Annals of Emerging Technologies in Computing (AETiC)
Vol. 5, No. 1, 2021

Research Article

Fuzzy-based Adaptive Framework for Module Advising Expert System

Obada Alhabashneh

Mutah University, Karak, Jordan
o.alhabashneh@mutah.edu.jo

Received: 3rd November 2020; Accepted: 22nd December 2020; Published: 1st January 2021

Abstract: In the enrolment process, selecting the right module and lecturer is very important for students. The wrong choice may put them in a situation where they may fail the module. This could lead to a more complicated situation, such as receiving an academic warning, being de-graded, as well as withdrawn from the program or the university. However, module advising is time-consuming and requires knowledge of the university legislation, program requirements, modules available, lecturers, modules, and the student's case. Therefore, the creation of effective and efficient systems and tools to support the process is highly needed. This paper discusses the development of a fuzzy-based framework for the expert recommender system for module advising. The proposed framework builds three main spaces which are: student-space (SS), module-space (MS), and lecturer-space (LS). These spaces are used to estimate the risk level associated with each student, module, and lecturer. The framework then associates each abnormal student case in the students' grade history with the estimated risk level in the SS, MS, and LS involved in that particular case. The fuzzy-based association-rule learning is then used to extract the dominant rules that classify the consequent situation for each eligible module if it is to be taken by the student for a specific semester. The proposed framework was developed and tested using real-life university data which included student enrollment records and student grade records. A five-fold cross-validation process was used for testing and validating the classifying accuracy of the fuzzy rule base. The fuzzy rule base achieved a 92% accuracy level in classifying the risk level for enrolling on a specific module for a specific student case. However, the average classifying accuracy achieved was 89.2% which is acceptable for this problem domain as it involves human behavior modeling and decision making.

Keywords: ***Intelligent Academic Advisor; Module Adviser; Expert System; Fuzzy Logic; Fuzzy Rule-based.***

---

1. Introduction

Every academic semester, students enroll in different modules offered by their universities. Enrolment might sound like a simple process, but it involves many complications and problems for students particularly in these universities where the credit hour system is adopted. According to this system, although there are some constraints such as the academic plan, the students still have the freedom to select the sections to enroll in based on their preferences including the module, timeslot, and lecturer of the offered sections in the timetable. However, the lack of knowledge about the modules and the lecturers, coupled with the contradictory advice they receive from their colleagues and the complexity of the academic plan of their programs may affect the students' ability to select the right choice. This creates a need for professional advice as taking a wrong choice may lead to unwanted situations such as failing the module, or more seriously, academic warning, program withdrawal or even leaving the university [1-3].

To address this problem, universities usually have academic advisors to help the students in selecting their modules and sections for the upcoming semester based on different factors including the student’s cumulative Grade Point Average (GPA), the available modules, timeslots, module’s lecturer, and the academic plan of the student. However, module advising is a challenging task as it is complicated, time-consuming, and needs qualified and knowledgeable human resources. This urges the need for developing intelligent systems for module-advising to aid the students in selecting the right modules at the right time to ensure a smooth passage for students throughout their academic plans to finish their degrees with fewer problems.

Intelligent Recommender Systems (RSs) are widely used in different application areas including online shopping, movies, social networking, and others. However, they are less common in education and this area is still under development [1-3]. RSs can provide the students with personalized recommendations on suitable modules based on the student’s academic history. Such systems can also incorporate previous student's experiences to provide an improved recommendation. Furthermore, they adapt to the changes in the data and give the recommendation based on that. The need for RSs and their requirements in the education sector has been discussed in [4-6]. More recently, the use of RSs in education has received more attention and their potential has been proven [7, 8].

This paper discusses the development of a fuzzy-based framework to be used in recommender systems for module advising. The proposed framework builds the recommendation based on creating three main spaces which are: student-space (SS), module-space (MS), and lecturer-space (LS). These spaces are used to estimate the risk level associated with each student, module, and lecturer. The framework then associates each abnormal student case in the students’ grade history with the estimated risk level in the SS, MS, and LS involved in that particular case. Fuzzy-based association-rule learning [9,10] is then used to extract and summarize a fuzzy rule base. The fuzzy rule base is used to predict the risk level associated with the combination of a specific student, module, and lecture and build the recommendation to students based on that. The proposed approach provides a novel mechanism to estimate the consequent risk associated with the student selection of a specific module that is taught by a specific lecturer. The risk estimation is based on creating and analyzing the three space elements.

2. Related Work

2.1. Course/Module Advising in Higher Education

Academic advising is vital for a good academic experience of students as it positively impacts their success and retention [11, 12]. However, it requires specific knowledge of the student's situation, history, program’s academic plan, modules, and lecturers [12]. Gordon *et al*. [13], defined academic advising as “situations in which an institutional representative gives insight or direction to a college student about an academic, social, or personal matter”. Module advising is one of the main tasks of the academic advisor, which refers to the process by which the academic advisor supports the student in selecting the right modules in which to enroll. Having the importance and the complexity of module advising, different researchers have argued that there is a need for developing intelligent systems to support this task [2, 4, 7, 14]. Developing such systems aims to minimize the demand for human advisors and gives them more time to focus on other important advising tasks including career development and financial issues. [12-16].

A novel approach for module long term plaining called Interactive Decision Support for Course Planning (IDiSC+) was proposed by Mohamed [12]. The approach used optimization techniques to support both the student and the advisor in building a recommended long-term academic plan (towards graduation) to be followed by the student. Laghari [14] developed an Automated Course Advising System (ACAS) for module planning. The system distributes the modules of the academic plan on different semesters based on the history of other students.

There has been significant research effort in applying information technology for module advising. Roushan *et al.* [15] introduced an Internet-based approach to support the module-advising process which integrated the process of advising with the enrolment taking into account the constraints of the program plan and the university policy. However, the system did not replace the human academic-advisor but rather facilitated the advisor role by providing an automated tool for communication and information exchange. A decision-support system was proposed to support academic advisors in preparing a pre-enrolment plan for the students and assist in the offering of appropriate modules for the upcoming semester [16]. Al-Ghamdi *et al.*[17] developed an advisor expert system (PAS) for postgraduate students. The proposed system was designed to assist them in the selection of the most relevant modules without referring to a human module-advisor. A web-based advising system [18] was proposed which supports three types of users, including students, advisors, and heads of department to make sure that a complete picture is available for the students. Mattei *et al.*[19] developed a decision-theory advising tool to enhance the advisor-student relationship. The tool allowed students to browse the module offerings, possible future scenarios, and their outcomes. Shatnawi *et al.* [20] used the enrolment and marking history from similar cases and applied an association rule-based system to give general recommendations when selecting the modules on which to enroll.

2.2. Content-based

Different researchers have used a content-based academic recommendation system for module selection. For example, Lin *et al*.[21] utilized a multi-agent approach and ontology to provide a dynamic and personalized recommended module list. The multi-agent approach, which included various agents, used a preference-driven planning algorithm supported by the ontology to build the recommendations. Darimola *et al*.[4] integrated Case-Based Reasoning (CBR) with Rule-Based Reasoning (RBR) techniques to provide an intelligent approach for module advising. The approach also used historical data to build a list of recommended modules for the following semester.

2.3. Collaborative filtering

Collaborative filtering was also used for academic advising. Huang *et al.* [22] proposed a score-based prediction approach for course recommendation. The approach used a user domain collaborative filtering to create the recommended course list. Courses were clustered based on the student feedback in [23] and the resulted clusters were combined with fuzzy based rule association technique to create the course recommendation. Nafea *et al*. [24] developed a learning-style-based collaborative-filtering approach for module recommendations which utilized different metrics to identify similar profiles including k-means, cosine, and person correlation. Chang *et al*. [25] proposed a user-based collaborative-filtering approach to predict student grades. Mortenson *et al*. [26] introduced a collaborative filtering approach for module selection which utilizes an artificial immune mechanism for the prediction. Bydžovská [27] investigated the effect of student and module features on the enrolment patterns and designed a collaborative-filtering-based system to predict the module grade. Yao *et al*. [28] attempted to increase the fairness of the module recommendation by addressing the biased-recommendation problem against minority groups of students. They developed four different fairness metrics that can be optimized using the learning objectives

2.4. Knowledge-based

Different knowledge-based recommendation approaches have been proposed for module selection. Xu *et al.* [29] developed a knowledge-based approach to offer a personalized module sequence to the new students. This approach utilized a dynamic learning algorithm that learns from the performance of other students on a specific module. Koutrika [30] argued that recommendation methods should not be 'hard-wired', but it should be flexible. In that sense, a new paradigm for the recommendation was introduced in which a recommendation approach can be defined declaratively as a high-level parameterized workflow comprising traditional relational operators and new operators that generate or combine recommendations. Keston *et al*. [31] utilized semantic web expert system technologies to build a knowledge base that is used by an intelligent web-based application to provide the required recommendations. Engin *et al*. [32] developed an expert rule-based system for module recommendation. The rules were captured from the real advisors and then injected into a rule-base using Oracle Policy Automation (OPA).

Hashemi and Blondin [33] included several factors to be taken into consideration when recommending modules for students such as the frequency of the module offering, balancing the module load, and shortening the graduation path. All these factors and others were included in a rule base which was used for recommending modules. Ayman [34] proposed an expert system for module selection which included both prescriptive and developmental advising models and utilized object-oriented databases for data and rule representation. Abdullah Al-Ghamdi *et al*. [17] proposed a rule-based advising system. The system was designed for postgraduate students and built recommendations to support the students in the selection of modules related to the topic of their research thesis. Nambia and Dutta [35] introduced a dynamic and flexible rule-based advising system in which the rules were separated from the execution to enable the student to try different scenarios by updating the XML file where the rules are stored. Nguyen *et al*. [36] proposed a knowledge-based framework that utilized a learning data-warehouse for discovering patterns in the student behavior including module selection and achievements. These patterns were then used to make the recommendations.

2.5. Hybrid

Other researchers proposed Hybrid approaches in which more than different perspectives were integrated. Daramola. *et al*. [4] designed an intelligent expert system for module advising which integrated rule-based reasoning (RBR) with Case-Based Reasoning (CBR) using the academic history of the students. Sobecki [37] applied Ant Colony Optimization (ACO) to provide an efficient module advisor system. The system predicted the final grade of the students in a module, based on a domain-specific representation integrated with the ACO. Abdulwahhab [38] integrated Genetic Algorithms (GAs) and a Decision Tree for short-term module-scheduling.

2.6. Fuzzy Based

A few researchers have used fuzzy logic to develop module advising systems. Goodarzi and Rafe [39] developed a fuzzy-based expert system for student advising. The proposed system was a web-based module that can be integrated with the university portal. The module fuzzifies the business rules and the GPA of the students to advise them on which modules should be taken in the following term. Adak [40] used Fuzzy techniques to recommend elective modules to students. It analyzed transcripts of students to extract fuzzy rules to relate the module to the students who intend to take them. Baloul and Williams [41] used the Order of Preference by Similarity to Ideal Solution (TOPSIS) technique to develop a fuzzy model for probation students to minimize the risk of taking the wrong modules in the early stage of their study.

3. The proposed approach

This paper discusses the development of a fuzzy-based framework of an expert system for module advising. The proposed framework assumes that recommendations are based on three main elements which are: student-space (SS), module-space (MS), and lecturer-space (LS). SS contains the current student status which includes their accumulative average (if there is any academic warning), the closest abnormal academic status and to which extent it's close to that status. It also contains the knowledge domains of the academic program and the progress of the students in each domain. MS contains the average module mark for the last five years and the knowledge domains that the module belongs to. LS contains the average of the marks for the lecturer for all modules and the average of the lecturer's marks for each module. The three spaces and the result of the final calculation are then combined in a matrix called the case-space (CS). Fuzzy based association-rule learning is then used to extract the dominant rules to classify the consequent case for each eligible module if it was taken by the student for a specific semester. The fuzzy logic is used to handle the uncertainty involved in modeling the human decision and to provide a transparent and interpretable mechanism module taking risk estimation.

The main purpose of the proposed framework is to estimate the consequent risk level (Low, Medium, and High) taking a specific module. The risk level is assigned based on a list of unwanted cases associated with the student failing or not progressing in the module. These cases include a GPA decrease, moving down from one GPA category to a lower one (degrading), receiving an academic warning, withdrawn from the academic program, withdrawn from the university, and graduation pending. In addition to the risk-level estimation, the framework provides the student with a justification (interpretation) for the estimated risk-level based on the student's situation, the targeted module, and the lecturer. The framework was developed based on real-life university data which included a historical record of enrolment associated, student's marks, cumulative GPA, module offerings, academic plans for the programs, and finally the knowledge domains for both the programs and the modules. The proposed approach is depicted in Figure 1

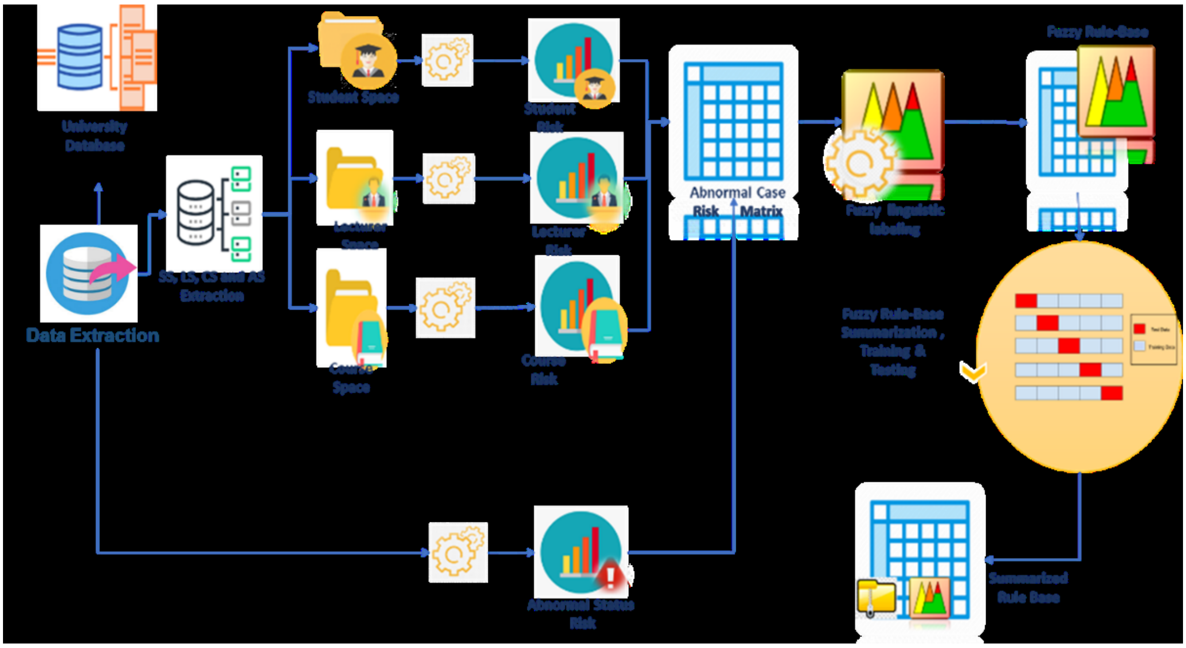


Figure 1: The proposed framework

Fig. 2 shows the use case diagram of the proposed system.

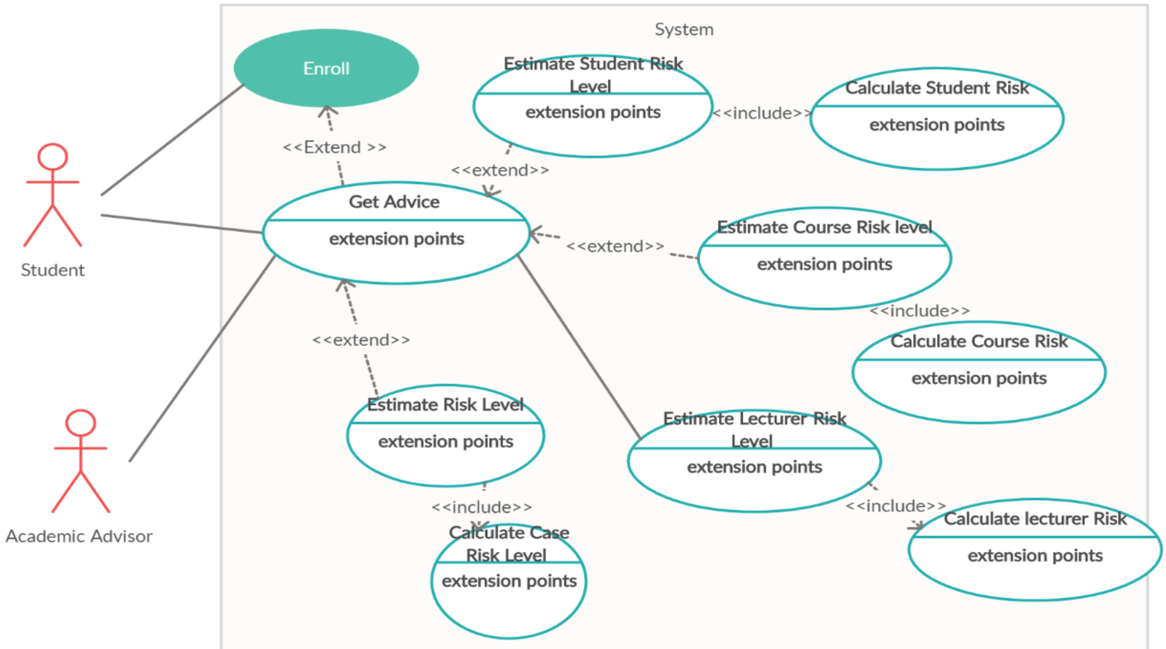


Figure 2: System use case diagram

The proposed approach consists of six main steps:

3.1. Step-1: Creating the Three Spaces and Abnormal Case Matrix

In this step, the student-space (SS), module-space (MS), lecturer-space (LS), and abnormal-case matrix are created as follows:

3.1.1. Step-1.A: Creating the three Spaces

The three spaces SS, MS, and LS are created as shown in Fig. 3**Error! Reference source not found.**. The spaces are extracted from the university database.

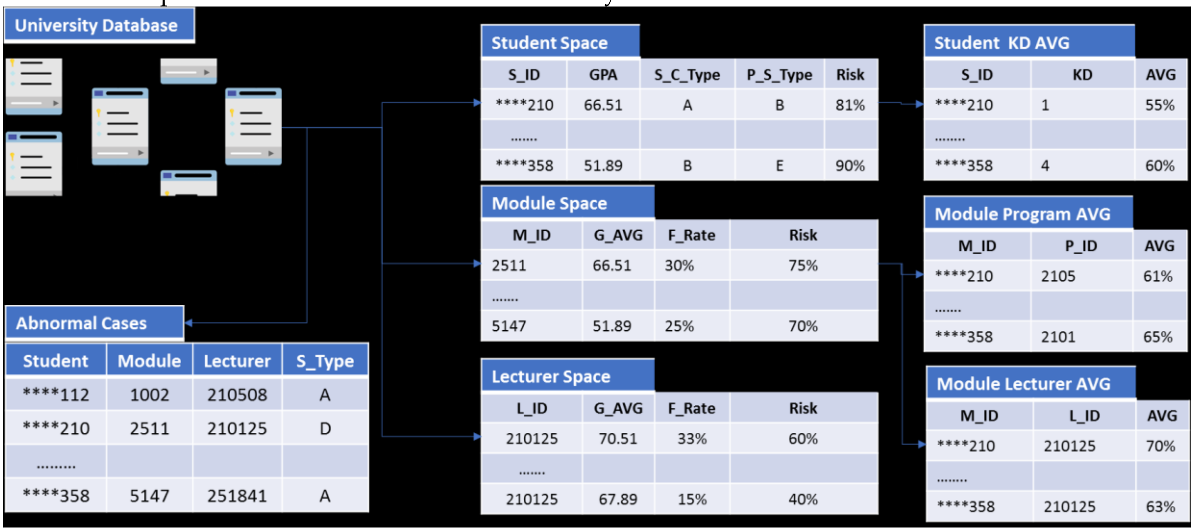


Figure 3: The three spaces and abnormal case matrix

Student space: this space contains information about the student as shown in Table 1.

Table 1: Student space

	Element	Description
1.	Student Id	A unique identification number for the student
2.	Cumulative GPA	The accumulative average of the student out of 100
3.	Student Average Mark for the Program Knowledge Domains	This is a matrix that includes the knowledge domains of the student academic program and the average mark of the student for the modules which belong to each domain.
4.	Potential Risk Situation	This refers to the approximate abnormal student situation for the current situation of the student. (i.e., the current abnormal situation of the student boundary between two GPA categories, such as "very good" and "satisfactory"). Each abnormal-situation type is given a code and a percent of 100 to indicate the level of risk which is given to the code.
5.	Current Student Situation	This refers to the student's current abnormal situation. i.e., If the current abnormal situation of the student boundary between two GPA categories, such as "very good" and "satisfactory", or the student's GPA has dropped. Each abnormal-situation type is given a code and a percent of 100 to indicate the level of risk which is given to the code.

Module Space: this space contains information about the module as shown in Table 2.

Table 2: Module space

	Element	Description
1.	Module Id	A unique identification number of the Module
2.	General Average of Grades	The average grades for the students in the Module
3.	Average grade for each program	A matrix includes the academic programs which contain the module and the average module grade for each program
4.	Average grade for each Lecturer who taught the module	A matrix which includes the lecturers who taught the module and the average grade for the module given by each Lecturer
5.	Fail Rate	The percentage of students who failed the module

Lecturer Space: This space contains information about the module as shown in Table 3.

Table 3: Lecturer space

	Element	Description
1.	Lecturer Id	A unique identification number of the lecturer
2.	General Average of Grades	The average grade of the lecturer
3.	Fail Rate	The percentage of students who failed the modules taught by the lecturer.

3.1.2. Step-1.B: Creating Abnormal Case Matrix

In this sub-step, the abnormal-case matrix is created by selecting the student id, module id, lecturer id, and the student abnormal situation for abnormal cases in the student enrolment records from the university database. The abnormal case is usually indicated by a flag or a symbol in the

Abnormal Cases

Student	Module	Lecturer	S_Type
****112	1002	210508	A
****210	2511	210125	D
………			
****358	5147	251841	A

Symbol	Case
A	GPA Drop
B	GPA Category Drop
C	Academic Warning
D	Program Withdrawal
E	University Withdrawal
F	Graduation Pending
….	….

database as shown in Figure **4.**

Abnormal Cases

Student	Module	Lecturer	S_Type
****112	1002	210508	A
****210	2511	210125	D
………			
****358	5147	251841	A

Symbol	Case
A	GPA Drop
B	GPA Category Drop
C	Academic Warning
D	Program Withdrawal
E	University Withdrawal
F	Graduation Pending
….	….

Figure 4: Abnormal-case matrix with case mapping

3.2. Step-2: Case Risk Analysis and Calculation

For each problematic case in the dataset, a risk weight is calculated for the three spaces: SS, MS, and LS and the associated abnormal situation. The product of this step is shown in Fig. 5. The risk was calculated for the three spaces using equations developed based on common sense and the opinion of field experts such as the academic registry members, lectures, and head of departments. The results than have been presented to the experts and evaluated by them.

Case Risk Analysis

S_ID	S_Risk	M_ID	M_risk	L_ID	L_Risk	S_Type	S_Risk
****112	60%	1002	80%	210508	40%	A	30%
****210	30%	2511	70%	210125	35%	D	80%
………….							
****358	20%	5147	20%	251841	80%	A	30%

Figure 5: Case-Risk Analysis and Calculation

Risk estimation of student-space:

$$\boldsymbol{SR(SS) = \left(100 - \left(\frac{GPA(SID) + KDA(MID, SID)}{2}\right) + PAC + CAC\right)/3}$$

Equation 1: Risk estimation of student-space

*Where **SR** is the risk estimation of the student space in relation to the current case. It takes the student space as input and returns a percentage value out of 100. SS is the student space. **GPA** is the student cumulative **GPA, SID** is the student Id, **KDA** is the student average mark of the knowledge domain to which the module belongs. PAC is the scaled weight (out 100) of the potential abnormal situation. **CAC** is the scaled weight (out 100) of the current abnormal situation of the student.*

Risk estimation of module-space:

$$\boldsymbol{MR}(\boldsymbol{MS}) = \left(\mathbf{100} - \frac{\boldsymbol{MAG}(\boldsymbol{MID}) + \boldsymbol{MPA}(\boldsymbol{MID}, \boldsymbol{PID}) + \boldsymbol{MLA}(\boldsymbol{MID}, \boldsymbol{LID})}{\mathbf{3}} + \boldsymbol{MFR}(\boldsymbol{MID})\right)/\mathbf{2}$$

Equation 2: Risk estimation of module-space

*Where **MR** is the risk estimation of the module in relation to the current student situation. It takes the Module Space (**MS**) as an input and returns a percentage out of 100, **MS** is the module space. **MAG** is the general average mark of the module. **MID** is the Module identification number; **MPA** is the average module mark for the student's program. **PID** is the identification number of the student's academic program. **LID** is the lecturer identification number. **CAI** is the average mark of the Module for the Lecturer. **MFR** is the Module fail rate which is a percentage of the students who failed the Module.*

Risk estimation of lecturer-pace:

$$\boldsymbol{LR}(\boldsymbol{LS}) = ((\mathbf{100} - \boldsymbol{LAG}(\boldsymbol{LID})) + \boldsymbol{LFR}(\boldsymbol{LID}))/\mathbf{2}$$

Equation 3: Risk Estimation of Lecturer Space

*Where **LR** is the risk estimation that comes from the lecturer side out 100, **LS** is the lecturer-space. **LAG** is the general average mark of the lecturer. **LID** is the lecturer-identification number. **LFR** is the lecturer fail rate which is a percentage of the students who failed the modules taught by the lecturer.*

Situation risk (SR) estimation:

The resulting situation types were provided to a set of experts (i.e., academic-registry staff, advisors, and heads of department) and they were asked to assign a specific weight which reflects the risk level. The risk level takes a value between 1 and 100. The net risk for each situation type is calculated by taking the average of the weights given by the experts.

3.3. Step-3: Linguistic Labelling

In this step, the final risk weights of the three spaces and the resulting situation are given linguistic labels of H-high, M-medium, and L-low. These linguistic labels are generated using Type-1 and based on predefined fuzzy sets based on the Mendel Wang method [10]. In the proposed approach, the member function of the fuzzy sets uses a triangle shape as shown in Figure 6 and is based on three parameters {a, b, c} as shown in Equation 4.

$$Triangle(x; a; b; c) = \begin{cases} 0, & x \leq a. \\ \dfrac{x-a}{b-a}, & a \leq x \leq b. \\ \dfrac{c-x}{c-b}, & b \leq x \leq c. \\ 0, & c \leq x. \end{cases}$$

Equation 4: Triangle fuzzy set

Where the parameters *{a, b, c}* (with *a*<*b*<*c*) determine the x coordinates of the three corners of the underlying triangular MF as shown in Fig.r (6).

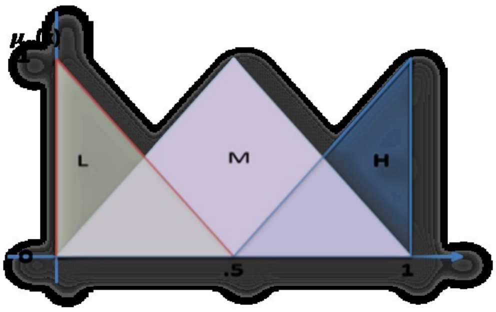


Figure 6: Fuzzy member function for input and output variables

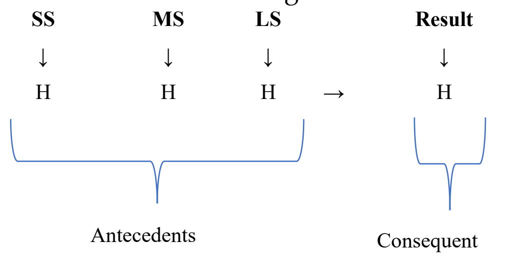
The outcome of this step is a linguistic label of **H-high**, **M-medium,** and **L-low** for each of the three spaces and the resulting situation associated with each abnormal case in the student record as shown in Figure 7.

Case Risk Linguistic Labels						
S_ID	S_Risk	M_ID	M_risk	L_ID	L_Risk	S_Type
****112	M	1002	H	210508	M	L
****210	L	2511	H	210125	L	H
…………						
****358	L	5147	L	251841	H	H

Figure 7: Case risk linguistic labels

3.4. Step-4: Fuzzy Rule-Base Creation

To create the fuzzy rule base, in this step and for each abnormal case, the linguistic labels resulting from the previous step are associated with each other in a form of antecedents and consequents (If-> Then) where the linguistic labels of the three spaces are the antecedents and the linguistic label of the resulting situation is the consequent as shown in Figure 8.



Rule Base	[redacted]			
S_Risk	M_risk	R_Risk	[redacted]	S_Type
M	H	M	[redacted]	H
L	H	L	[redacted]	M
…….			[redacted]	
L	L	H	[redacted]	M

Figure 8: Rule base creating

3.5. Step-5: Fuzzy Rule-base Compression and Validation

The rule base extracted in the previous step could contain many rules based on the size of the dataset. It may have a large amount of repetition (i.e., Repeated rules) and contradiction. The contradiction here means rules that have the same antecedent with different consequents such as shown in the following example in Table 4:

Table 4: Contradictory rule patterns

R(SS)	(MS	IS		Result
↓	↓	↓		↓
H	**H**	**H**	→	**H**
H	**H**	**H**	→	**L**
H	**H**	**H**	→	**M**

To address these issues a five-fold cross-validation [42] process was used to train the rule base to summarize it to the most dominant unique patterns. In this validation method the dataset is divided into 5 equal-size sets (D1, .., D5).

For each fold the following steps are applied:

1. One of the subsets $\mathbf{D_n}$ is selected as a testing set T_n and the other subsets are groups in one training set **D^**.
2. The rule compression(summarization) technique [43] is applied to the training set to produce a summarized rule-base
3. The summarized rule-base is applied to the test set to predict the risk level.
4. The predicted risk levels are compared with the actual risk levels from the dataset, to identify the accuracy of the rule-base in predicting the risk level

The compression technique: This technique uses two measures for rule quality, which are "generality" and "reliability" and are used to identify rule patterns with maximum firing strength. Generality measures the number of instances in the extracted rule base which supports each rule pattern. Reliability measures the confidence level of each rule pattern [43].

In the proposed approach, generality is calculated using scaled fuzzy support, and the reliability is calculated by multiplying the scaled fuzzy support by the firing strength of the rule pattern. The support of the rule pattern refers to the number of the rules in the rule-base that the pattern represents. The "confidence" refers to the strength of a specific rule pattern against the contradictory patterns (i.e. because other rule patterns have the same antecedent and a different consequent) [43].

Fuzzy support is used to identify the unique rule-patterns with their occurrence in the extracted rule base. The fuzzy support can be scaled for each unique rule-pattern using the total number of instances in the rule-base which have the same consequent. Equation 5 shows how the scaled fuzzy support for a unique rule pattern is calculated:

$$scFuzzSup(\underline{RP_l}) = \frac{N_{\underline{RP_l}}}{N_{\underline{RP_l}} + N_{\widehat{RP}_l}}$$

Equation 5: Fuzzy support

Where l =1 to M, l is the index of the rule pattern, $\underline{RP_i}$ is a unique rule pattern i.e. (H, H, H →M), $N_{\underline{RP_l}}$ is the number of instances in the extracted rule base supporting the rule pattern **RPi**. $N_{\widehat{RP}_l}$ is the number of instances in the extracted rule base which support other patterns with the same consequent i.e. ({M, H, H →M}, {M, M, H →M},…).

The "confidence" is a measure of the uniqueness of the pattern as it indicates its strength against the contradictory patterns, which are the other patterns having the same antecedents but a different consequent. Equation 6 shows how confidence is calculated

$$scConf(\underline{RP_l}) = \frac{scFuzzSup(\underline{RP_l})}{Co_{\underline{RP_l}}}$$

Equation 6: Confidence

Where $Co_{\underline{RP_l}}$ is the number of instances in the extracted rule base which supports the contradictory rule-patterns with $\underline{RP_l}$.

The final scaled-weight of the rule pattern is calculated as the product of the fuzzy support and the fuzzy confidence as shown in Equation 7.

$$scWi = scFuzzSup \times scConf$$

Equation 7: Final scaled weight

Each of the generated unique rule patterns is assigned the scaled fuzzy weight measure scWi as follows:

Table 5: The scaled fuzzy weight of the unique rule-patterns

SS	MS	LS		Result	*scWi*
↓	↓	↓		↓	
H	**H**	**H**	→	**H**	**0.35**
H	**H**	**H**	→	**L**	**0.02**
H	**H**	**H**	→	**M**	**0.12**

The scaled fuzzy weight of the unique rule patterns is then used to select the rules with the highest weights among the contradictory-patterns set. The result of the compression process is a summarized rule-base that contains dominants and consistent rule patterns. The resulting rule base will be used later to estimate the risk level considering a specific module, lecturer, and student.

3.6. Step-6, Compressed Rule Base Selection

The accuracy levels of the five compressed rule-bases resulting from the previous step are compared and the compressed rule-base with the highest accuracy level was selected to be included in the system.

4. Experiment and Results

4.1. Dataset

A real-life university data was used as a dataset for this paper. The dataset included student-enrolment records, student-mark records, module records, academic program records, and records for lecturers. The dataset consisted of the records of 5000 students who faced problems during their studies. These problems included academic warnings, program withdrawals, program changes, and cumulative average grade down-gradings.

4.2. Creating the Three Spaces (SS, MS, and LS)

The university uses an Oracle database for the student information system (SIS), which was used to create a view for each of the three space types. A view was also created to include the risk estimation for each space associated with the risk estimation provided by the expert.

4.3. Creating the Fuzzy Rule-Base

To create the fuzzy rule base, ***MatLab*** Fuzzy Logic Tool Kit was used to create the fuzzy sets, which were then used to create the linguistic labels for the data extracted from the three spaces.

4.4. Fuzzy Rule-Base Training and Compression

A five-fold cross-validation technique was used to train the rule-base, then, for each fold, the fuzzy rule-base was divided into two different subsets, which were training and testing. The 80%-20% rule was used to identify the size of each set. The compression technique discussed in section (3) was applied on the training set to produce the compressed rule-base. The resulting compressed rule-base was applied to the test set to predict the risk level. The predicted risk-levels were compared with the actual risk-levels provided by the experts to determine the accuracy of the rule-base in predicting the risk level as shown in Table 6.

Table 6: Rule base classifying accuracy sample

S_ID	S_Risk	M_ID	M_risk	L_ID	L_Risk	Situation Type	Actual	Predicted	Passed?
****112	M	1002	H	210508	M	A	L	L	1
****210	H	2511	H	210125	L	D	H	M	0
****525	H	1002	M	030410	H	E	H	H	1
****402	L	5147	L	120215	H	D	H	H	1
****111	M	1002	H	130514	M	B	M	M	1
****237	M	2142	L	010816	M	F	M	M	1
****252	H	2151	M	070516	H	D	H	H	1
****332	H	2101	H	220589	L	D	H	M	0

The accuracy level of the test results for each fold is shown in Table 7. As shown in the table the proposed approach achieved 92% classifying accuracy and 89.2 as an average classifying accuracy. The comparison between the proposed approach and the other approaches is not a straightforward task as the main focus of the proposed approach differs from those of the others. The proposed approach aims to address the risk estimation while some others focused on long-term planning or supported the process of the academic advisory as a whole. However, the results can still be compared to indicate the performance of the proposed approach. For example, the multi-agent system proposed in [22] achieved only 60% user satisfaction with its effectiveness. Besides, the user satisfaction with the intelligent advising system proposed in [4] was 77.8% which indicates at the performance proposed system in this paper is acceptable.

Table 7: 5-Fold accuracy test results

Fold	*Number of Rules*	*Accuracy*
1	24	88%
2	25	91%
3	27	92%
4	23	86%
5	23	89%
Average		89.2%

4.5. Best Compressed Rule Base Selection

The accuracy levels of the five compressed rules bases resulting from the previous step were compared and the compressed rule-base with the highest accuracy level was selected to be included in the system as shown in Table 8.

Table 8: Best rule set

I	*SS*	*MS*	*IS*		*Result*	*scWi*
	↓	↓	↓		↓	
1	*H*	*H*	*H*	→	*H*	*0.352*
2	*H*	*H*	*M*	→	*H*	*0.311*
3	*H*	*H*	*L*	→	*M*	*0.273*
4	*H*	*M*	*H*	→	*H*	*0.218*
5	*H*	*M*	*M*	→	*H*	*0.346*
6	*H*	*M*	*L*	→	*M*	*0.438*
7	*H*	*L*	*H*	→	*H*	*0.517*
8	*H*	*L*	*M*	→	*M*	*0.593*
9	*M*	*L*	*L*	→	*L*	*0.433*
10	*M*	*H*	*H*	→	*H*	*0.214*
11	*M*	*H*	*M*	→	*M*	*0.511*
12	*M*	*H*	*L*	→	*M*	*0.162*
13	*M*	*M*	*H*	→	*M*	*0.169*
14	*M*	*M*	*M*	→	*M*	*0.283*
15	*M*	*M*	*L*	→	*M*	*0.364*
16	*M*	*L*	*H*	→	*M*	*0.407*
17	*M*	*L*	*M*	→	*M*	*0.502*
18	*L*	*L*	*L*	→	*L*	*0.584*
19	*L*	*H*	*H*	→	*H*	*0.224*
20	*L*	*H*	*M*	→	*M*	*0.436*
21	*L*	*H*	*L*	→	*M*	*0.365*
22	*L*	*M*	*H*	→	*M*	*0.156*
23	*L*	*M*	*M*	→	*M*	*0.132*
24	*L*	*M*	*L*	→	*L*	*0.214*
25	*L*	*L*	*H*	→	*L*	*0.375*
26	*L*	*L*	*M*	→	*L*	*0.325*
27	*L*	*L*	*L*	→	*L*	*0.154*

5. Conclusion

This paper has presented a fuzzy-logic based approach of an expert system for module advising. The approach creates three spaces which are student-space, module-space, and lecturer space, and associate them with the abnormal situation type for each abnormal case. Each space and the abnormal situation type is then given a linguistic label using fuzzy sets. The linguistic labels are then associated with each other for each case to generate a fuzzy rule. The fuzzy rules for all cases are combined in one rule-base and then compressed to extract those rules with the highest firing strength. The fuzzy logic was used to handle the uncertainty implied in the human judgment of the student case as well as to provide a transparent and interpretable mechanism for predicting the risk level of enrolling a student on a module.

The approach was developed using real-life university data and achieved an acceptable level of accuracy of 92% which is expected to improve as more data is captured and used to train the rule base. Although the achieved accuracy might sound like it needs a bit of enhancement but having that the machine learning approach used for training the rule base the accuracy is expected to enhance as more data instances are included. Also, the accuracy level acceptance depends on the problem and context, especially with cases in which are molding human decisions or behavior. Although there are different approaches have been proposed in this area, this paper introduces a novel mechanism that creates three spaces to estimate the risk of the student situation associated with a specific module. The three spaces which are namely: Students, Lecturer, and Module provide a multi-angle view on the student case and makes the estimation more realistic. Also, applying the fuzzy logic provides a means to handle the uncertainty included in the human decision-making regarding module selection.

The fuzzy rule base also provides a transparent mechanism to make the recommendation which means it doesn't provide the risk level only but also the justification of that recommendation based on the risk level of the three spaces.

References

[1]. Lam, S. S. and Choi, S. P. M. (2013). Implementing an efficient preference-based academic advising system. International Journal of Applied Management Science, 5(4), pp 297–321, DOI: 10.1504/IJAMS.2013.057110.

[2]. Almutawah, K. A. (2014). A decision support system for academic advisors. International Journal of Business Information Systems, 16(2), pp 177. DOI 10.1504/IJBIS.2014.062837.

[3]. Daramola, O., Emebo, O., Afolabi, I. and Ayo, C. (2014). Implementation of an Intelligent Course Advisory Expert System Cased-Based Course Advisory Expert System. In (IJARAI) International Journal of Advanced Research in Artificial Intelligence, 3(5), Available: www.ijarai.thesai.org

[4]. Bendakir, N. and Aïmeur, E. (2006). Using association rules for course recommendation. AAAI Workshop - Technical Report, vol. WS-06-05, pp 31–40, Available: https://www.aaai.org/Papers/Workshops/2006/WS-06-05/WS06-05-005.pdf

[5]. O'Mahony, M. P. and Smyth, B. (2007). A recommender system for on-line course enrolment: An initial study. RecSys'07: Proceedings of the 2007 ACM Conference on Recommender Systems, pp 133–136. DOI: 10.1145/1297231.1297254.

[6]. Sandvig, J. and Burke, R. (2005). Aacorn: A CBR recommender for academic advising. Technical Report TR05-015, Available: http://facweb.cs.depaul.edu/research/techreports/TR05-015.doc.

[7]. Ajanovski, V. V. (2017). Guided Exploration of the Domain Space of Study Programs. In 4th Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS), Available: http://ceur-ws.org/Vol-1884/paper8.pdf.

[8]. Young, P. (2017). A Recommender System for Personalized Exploration of Majors, Minors, and Concentrations *, RecSys 2017 Poster Proceedings 27 (2017), Available: http://ceur-ws.org/Vol-1905/recsys2017_poster12.pdf.

[9]. Sharma, A. and Tiwari, N. (2013). Design and Analysis of Fuzzy based Association Rule Mining. International Journal of Computer Applications and Information Technology, 3(I), pp 12, Available: https://www.ijcait.com/IJCAIT/31/314.pdf.

[10]. Wang, L.-X. X. and Mendel, J. M. (1992). Generating Fuzzy Rules by Learning from Examples. IEEE Transactions on Systems, Man and Cybernetics, 22(6), pp 1414–1427. DOI: 10.1109/21.199466.

[11]. Young-Jones, A. D., Burt, T. D., Dixon, S. and Hawthorne, M. J. (2012). Academic Advising: Does it Really Impact Student Success? In Quality Assurance in Education. vol. 21, Available: www.emeraldinsight.com.

[12]. Mohamed, A. (2015). A decision support model for long-term course planning. Decision Support Systems, vol. 74, pp 33–45. DOI: 10.1016/j.dss.2015.03.002.

[13]. Harding, B. (2008). Students with specific advising needs. In V. N. Gordon, W. R. Habley and T. J. Grites (Eds.), Academic advising: A comprehensive handbook (2nd ed.), pp. 189–203, Jossey-Bass.

[14]. Laghari, M. S. (2014). Automated Course Advising System. International Journal of Machine Learning and Computing, pp 47–51. DOI: 10.7763/ijmlc.2014.v4.384.

[15]. Roushan, T., Chaki, D., Hasdak, O., Chowdhury, M. S., Rasel, A. A., Rahman, M. A. and Arif, H. (2014). University course advising: Overcoming the challenges using decision support system. 16th Int'l Conf. Computer and Information Technology, ICCIT 2013, pp 13–18. DOI: 10.1109/ICCITechn.2014.6997355.

[16]. Talal Al-Nory, M. (2012). Simple decision support tool for university academic advising. Proceedings of 2012 International Symposium on Information Technologies in Medicine and Education, ITME 2012, vol. 1, pp 53–57. DOI: 10.1109/ITiME.2012.6291245.

[17]. Abdullah, A.-G., Sumaia, A.-G., Fadel, A., AL-Ruhaili, F. and Thamary, A.-A. (2012). An Expert System for Advising Postgraduate Students. International Journal of Computer Science and Information Technology, 3(3), pp 4529–4532.

[18]. Albalooshi, F. and Shatnawi, S. (2010). HE-Advisor: A Multidisciplinary Web-Based Higher Education Advisory System. In Global Journal of Computer Science and Technology, 10(7), pp. 37-49.

[19]. Mattei, N., Dodson, T., Guerin, J. T., Goldsmith, J. and Mazur, J. M. (2014). Lessons Learned from Development of a Software Tool to Support Academic Advising. In Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education - "Engineering Education: Industry Involvement and Interdisciplinary Trends", ASEE Zone 1 2014. IEEE Computer Society. DOI: 10.1109/ASEEZone1.2014.6820659.

[20]. Shatnawi, R., Althebyan, Q., Ghalib, B. and Al-Maolegi, M. (2014). Building A Smart Academic Advising System Using Association Rule Mining. arXiv, Available: http://arxiv.org/abs/1407.1807

[21]. Lin, F., Dunwei, S. L., Frank, W., Kinshuk, Z. and Mcgreal, R. (2007). e-Advisor: A Multi-agent System for Academic Advising, Available: http://io.acad.athabascau.ca/~oscarl/pub/ABSHL2007.pdf
[22]. Huang, L., Wang, C. D., Chao, H. Y., Lai, J. H. and Yu, P. S. (2019). A Score Prediction Approach for Optional Course Recommendation via Cross-User-Domain Collaborative Filtering. IEEE Access, vol. 7, pp 19550–19563, DOI: 10.1109/ACCESS.2019.2897979.
[23]. Asadi, S. and Shokrollahi, Z. (2019). Developing a Course Recommender by Combining Clustering and Fuzzy Association Rules. Journal of AI and Data Mining, 7(2), pp. 249–262. DOI: 10.22044/jadm.2018.6260.1739.
[24]. Nafea, S. M., Siewe, F. and He, Y. (2019). A Novel Algorithm for Course Learning Object Recommendation Based on Student Learning Styles. Proceedings of 2019 International Conference on Innovative Trends in Computer Engineering, ITCE 2019, pp 192–201, DOI: 10.1109/ITCE.2019.8646355.
[25]. Chang, P. C., Lin, C. H. and Chen, M. H. (2016). A hybrid course recommendation system by integrating collaborative filtering and artificial immune systems. Algorithms, 9(3), DOI: 10.3390/a9030047.
[26]. Newell, C. and Miller, L. (2013). Design and evaluation of a client-side recommender system. RecSys 2013 - Proceedings of the 7th ACM Conference on Recommender Systems, pp 473–474. DOI: 10.1145/2507157.2508220.
[27]. Bydžovská, H. (2015). Are collaborative filtering methods suitable for student performance prediction? Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9273, pp. 425–430. DOI: 10.1007/978-3-319-23485-4_42.
[28]. Yao, S. and Huang, B. (2017). Beyond Parity: Fairness Objectives for Collaborative Filtering. Advances in Neural Information Processing Systems, vol. 2017-Decem, pp. 2922–2931, http://arxiv.org/abs/1705.08804.
[29]. Xu, J., Xing, T. and van der Schaar, M. (2015). Personalized Course Sequence Recommendations. IEEE Transactions on Signal Processing, 64(20), pp. 5340 - 5352, DOI: 10.1109/TSP.2016.2595495.
[30]. Koutrika, G., Bercovitz, B. and Garcia-Molina, H. (2009). FlexRecs: Expressing and combining flexible recommendations. SIGMOD-PODS'09 - Proceedings of the International Conference on Management of Data and 28th Symposium on Principles of Database Systems, pp 745–757. DOI: 10.1145/1559845.1559923.
[31]. Keston, L. and Goodridge, W. (2015). AdviseMe: An Intelligent Web-Based Application for Academic Advising. International Journal of Advanced Computer Science and Applications, 6(8). DOI: 10.14569/ijacsa.2015.060831.
[32]. Engin, G., Aksoyer, B., Avdagic, M., Bozanli, D., Hanay, U., Maden, D. and Ertek, G. (2014). Rule-based expert systems for supporting university students. Procedia Computer Science, vol. 31, pp 22–31. DOI: 10.1016/j.procs.2014.05.241.
[33]. Hashemi, R. R. and Blondin, J. (2010). SASSY: A petri net based student-driven advising support system. ITNG2010 - 7th International Conference on Information Technology: New Generations, pp 150–155. DOI: 10.1109/ITNG.2010.57.
[34]. Ayman, M. (2011). A Prototype Student Advising Expert System Supported with an Object-Oriented Database. International Journal of Advanced Computer Science and Applications, 1(3). DOI: 10.14569/specialissue.2011.010316.
[35]. Nambiar, A. N. and Dutta, A. K. (2010). Expert system for student advising using JESS. ICEIT 2010 - 2010 International Conference on Educational and Information Technology, Proceedings, vol 1. DOI: 10.1109/ICEIT.2010.5607701.
[36]. Thanh Binh, N., Thi Anh Duong, H., Hieu, T., Duc Nhuan, N. and Hong Son, N. (2008). An integrated approach for an academic advising system in adaptive credit-based learning environment. VNU Journal of Science, Natural Sciences and Technology, vol. 24, pp 110–121, Avaiable: https://repository.vnu.edu.vn/bitstream/11126/4690/3/TC_02468.pdf.
[37]. Sobecki, J. and Tomczak, J. M. (2010). Student courses recommendation using ant colony optimization. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5991 LNAI(PART 2), pp 124–133, DOI: 10.1007/978-3-642-12101-2_14.
[38]. Abdulwahhab, R. S., Al Makhmari, H. S. and Al Battashi, S. N. (2015, March 12). An educational web application for academic advising. 2015 IEEE 8th GCC Conference and Exhibition, GCCCE 2015, DOI: 10.1109/IEEEGCC.2015.7060084.
[39]. Goodarzi, M. H. and Rafe, V. (2012). Educational Advisor System Implemented by Web-Based Fuzzy Expert Systems. Journal of Software Engineering and Applications, 05(07), pp 500–507, DOI: 10.4236/jsea.2012.57058.
[40]. Adak, M. F., Yumusak, N. and Taskin, H. (2016, April 28). An elective course suggestion system developed in computer engineering department using fuzzy logic. 2016 International Conference on Industrial Informatics and Computer Systems, CIICS 2016. DOI: 10.1109/ICCSII.2016.7462394.

[41]. Baloul, F. M. and Williams, P. (2013). Fuzzy academic advising system for on probation students in colleges of applied sciences - OMN. Proceedings - 2013 International Conference on Computer, Electrical and Electronics Engineering: "Research Makes a Difference", ICCEEE 2013, pp. 372–377, DOI: 10.1109/ICCEEE.2013.6633965.
[42]. Moss, H. B., Leslie, D. S. and Rayson, P. (2018). Using J-K fold Cross Validation to Reduce Variance When Tuning NLP Models. arXiv, vol. 1, pp 2978–2989, Available: http://arxiv.org/abs/1806.07139.
[43]. Wu, D., Mendel, J. M. and Joo, J. (2010). Linguistic summarization using IF-THEN rules. 2010 IEEE World Congress on Computational Intelligence, WCCI 2010, DOI: 10.1109/FUZZY.2010.5584500.