2019/12/03/truecaller-insights-top-20-countries-affected-by-spam-calls-sms-in-2019/ (accessed Jun. 14, 2022).

[6] J. M. G. Hidalgo, G. C. Bringas, E. P. Sánz, and F. C. García, "Content based SMS spam filtering," in *Proceedings of the 2006 ACM Symposium on Document Engineering, DocEng 2006*, 2006, vol. 2006, pp. 107–114. doi: 10.1145/1166160.1166191.

[7] N. A. Setifani *et al.*, "PERBANDINGAN ALGORITMA NAÏVE BAYES, SVM, DAN DECISION TREE UNTUK KLASIFIKASI SMS SPAM," 2020.

[8] H. Herwanto, N. L. Chusna, and M. S. Arif, "Klasifikasi SMS Spam Berbahasa Indonesia Menggunakan Algoritma Multinomial Naïve Bayes," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 4, p. 1316, Oct. 2021, doi: 10.30865/mib.v5i4.3119.

[9] A. Theodorus, T. K. Prasetyo, R. Hartono, and D. Suhartono, "Short Message Service (SMS) Spam Filtering using Machine Learning in Bahasa Indonesia," in *3rd 2021 East Indonesia Conference on Computer and Information Technology, EIConCIT 2021*, Apr. 2021, pp. 199–202. doi: 10.1109/EIConCIT50028.2021.9431859.

[10] V. G. Tandra, Y. Yowen, R. Tanjaya, W. L. Santoso, and N. Nurul Qomariyah, "Short Message Service Filtering with Natural Language Processing in

Indonesian Language," Aug. 2021. doi: 10.1109/ICISS53185.2021.9532503.

[11] G. Sethi and V. Bhootna, "SMS Spam Filtering Application Using Android." [Online]. Available: www.ijcsit.com

[12] A. K. Uysal, S. Gunal, S. Ergin, and E. S. Gunal, "A novel framework for SMS spam filtering," 2012. doi: 10.1109/INISTA.2012.6246947.

[13] K. Yadav, P. Kumaraguru, A. Goyal, A. Gupta, and V. Naik, "SMSAssassin: Crowdsourcing driven mobile-based system for SMS spam filtering," in *HotMobile 2011: The 12th Workshop on Mobile Computing Systems and Applications*, 2011, pp. 1–6. doi: 10.1145/2184489.2184491.

[14] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[15] T. Mikolov, A. Joulin, S. Chopra, M. Mathieu, and M. 'A Ranzato, "Learning longer memory in recurrent neural networks," 2015.

[16] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2014, pp. 1532–1543. doi: 10.3115/v1/d14-1162.

[17] USENIX Association., ACM SIGMOBILE., ACM Special Interest Group in Operating Systems., and ACM Digital Library., *Papers presented at the Workshop on Wireless Traffic Measurements and Modeling : June 5, 2005, Seattle, WA, USA*. USENIX Association, 2005.

[18] B. M. Randles, I. v. Pasquetto, M. S. Golshan, and C. L. Borgman, "Using the Jupyter Notebook as a Tool for Open Science: An Empirical Study," Jul. 2017. doi: 10.1109/JCDL.2017.7991618.

[19] "Google Colab." https://research.google.com/colaboratory/faq.html (accessed Jun. 14, 2022).

[20] T. Carneiro, R. V. M. da Nobrega, T. Nepomuceno, G. bin Bian, V. H. C. de Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018, doi: 10.1109/ACCESS.2018.2874767.

[21] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," Jul. 2012, [Online]. Available: http://arxiv.org/abs/1207.0580

[22] J. Tompson, R. Goroshin, A. Jain, Y. Lecun, and C. Bregler, "Efficient Object Localization Using Convolutional Networks."

[23] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning," May 2015, [Online]. Available: http://arxiv.org/abs/1506.00019

[24] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, "A C-LSTM Neural Network for Text Classification," Nov. 2015, [Online]. Available: http://arxiv.org/abs/1511.08630

[25] A. Graves, N. Jaitly, and A. R. Mohamed, "Hybrid speech recognition with Deep Bidirectional LSTM," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2013 - Proceedings*, 2013, pp. 273–278. doi: 10.1109/ASRU.2013.6707742.

[26] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines."

[27] V. L. Helen Josephine, A. P. Nirmala, and V. L. Alluri, "Impact of Hidden Dense Layers in Convolutional Neural Network to enhance Performance of Classification Model," *IOP Conference Series: Materials Science and Engineering*, vol. 1131, no. 1, p. 012007, Apr. 2021, doi: 10.1088/1757-899x/1131/1/012007.

[28] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*.

[29] R. Kohavi, "Glossary of Terms Special Issue on Applications of Machine Learning and the Knowledge Discovery Process," 1998. [Online]. Available: http://robotics.stanford.edu/~ronnyk/glossary.html

[30] X. Deng, Q. Liu, Y. Deng, and S. Mahadevan, "An improved method to construct basic probability assignment based on the confusion matrix for classification problem," *Information Sciences*, vol. 340–341, pp. 250–261, May 2016, doi: 10.1016/j.ins.2016.01.033.

[31] "TensorFlow Lite." https://www.tensorflow.org/lite/guide (accessed Jun. 14, 2022).

Peer Reviewer

The Board of Editors greatly appreciate the participation of the following reviewers that help during the review process for the publication of Khazanah Informatika since 2021.

1.	Aam Amrullah	Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia
2.	Adi Supriyatna	Universitas Bina Sarana Informatika, Bandung, Indonesia
3.	Afandi Nur Aziz Thohari	Politeknik Negeri Semarang, Semarang, Indonesia
4.	Afrig Aminuddin	Universitas Amikom Yogyakarta, Yogyakarta, Indonesia
5.	Ahmad Yusuf Ismail	Kunsan National University, Gusan, Republic of Korea
6.	Akmal Junaidi	Universitas Lampung, Lampung, Indonesia
7.	Alwis Nazir	Universitas Islam Negeri Sultan Syarif Kasim, Riau, Indonesia
8.	Ardi Pujiyanta	Universitas Ahmad Dahlan, Yogyakarta, Indonesia
9.	Aris Rakhmadi	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
10.	Arkham Zahri Rakhman	Institut Teknologi Sumatera, Lampung, Indonesia
11.	Auzi Asfarian	Institut Pertanian Bogor University, Bogor, Indonesia
12.	Bana Handaga	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
13.	Budi Nugroho	Research Center for Informatics, National Agency for Research and Innovation, Jakarta, Indonesia
14.	Dedi Gunawan	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
15.	Devi Afriyanti Puspa Putri	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
16.	Dewi Faroek	Universitas Ahmad Dahlan, Yogyakarta, Indonesia
17.	Diah Priyawati	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
18.	Dimas Aryo Anggoro	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
19.	Dwi Ely Kurniawan	Politeknik Negeri Batam, Batam, Indonesia
20.	Eka N Kencana	Universitas Udayana, Denpasar, Bali, Indonesia
21.	Endah Sudarmilah	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
22.	Endang Wahyu Pamungkas	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
23.	Fajar Suryawan	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
24.	Fata Nidaul Khasanah	Universitas Bina Insani, Bekasi, Indonesia
25.	Fatah Yasin Al Irsyadi	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
26.	Favian Dewanta	Telkom University, Bandung, Indonesia
27.	Frieyadie	STMIK Nusa Mandiri, Jakarta, Indonesia
28.	Gunawan Ariyanto	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
29.	Hilal Nuha	Telkom University, Bandung, Indonesia
30.	Irma Yuliana	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
31.	Iwan Awaludin	Politeknik Negeri Bandung, Bandung, Indonesia
32.	Lasmedi Afuan	Universitas Jenderal Soedirman, Purwokerto, Indonesia
33.	Leonard Goeirmanto	Universitas Mercu Buana, Jakarta, Indonesia
34.	Lutfiyah Dwi Setia	Politeknik Negeri Madiun, Madiun, Indonesia
35.	Maryam	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
36.	Meyti Eka	Politeknik Negeri Malang, Malang, Indonesia
37.	Naufal Azmi Verdikha	Universitas Muhammadiyah Kalimantan Timur, Samarinda, Indonesia
38.	Nurgiyatna	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
39.	Pardomuan Robinson Sihombing	Badan Pusat Statistik (Statistics Indonesia), Jakarta, Indonesia

40.	Prajanto Wahyu adi	Universitas Diponegoro, Semarang, Indonesia
41.	Rajif Agung Yunmar	Institut Teknologi Sumatera, Lampung, Indonesia
42.	Ridho Ananda	Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia
43.	Rizki Wahyudi	Universitas Amikom Purwokerto, Purwokerto, Indonesia
44.	Sayekti Harits Suryawan	Universitas Muhammadiyah Kalimantan Timur, Samarinda, Indonesia
45.	Sitaresmi Wahyu Handani	Universitas Amikom Purwokerto, Purwokerto, Indonesia
46.	Solikhun	AMIK & STIKOM Tunas Bangsa Pematangsiantar, Pematang Siantar, Indonesia
47.	Sri Karnila	Institut Informatika Dan Bisnis Darmajaya, Bandar Lampung, Indonesia
47.	Tati Ernawati	Politeknik TEDC Bandung, Bandung, Indonesia
48.	Umi Fadlilah	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia
49.	Ventje Jeremias Lewi Engel	Institut Teknologi Harapan Bangsa, Bandung, Indonesia
50.	Wiwit Supriyanti	Politeknik Indonusa Surakarta, Surakarta, Indonesia
51.	Yogiek Indra Kurniawan	Universitas Jenderal Soedirman, Purwokerto, Indonesia
52.	Yusuf Sulistyo Nugroho	Universitas Muhammadiyah Surakarta, Surakarta, Indonesia