information

Article

Vertical Integration Decision Making in Information Technology Management

Menekse Gizem Gorgun *, Seckin Polat and Umut Asan *

Department of Industrial Engineering, Istanbul Technical University, Macka, Istanbul 34367, Turkey; polatsec@itu.edu.tr
* Correspondence: saygim@itu.edu.tr (M.G.G.); asanu@itu.edu.tr (U.A.)

Abstract: Vertical integration, also known as make-or-buy, defines whether activities are conducted by company or provided by external parties. There are different models to support decision making for vertical integration in the literature. However, they ignore the uncertainty aspect of vertical integration. As a strategic decision, vertical integration is multidimensional and less frequent. This study contributes a new data-driven model that includes all these characteristics of vertical integration decisions. In this study, a methodology is suggested that benefits from the models in the literature and assesses the results with data obtained from real IT cases. Different methodologies were followed to reach a model that accurately predicts make-or-buy decisions in IT projects at a retail company. Firstly, three different knowledge-based generic models derived from the literature were applied to predict decisions for twenty-one different make-or-buy cases in IT. The highest accuracy rate reached among these knowledge-based models was 76%. Secondly, the same cases were also analyzed with Naïve Bayes using factors originally introduced by these generic models. The Naïve Bayes algorithm can represent the uncertainty inherent in the decision model. The highest accuracy rate obtained was 67%. Thirdly, a new data-driven model based on Naïve Bayes using IT-related factors was proposed for the decision problem of vertical integration. The data-driven model correctly classified 86% of the decisions.

Keywords: make-or-buy decision; information technology; vertical-integration models; Naïve Bayes

check for updates

Citation: Gorgun, M.G.; Polat, S.; Asan, U. Vertical Integration Decision Making in Information Technology Management. *Information* **2022**, *13*, 341. https://doi.org/10.3390/info13070341

Academic Editor: Katerina Kabassi

Received: 24 May 2022
Accepted: 11 July 2022
Published: 15 July 2022

1. Introduction

Companies determine their fields of activity which are established by organizational structure and ecosystems [1,2]. Defining activities of companies provides cost-reduction and economic advantages. Additionally, companies often encounter supply-chain problems such as inadequate by-products, delays caused by suppliers, and supply-chain disruptions. On the other hand, companies are faced with various challenges when they insource all activities.

The fields of activity limit the boundaries of companies, which define what the company does. While companies determine their boundaries, they must compare the benefits and costs of using the market instead of performing the activity in-house. Companies have horizontal, vertical and corporate boundaries. Vertical boundaries separate activities as operations that the company makes with internal sources and operations that are purchased or procured from other firms. This separation is determined by vertical-integration decisions, which are decisions about activities completed by the company itself or purchased from a specialist firm [3]. Several studies have looked into the benefits and drawbacks of vertical integration [2–7]. Vertical-integration alternatives (make-or-buy decisions, joint ventures, strategic cooperation, mergers and acquisitions) have a major impact on firm success, according to the findings [1–11].

Vertical integration was originally underlined in 1937 by Coase's major paper [12]. The question posed by Coase's study was why so many firms existed in the industry. At the beginning of the 1970s, there was a growing interest in addressing this issue. Williamson's studies in this field made substantial contributions, and he developed a comprehensive theory, namely transaction cost economics (TCE) [13]. This theory is linked to the phenomenon of vertical integration, often known as make-or-buy. These studies [12,13] are essentially descriptive and generic. TCE reflects the economic point of view, and it has since been the subject of various studies. Other theories such as option theory, agency theory, and resource-based view were applied to describe the behavior of companies.

In the literature, prescriptive models for make-or-buy decision-making have been proposed [14–17]. Before the 1980s, the industry was the primary focus of vertical integration studies. However, after the 1980s, make-or-buy decision strategies were also investigated in IT, human resources (HR), sales and similar areas. In IT, "make" or "build" means in-house software development, and "buy" refers to externally supplying it. Open-source software (OSS) is part of the "make" decision since an effort needs to be made internally to adapt OSS to the product. In fact, "Commercial of the Shelf" is part of the "buy" decision in IT. In HR, "make" implies preparing people in the organization for a future task, whereas "buy" represents hiring people from the labor market.

According to their focus area, they may be divided into two categories. The first category includes generic make-or-buy decision models that may be used in any make-or-buy context. Models proposed by Harrigan [2] and Mahoney [7] are two generic model examples. Some mathematical models focusing on optimization are generic models developed by Montgomery et al. [18] and Gelderman and Weele [19]. Function-specific models are the second category of models that are developed for specific contexts. For IT make-or-buy decisions, Buchowicz [14], Rand [15], Cortellessa, Marinelli and Potena [16], and Kramer and Heinzl [17] are good instances of function-specific models. Qualitative frameworks are provided by these investigations that support the make-or-buy decision process using multiple factors.

Studies in the IT environment have become more prominent with the emergence of Industry 4.0 [20]. Make-or-buy decisions in IT are also one of the most important issues affecting the efficiency of companies [21–35]. Inadequate tools and methods for software make-or-buy decisions could influence firms' capability to use IT competitively [21]. There is a strong need to decrease software costs for most IT companies [22].

Several methods in the literature are applied for data collection and data analysis. Interviewing with IT experts is one method to obtain data [23,24]. Using survey results is another way to obtain data for vertical-integration decision studies [25,26]. A few studies [16,27–31] focused on handling vertical-integration decisions via optimization models. Bali et al. [30] and Kalantari et al. [31] included a fuzzy approach in their optimization studies. On the other hand, Buchowicz [14] and Rand [15] evaluated strategic make-or-buy decisions with multistage decision models. Jang and Huang [32], Wang and Yang [33] and Yang et al. [34] studied on multicriteria decision-making techniques to take the strategic decisions for make-or-buy. Gorgun et al. [35] developed a data-driven model that predicts make-or-buy decisions of real IT cases.

The literature review shows that most models, especially those suggesting a generic framework, are knowledge-based studies. Classification of the models are represented in Table 1. Vertical integration as a strategic decision is multidimensional and less frequent. It involves uncertainty, and therefore, it is essential to indicate uncertainty in the models of vertical integration. However, the models suggested in the literature ignore the uncertainty aspect.

Table 1. Classification of make-or-buy decision models.

Studies in the Literature	Generic/Function Specific	Methodology	Application Area
Harrigan (1984), [2] Mahoney (1990), [8] Venkatesan (1992), [4] Welch and Nayak (1992), [9] Humphreys et al. (2000), [6] McIvor (2000), [11] Humphreys et al. (2002), [7] McIvor (2008), [36]	Generic	Knowledge-Based Model	-
McIvor et al. (1997), [5] Canez et al. (2000), [11]			Manufacturing
Gelderman and Weele (2005), [19] Montgomery et al. (2017), [18]		Optimization Model	-
Buchowicz (1991), [14] Rand (1993), [15]	Function Specific	Knowledge-Based Model -Multistage Decision	IT-Software
Cortellessa et al.(2008), [16] Jha et al. (2013), [29] Jha at al. (2014), [28]		Optimization Model	
Daneshgar et al. (2013), [23] Shahzad et al. (2017), [24]		Interviewing	
Sena et al. (2011), [25] Borg et al. (2019), [26]		Survey	
Bali et al. (2014), [30] Kalantari et al. (2021), [31]		Optimization Model-Fuzzy Approach	
Yang and Huang (2000), [32] Wang and Jang (2006), [33] Yang et al. (2006), [34]		Multicriteria Decision-making Model	

Although several knowledge-based studies [2,7,14–17] have been published for vertical-integration decisions, there is a lack of data-driven models that have been developed for vertical-integration decisions.

According to the literature review, a new decision-making approach is required to handle the above-mentioned characteristics of vertical-integration decisions. In an effort to fill this gap, a methodology is followed that benefits from the models in the literature and assesses the results of data obtained from real IT cases. This methodology deals with characteristics of vertical integrations, which are uncertainty, multidimensionality and limited dataset.

Firstly, three different knowledge-based generic models from literature [8,9,36] were applied in datasets collected from the IT environment. Accuracy results of the three models were compared.

Secondly, a learning-based model, which is probabilistic and quantitative, was offered for make-or-buy decisions for IT projects in the retail industry. The proposed model is developed by using the Naïve Bayes classifier, which works with probabilities to represent uncertainty. Vertical-integration decisions are strategic decisions that are not taken very frequently in companies. Naïve Bayes does not require large datasets. Additionally, the proposed data-driven model in this study includes high-dimensional data. Naïve Bayes is advantageous in case of high-dimensional datasets [37–39]. Due to the above-mentioned features of the Naïve Bayes classifier, it was used in this study.

Generic models in this study have a smaller number of parameters. Naïve Bayes is also capable of learning with a smaller number of parameters. Thirdly, the parameters in the knowledge-based generic models were also evaluated with the Naïve Bayes technique.

The rest of the structure of this study is as follows. The second section introduces the proposed methodology, which includes knowledge-based models, the Naïve Bayes approach, and their application in IT cases. The third section provides the details of the findings and a discussion of the proposed data-driven model compared to the knowledge-based models. The final section involves conclusions from this study and presents further suggestions.

2. Methodology

In this study, the following approach, consisting of four steps was applied to a leading retail company (see Figure 1). In the first step, twenty-one IT project cases that successfully evolved into software products were identified. In other words, these decisions had already been made before this study, and the products of these cases are currently in use. Details about the products (cases) are:

- 4 Web-based software (SW) development (1 make and 3 buy decision)
- 6 Web-based SW together with mobile application development (2 make and 4 buy decision)
- 2 Web service development (2 make decision)
- 1 Web service development and its integration to third party (make decision).
- 2 ERP development (1 make and 1 buy decision)
- 2 CRM development (1 make and 1 buy decision)
- 2 Chatbot (1 make and 1 buy decision)
- 1 Robotic process automation (RPA) platform development (make decision)
- 1 Working-hours registration platform development (make decision).

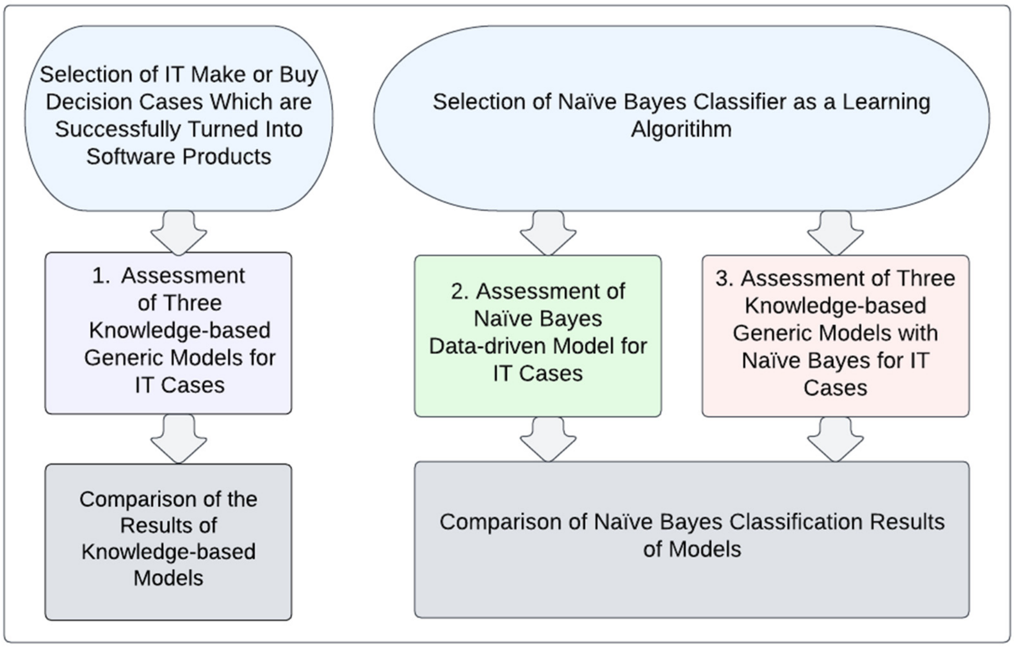

Figure 1. A summary for methodology.

The data related to these make-or-buy decision cases were collected from the retail company.

In the second step, three knowledge-based generic models from the literature [8,9,36] were applied to recommend decisions for twenty-one cases. Based on these cases, questions related to chosen models were asked to experts in IT. These experts are the leading decision-makers in the cases. The recommended decisions obtained from three knowledge-based models were compared with previous decisions that had already been made for twenty-one cases.

In the third step, a data-driven vertical integration IT model was developed by adapting the Naïve Bayes algorithm to twenty-one cases. The factors were adopted from the study of Badampudi et al. [40], which focused on making make-or-buy decisions for IT projects. The developed method was applied to predict decisions for each case.

In the last step, to make comparisons with the data-driven vertical integration IT model, the factors of the three knowledge-based models were analyzed in twenty-one cases by applying the Naïve Bayes algorithm. Figure 1 represents each step of the methodology followed in this study.

2.1. Knowledge-Based Models in Vertical-Integration Decisions

In this stage, three different well-known knowledge-based generic models were assessed for their effectiveness on make-or-buy decisions with IT case studies in an IT environment.

Model 1 is the study of Welch and Nayak [9], which evaluates the make-or-buy decisions under three factors: "maturity process technology across industries", "process technology relative to competitors" and "significance of process technology for competitive advantage". The factor levels and corresponding decisions for Model 1 are tabulated in Figure 2.

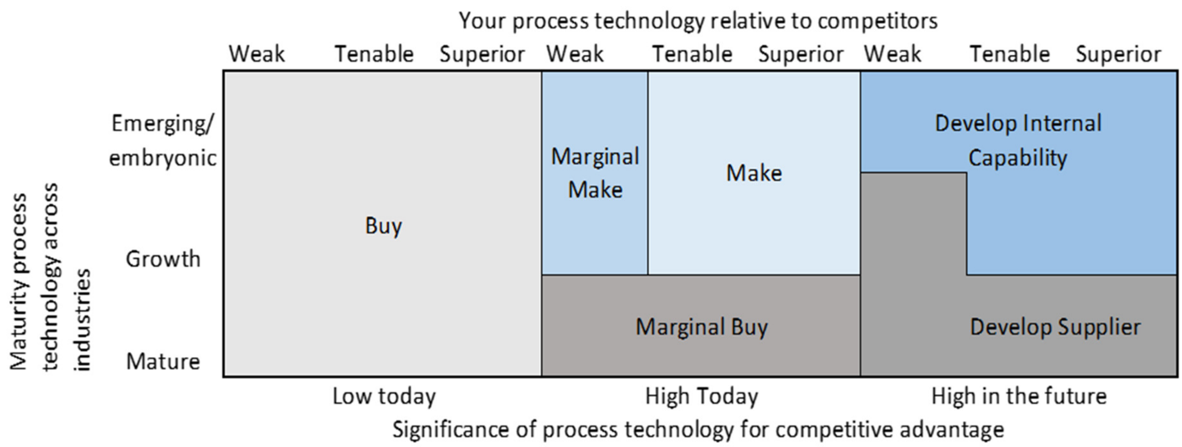

Figure 2. Knowledge-based Model 1: Process Technology [9].

"Marginal make" "Make" and "Develop internal capability" strategies were considered as "Make (Insource)" decisions in this study. Moreover, "Marginal buy", "Buy" and "Develop supplier" strategies were considered as "Buy (Outsource)" decisions.

In Model 2, McIvor [36] evaluates the make-or-buy decisions by considering other three factors as "Contribution to competitive advantage", "Relative capability position", and "Opportunism potential". The factors, levels and decisions are tabulated in Figure 3. Model 2 consists of "Invest to perform internally" and "Keep internal" strategies, which were categorized as "Make (Insource)", whereas "Outsource" was considered as a "Buy (Outsource)" decision.

Model 3, developed by Mahoney [8], considers the factors "Task programmability", "Asset specificity" and "Separability" in make-or-buy decisions. Figure 4 shows the factors and decisions that belong to Model 3. "Spot market", "Long-term contract", "Relational contract", and "Joint venture" strategies were considered as "Buy (Outsource)" in this study, while "Make (Insource)" decisions were assumed as Clan (Hierarchy")", "Inside contract" and "Hierarchy" strategies.

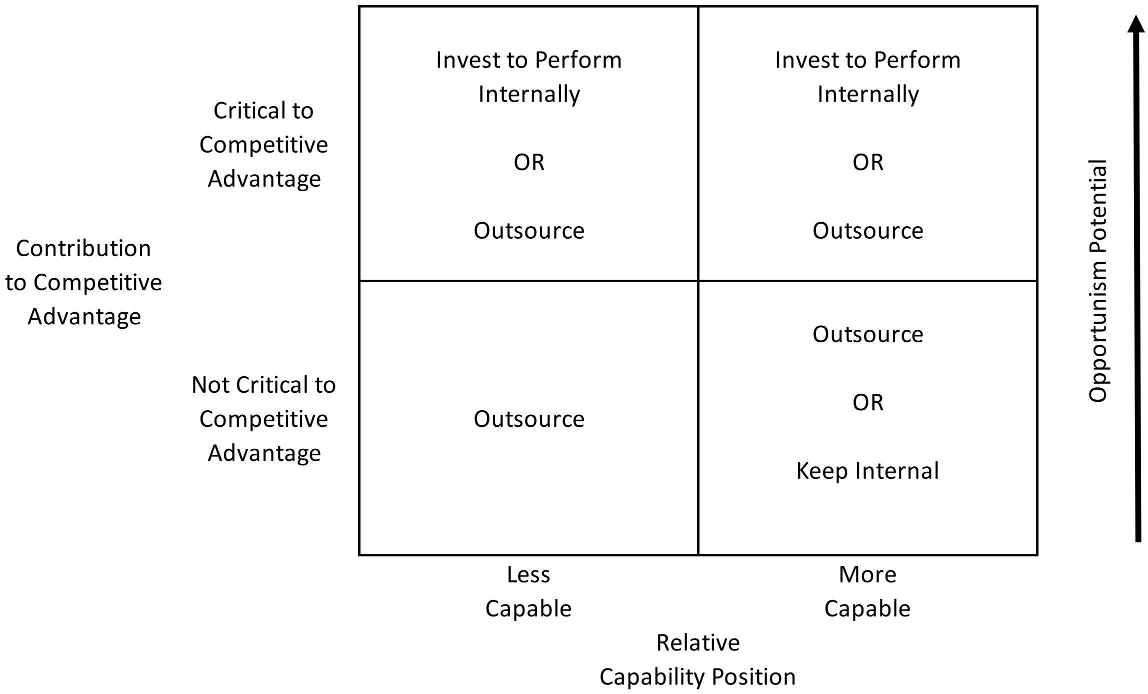

Figure 3. Knowledge-based Model 2: Competitiveness, Capability and Opportunism [36].

	Low Task Programmability		High Task Programmability	
	Low Asset Specificity	High Asset Specificity	Low Asset Specificity	High Asset Specificity
Low Non-separability	1. Spot Market	2. Long-Term Contract	5. Spot Market	6. Joint Venture
High Non-separability	3. Relational Contract	4. Clan	7. Inside Contract	8. Hierarcy

Figure 4. Knowledge-based Model 3: Task Programmability, Separability and Asset Specificity [8].

Twenty-one cases were used to test the knowledge-based models. Cases consist of data and decisions obtained from an IT department of a retail company. The data sets in these cases are related to a variety of software development projects such as ERP modules, CRM tools, institutional websites, and mobile applications. For each model and case, the questions given below were asked to the decision makers in order to clarify the levels of each factor.

Model 1—Process Technology:

- Maturity process technology across industries: What is the maturity level of this software in comparison with other industries? (Answer options: a—emerging/embryonic; b—growth; c—mature)
- Your process technology relative to competitors: What is the (superiority) status of this software compared to competitors? (Answer options: a—weak; b—tenable; c—superior)
- Significance of process technology for competitive advantage: What is the importance of this software in terms of competitive advantage? (Answer options: a—low today; b—high today; c—high in the future)

Model 2—Competitiveness, Capability and Opportunism:

- Contribution to Competitive Advantage: What is the contribution of this software to competitive advantage? (Answer options: a—not critical; b—critical)
- Relative Capability Position: What are the capabilities of the company to develop this software in comparison with other companies? (Answer options: a—less capable; b—more capable)
- Opportunism Potential: What is the opportunism potential created by this software? (Answer options: a—low; b—middle; c—high)

Model 3—Task Programmability, Separability and Asset Specificity:

- Non-separability: Is this software (product/output) sufficient to measure the developer's success? (Answer options: a—low; b—high)
- Task-programmability: Is effort/input sufficient to measure the success of the software? (Answer options: a—low; b—high)
- Asset Specificity: Can assets (internal and external) to develop this software be used for other purposes? (Answer options: a—low; b—high)

2.2. Naïve Bayes Classifier

In this study, the proposed model for IT make-or-buy decision cases using a Naïve Bayes classifier is a probabilistic-based classification algorithm based on Bayes' theorem and a strong (naïve) assumption of conditional independence among features [41]. In comparison to other classification techniques, Naïve Bayes does not enforce any specific scaling or distributional assumptions. Moreover, it is capable of learning with a smaller number of parameters, and it is more compatible with high-dimensional data. It works with probabilities to represent uncertainty [37–39]. It has high performance and is efficient in terms of time and memory, despite this strong assumption [42]. Additionally, prior to learning, Naïve Bayes classifier does not need a large-scale dataset [43].

A vector of n independent characteristics $\boldsymbol{x} = (x_1, \ldots, x_n)$ is used to represent a problem case for classification. The Bayes formula takes the following form [44], if c_j is the j-th class label and $\boldsymbol{x}$ is the vector specifying the instance to be categorized:

$$P(c_j|\boldsymbol{x}) = \frac{P(\boldsymbol{x}, c_j)}{P(\boldsymbol{x})} = \frac{P(\boldsymbol{x}|c_j)P(c_j)}{P(\boldsymbol{x})} \tag{1}$$

The following steps are involved in Naïve Bayes classification:

1. Determine significant factors and factor levels;
2. Establish the training set;
3. Indicate each instance in the training set in vector form $\boldsymbol{x} = (x_1, \ldots, x_n)$;
4. Compute $P(x_i|c_j)$ for each x_i and each class c_j by using the relative frequency of x_i among the training instances belonging to c_j;
5. Determine $P(c_j)$;
6. Based on the conditional independence assumption, calculate

$$P(\boldsymbol{x}|c_j) = \prod_{i=1}^{n} P(x_i|c_j) \tag{2}$$

7. For each class, compute $P(c_j) \cdot P(\boldsymbol{x}|c_j)$;
8. Select the class with the highest $P(c_j) \cdot P(\boldsymbol{x}|c_j)$;
9. Assess the accuracy.

2.3. Data-Driven Vertical-Integration Model for IT

A learning-based vertical-integration decision model was developed with the Naïve Bayes classification approach. This approach was used for make-or-buy decisions made by a retail company in software development cases. The factors were selected from the study by Badampudi et al. [40], which assiduously involves make-or-buy decision strategies in the IT field. The "Requirements" factor was represented as two factors to make a more comprehensive analysis: "Requirements complexity" and "Requirements certainty". The levels of the factors were adopted from a previous study [35], where the levels had been determined by decision makers covering software decisions of the same retail company. Table 2 shows selected factors and their levels.

Table 2. Selected factors.

Factor Name	Definition	Factor Levels
Time (T)	Time spent on development, testing and integration in total.	Less than 90 days, 90–270 days, More than 270 days.
Cost (C)	Any project-related expenses.	Less than average, Average, More than average.
Effort (E)	Effort for development and/or decision making and application.	Less than 2 man-months, 2–6 man-months, More than 6 man-months.
Quality (Q)	Expectations for quality.	Low, Middle, High.
Market Trend (Mt)	The product's availability in the marketplace.	Growing, Fixed, Shrinking, Specific product (No trend).
Availability of Source Code (Sc)	Defining source code availability.	Open, Licensed, N/A.
Technical Support (Ts)	Support, bug fixes and feature updates are all required.	Low, Average, High.
License (L)	Fees and obligations for license.	Yes, No.
Integration (I)	Simplicity of combining process.	Simple, Hard.
Complexity of Requirements (Rco)	Defines product requirement complexity.	Complex, Uncomplex.
Certainty of Requirements (Rce)	Defines product requirement certainty.	Certain, Uncertain.
System Maintenance (M)	Easiness level for maintenance.	Easy, Middle, Difficult.

The data obtained from the real IT cases were used as the training set. Ten-fold cross-validation was utilized to analyze the training dataset in Weka. In this study, decision options were simply defined as insourcing and outsourcing. In the study by Badampudi et al. [40], the decisions are categorized as insourcing, outsourcing, COTS (Commercial of the Shelf Product) and OSS (Open Source Software). In the proposed data-driven model, COTS was considered as an outsourcing strategy. However, OSS was not included since this strategy was not considered an option to evaluate previous make-or-buy decisions in this study.

3. Results and Discussion

The above-mentioned three knowledge-based generic models were adapted from previous studies to predict the decisions of twenty-one cases in the IT environment. The validity of these knowledge-based models was evaluated based on the make-or-buy decisions made by the retail company. Model 1 predicted fourteen make-or-buy decisions correctly out of twenty-one decisions. Model 2 gave sixteen decisions the same as the retail company's decisions. Model 3 only matched twelve strategic decisions correctly. The number of recommendations consistent with the company's decisions and accuracy rates are shown in Table 3, while Model 1 recommended outsourcing decisions more, Model 3 recommended insourcing decisions more for the same cases.

Table 3. Accuracy rates of knowledge-based generic models.

	Number of Recommendations Consistent with the Company's Make-or-Buy Decisions	Accuracy Rate
Model 1: Process Technology	14	67%
Model 2: Competitiveness, Capability and Opportunism	16	76%
Model 3: Task Programmability, Separability and Asset Specificity	12	57%

The results indicate that Model 2 is the most accurate (76%) in predicting decisions for twenty-one cases compared to the previous decisions made by the IT experts. On the other hand, Model 1 predicted decisions with an accuracy rate of 67%. With a rate of 57%, Model 3 made less accurate recommendations than the others. The reason why Model 2 produced the most accurate recommendations is that it involves more generic factors than the other two models and is more adaptable to IT cases. The factors of Opportunism Potential, Relative Capability Position and Contribution to Competitive Advantage could be discussed in any area and context. In comparison to other models, Model 3 has only internal factors, whereas others have both internal and external factors. This may result in a lower match with IT experts' decisions.

After assessing knowledge-based models, the data-driven vertical-integration model was evaluated in Weka software. The dataset was analyzed with a Naïve Bayes classifier [45]. Weka was chosen since it is open-source and easy to use. Table 4 demonstrates the instances that are correctly and incorrectly classified with the data-driven vertical integration model. The Naïve Bayes approach correctly classified 10 of 11 (90.9%) cases for insourcing decisions, while it correctly predicted 8 of 10 (80%) cases for outsourcing decisions. In total, 18 of 21 cases were correctly classified with the data-driven vertical integration model, which corresponds to an accuracy rate of 85.71%.

Table 4. Confusion matrix for the data-driven vertical-integration model.

	Classified as Insourcing	Classified as Outsourcing
Insourcing (a)	10	1
Outsourcing (b)	2	8

The area under the ROC curve (AUC), another metric to measure classification performance, was calculated as 0.836. It is close to one, which means that the model has a great discriminating ability. Table 5 presents other related assessment indicators for the classification performance of the data-driven vertical-integration model. TP (True Positive), FP (False Positive), Precision and Recall metrics indicate good classification performance. Although the TP rate for the insourcing class was higher than the outsourcing class, the precision of the insourcing class was lower than the outsourcing class. The TP rate was higher for insourcing than outsourcing class because more insourcing instances were correctly classified.

Table 5. Detailed accuracy by class for the data-driven vertical-integration model.

Evaluation Metric	Insourcing Class	Outsourcing Class	Weighted Average
TP Rate	0.909	0.800	0.857
FP Rate	0.200	0.091	0.148
Precision	0.833	0.889	0.860
Recall	0.909	0.800	0.857
F-Measure	0.870	0.842	0.856

The reason for the three inaccurately classified cases might be the subjective evaluation of the relevant decision makers. Two of these three cases were older cases, which indicates that the time the cases were originally evaluated might have had an impact on the assessment of factor levels or decisions.

In order to provide a valid comparison between the three knowledge-based generic models and the proposed data-driven vertical integration model, the cases were also evaluated by the knowledge-based generic models using the Naïve Bayes algorithm. Expert responses were obtained to the questions raised in Section 2.1. The knowledge-based models were assessed with the data-driven vertical integration model in Weka software under the same conditions. The accuracy comparison of the models is given in Table 6.

Table 6. Accuracy comparison of the models predicted by Naïve Bayes.

	Model 1: Process Technology	Model 2: Competitiveness, Capability and Opportunism	Model 3: Task Programmability, Separability and Asset Specificity	Data-Driven Model
Accuracy	67%	62%	48%	86%

Model 1 and 2's accuracies were close to each other when they were evaluated with the Naïve Bayes Approach. The accuracy of Model 3 (similar to the result of the knowledge-based model assessment) was lower than the others. The lower accuracy rate of Model 3 may result from the fact that Model 3 contains only internal factors, while other models include both internal and external factors.

Comparisons of the accuracy rates of the three models calculated for recommendations (Table 3) and Naïve Bayes predictions (Table 6) showed that predictions of Naïve Bayes were less accurate. According to the learning mechanism of Naïve Bayes, number of factors and number of factor levels might be inadequate. It was found that all three models from the literature have lower accuracy than the data-driven vertical integration model. While the number of factors of the knowledge-based models is only three, the data-driven vertical-integration model incorporates twelve different IT-related factors. This might explain why the data-driven vertical integration model yielded better predictions.

In order to gain more insight into the experts' decision mechanism, the decision makers who are responsible for the cases in the retail company were asked to determine the relevant factors. Thus, meetings were organized with IT experts from the company to specify the decision factors they considered as relevant. After discussions, experts pointed out four main factors: Business Know-How, Technical Competence, Core Business Activity and Capacity Availability.

Business Know-How was evaluated as the primary assessment factor for experts in making strategic make-or-buy decisions. They emphasized the importance of keeping know-how inside while developing software. Experts implied that they need adequate technical competence to develop software; therefore, Technical Competence is determined as a decisive factor. Furthermore, experts refer that software has to serve as a Core Business Activity since they do not want to preserve valuable resources for support activities. The last assessment factor is determined as Capacity Availability for software development.

Factors of the models applied and decision factors suggested by IT experts are shown in Table 7. Factors of knowledge-based Model 2 have similarities with the decision factors of IT experts. The "Relative Capability Position" factor in Model 2 consists of "Capacity Availability" and "Technical Competence" factors specified by the experts. Additionally, "Contribution to Competitive Advantage" and "Opportunism Potential" are highly related to "Business Know-How" and "Core-Supportive Business Activity" decision factors. The similarity between the factors of "Knowledge-based Model 2" and "Decision Factors Suggested by IT Experts" could be one of the reasons why Model 2 has a higher accuracy rate (76%) than other knowledge-based models.

Table 7. Comparison of decision factors.

	Factors Considered in Models		**Decision Factors Suggested by IT Experts**
	Knowledge-Based Models	**Data-Driven IT Model**	
Model 1	Maturity process technology across industries	Time	Business Know-How
	Your process technology relative to competitors	Cost of Product and Maintenance	Core-Supportive Business Activity
	Significance of process tech. for competitive advantage	Effort	Technical Competence
Model 2	Contribution to Competitive Advantage	Quality	Capacity Availability
	Relative Capability Position	Market Trend	
	Opportunism Potential	Availability of Source code	
Model 3	Nonseparability	Technical Support	
	Task programmability	License	
	Asset Specificity	Integration	
		Complexity of Requirements	
		Certainty of Requirements	
		System Maintenance	

4. Conclusions

Strategies of vertical integration, including make-or-buy decisions, are crucial for companies to protect their competitive advantages in the market. Therefore, make-or-buy decision strategies are a crucial research area, especially for the IT field. Although there are several studies on make-or-buy decision strategies in the literature, only a limited number of decision-making models have been developed for the IT environment. The decision-making models that take place in the literature are not data-driven and do not explicitly consider uncertainty. Uncertainty is the core characteristic of vertical-integration strategies because these strategies are multidimensional.

This study evaluated three different knowledge-based generic models for the IT environment with real decision cases. Among these models, the recommendation accuracy of Model 2 is higher than the others. The higher accuracy might be explained by the factors used in Model 2, which are more generic than the factors of the other models and adaptable to IT cases.

Moreover, a novel data-driven decision-making model was proposed that takes into account both uncertainty and multidimensionality. In a retail company, the Naïve Bayes model was used for software make-or-buy decision-making instances. Eighteen out of twenty-one examples were accurately classified using the Naïve Bayes model. Inaccurately classified cases might have resulted from subjective assessments of the IT experts in the company. Additionally, the TP rates and precision were high. The findings showed that predictions of Naïve Bayes provided good agreement with the decisions that had already been made for the IT cases.

Factors of the three knowledge-based models were also assessed with the Naïve Bayes approach to compare with the data-driven vertical integration model. The results indicate that all three models could not reach the accuracy level of the data-driven vertical integration model. The data-driven vertical integration model has a higher number of factors than the other three models. This is an indication that including a higher number of factors increases the accuracy. The data-driven vertical-integration model had twelve different factors, while the number of factors in three knowledge-based models was only three. Through the high number of factors, the Naïve Bayes approach predicted decisions with better accuracy.

Considering several factors in make-or-buy decisions led to uncertainty for companies. As for the managerial implications, it is suggested to use models covered in this study to cope with uncertainty as a result of considering several factors and to increase efficiency and objectivity. This study shows that models that include both internal and external factors give more accurate results. Therefore, companies may consider both internal and external factors in their make-or-buy decision models.

This study includes limitations as follows. Vertical-integration decisions are strategical decisions that are not taken very frequently in companies. Therefore, the number of cases for vertical integration is generally limited. In this study, twenty-one cases were studied. For this reason, other classification algorithms to implement IT cases could not be considered. Furthermore, the study covers only cases for IT in retail industry. This limits generalization of the results of the study to other industries. Finally, the cases cover only make-or-buy decisions. Thus, vertical integration alternatives such as joint ventures, strategic cooperation, mergers and acquisitions were not considered in this study.

Another classification method besides Naïve Bayes may be studied and applied for verification purposes in future investigations. Furthermore, to evaluate strategic make-or-buy decisions, multistage-type decision models may also be applied for IT cases. Finally, this study solely focuses on make-or-buy decisions on vertical integration strategies. Supplier-selection methodologies for "buy" decisions can be a potential future research topic. For "make" decisions in the future, research can take into consideration IT development strategies such as agile, waterfall (predictive) or hybrid.

Author Contributions: Conceptualization, M.G.G., S.P. and U.A.; methodology, M.G.G., S.P. and U.A.; software, M.G.G.; validation, M.G.G., S.P. and U.A.; formal analysis, M.G.G. and U.A.; investigation, M.G.G.; resources, M.G.G.; data curation, M.G.G.; writing—original draft preparation, M.G.G.; writing—review and editing, M.G.G., S.P. and U.A.; visualization, M.G.G.; supervision, S.P. and U.A.; project administration, M.G.G. and S.P.; funding acquisition, M.G.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by Migros Ticaret A. S.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Quinn, J.B.; Hilmer, F.G. Strategic Outsourcing. *Mckinsey Q.* **1995**, *1*, 48–70.
2. Harrigan, K.R. Formulating Vertical Integration Strategies. *Acad. Manag. Rev.* **1984**, *9*, 638–652. [CrossRef]
3. Besanko, D.; Dranove, D.; Shanley, M.; Schaefer, S. *Economics of Strategy*, 6th ed.; John Wiley: Hoboken, NJ, USA, 2013.
4. Venkatesan, R. Strategic Sourcing: To Make or Not to Make. *Harv. Bus. Rev.* **1992**, *70*, 97–107.
5. McIvor, R.T.; Humphreys, P.K.; McAleer, W.E. A Strategic Model for the Formulation of an Effective Make or Buy Decision. *Manag. Decis.* **1997**, *35*, 169–178. [CrossRef]
6. Humphreys, P.K.; Lo, V.H.Y.; McIvor, R.T. A Decision Support Framework for Strategic Purchasing. *J. Mater. Process. Technol.* **2000**, *107*, 353–362. [CrossRef]
7. Humphreys, P.; McIvor, R.; Huang, G. An Expert System for Evaluating the Make or Buy Decision. *Comput. Ind. Eng.* **2002**, *42*, 567–585. [CrossRef]
8. Mahoney, J.T. The Choice of Organizational Form: Vertical Financial Ownership Versus Other Methods of Vertical Integration. *Strateg. Manag. J.* **1990**, *13*, 559–584. [CrossRef]

9. Welch, J.A.; Nayak, P.R. Strategic Sourcing: A Progressive Approach to the Make-or-Buy Decision. *Acad. Manag. Exec.* **1992**, *6*, 23–31. [CrossRef]
10. Canez, L.E.; Platts, K.W.; Probert, D.R. Developing a Framework for Make-or-buy Decisions. *Int. J. Oper. Prod. Manag.* **2000**, *20*, 1313–1330. [CrossRef]
11. McIvor, R. A Practical Framework for Understanding the Outsourcing Process. *Supply Chain. Manag. Int. J.* **2000**, *5*, 22–36. [CrossRef]
12. Coase, R.H. The nature of the firm. *Economica* **1937**, *4*, 386–405. [CrossRef]
13. Williamson, O.E. *Markets and Hierarchies*; Free Press: New York, NY, USA, 1975.
14. Buchowicz, B.S. A Process Model of Make-vs.-Buy Decision-Making; The Case of Manufacturing Software. *IEEE Trans. Eng. Manag.* **1991**, *38*, 24–32. [CrossRef]
15. Rand, T. A framework for managing software make or buy. *Manag. Sci. Inf. Syst.* **1993**, *2*, 273–282. [CrossRef]
16. Cortellessa, V.; Marinelli, F.; Potena, P. An optimization framework for build-or-buy decisions in software architecture. *Comput. Oper. Res.* **2008**, *35*, 3090–3106. [CrossRef]
17. Kramer, T.; Heinzl, A.; Spohrer, K. Should this software component be developed inside or outside our firm? A design science perspective on the sourcing of application systems. In *New Studies in Global IT and Business Service Outsourcing. 5th Global Sourcing Workshop Courchevel, France*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 115–132.
18. Montgomery, R.; Ogden, J.; Boehmke, B. A quantified Kraljic Portfolio Matrix: Using decision analysis for strategic purchasing. *J. Purch. Supply Manag.* **2017**, *24*, 192–203. [CrossRef]
19. Gelderman, C.; Weele, A. Purchasing Portfolio Models: A Critique and Update. *J. Supply Chain Manag.* **2005**, *41*, 19–28. [CrossRef]
20. Lasi, H.; Kemper, H. Industry 4.0. *Bus. Inf. Syst. Eng.* **2014**, *6*, 239–242. [CrossRef]
21. Rand, T. The key role of applications software make-or-buy decisions. *J. Strateg. Inf. Syst.* **1992**, *1*, 215–223. [CrossRef]
22. Jones, C. Build, buy, or outsource? *Softw. Product. Res.* **1994**, *27*, 77–78. [CrossRef]
23. Daneshgar, F.; Low, G.C.; Worasinchai, L. An investigation of 'build vs. buy' decision for software acquisition by small to medium enterprises. *Inf. Softw. Technol.* **2013**, *55*, 1741–1750. [CrossRef]
24. Shahzad, B.; Abdullatif, A.M.; Ikram, N.; Mashkoor, A. Build Software or Buy: A Study on Developing Large Scale Software. *IEEE Access* **2017**, *5*, 24262–24274. [CrossRef]
25. Sena, M.; Sena, J. Make or Buy: A Comparative Assessment of Organizations that Develop Software Internally Versus those that Purchase Software. *J. Inf. Syst. Appl. Res. (JISAR)* **2011**, *4*, 38–51.
26. Borg, M.; Chatzipetrou, P.; Wnuk, K.; Alégroth, E.; Gorschek, T.; Papatheocharous, E.; Ali Shah, S.M.; Axelsson, J. Selecting Component Sourcing Options: A Survey of Software Engineering's Broader Make-or-Buy Decisions. *Inf. Softw. Technol.* **2019**, *112*, 18–34. [CrossRef]
27. Researchgate. Available online: https://www.researchgate.net/publication/228412936_Automated_support_for_make-or-buy_decisions_in_component_based_software_systems (accessed on 26 December 2021).
28. Jha, P.C.; Bali, V.; Narula, S.; Kalra, M. Optimal component selection based on cohesion & coupling for component based software system under build-or-buy scheme. *J. Comput. Sci.* **2014**, *5*, 233–242.
29. Jha, P.C.; Kaur, R.; Bali, S.; Madan, S. Optimal component selection Approach for Fault-Tolerant Software System Under CRB Incorporating Build-or-Buy Decision. *Int. J. Reliab.* **2013**, *20*, 1350024. [CrossRef]
30. Bali, S.; Jha, P.C.; Kumar, U.D.; Pham, H. Fuzzy multi-objective build-or-buy approach for component selection of fault tolerant software system under consensus recovery block scheme with mandatory redundancy in critical modules. *Int. J. Artif. Intell. Soft Comput.* **2014**, *4*, 98–119. [CrossRef]
31. Kalantari, S.; Motameni, H.; Akbari, E.; Rabbani, M. Optimal components selection based on fuzzy-intra coupling density for component-based software systems under build-or-buy scheme. *Complex Intell. Syst.* **2021**, *7*, 3111–3134. [CrossRef]
32. Yang, C.; Huang, J. A decision model for IS outsourcing. *Int. J. Inf. Manag.* **2000**, *20*, 225–239. [CrossRef]
33. Wang, J.; Yang, D. Using a hybrid multi-criteria decision aid method for information systems outsourcing. *Comput. Oper. Res.* **2006**, *34*, 3691–3700. [CrossRef]
34. Yang, D.; Kim, S.; Min, J. Developing a decision model for business process outsourcing. *Comput. Oper. Res.* **2006**, *34*, 3496–3778. [CrossRef]
35. Gorgun, M.; Polat, S.; Asan, U. A Learning Based Vertical Integration Decision Model. In *International Conference on Intelligent and Fuzzy Systems*; Springer: Cham, Switzerland, 2021.
36. McIvor, R. What is the right outsourcing strategy for your process? *Eur. Manag. J.* **2008**, *26*, 24–34. [CrossRef]
37. Zhang, X.; Li, M.; Zhang, Y.; Ning, J. Cost-sensitive Naïve Bayes Classification of Uncertain Data. *J. Comput.* **2014**, *9*, 1897–1903. [CrossRef]
38. Ren, J.; Lee, S.D.; Chen, X.; Kao, B.; Cheng, R.; Cheung, D. Naive Bayes Classification of Uncertain Data. In Proceedings of the 2009 Ninth IEEE International Conference on Data Mining, Miami Beach, FL, USA, 6–9 December 2009.
39. Maia, M.; Plastino, A.; Freitas, A. An Ensemble of Naive Bayes Classifiers for Uncertain Categorical Data. In Proceedings of the 2021 IEEE International Conference on Data Mining (ICDM), Auckland, New Zealand, 7–10 December 2021.
40. Badampudi, D.; Wohlin, C.; Petersen, K. Software Component Decision-Making: In- house, OSS, COTS or Outsourcing- A Systematic Literature Review. *J. Syst. Softw.* **2016**, *121*, 105–124. [CrossRef]
41. Witten, I.H.; Frank, E.; Hall, M.A.; Pal, C.J. Practical machine learning tools and techniques. *Morgan Kaufmann* **2005**, *2*, 578.

42. Hamine, V.; Helman, P.A. Theoretical and Experimental Evaluation of Augmented Bayesian Classifiers. In *American Association for Artificial Intelligence*; The University of New Mexico: Albuquerque, NM, USA, 2006.
43. Stern, M.; Beck, J.; Woolf, B.P. *Naïve Bayes Classifiers for User Modeling*; Center for Knowledge Communication, Computer Science Department, University of Massachusetts: Amherst, MA, USA, 1999.
44. Kubat, M. Probabilities: Bayesian Classifiers. In *An Introduction to Machine Learning*; Springer: Cham, Switzerland, 2017; pp. 19–41.
45. Weka Homepage. Available online: https://www.cs.waikato.ac.nz/ml/weka/index.html (accessed on 1 November 2021).