Necmettin Erbakan Üniversitesi
Ereğli Eğitim Fakültesi Dergisi
Ereğli Faculty of Education Journal (NEUEEFD)
DOİ: 10.51119/ereegf.2023.29
Cilt:5 Sayı:1 Yıl:2023 e-ISSN: 2687-1831

Investigating Computational Identity and Empowerment of The Students Studying Programming: A Text Mining Study

Nilüfer ATMAN USLU [1], Aytuğ ONAN [2]

[1] *Manisa Celal Bayar Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, Manisa, Türkiye, e-posta*: *atmanuslu@gmail.com*
[2] *İzmir Katip Çelebi Üniversitesi, Bilgisayar Mühendisliği Bölümü, İzmir, Türkiye, e-posta: aytug.onan@ikc.edu.tr*

Makale Bilgileri

Research Article

Article History
Received: 11.05.2023
Accepted: 16.06.2023
Published: 30.06.2023
Keywords:
Computational identity,
Programming,
Empowerment,
Text-mining

ABSTRACT

In this study, it is aimed to predict the data obtained from the answers given by the students who receive programming education to open-ended questions with text mining algorithms. Thus, text-based data on computational identity and programming empowement were analyzed and the performances of different algorithms were compared. The participants of the research consisted of 646 students whose age range was between 12-20 and who received programming education. An electronic form consisting of open-ended questions was prepared to collect the opinions of the students who received programming education. A total of six open-ended questions have been prepared about computational identity and (3 questions) and programming empowerment (3 questions). The text mining process was followed in the analysis of the data set. Analyzes were made in Python 3.8 program. In the study, the performance of Word2vec (W2v) and Term Frequency-Inverse Document Frequency (TF-IDF) word representation methods with five machine learning algorithms were compared: (a) Logistic regression, (b) Decision tree, (c) Support Vector Machines, (d) Random Forest, (e) Neural Network. Regarding computational identity, the highest prediction accuracy was found in artificial neural network (tf-idf) and logistic regression (tf-idf) algorithms. These algorithms have an accuracy rate of 93% regarding computational identity. It was determined that the logistic regression (tf-idf) method reached the highest accuracy prediction rate (96%) in programming empowement. Following this method, the accuracy rate of random forest (tf-idf), support vector machine (tf-idf) and artificial neural network (tf-idf) algorithms was 94%. The fact that these obtained values are above 90% indicates that the estimation performance is sufficient.

Legal Permissions: Ethics Commitee: İzmir Katip Çelebi University Rectorate Graduate Education Institute Ethics Committee Scientific Research Ethics Committee, Date: 27.03.2023.

Atıf/Citation: Atman Uslu, N. & Onan, A. (2023) Investigating Computational Identity and Empowerment of The Students Studying Programming: A Text Mining Study. *Necmettin Erbakan Üniversitesi Ereğli Eğitim Fakültesi Dergisi, 5*(1), 29-45. https://doi.org/10.51119/ereegf.2023.29

Programlama Eğitimi Alan Öğrencilerin Bilgi İşlemsel Kimlikleri ve Yetkilendirilmelerinin İncelenmesi: Bir Metin Madenciliği Çalışması

Article Info

Araştırma Makalesi

Makale Geçmişi
Geliş: 11.05.2023
Kabul: 16.06.2023
Yayın: 30.06.2023

Anahtar Kelimeler:
Bilgi-işlemsel kimlik,
Programlama,
Yetkilendirme,
Metin madenciliği

ÖZ

Bu çalışmada programlama eğitimi alan öğrencilerin açık uçlu sorulara verdikleri cevaplardan elde edilen verilerin metin madenciliği algoritmaları ile tahmin edilmesi amaçlanmıştır. Böylece bilgi-işlemsel kimlik ve programlamada yetkilendirilme ile ilgili metin tabanlı veriler analiz edilmiş ve farklı algoritmaların performansları karşılaştırılmıştır. Araştırmanın katılımcılarını, yaş aralığı 12-20 arasında değişen ve programlama eğitimi alan 646 öğrenci oluşturmuştur. Programlama eğitimi alan öğrencilerin görüşlerini toplamak için açık uçlu sorulardan oluşan elektronik bir form hazırlanmıştır. Bilgi-işlemsel kimlik ve (3 soru) ve programlamada yetkilendirme (3 soru) ile ilgili toplam altı açık uçlu soru hazırlanmıştır. Veri setinin analizinde metin madenciliği süreci izlenmiştir. Analizler Python 3.8 programında yapılmıştır. Çalışmada Word2vec (W2v) ve Terim Frekans-Ters Doküman Frekansı (TF-IDF) kelime temsil yöntemleri ile beş makine öğrenme algoritmasının performansı karşılaştırılmıştır: (a) Lojistik regresyon, (b) Karar ağacı, (c) Destek Vektör Makineleri, (d) Rastgele Orman, (e) Yapay Sinir Ağı. Bilgi işlemsel kimlik ile ilgili olarak, en yüksek tahmin doğruluğunun yapay sinir ağı (tf-idf) ve lojistik regresyon (tf-idf) algoritmasında olduğu görülmüştür. Bu algoritmalar, bilgi işlemsel kimlik ile ilgili olarak % 93'lük bir doğruluk oranına sahiptir. Programlamada yetkilendirmede, lojistik regresyon (tf-idf) yönteminin en yüksek doğruluk tahmin oranına (%96) ulaştığı belirlenmiştir. Bu yöntemin ardından rastgele orman (tf-idf), destek vektör makinesi (tf-idf) ve yapay sinir ağı (tf-idf) algoritmalarının doğruluk oranı %94 olarak saptanmıştır. Elde edilen bu değerlerin %90'ın üzerinde olması tahmin performansının yeterli olduğuna işaret etmektedir.

INTRODUCTION

There is a growing trend globally for the development of programming skills of children and young adults. This trend has been reflected in research especially in the last ten years. Accordingly, it is seen that researches on programming education are carried out in a wide range from early childhood (Angeli & Valanides, 2020; Kazakof et al., 2013; Papadakis et al., 2016), primary and secondary education (Atman-Uslu et al., 2018; Chen et al., 2017; Korkmaz et al., 2020; Oluk et al., 2018) to high school (Saritepeci, 2020) and university level (Mouza et al., 2017; Romero et al., 2017). Many reviews have attempted to categorize and classify research on computational thinking (CT) and programming skills. For example, Sun et al. (2022), made a classification in the studies as teaching methods, tools and assessment. Tikva and Tambouris (2021) revealed the categories of learning strategies, tools, assessment, factors, capacity building and knowledge base related to the teaching of CT. As a result, tools, pedagogies, assessment are among the main research branches of computational thinking research. While research continues on the tools and learning approaches to teach this skill, studies on how to define CT and which sub-components it contains is still at the center of studies in this area. Many frameworks have been proposed in the literature regarding the sub-components of CT and programming skills. Cognitive components such as breaking the problem into small parts, abstracting, testing are included in these frameworks. However, there has been a focus on components of CT such as dispositions, attitudes, and perspectives. Perspectives on CT are defined by Brennan and Resnick (2012) as students' making sense of themselves, the technological context, and the relationships between them. In this context, two concepts come to the fore: computational thinking identity and programming empowerment.

Kong and Lai (2022) define the concept of computational identity *"...as the ongoing mental construction of the self in relation to personal and collective involvement with programming and computational activities at school."*. Computational identity is a component to consider in motivating students to become interested in programming (Brousseau, & Sherman, 2019). Drawing on Kong and Lai (2022), this study considers three components of computational identity (a) engagement, (b) imagination, (c) affiliation. Engagement refers to an individual's currently active involvement in programming (Kong & Wang, 2020). Imagination is about the commitment to programming activities as a future career orientation (Capobianco et al., 2012; Sfard & Prusak, 2005). Affiliation refers to a sense of commitment and belonging to a programming-related group (Kong & Wang, 2020). Programming education should focus on empowering students as well as forming their computational identities (Kong & Lai, 2022). According to Page and Czuba (1999), empowerment refers to power that includes a process of change and this power can be expanded. Empowered students feel more competent, have higher motivation to perform learning tasks, and find these tasks meaningful (Houser & Frymier, 2009). There are three components of learner empowerment in programming education: (a) Meaningfulness, (b) Impact, (c) self-efficacy (Kong et al., 2018; Kong & Lai, 2022). In other words, empowered learners find a task meaningful, feel competent in performing it, and believe that their efforts have an impact (Frymier et al., 1996). Accordingly, Kong et al. (2018), states that programming empowerment includes the individuals' seeing a programming-related task as meaningful, believing that they can complete this task and that it has an effect.

Studies on computational identity and empowerment are mostly carried out using quantitative methods in programming education. The structural model tested by Kong and Wang (2020) revealed that, students' expressing and connecting abilities in programming activities had a positive effect on their computational identity formations. Kong et al. (2018) found the positive role of collaboration and interests on programming empowerment components. More recently, Kong and Lai (2022) reported that the components of programming empowerment (meaningfulness, impact and self-efficacy) have an positive effect on computational identity. Atman-Uslu (2022) examined the latent profiles of secondary school students according to their computational identity and academic resilience in programming. This study found that profiles characterized by higher identity had significantly higher CT performance and CT self-efficacy scores.

Although these studies reveal the relationships between identity and empowerment and emphasize its role in programming education, it can be argued that studies on this topic are at an early stage. As a matter of fact, the limitation of qualitative studies on this issue is also noteworthy. In this context, examining computational identity and empowerment with text mining can open interesting ways. Text mining makes it possible to discover previously unknown information by automatically extracting text data (Hearst, 2003). This method tries to extract meaningful information by analyzing natural language texts (Kumar & Bhatia, 2013). With the digitization of learning and teaching processes, extracting meaningful and useful information from a large amount of text data obtained from forums, chat or social networks presents many challenges (Ferreira-Mello et al., 2019). Therefore, it is seen that there has been an increasing interest in educational text mining studies in recent years. Based on these points, this study aimed to predict the texts obtained from the answers given by the students receiving programming education to open-ended questions, with text mining algorithms. Thus, an attempt was made to analyze text-based data in research on computational identity and programming empowerment and to compare the performances of different algorithms.

METHOD

Participants

The participants of the study consisted of 646 students studying programming. When the education level of the participants is examined, it is seen that 68.7% of these students are educated at secondary education level. In order to collect data from a wide range of participants in terms of education level, data were also collected from first-year university students studying computer science. Accordingly, 31.3% of the participants were first-year undergraduate students. As a result, the age range of the participants varies between 12-20.

Research Instruments and Processes

An electronic form consisting of open-ended questions was prepared to collect the opinions of the students who received programming education. There are a total of six open-ended questions about computational identity (3 questions) and empowerment (3 questions). Factors in the instrument developed by Kong and Lai (2022) were taken into account in the preparation of the questions about computational identity. Questions related to computational identity are listed below.

- In what ways do you find programming engaging? (Engagement)
- What do you dream of doing about programming in the future? (Imagination)
- What are your views on programming with a peer or a team of your peers? (Affiliation)

These questions were formed from the sub-components of computational identity defined in the literature. Three open-ended questions were prepared in order to get the opinions about programming empowerment. Factors in the instrument developed by Kong, Chiu and Lai (2018) were taken into account in the preparation of the questions about programming empowerment. These questions are listed below:

- What are the benefits and contributions of programming to you? (Meaningfulness)
- What difference do you want to make when programming, in making the lives of others easier, improving, and solving the world's problems? (Impact)
- In which aspects do you consider yourself competent in programming? (Self-efficacy)

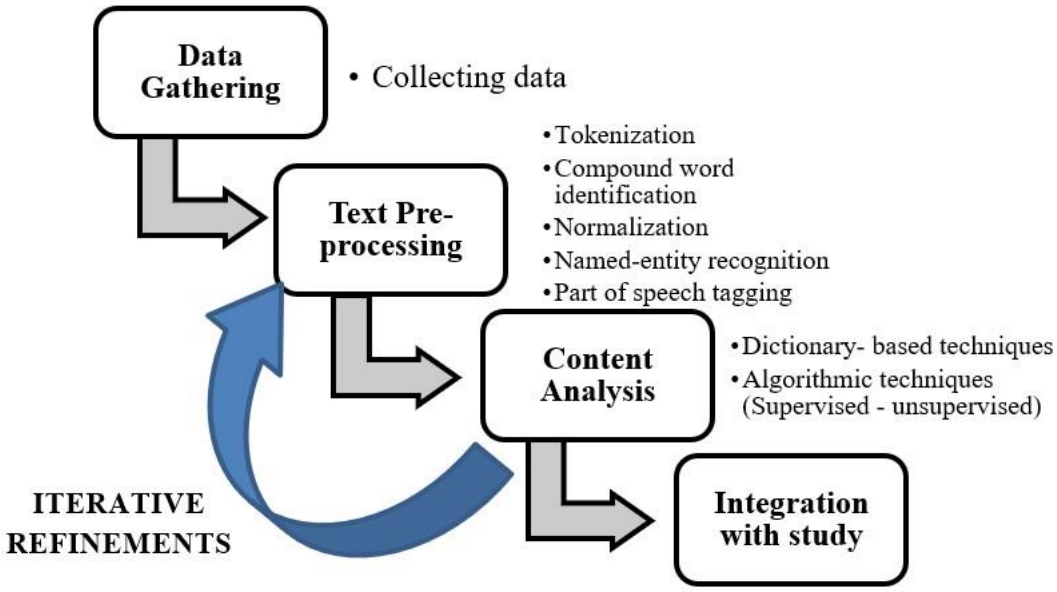
As a result of the answers given by the students to the questions, a data set of 3876 sentences was created.

Data Analysis

The text mining process was followed in the analysis of the data set. This process is given in Figure 1 (Antons et al., 2020).

Figure 1.

Text-mining process (Adapted from Antons et al., 2020).



By following the process shown in Figure 1, analyzes were carried out in Python 3.8 program. First of all, the data set was created by obtaining the answers given by the students to the open-ended answers. The sentences in the created data set were read one by one and labeled as positive and negative (1: positive; 0: negative). In our study, manual coding was indeed used to label the open-ended responses provided by the students. To ensure consistency between coders, a detailed coding protocol and guidelines were developed prior to the coding process. This protocol included specific instructions on how to identify and label positive and negative statements related to computational identity and programming empowerment. To establish inter-rater reliability, two independent coders, who were trained in the coding protocol, individually reviewed and labeled a subset of the responses. The coders then compared their results and discussed any discrepancies or disagreements to reach a consensus. This iterative process was conducted until a high level of agreement was achieved between the coders.

In order for the data to have a certain standard, text-processing has been done. In this process, cleaning of punctuation marks, case folding, tokenization, cleaning of stopwords, stemming and retyping were done. Next, feature extraction was applied. Feature extraction is the process of extracting a word list from text data and creating a feature set for classification from it (Aninditya et al., 2019). In this study, Word2vec (W2v) and Term Frequency-Inverse Document Frequency (TF-IDF) word representation methods were used. The word2vec algorithm uses a neural network model to learn word associations from the textual dataset (Uday et al., 2022). TF-IDF calculates the values of each word in a document as the inverse of the frequency of the word in a particular document and the percentage of documents in which the word appears (Yaman, 2022). After the preprocessing process, machine learning algorithms were used. Machine learning is an area of artificial intelligence concerned with the development of techniques that enable computers to learn through the analysis of data sets (Hotho et al., 2005). Five machine learning algorithms compared in this study: (a) Logistic regression, (b) Decision tree, (c) Support Vector Machines, (d) Random Forest, (e) Artificial Neural Network. Accuracy, precision, recall and F1 score were used to measure the performance of the algorithms. The mathematical formulas of these metrics are given below:

$$Accuracy = \frac{(TruePositives + TrueNegatives)}{TruePositives + TrueNegtives + FalsePositives + FalseNegatives}$$

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives}$$

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives}$$

$$F1 - Score = \frac{2 * (recall * precision)}{recall + precision}$$

Ethic

Ethical permission was obtained from İzmir Kâtip Çelebi University to conduct the research (27.03.2023 / Document Verification Code: 33ECA7T).

FINDINGS

First of all, data related to computational identity were analyzed. The word cloud for positive comments is presented in Figure 2. The most common unigrams are given in Figure 3.

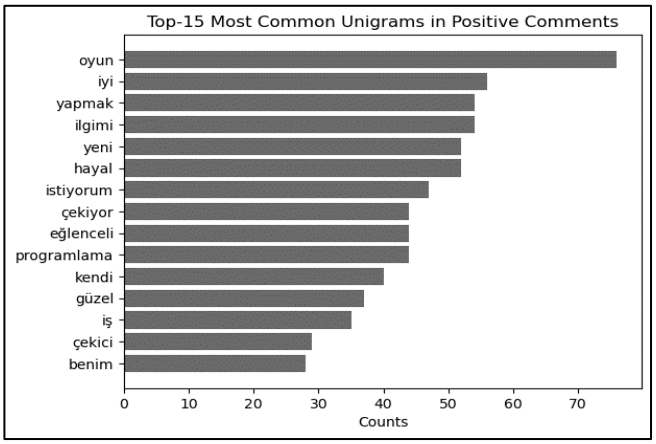
Figure 2.

Wordcloud of positive comments regarding computational identity

As seen in Figures 2 and 3, words such as game, good, doing, interest, new, dream, want, fun, programming are frequently used in positive comments. It has been found that these expressions are related to the dimensions of the computational identity.

Figure 3.

Most common unigrams in positive comments regarding computational identity



Positive and negative labeled texts for the three dimensions of computational identity are given in Table 1.

Table 1.

Positive and negative statement quotes regarding computational identity components

Computational identity component		**Statement**
Engagement	Negative (0)	*"Programming is pretty boring, I don't see the interesting side of creating characters in Mblock."*
	Positive (1)	*"Because while programming, we are trying to realize a scenario (algorithm) that we have set up in our minds, we do not even know that it will work most of the time when we do it for the first time, but despite that, when we can realize what we have in mind, the feeling of satisfaction is very sufficient."*
		"For me, programming is enjoyable because when there is a problem, the steps leading to its solution are converted into code blocks. That's why it's engaging"
		"I find it engaging because there is no limit to what people with unlimited imagination can do."
		"I was wondering what kind of code is inside the programs.
		"Making arduino circuits and seeing it working makes me happy""
Imagination	Negative (0)	*"I don't have a programming-related career goal or dream"*
		"I don't plan to advance in programming."
	Positive (1)	*"I dream of determining the right software field to work, working in a suitable job and improving myself day by day and reaching the best point I can come to."*
		"I will start a company. I will focus on artificial intelligence and quantum computer technologies."
Affiliation	Negative (0)	*"I do not think that working with peers is a very useful teamwork because I do not think that people of the same age group, who are in the same environment and receive the same education, can add a lot to each other in a joint project."*
		"I don't think it has positive aspects, the margin of error in the

	things I do as a team is high because both people have it. When I do it alone, I think it's better because I'm the only one with the margin of error."
Positive (1)	*"Working as a team with my peers allows me to see multiple perspectives, broaden my horizons and learn new things. At the same time, I believe that I learned programming better while working as a team".*

By deducing the meaning of the answers given by the students, the positive ones were labeled as 1 and the negative ones were labeled as 0. In this way, statements about computational identity in the dataset are generally labeled as positive and negative, not on a dimension basis. The findings regarding the performance of machine learning algorithms used in the prediction of computational identity are given in Table 2.

Table 2.

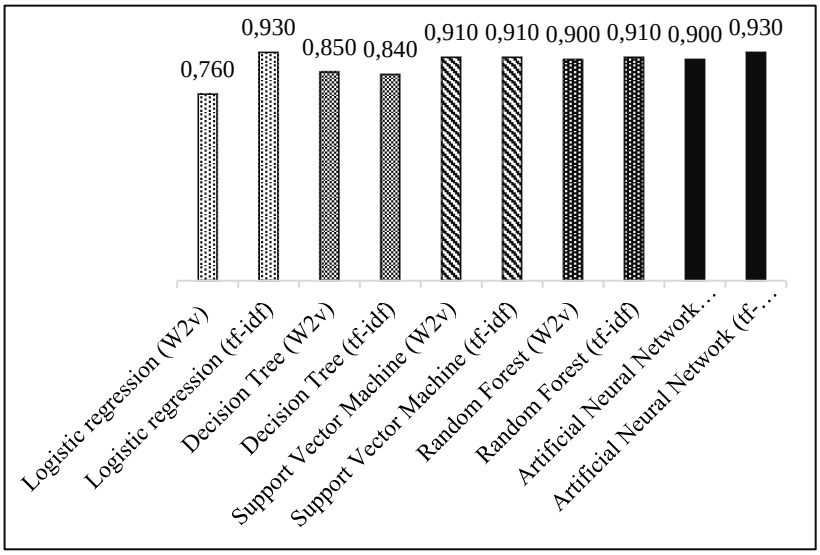
Precision, recall, F1-score and accuracy values of machine learning algorithm regarding computational identity

		Precision	**Recall**	**F1 score**	**Accuracy**
Logistic regression (W2v)	Negative (0)	0.00	0.00	0.00	0.76
	Positive (1)	0.76	1.00	0.86	
Logistic regression (tf-idf)	Negative (0)	0.97	0.74	0.84	0.93
	Positive (1)	0.92	0.99	0.96	
Decision Tree (W2v)	Negative (0)	0.70	0.64	0.67	0.85
	Positive (1)	0.89	0.91	0.90	
Decision Tree (tf-idf)	Negative (0)	0.61	0.91	0.73	0.84
	Positive (1)	0.97	0.82	0.88	
Support Vector Machine (W2v)	Negative (0)	0.95	0.64	0.77	0.91
	Positive (1)	0.90	0.99	0.94	
Support Vector Machine (tf-idf)	Negative (0)	0.89	0.74	0.81	0.91
	Positive (1)	0.92	0.97	0.94	
Random Forest (W2v)	Negative (0)	0.95	0.62	0.75	0.90
	Positive (1)	0.89	0.99	0.94	
Random Forest (tf-idf)	Negative (0)	0.89	0.74	0.81	0.91
	Positive (1)	0.92	0.97	0.94	
Artificial Neural Network (W2v)	Negative (0)	0.95	0.62	0.75	0.90
	Positive (1)	0.89	0.99	0.94	
Artificial Neural Network (tf-idf)	Negative (0)	0.85	0.86	0.86	0.93
	Positive (1)	0.96	0.95	0.95	

When Table 2 is examined, it is seen that accuray is calculated as 76% for the w2v word representation method of the logistic regression algorithm and 93% for TF-IDF. In the Decision tree algorithm, the accuracy value was found to be 85% for W2v and 84% for TF-IDF. In the support vector machine algorithm, the accuracy values for the w2v and TF-IDF methods were 91%. It is seen that the random forest algorithm calculates accuray as 90% for the w2v word representation method and 91% for TF-IDF. In the artifical neural network algorithm, the accuracy value was found to be 90% for W2v and 93% for TF-IDF.
As a result, the accuracy comparisons of the algorithms are given in Figure 4.

Figure 4.

Accuracy comparisons of machine learning regarding computational identity



Later, data related to programming empowerment were analyzed. The word cloud for positive comments is presented in Figure 5, and the most common unigrams are given in Figure 6.

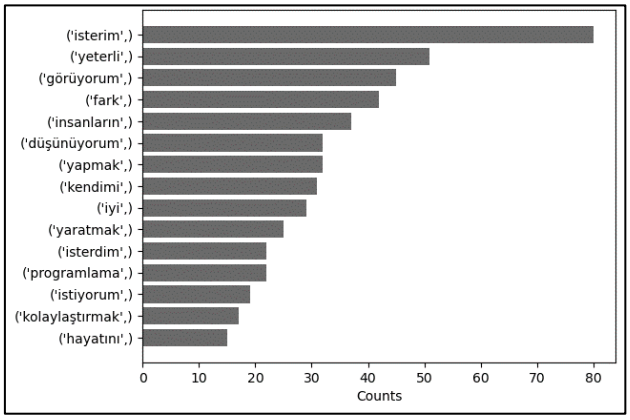
Figure 5.

Wordcloud of positive comments regarding programming empowerment

As seen in Figures 5 and 6, words such as want, competent, see, difference, people, think, doing, programming, are frequently used in positive comments.

Figure 6.

Most common unigrams in positive comments regarding programming empowerment



It has been found that these expressions are related to the dimensions of the programming empowerment. Positive and negative labeled texts for the three dimensions of programming empowerment are given in Table 3.

Table 3.

Positive and negative statement quotes regarding programming empowerment components

Computational identity component		**Statement**
Meaningfulness	Negative (0)	*"I don't think learning programming will be useful and important to me in the future."*
		"There are more important lessons than coding, in my opinion it is unnecessary."
	Positive (1)	*"Making progress in my programming skills will enable me to get a good job in the future."*
		"Learning to code helps me to solve the problems I encounter in daily life."
Impact	Negative (0)	*"I have no intention of making a difference in the world with programming."*
		"I don't find it necessary to deal with programming. More precisely, I deal more with the jobs with the highest return on money."
	Positive (1)	*"In the software field, I think we can go to a very advanced level in facilitating human life. At least a sufficient difference can be made in solving problems related to the world."*
		"I can establish a digital platform that will direct people to produce, not consume, and enable them to use their time efficiently."
		"I dream of developing technologies to help people with disabilities."
		"I want to make games that teach kids about real life."
		"I would like to take part in many projects to make life easier."

Self-efficacy	Negative (0)	*"I don't see myself enough in terms of patience in programming, I get very stressed"*
		"I am inadequate in every way. I don't even see myself at the beginning of the road."
	Positive (1)	*"I am confident in terms of my way of thinking and game development."*
		"I have the technical infrastructure necessary for my future career at my current level. I consider myself sufficient that I will have more advanced competencies by working harder."
		"I can solve any complex problem more quickly now, I think I am proficient in problem solving."

By deducing the meaning of the answers given by the students, the positive ones were labeled as 1 and the negative ones were labeled as 0. In this way, statements about programming empowerment in the dataset are generally labeled as positive and negative. The findings regarding the performance of machine learning algorithms used in the prediction of programming empowerment are given in Table 4.

Tablo 4.

Precision, recall, F1-score and accuracy values of machine learning algorithm regarding programming empowerment

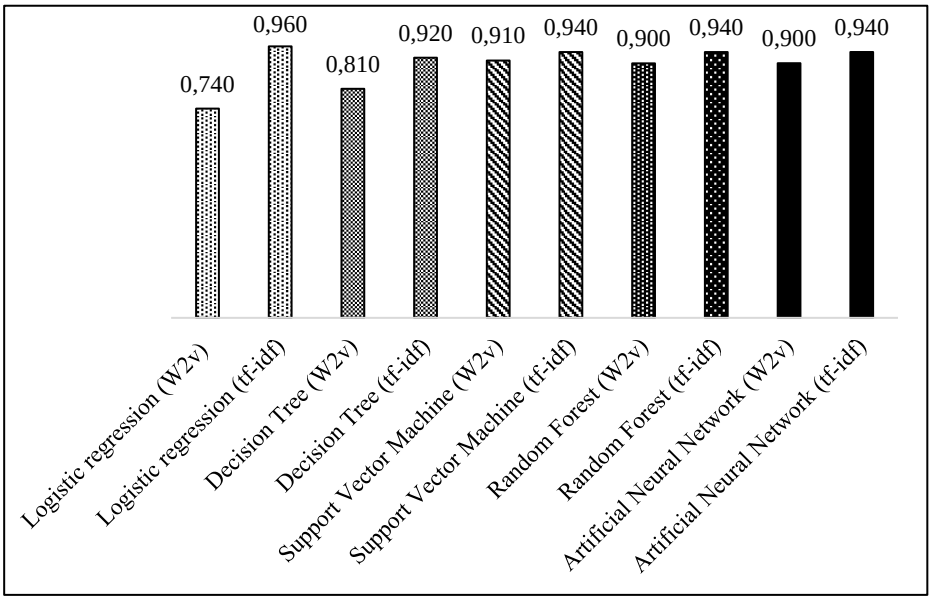
		Precision	**Recall**	**F1 score**	**Accuracy**
Logistic regression (W2v)	Negative (0)	0.00	0.00	0.00	0.74
	Positive (1)	0.74	1.00	0.85	
Logistic regression (tf-idf)	Negative (0)	0.96	0.87	0.91	0.96
	Positive (1)	0.95	0.99	0.97	
Decision Tree (W2v)	Negative (0)	0.62	0.71	0.66	0.81
	Positive (1)	0.89	0.85	0.87	
Decision Tree (tf-idf)	Negative (0)	0.88	0.81	0.84	0.92
	Positive (1)	0.93	0.96	0.95	
Support Vector Machine (W2v)	Negative (0)	1.00	0.65	0.79	0.91
	Positive (1)	0.89	1.00	0.94	
Support Vector Machine (tf-idf)	Negative (0)	1.00	0.75	0.86	0.94
	Positive (1)	0.94	0.94	0.93	
Random Forest (W2v)	Negative (0)	0.94	0.63	0.76	0.90
	Positive (1)	0.89	0.99	0.93	
Random Forest (tf-idf)	Negative (0)	1.00	0.75	0.86	0.94
	Positive (1)	0.92	1.00	0.96	
Artificial Neural Network (W2v)	Negative (0)	0.94	0.63	0.76	0.90
	Positive (1)	0.89	0.99	0.93	
Artificial Neural Network (tf-idf)	Negative (0)	0.95	0.81	0.88	0.94
	Positive (1)	0.94	0.99	0.96	

When Table 4 is examined, it is seen that accuray is calculated as 74% for the w2v word representation method of the logistic regression algorithm and 96% for TF-IDF. In the Decision tree algorithm, the accuracy value was found to be 81% for W2v and 92% for TF-IDF. In the support vector machine algorithm, the accuracy values for the w2v was 91% and fort he TF-IDF methods was 94%. It is seen that the random forest algorithm calculates accuray as 90% for the w2v word representation method and 94% for TF-IDF. In the artifical neural network algorithm, the accuracy value was found to be 90% for W2v and 94% for TF-IDF.

As a result, the accuracy comparisons of the algorithms are given in Figure 7.

Figure 7.

Accuracy comparisons of machine learning regarding programming empowerment



The observed variations in precision, recall, F1 score, and accuracy between the different machine learning algorithms used in our study provide valuable insights into their performance and highlight potential factors contributing to the differences. One factor that can contribute to the variation in performance is the inherent characteristics of each algorithm. Logistic regression, for instance, is a linear classifier that models the relationship between input features and the binary outcome. Decision tree algorithms create hierarchical structures based on specific conditions, while support vector machines aim to find an optimal hyperplane for class separation. Random forest combines multiple decision trees, and artificial neural networks capture nonlinear relationships in the data. These algorithmic differences can impact their ability to effectively predict students' perspectives on computational identity and programming empowerment. Another influential factor is the choice of word representation methods, such as Word2vec (W2v) and Term Frequency-Inverse Document Frequency (TF-IDF). W2v represents words as dense vectors, capturing semantic relationships, while TF-IDF calculates the importance of each word based on its frequency and distribution across the dataset. These different representations can affect the algorithms' performance by influencing their ability to extract meaningful information from the text data. The characteristics of the dataset itself can also play a role. Factors such as class distribution imbalance, noisy or ambiguous statements, or variations in the complexity of language used by students can impact the performance of the algorithms. Imbalanced class distributions, for example, may lead to biased predictions or lower performance on the minority class. Similarly, the presence of noisy or ambiguous statements can introduce challenges for certain algorithms, affecting their precision, recall, F1 score, and accuracy. Furthermore, the multidimensional nature of computational identity and programming empowerment, influenced by individual experiences, backgrounds, and motivations, adds complexity to the prediction task. Different algorithms may capture and weigh these factors differently, leading to variations in their performance.

DISCUSSION, CONCLUSION, RECOMMENDATIONS

A text-based data set consisting of open-ended answers to questions about individuals' tendencies and views on programming was created in this study. The answers of the students to the questions about

computational identity and programming empowerment were predicted using machine learning algorithms. Concerning computational identity, it was found that the highest estimation accuracy was in artificial neural network (tf-idf) and logistic regression (tf-idf) algorithm. These algorithms have an accucary rate of 93% regarding computational identity. When the text-data related to programming empowerment was analyzed, it was determined that the logistic regression (tf-idf) method reached the highest accuracy prediction rate (96%). Following this method, random forest (tf-idf), support vector machine (tf-idf) and artificial neural network (tf-idf) algorithms predicted with 94% accuracy. The fact that these obtained scores are above 90% can be interpreted as sufficient estimation performance.

Regarding computational identity, it was determined that the words with the highest frequency in the dataset were game, good, doing, interest, new, dream, want, fun, programming. In particular, it is interesting that statements about game development and gaming relate to computational identity. It may be helpful for teachers and lecturers to support computtional identity formation, with talk about programming-related imaginations, the relationship between games and coding, and to hold discussion sessions on the factors that attract them to programming. Regarding programming empowerment, the expressions want, competent, see, difference, people, think, doing, programming are frequently used. Most of the answers of the students about empowerment included statements about making a difference about programming, the benefits of programming for them and their competencies. In order to train empowered learners, it is necessary to support students to find programming activities meaningful, to feel competent and to believe that their activities will have an impact.

In the literature, there are a limited number of studies on the use of educational text mining algorithms in programming education. In studies on this subject, it is noteworthy that methods related to data mining are used for recommendation systems. For example, Lin and Chen (2020) found that a deep learning-based augmented reality system is more effective for student performance in learning experiences related to programming and computational thinking. Moon et al. (2020) proposed a framework for how to integrate learning analytics and data mining to support personalized learning in open educational resources related to programming education. The current study presents a trained dataset to predict student views on computational identity and programming empowerment. The findings of this study can provide a starting point for recommendation systems to promote personolized learning in programming education. It can also be useful for automated feedback to learner reflections in open learning resources and online learning environments that will be designed in the future.

In our study, the selection of the Logistic regression, Decision tree, Support Vector Machines, Random Forest, and Artificial Neural Network algorithms was based on their well-established effectiveness and extensive utilization in text classification tasks (Onan et al., 2016; Kowsari et al., 2019). These algorithms have been extensively studied and applied in various natural language processing and text mining domains, including sentiment analysis, document classification, and text categorization (Gupta and Lehal, 2009). Logistic regression is a widely adopted algorithm known for its interpretability and simplicity in modeling the relationship between input features and the probability of binary outcomes. Decision trees are renowned for their ability to handle nonlinear relationships and capture intricate decision boundaries, making them suitable for capturing complex patterns in text data. Support Vector Machines (SVM) have gained popularity due to their ability to find optimal hyperplanes that maximize the separation between classes in the feature space. Random Forest, on the other hand, leverages ensemble learning by combining multiple decision trees to enhance the overall prediction accuracy and effectively handle high-dimensional data. Artificial Neural Networks (ANN) have demonstrated their power in modeling complex relationships and extracting intricate patterns through interconnected layers of neurons. They are capable of capturing both linear and nonlinear relationships in text data, making them well-suited for text classification tasks. While we acknowledge that

there are numerous other algorithms that could potentially be employed for text classification, the selection of these specific algorithms was based on their proven performance and wide adoption in the field. The extensive literature on text classification consistently demonstrates the effectiveness of these algorithms, further supporting their suitability for our study on computational identity and programming empowerment analysis. The chosen algorithms offer a robust and diverse set of techniques to analyze and predict computational identity and programming empowerment from textual data. Their selection was guided by their established effectiveness, widespread usage, and their ability to handle the complexities inherent in text classification tasks. By leveraging these algorithms, our study contributes to the existing literature by providing valuable insights into the factors influencing computational identity and programming empowerment in educational contexts.

The findings of this study provide valuable insights into students' perspectives on computational identity and programming empowerment, as well as the performance of various text mining algorithms in predicting these perspectives. By analyzing the open-ended responses of 646 programming students, we gained a deeper understanding of their engagement, imagination, and affiliation in relation to computational identity, as well as their perceptions of meaningfulness, impact, and self-efficacy in programming empowerment. These findings suggest that students are motivated and interested in programming activities, which aligns with the literature emphasizing the importance of computational identity in fostering students' interest and involvement in programming. In terms of programming empowerment, positive comments emphasized the benefits of programming, the desire to make a difference and impact with programming skills, and self-perceived competence in programming tasks. These findings align with the notion that empowered learners find programming tasks meaningful, believe in their abilities, and perceive their efforts as impactful. It is encouraging to see that students recognize the potential of programming to solve real-world problems, improve lives, and contribute to positive societal change. These aspects of programming empowerment are essential for promoting students' motivation, confidence, and sense of purpose in their programming endeavors. The performance of the text mining algorithms in predicting computational identity and programming empowerment was also assessed. Logistic regression and TF-IDF representation achieved the highest accuracy rates for both computational identity (93%) and programming empowerment (96%). These results suggest the potential effectiveness of these algorithms in analyzing and predicting students' perspectives on these constructs. However, it is important to note that other algorithms, such as decision tree, support vector machines, random forest, and artificial neural networks, also demonstrated relatively high accuracy rates, ranging from 81% to 94%. These findings indicate that multiple algorithms can be utilized for predicting students' perspectives, and the choice of algorithm may depend on specific requirements and preferences. Despite the promising findings, there are several limitations to consider. Firstly, the dataset predominantly consisted of positive comments, which may not fully capture the range of students' experiences and perspectives. Future research could address this limitation by incorporating a more balanced dataset, including negative comments or contrasting viewpoints. Additionally, the dataset was obtained through self-reported responses from students, which may be subject to bias or influenced by social desirability. Combining the text mining approach with qualitative methods, such as interviews or observations, could provide a more comprehensive understanding of students' computational identity and programming empowerment. Furthermore, the generalizability of the findings may be limited to the specific context and sample of this study. The participants were programming students from secondary education and first-year university levels, which may not fully represent the diversity of programming learners. Future research could involve a more diverse sample, including learners from different age groups, educational backgrounds, and programming experiences. This would provide a more comprehensive understanding of how computational identity and programming empowerment evolve across various learning stages. In conclusion, this study contributes to the understanding of computational identity and programming empowerment by employing text mining algorithms to analyze students' perspectives. The high accuracy rates achieved by the text mining algorithms

suggest their potential in predicting students' perspectives, thereby facilitating personalized learning and support in programming education. However, further research is needed to address the limitations and refine the approaches used in this study. By doing so, we can advance our understanding of students' experiences and develop effective interventions to foster their computational identity and programming empowerment.

Information Note

This study is derived from a master's thesis written by the first author under the supervision of the second author.

Author Contributions: Author 1: 60%-Research design, literature review, method, analysis, findings, and conclusions, Author 2: 40 %- Research design, discussion and conclusion.

Ethical Statement and Conflict of Interest

Scientific ethical principles and rules were taken as the basis in all stages of this research, including preparation, data collection and analysis, and reporting. The ethical standards and conditions of the Committee on Publication Ethics (COPE) have been accepted and acted accordingly. The study did not receive funding from an institution or organization. There is no conflict of interest in the article.

BİLGİ NOTU

Bu çalışma ikinci yazarın danışmanlığında ilk yazar tarafından yazılmış yüksek lisans tez çalışmasından üretilmiştir.

Yazar Katkıları: Yazar 1: %60-Araştırma tasarımı, literatür tarama, yöntem, analiz, bulgu ve sonuçlar, Yazar 2: %40-Araştırma tasarımı, tartışma ve sonuç

Etik Beyan ve Çıkar Çatışması

Bu araştırmanın hazırlık, verilerin toplanması ve analizi, raporlama olmak üzere tüm aşamalarında bilimsel etik ilke ve kuralları temel alınmıştır. Committee on Publication Ethics (COPE)' in etik standartları ve koşullarını kabul edilmiş ve buna uygun davranılmıştır. Çalışma, bir kurum veya kuruluş tarafından fon desteği almamıştır. Makalede çıkar çatışması bulunmamaktadır.

REFERENCES

Angeli, C., & Valanides, N. (2020). Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in Human Behavior, 105,* 105954

Aninditya, A., Hasibuan, M. A., & Sutoyo, E. (2019, November). Text mining approach using TF-IDF and naive Bayes for classification of exam questions based on cognitive level of bloom's taxonomy. In 2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS) (pp. 112-117). IEEE.

Antons, D., Grünwald, E., Cichy, P., & Salge, T. O. (2020). The application of text mining methods in innovation research: current state, evolution patterns, and development priorities. *R&D Management, 50*(3), 329-351.

Atman-Uslu, N., Mumcu, F., & Eğin, F. (2018). Görsel programlama etkinliklerinin ortaokul öğrencilerinin bilgi-işlemsel düşünme becerilerine etkisi. *Ege Eğitim Teknolojileri Dergisi 2*(1), 19-31.

Atman Uslu, N. (2022). How do computational thinking self-efficacy and performance differ according to secondary school students' profiles? The role of computational identity, academic resilience, and gender. *Education and Information Technologies*, 1-25.

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada (Vol. 1, p. 25).

Brousseau, E., & Sherman, M. (2019, October). Position: The Role of Blocks Programming in Forming Computational Identity. In 2019 IEEE Blocks and Beyond Workshop (B&B) (pp. 15-17). IEEE.

Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & education, 109*, 162-175.

Capobianco, B. M., French, B. F., & Diefes-Dux, H. A. (2012). Engineering identity development among pre-adolescent learners. *Journal of Engineering Education, 101*(4), 698–716. https://doi.org/10.1002/j.2168-9830.2012.tb01125.x

Frymier, A. B., Shulman, G. M. and Houser, M. L. 1996. The development of a learner empowerment measure. *Communication Education, 45*, 181–199.

Gupta, V., & Lehal, G. S. (2009). A survey of text mining techniques and applications. *Journal of emerging technologies in web intelligence, 1*(1), 60-76.

Hotho, A., Nürnberger, A., & Paaß, G. (2005). A brief survey of text mining. *Journal for Language Technology and Computational Linguistics, 20*(1), 19-62.

Houser, M. L., & Frymier, A. B. (2009). The role of student characteristics and teacher behaviors in students' learner empowerment. *Communication Education, 58*(1), 35-53.

Kazakof, E. R., Sullivan, A., & Bers, M. U. (2013). The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood. *Early Childhood Education Journal, 41*(4), 245–255.

Kong, S. C., & Wang, Y. Q. (2020). Formation of computational identity through computational thinking perspectives development in programming learning: A mediation analysis among primary school students. *Computers in Human Behavior, 106*, 106230.

Kong, S. C., Chiu, M. M., & Lai, M. (2018). A study of primary school students' interest, collaboration attitude, and programming empowerment in computational thinking education. *Computers & education, 127*, 178-189.

Kong, S. C., & Lai, M. (2022). Computational identity and programming empowerment of students in computational thinking development. British Journal of Educational Technology, 53(3), 668-686.

Korkmaz, Ö., Balcı, H., Çakır, R., & Erdoğmuş, F. U. (2020). Görsel programlama ortamlarında yapılan oyun geliştirme etkinliklerinin etkililiği. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi, (57),* 52-73.

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). *Text classification algorithms: A survey. Information, 10*(4), 150

Lin, P. H., & Chen, S. Y. (2020). Design and evaluation of a deep learning recommendation based augmented reality system for teaching programming and computational thinking. *IEEE Access, 8*, 45689-45699.

Moon, J., Do, J., Lee, D., & Choi, G. W. (2020). A conceptual framework for teaching computational thinking in personalized OERs. *Smart Learning Environments, 7*(1), 1-19.

Mouza, C., Yang, H., Pan, Y. C., Ozden, S. Y., & Pollock, L. (2017). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). *Australasian Journal of Educational Technology, 33*(3).

Oluk, A., Korkmaz, Ö., & Oluk, H. A. (2018). Scratch'ın 5. sınıf öğrencilerinin algoritma geliştirme ve bilgi-işlemsel düşünme becerilerine etkisi. *Turkish Journal of Computer and Mathematics Education (TURCOMAT), 9*(1), 54-71.

Onan, A., Korukoğlu, S., & Bulut, H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications, 57*, 232-247.

Page, N., & Czuba, C. E. (1999). Empowerment: What is it. *Journal of extension, 37*(5), 1-5.

Papadakis, S., Kalogiannakis, M., & Zaranis, N. (2016). Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: A case study. *International Journal of Mobile Learning and Organisation, 10*(3), 187–202

Romero, M., Lepage, A., & Lille, B. (2017). Computational thinking development through creative programming in higher education. *International Journal of Educational Technology in Higher Education, 14*(1), 1-15.

Saritepeci, M. (2020). Developing computational thinking skills of high school students: Design-based learning activities and programming tasks. *The Asia-Pacific Education Researcher, 29*(1), 35-54.

Sfard, A., & Prusak, A. (2005). Telling identities: In search of an analytic tool for investigating learning as a culturally shaped activity. *Educational Researcher, 34*(4), 14–22. https://doi.org/10.3102/0013189X034004014

Sobral, S. R. (2021). Teaching and Learning to Program: Umbrella Review of Introductory Programming in Higher Education. *Mathematics, 9*(15), 1737.

Sun, L., Guo, Z., & Zhou, D. (2022). Developing K-12 students' programming ability: A systematic literature review. *Education and Information Technologies, 27*(5), 7059-7097.

Tikva, C., & Tambouris, E. (2021). Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature Review. *Computers & Education, 162*, 104083.

Uday, S. S., Pavani, S. T., Lakshmi, T. J., & Chivukula, R. (2022). COVID-19 literature mining and retrieval using text mining approaches. arXiv preprint arXiv:2205.14781.

Yaman, U. C. (2022). *Metin madenciliği teknikleri ile Türkçe müşteri yorumlarının analizi* (Master's thesis, Eskişehir Teknik Üniversitesi).