

Programmer Selection Using Modified Fuzzy Mamdani Method

Abdul Manan^{a1}, Victor Wiley^{a2}, Thomas Lucas^{a3}

^aInformatics Engineering, STMIK Swadharma
Jl. Malaka No.3, RT.6/RW.2, Roa Malaka, Tambora, Kota Jakarta Barat

¹abdmanan8@gmail.com

²codingvictor@gmail.com

³thomasstimik@gmail.com

Abstract

Selection of candidate of the programmer is a complex and tiring process. Software development manager must work hard to guarantee that only qualified candidates will be selected. This study the parameters needed by the programmer are proper and adequate knowledge, skills, attitudes, and productivity. Knowledge, skills, attitudes, and productivity are the four competencies that every programmer must-have. The four components above are very important in developing an IT company. This study proposes a classification model of programmer selection based on certain criteria, parameters, and attributes. This study modifies the Fuzzy Mamdani Method as the approach for determining the feasibility of the programmer. The proposed model has satisfied result of percent of accuracy with 75.57% level. The result indicates that the proposed model has produced a sufficient solution to be used in the real situation for selecting the feasible programmer.

Keywords: *Programmer Candidates, Knowledge, Skills, Attitudes, Productivity, Fuzzy Mamdani*

1. Introduction

Software development companies need programmers with adequate knowledge, skill and attitude to provide feasible productivity in managing software projects [1]. However, there is various ability and characteristics of the programmer (i.e., knowledge, skills, and attitudes of behavior) determine the team productivity and success [2]. In the meantime, selecting candidate programmers is often a complex and tiring process. Therefore, it is necessary to build an approach to choose the candidate of programmer based on certain criteria, parameter, and attributes. The selection process must be carried out effectively by filtering the individual competency criteria in order to assemble a development team with high productivity [3]. In fact, a new applicant or candidate of programmer has various characteristics which not easy to be detected. The selection can be very varied and burdened the software project. In order to mitigate the issue, it needs to be a tool that can simplify the candidate selection [4]. This paper proposes an approach based on the Fuzzy Mamdani method for selecting the candidate of a programmer. The method is based on fuzzy logic values filtering the candidate's attributes and parameters [5]. Through mathematical calculations of three parameters (e.g., knowledge, skill, and attitude), the test results of the candidates are simulated to be assigned into different fuzzy set memberships [6]. Their memberships are based on the priority values and the largest percentage of the assignment result.

[7] considers the estimation of the final (i.e., successful or unsuccessful) status of the project by applying the Bayesian classifier as a metric of data collected from the project. However, Naïve Bayes has the disadvantage of being very sensitive in the selection of features [8]. Another disadvantage of Naïve Bayes is that there are too many features, not only increases calculation time but also decreases classification accuracy [9]. While the use of fuzzy methods is able to handle very complex processes, which are represented by inaccurate, uncertain and qualitative information. Usually, fuzzy methods are based on linguistic rules of the type "if conditions are

then actions", where fuzzy set theory and fuzzy logic provide the mathematical basis needed to handle information that is inaccurate and with linguistic rules.

The proposed model is useful for making decisions among the manager to determine the best candidate of programmers based on the parameter values. In addition, a comparison between Mamdani fuzzy calculations and manual fuzzy calculations is also conducted and explained. The advantage of using fuzzy-based mathematical methods is also given. The conclusion and suggestion are given in the final part.

2. Theoretical Review

2.1. Fuzzy Sets

Fuzzy set theory is a calculation of fuzzy inference system in order to determine the range of criteria values for a selection of candidate programmers [10]. The data of criteria of a programmer is collected from a survey of information technology companies. The data contains a range of criteria values and fuzzy membership degrees [6]. The data represents a form of fuzzy set that represents the state of the candidate before and after recruitment. In the form of fuzzy variables, the set of fuzzy candidate programmers is divided into two linguistic variables namely "Pass" and "Not Pass" of exam testing [10][8]. The formation of this fuzzy set is adjusted based on the opinion of the agile project manager.

2.2. Fuzzy Inference System

A system that performs calculations based on the concepts of fuzzy set theory, fuzzy rules, and the concept of fuzzy logic, namely the Fuzzy Inference System (FIS) [11]. In a fuzzy inference system, there are fuzzy inputs in the form of crisp values[12]. The crisp value will be calculated based on the rules that have been made to produce a fuzzy quantity called the fuzzification process [13].

The Mamdani fuzzy method inference forms a rules-based or rule basis in the form of "cause-effect" or "if-then"[14][15]. The first step in calculating the Mamdani fuzzy method is to make a fuzzy rule or rule. The next step calculated the degree of membership in accordance with the rules that have been made. After knowing the value of the degree of membership of each fuzzy rule, it can be determined the alpha value of the predicate by using fuzzy set operations [16][17].

3. Research Methods

This research is planned to be conducted in two cities, namely in Jakarta and Solo. The research participants are project managers, programmers, software development companies that are still under-5-years startups. Each participant was distributed a survey questionnaire to fill in the projects they had been working on. Prioritized software projects were mobile creation. From all participants will be measured knowledge, competence, attitudes and resources of time and cost, number of teams, number of meetings, work schedules that they allocate to each project.

3.1. Research Measurement Method

In this study, the model is established based of the calculation of some parameters with the steps as below:

1. Recapitulation of the data for the Allocation Team Qualification in accordance with the parameters needed to detect it.
2. Processing the fuzzy data for the Allocation Team Qualification using the Mamdani method.
3. Conducting a comparison of the results of the Mamdani method with the Quality Team data sample.

4. If the results of the Mamdani method are in accordance with the results of the sample data obtained, the results are considered ACCURATE.
5. If the results of the Mamdani method are not in accordance with the results of the sample data obtained when the results are considered NOT ACCURATE.
6. Finally, the percent accuracy of the Mamdani method is calculated by the formula: % Accuracy = (Accurate Data Amount / Sample Total) * 100

3.2. Data Analysis Method

From the interviews with the agile project managers, it was assumed that the manager needs a programmer with feasible and adequate knowledge, skills, attitude and productivity [8]. Knowledge, skill, attitude, and productivity are the four competencies that every programmer should have. The four components above are very important in the development of an IT company. These four parameters have become main attention among the manager to maintain their team productivity. So, the four parameters will be used as input for the designed system. After reviewing the literature of fuzzy sets, we determine the parameters for fuzzification input and output as below:

1. Knowledge of the developer has three linguistic values (high, medium, low)
2. Skill has three linguistic values (high, medium, low)
3. An attitude of the developer has three linguistic values (high, medium, low)
4. Productivity Developers have three linguistic values (high, medium, low)

In this study our lowest range value does not use a zero because it is considered that every prospective programmer already has the basics of knowledge, skills and attitude. for the criteria of knowledge, skill, and attitude we divide into three low, medium and high ranges, while our specific productivity is only divided into two ranges, namely low and high, this is because in this case we only assume productivity in IT development companies, we simplify only high and low.

Table 1. Criteria Details

Criteria	Value Range		
	Low	Medium	High
Knowledge	25-50	65-85	80-100
Skill	25-50	65-85	80-100
Attitude	25-50	65-85	80-100
Productivity	65-80	-	80-100

The next step in the Fuzzy calculation process is to form Fuzzy rules as shown in Table 2.

Table 2. Details of Fuzzy Rules

No	VARIABLES			
	Knowledge	Skill	Attitude	Productivity
1	High	High	High	High
2	High	High	Low	High
3	High	High	Medium	High
4	High	Medium	High	High
5	High	Medium	Low	Medium
6	High	Medium	Medium	High
7	High	Low	High	High
8	High	Low	Medium	Medium
9	High	Low	Low	Medium
10	Medium	High	High	High
11	Medium	High	Low	Medium
12	Medium	High	Medium	High
13	Medium	Medium	High	Medium

No	Knowledge	Skill	Attitude	Productivity
14	Medium	Medium	Medium	Medium
15	Medium	Medium	Low	Medium
16	Medium	Low	High	Medium
17	Medium	Low	Low	Medium
18	Medium	Low	Low	Medium

The programmers are assigned to be in a position of high productivity if they have a final membership value of 80-100, or unproductive if their final membership value is less than 80. Similar steps are repeated for other membership values of each variable as shown in Fig. 1. The process of fuzzification is the calculation of crisp value or input value into the degree of membership. Calculations in the fuzzification process are based on the limits of membership functions. The following is the fuzzy set membership function with 4 input criteria:

1. Fuzzy Set of Knowledge Test ($\mu_{KT_a}; \mu_{KT_a}$)

Each programmer is assigned into a knowledge test. Their test results are then recorded as input values into the fuzzy set member. The results of their tests are given in Figure. 1 which represents the knowledge test result. Each candidate test result is entered into the membership function plot. In the membership function plot; there are three membership groups, namely, low, medium and high. In this study, Mamdani fuzzy logic was used to get the output in the form of a decision in the selection of prospective programmers in IT developer companies. This is supported by research by Jayanti, S., & Hartati [19] who examined the Decision Support System for Adult Choir Members Selection Using the Fuzzy Mamdani Method. According to [19] using Fuzzy Mamdani Logic reasoning in processing input and output data, as well as supporting information in the form of ranking that is very supportive in decision making to determine someone to become a member of the adult choir. Based on the above research, this research uses Fuzzy Mamdani Logic reasoning in processing prospective programmers' selection in IT developer companies.

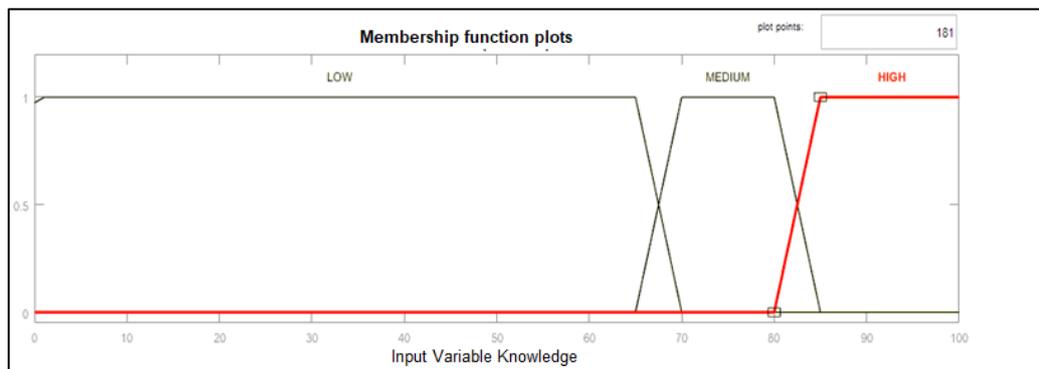


Figure 1. Result of Knowledge test

a) Low Knowledge Level (μ_{KT_a}):

$$\mu_{KT_a}(x) = \begin{cases} 1; & x \leq 65 \\ \frac{70-x}{70-65}; & 65 < x < 70 \\ 0; & x \geq 70 \end{cases} \quad (1)$$

b) Medium Knowledge Level (μ_{KT_b}):

$$\mu_{KT_b}(x) = \begin{cases} 0; & x \leq 65 \text{ or } \geq 85 \\ \frac{x - 65}{85 - 65}; & 65 \leq x \leq 85 \\ \frac{85 - x}{85 - 80}; & 80 \leq x \leq 85 \end{cases} \quad (2)$$

c) High Knowledge (μ_{KT_c}):

$$\mu_{KT_c}(x) = \begin{cases} 0; & x \leq 80 \\ \frac{x - 80}{100 - 80}; & 80 \leq x \leq 100 \\ 1; & x \geq 100 \end{cases} \quad (3)$$

2. Fuzzy set of skill Test (μ_{ST_a} ; μ_{ST_c})

Similar to a skill test, the program candidates are given a skill test. The fuzzy set input value of the skill test was obtained from the results of the candidates' tests. The input value is then recorded in the membership plot as shown in Figure 2. The result of the skill test is entered into the membership function plot. In the membership function plot; there are three membership groups, namely, low, medium and high.

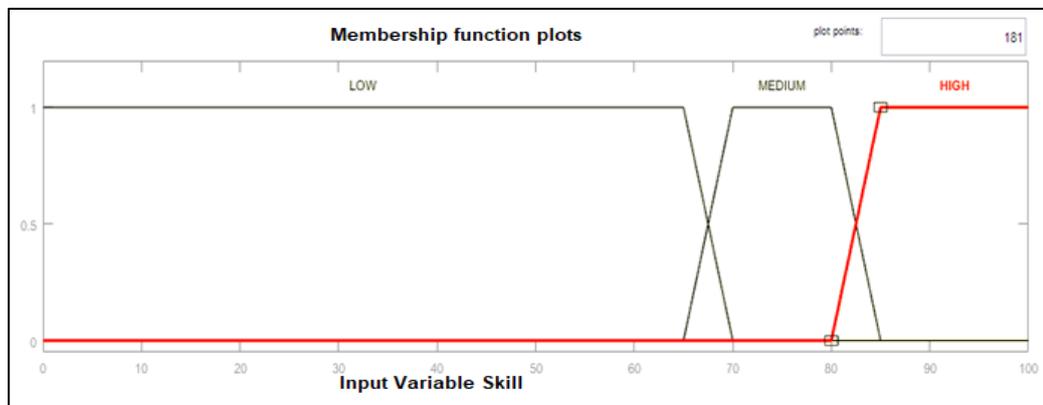


Figure 2. Result of Skill test

a) Low skill level (μ_{ST}):

$$\mu_{ST_a}(x) = \begin{cases} 1; & x \leq 65 \\ \frac{70 - x}{70 - 65}; & 65 < x < 70 \\ 0; & x \geq 70 \end{cases} \quad (4)$$

b) Degree of moderate skill (μ_{ST_b}):

$$\mu_{ST}(x) = \begin{cases} 0; & x \leq 65 \text{ or } \geq 85 \\ \frac{x - 65}{85 - 65}; & 65 \leq x \leq 85 \\ \frac{85 - x}{85 - 80}; & 80 \leq x \leq 85 \end{cases} \quad (5)$$

c) High skill degree (μ_{ST_c}):

$$\mu_{ST_c}(x) = \begin{cases} 0; & x \leq 80 \\ \frac{x - 80}{100 - 80}; & 80 \leq x \leq 100 \\ 1; & x \geq 100 \end{cases} \quad (6)$$

3. Attitude Fuzzy Test Set ($\mu_{AT_a}; \mu_{AT_c}$)

In the variable when fuzzy set membership will also be formed for the aptitude test. Here the input is obtained from the results of attitude skills from candidate programs. The membership function is formed by the antecedents and consequences of attitude rules. By collecting the membership referred to by the antecedents of the attitude rule, three aggregate weighted groups will be formed, namely low, medium, and high. The input value of the aptitude test results will be mapped as the input attitude variable as shown in Figure. 3.

We determine the rule that to be accepted into the membership function plot, the candidate must obtain a position from 0.5 to 1. Lift 1 if the candidate is between 0.5 and above it can still be accepted in the group. Candidates are fully grouped into antecedents set according to these limits. From the results of the candidate attitude test, three types of membership plots were obtained, namely low, medium, and high. The highest limit to be fully accepted is value 1 while for the centroid limit of 0.5 the meeting between low and medium are only accepted into the fuzzy set with the centroid 0.5, the rest the candidates are rejected.

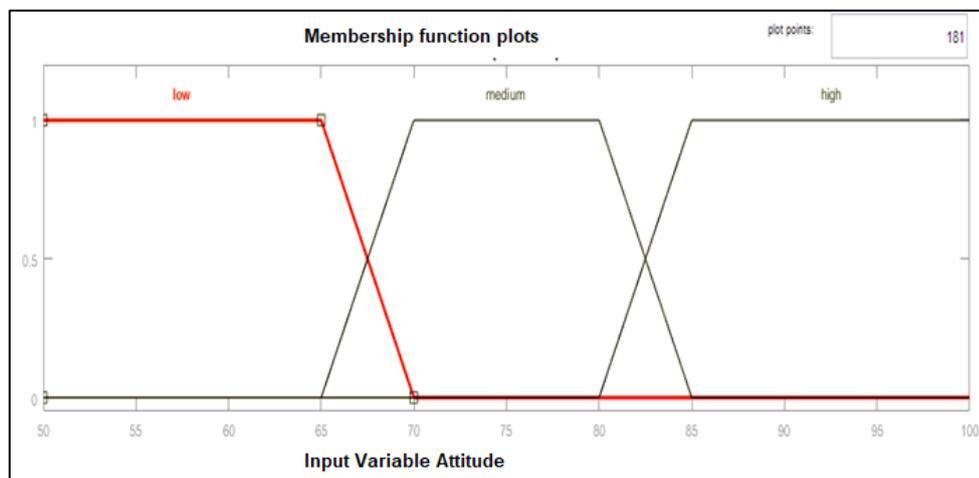


Figure 3. Attitude Variable

a) Low attitude degree (μ_{AT_a}):

$$\mu_{AT_a}(x) = \begin{cases} 1; & x \leq 65 \\ \frac{70-x}{70-65}; & 65 < x < 70 \\ 0; & x \geq 70 \end{cases} \quad (7)$$

b) Degree of moderate attitude (μ_{AT_b}):

$$\mu_{AT_b}(x) = \begin{cases} 0; & x \leq 65 \text{ or } \geq 85 \\ \frac{x-65}{85-65}; & 65 \leq x \leq 85 \\ \frac{85-x}{85-80}; & 80 \leq x \leq 85 \end{cases} \quad (8)$$

c) Degree of attitude is high (μ_{AT_c}):

$$\mu_{AT_c}(x) = \begin{cases} 0; & x \leq 80 \\ \frac{x-80}{100-80}; & 80 \leq x \leq 100 \\ 1; & x \geq 100 \end{cases} \quad (9)$$

4. Fuzzy set of Productivity (μ_{PT_a} ; μ_{PT_c})

In accordance with the purpose of this study, which is to measure the highest productivity by selecting candidate programs, productivity is considered very important. For this reason, the productivity variables are divided into two groups, namely moderate and high. For this reason, an index line is created representing productivity across the membership function line which determines the extent to which the productivity rules must range from moderate to high to be activated.

The two rules form a row of productivity plots. By looking at the antecedents of each rule, it was determined that three plots of moderate productivity and high productivity could be obtained.

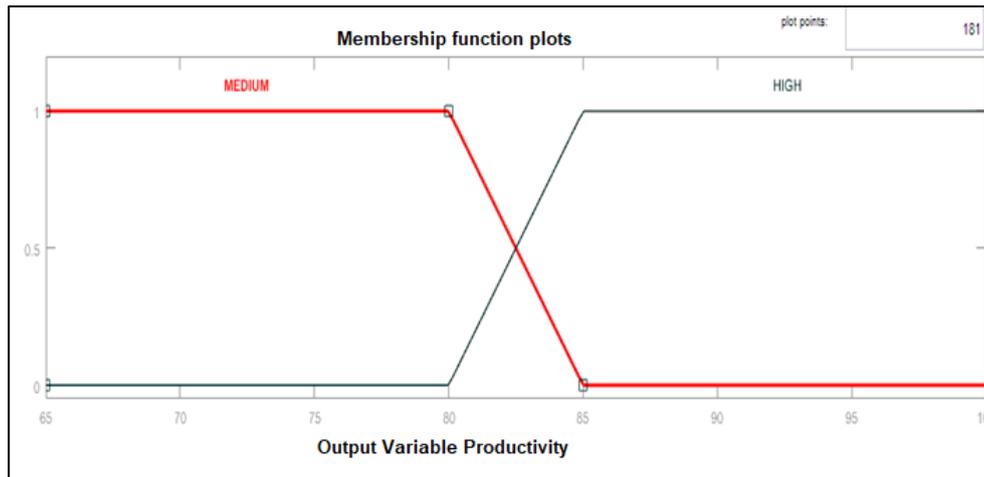


Figure 4. Productivity variables

a) Degree of moderate productivity (μ_{PT_a}):

$$\mu_{PT_a}(x) = \begin{cases} 0; & x \leq 65 \text{ or } \geq 80 \\ \frac{x - 65}{80 - 65}; & 65 \leq x \leq 80 \end{cases} \quad (10)$$

b) Degree of high productivity (μ_{PT_b}):

$$\mu_{PT_b}(x) = \begin{cases} 0; & x \leq 80 \\ \frac{x - 80}{100 - 80}; & 80 \leq x \leq 100 \\ 1; & x \geq 100 \end{cases} \quad (11)$$

3.3. Defuzzification

The final step in the Fuzzy Mamdani method is to find the output value in the form of a crisp (z) value known as the defuzzification process. The method used in this process is the Center Average Defuzzifier method. The method is explained in the equation below.

$$Z = \frac{\sum(\alpha_{p_i} * z_i)}{\sum \alpha_{p_i}} \quad (12)$$

where:

Z = defuzzification of centered average (result)

α_p = alpha predicate value (minimum value of membership degree)

Z_i = crisp value obtained from the results of inference.

i = number of fuzzy rules

We also provide the manual calculation of Fuzzy Mamdani method. We want to know that the proposed model will work in real situation. We therefore calculate the input value to get output crisp (z) values (A1, A2, A3) as below.

$$A1 = (67.5-65) \times 0.5 / 2 = 0.625$$

$$A2 = (70-67.5) \times 0.5 / 2 = 0.625$$

$$A3 = (82.5-80) \times 0.5 / 2 = 0.625$$

The last step is to do the defuzzification process using the method of centroid where using the equation $X = M1 + M2 + M3 / A1 + A2 + A3$,

where $M1 = 0$,

$$M2 = (0.08 * 82.25^2) - (0.08 * 80^2) = 29.20,$$

$$M3 = (0.15 * (82.5^2)) - (0.15 * (80^2)) = 60$$

$$X = 0 + 29.20 + 60 / 0.625 + 0.625 + 0.625$$

$$X = 75.57$$

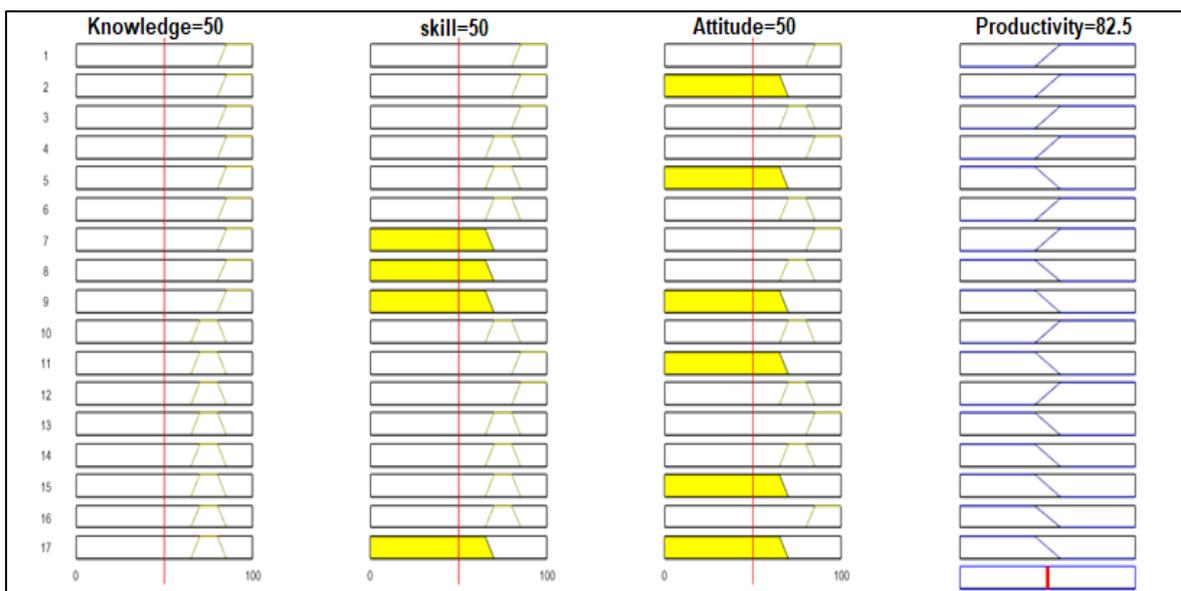


Figure 5. Comparison result of the effect of knowledge skill attitude toward productivity

There are 17 candidates who meet the requirements and pass the tests of attitude, skill, and knowledge ($n = 17$). They are then combined to produce the highest production value. In Figure. 5 it shows the middle boundary of three fuzzy sets, namely set, attitude, skill, knowledge, and productivity with each has values of 50, 50, 50, and 82.5, respectively. By following fuzzy rules, rules are determined as a road map of all fuzzy inference processes. This is based on the fuzzy inference diagram described in the previous section. The picture above shows the composition of each variable with an input that can be seen in the yellow input box. The red line color is a line to change the input value and produces a new output response. The output is in the rightmost box that is blue. So, the output can be directly displayed based on the input entered. The result shows a number which is the amount of productivity.

The membership function is determined based on the antecedents and consequences of the rules of knowledge, skill, attitudes. Each rule forms productivity plot. By looking at the antecedents of each rule, it is determined that three variables are membership functions referenced by the antecedents of each rule. Furthermore, productivity plots represent aggregate weighted decisions for the proposed inference system. This decision will depend on the input value of the candidate test results into the system. The candidate test results are then mapped as three parameters (e.g., knowledge, skill, and attitude) to predict productivity. The results are given in Figure 6 that there are two groups of productivity, namely moderate productivity, and high productivity.

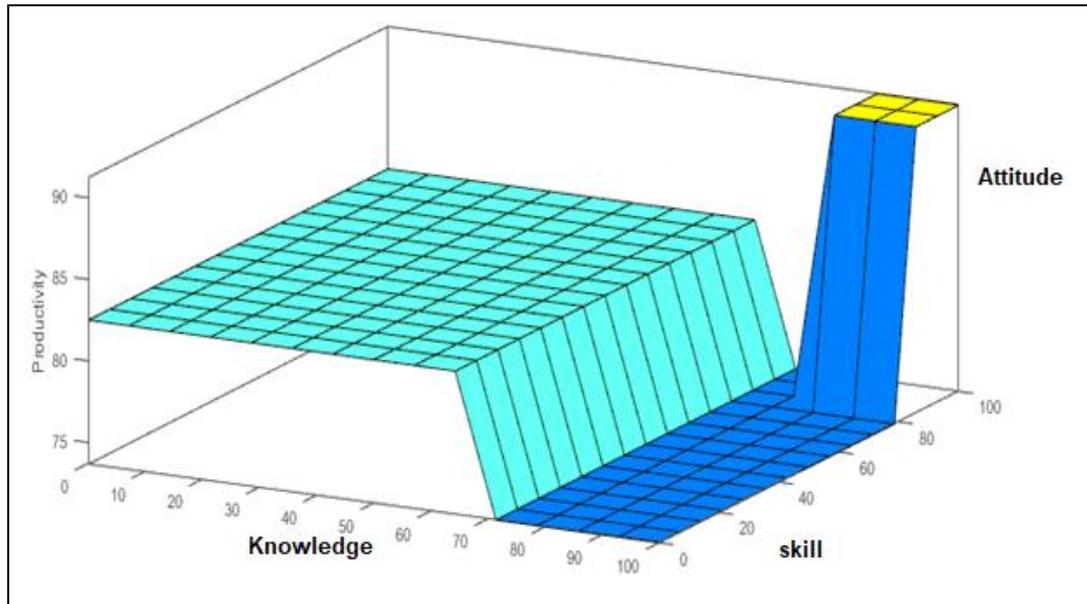


Figure 6. Comparison of knowledge, skills, and productivity

To produce productivity between 80 and 85, programmers must have knowledge above 73, average skills are also above 80 and attitude reaches 90 and above. The highest productivity will be achieved if the recruited programmers have knowledge above 70. This is because the index line representing productivity crosses the knowledge membership function line in the left plot, so it determines the extent to which the candidate programmers who have minimum knowledge will be activated. The light blue patch under the actual membership function curve shows the value of fuzzy membership visually. The yellow patch under the actual attitude membership function curve shows the value of fuzzy membership for variable attitude. From Fig. 6, it is known that attitude is only owned by a small number of candidate programs, namely only 4 people (according to four yellow boxes).

Productivity variables are formed by input index lines of knowledge and skills. In this way, it can be seen that production ranges from 80 to 85 which means that the project manager must prioritize the programmer's programmer who has knowledge above 70 and skill of at least 80. Although candidate skills can be very high (blue) up to 100 percent, their productivity still only around 80 to 85 percent.

3.4. Quality of the model

Thus, the percent accuracy of the Mamdani method can be calculated to determine the prediction of programmer productivity with the equation:

$$\begin{aligned} \% \text{ Accuracy} &= (\text{Accurate Data Amount} / \text{Total Samples}) \times 100 \\ \text{Accuracy} &= (75.57 / 100) \times 100 = 75.57\% \end{aligned}$$

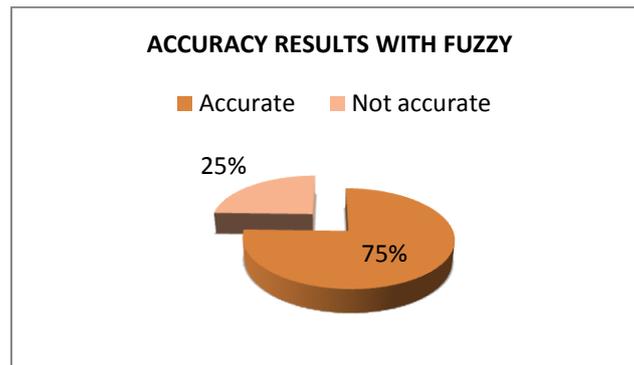


Figure 7. Graph of Mamdani Method Accuracy Results for Programmer Productivity

4. Conclusion

The results of this study use the fuzzy Mamdani method so that it can be implemented to a company to determine the selection of candidate programmers with the results of a comparison between expert ranking and system ranking that produces different values. In testing the system to obtain accurate results, the in this test the accuracy value is 75.57% which indicates that the system is functioning accurately.

References

- [1] F. A. Lopes, M. Santos, R. Fidalgo, S. Fernandes, and S. Member, "A Software Engineering Perspective on SDN Programmability," *IEEE Communications Surveys & tutorials*, vol. 18, no. 2, pp. 1255–1272, 2015.
- [2] I. Couso, C. Borgelt, E. Hüllermeier, and R. Kruse, "Fuzzy Sets in Data Analysis: From Statistical Foundations," *IEEE Computational Intelligence Magazine*, vol. 14, no.1 February 2019, pp. 31–44, 2019.
- [3] F. Bobillo and U. Straccia, "International Journal of Approximate Reasoning Generalizing type-2 fuzzy ontologies and type-2 fuzzy description logics ☆," *International Journal of Approximate Reasoning*, vol. 1, pp. 1–27, 2017.
- [4] A. Sampson, B. Ransford, and L. Ceze, "A CCEPT: A Programmer-Guided Compiler Framework for Practical Approximate Computing," vol. 1, no. 14, pp. 1–14, 2015.
- [5] S. Vesely, C. A. Klöckner, and M. Dohnal, "Predicting recycling behaviour: Comparison of a linear regression model and a fuzzy logic model," *WASTE Management*, vol. 49, March, pp. 530–536, 2016.
- [6] B. M. & H. P. Subhashis Chatterjee, "A fuzzy rule-based generation algorithm in interval type-2 fuzzy logic system for fault prediction in the early phase of software development," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 31, issue 3, pp. 369–391, 2018.
- [7] N. Cerpa, M. Bardeen, C. A. Astudillo, and J. Verner, "Evaluating different families of prediction methods for estimating software project outcomes," *J. Syst. Softw.*, vol. 112, pp. 48–64, 2016.
- [8] M. Panda, "Developing an Efficient Text Pre-Processing Method with Sparse Generative Naive Bayes for Text Mining," *International Journal Modern Education and Computer Science*, vol. 10, no. 9, pp. 11–19, 2018.
- [9] A. Benavoli, G. Corani, J. Demsar, and M. Zaffalon, "Time for a change: a tutorial for comparing multiple classifiers through Bayesian analysis," *Journal of Machine Learning Reseach*, vol. 18, pp. 1–36, 2016.
- [10] A. Fallahpour, E. Udony, O. Siti, and N. Musa, "An integrated model for green supplier selection under fuzzy environment: application of data envelopment analysis and genetic programming approach," *Neural Computer and Application*, April 2015.

- [11] F. Camastra *et al.*, "Expert Systems with Applications A fuzzy decision system for genetically modified plant environmental risk assessment using Mamdani inference," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1710-1716, Februari, 2015.
- [12] F. Rudziński, "A multi-objective genetic optimization of interpretability-oriented fuzzy rule-based classifiers.," *Applied Soft Computing*, vol. 38, pp. 118–133, January, 2016.
- [13] P. Ghadimi, A. Dargi, and C. Heavey, "Sustainable supplier performance scoring using audition check-list based fuzzy inference system: a case application in automotive spare part industry," *Computers & Industria Engineering*, vol. 105, pp. 12-17, March 2017.
- [14] X. Wang, X. Liu, W. Pedrycz, and L. Zhang, "Fuzzy rule based decision trees," *Pattern Recognition*, vol. 48, no. 1, pp. 50–59, 2015.
- [15] J. A. M. R. Wikström and C. Carlsson, "Mobile Decision Support with Fuzzy Ontology," *Decision Support Systems*, vol. 81, pp. 66–75, January 2016.
- [16] S. Rajak and S. Vinodh, "Application of fuzzy logic for social sustainability performance evaluation: a case study of an Indian automotive component manufacturing organization," *Journal of Cleaner Production*, vol. 108, pp. 1–9, 2015.
- [17] K. Grzegorz, A. Gola, and Ś. Antoni, "Application of Fuzzy Logic in Assigning Workers to Production Tasks," *Adv. Intell. Syst. Comput.*, vol. 13, no. 474, pp. 505–506, 2016.
- [18] P. Serrador and J. K. Pinto, "ScienceDirect Does Agile work ? — A quantitative analysis of agile project success," *International Journal of Project Management*, vol. 33, no. 5, pp. 1040–1051, July, 2015.
- [19] S. Jayanti and S. Hartati, "Sistem Pendukung Keputusan Seleksi Anggota Paduan Suara Dewasa Menggunakan Metode Fuzzy Mamdani," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 6, no. 1, 2012.