A Supervised Hybrid Weighting Scheme for Bloom's Taxonomy Questions using Category Space Density-based Weighting

Sucipto
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia | Department Information System, Universitas Nusantara PGRI Kediri, Indonesia
sucipto.2305349@students.um.ac.id

Didik Dwi Prasetya
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia
didikdwi@um.ac.id (corresponding author)

Triyanna Widiyaningtyas
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia
triyannaw.ft@um.ac.id

Received: 27 January 2025 | Revised: 27 February 2025 | Accepted: 6 March 2025

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | *DOI: https://doi.org/10.48084/etasr.10226*

ABSTRACT

Question documents organized based on Bloom's taxonomy have different characteristics than typical text documents. Bloom's taxonomy is a framework that classifies learning objectives into six cognitive domains, each having distinct characteristics. In the cognitive domain, different keywords and levels are used to classify questions. Using existing category-based term weighting methods is less relevant because it is only based on word types and not on the main characteristics of Bloom's taxonomy. This study aimed to develop a more relevant term weighting method for Bloom's taxonomy by considering the term density in each category and the specific keywords in each domain. The proposed method, called Hybrid Inverse Bloom Space Density Frequency, is designed to capture the unique characteristics of Bloom's taxonomy. Experimental results show that the proposed method can be applied to all question datasets, considering term density in each category and keywords in each cognitive domain. Furthermore, the accuracy of the proposed method was superior on all datasets using machine learning model evaluation.

Keywords-term; question; Bloom's taxonomy; space density; machine learning

I. INTRODUCTION

Education is a process that emphasizes the provision of knowledge and develops critical and analytical thinking skills [1]. In this context, Bloom's Taxonomy (BT) plays a very important role, as it provides a framework that helps educators design learning objectives that focus on student cognitive development [1-2]. By dividing the learning process into six levels, including knowledge, comprehension, application, analysis, synthesis, and evaluation, this method allows for more comprehensive and holistic teaching. Each level of this taxonomy challenges students to memorize information and understand, make connections, and develop creative solutions to their problems [3]. In addition, BT serves as a guide for developing assessment methods that accurately measure students' cognitive progress. It also facilitates the implementation of adaptive learning strategies, ensuring that instructional materials align with individual learning needs. As educational technology advances, integrating BT into automated learning systems can enhance personalized learning experiences and improve overall educational outcomes.

In Natural Language Processing (NLP), understanding and extracting relevant information is essential for many applications, ranging from information retrieval to text classification. Text weighting using Term Frequency - Inverse Text Frequency (TF-IDF) is one of the most widely used methods to achieve this goal [4-6]. TF-IDF helps to identify words that best describe the nature of the document. This approach emphasizes rare but important words, allowing the system to avoid common words that do not provide much information in search or classification [7-9].

At the same time, in addition to text processing, the world of data often requires us to analyze hidden patterns in the distribution of data points [10-11]. Spatial density methods, particularly the Inverse Class Spatial Density Distribution

(ICSD) method, provide a novel approach to data segmentation based on density patterns. ICSD uses inversion layers to detect high-density areas, helping to identify patterns that are not easily visible by traditional methods [12]. This approach benefits data that are unevenly distributed or have significant noise. Although coming from different fields, these three concepts complement each other and contribute to new ways of understanding and managing information, whether in education, text analysis, or complex data processing.

Several studies have been conducted to obtain the best results in BT classification. Early research on Machine Learning (ML) models used standard text processing and the Support Vector Machine (SVM) algorithm [12]. Later, other studies adopted the dataset used in [12] to improve BT's classification performance [13]. Later studies introduced a refinement by assigning impact factors to words using part-of-speech markers (ETF-IDF) [14-15]. Further research introduced a new term weighting scheme called ETFPOS-IDF, which gives higher weights to verbs in BT than supporting verbs. In addition, other studies developed weighting using the Inverse Class Space Density Frequency (ICSδF) method [10-12], which calculates the density of documents in the category space based on each term used, allowing for more accurate pattern identification in BT classification.

Based on the theory and findings of previous studies, this one proposes a new approach to improve BT classification by modifying the TF-IDF and ICSδF models according to the meaning of BT categories. This proposal aims to identify BT keywords using thematic keyword terms. In addition, the new approach also proposes using a Guided Hybrid Weighting Scheme for BT Questions, utilizing Category Space Density (CSD) to improve accuracy in BT classification.

II. METHODS

A. Dataset

Term weighting research uses a taxonomy-based keyword approach, or thematic keywords, to analyze and determine the weight of words in a question text dataset. This research uses data from various studies, as shown in Table I.

TABLE I. COGNITIVE BLOOM'S TAXONOMY DATASET

Dataset	Cognitive Bloom's Taxonomy					
	1	2	3	4	5	6
1 [14]	26	23	15	23	30	24
2 [13]	100	100	100	100	100	100
3 [16]	271	300	300	300	300	300
4 [17]	9237	1431	3262	4333	1227	1705

The dataset in Table I consists of six classes according to BT. The collection of datasets varies from hundreds to tens of thousands of data. Four datasets have varying class distributions according to the data used in previous studies [13-14, 18]. The dataset is specific to BT classification and has a question sentence type. Table I presents four datasets classified based on BT, consisting of six cognitive complexity levels (1 to 6). Each dataset varies in the number of instances per category. Dataset 1 [14] has a relatively small and balanced number of instances, ranging from 15 to 30 per level. Dataset 2 [13] is entirely balanced, with 100 instances per category, which makes it ideal for studies that require equal representation. Dataset 3 [16] is moderately sized, with 271 to 300 instances per level, ensuring a more generalized distribution. Dataset 4 [17] is the largest dataset, with a significantly high number of instances in lower cognitive levels (9,237 in level 1) and fewer instances in higher levels (as low as 1,227 in level 5). This suggests a dominance of lower-order cognitive tasks.

Given its large size, Dataset 4 is more suitable for ML, particularly for models that benefit from vast data to improve classification accuracy. Datasets 1 and 2, being relatively small, are better suited for quick experiments or controlled studies, while Dataset 3, with its balanced distribution, provides better generalization. The uneven distribution in Dataset 4 may reflect educational biases where lower-order thinking tasks are more prevalent than higher-order cognitive skills. Selecting the appropriate dataset is crucial for BT classification using the space density-based method, where data distribution plays a key role in classification performance.

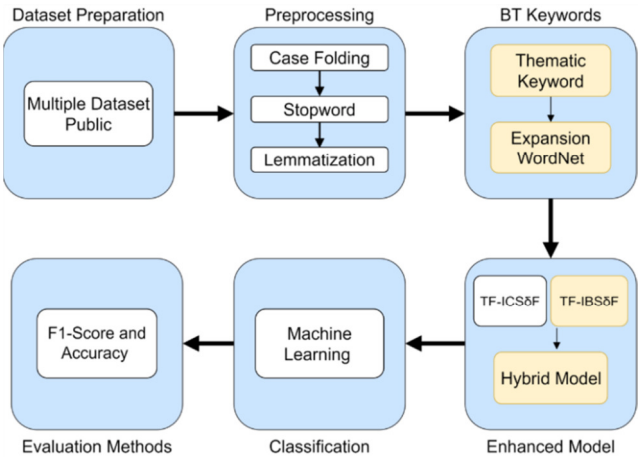

Fig. 1. Stages of the proposed space density-based question modification.

B. Data Preprocessing

This research design is experimental by modifying the Term Frequency (TF) algorithm as the primary method for word weighting [19-20]. Figure 1 presents the flow of this investigation. It begins with the selection of public datasets [13-14, 16, 17]. In the text analysis process, case folding, stop word removal, and lemmatization are the three main techniques used, as in some previous studies [19-22]. Case folding is converting all letters in the question text to lowercase to ensure consistency and reduce word variation due to capitalization differences. Stop-word removal involves removing common question words such as "and," "or," and "and in," which frequently appear in the question text but do not contribute significantly to the analysis. Lemmatization is a technique of converting words to their base form or lemma, such as converting "ran" to "run" so that different forms of words can be analyzed as the same entity. By applying these three techniques, text analysis becomes more accurate and efficient.

The novelty of the proposed model lies in the optimization of data preprocessing, utilizing a TF-IDF modification that uses

keyword expansion on BT base words with the term "thematic keyword" collected and validated by three experts in the field of BT. BT root words were expanded using WordNet [21, 23-26].

C. Modeling

This research is based on term weighting, a procedure to calculate the weight of each term searched in each document. This is performed to determine the availability and similarity of a term in the document [27-29].

1) Term Frequency - Inverse Document Frequency (TF-IDF)

Each term is assumed to have an importance proportional to the number of times it appears in the document, which is known as Term Frequency (TF).

$$TF = \begin{cases} 1 + \log_{10}(f_{t,d}), & f_{t,d} > 0 \\ 0, & f_{t,d} = 0 \end{cases} \quad (1)$$

where $f_{t,d}$ is the frequency of term t in document d.

Inverse Document Frequency (IDF) considers the occurrence of a term in a set of documents. The IDF function gives the lowest score to a term that appears in many documents in the document space $D = d_1, d_2, \ldots, d_n$.

$$df(t_i) = \sum d(t_i) \quad (2)$$

where $df(t_i)$ is the frequency of documents containing the term i, and $d(t_i)$ is the document containing the term i.

$$idf = \log \frac{D}{df_{(t_i)}} \quad (3)$$

where idf is the inverse of the document frequency $df_{(t_i)}$, and D is the total number of documents. Then, IDF is combined with TF by:

$$W_{TF*IDF(t_i,d_j)} = tf_{t_i,d_j} \times \left(\log \frac{D}{df_{(t_i)}} \right) \quad (4)$$

where $W_{TF*IDF(t_i,d_j)}$ is the weight of term i in document j, tf_{t_i,d_j} is the number of terms i in document j, D is the total number of documents, and $df_{(t_i)}$ is the total number of documents containing term i.

2) Inverse Class Frequency (ICF)

ICF is adapted from the IDF method, using the inverse ratio of the number of categories to the number of categories containing terms. In the case of class-oriented indexing, a subset of documents from the document space $D = d_1, d_2, \ldots, d_n$ is allocated to a particular category. So, the more often a term appears in documents in that category, the weight of the term is closer to 0. The ICF function gives the lowest score of terms that appear in several categories in the category space $C = C_1, C_2, \ldots, C_n$. ICF is given by (5) and (6):

$$cf_{(t_i)} = \sum c_{(t_i)} \quad (5)$$

where $cf(t_i)$ is the frequency of categories containing the term i, and $c_{(t_i)}$ is the category containing the term i.

$$icf = \log \frac{c}{cf_{(t_i)}} \quad (6)$$

where icf is the inverse of the class frequency $cf_{(t_i)}$, and C is the number of categories.

Therefore, a term's numerical representation is the product of TF (local parameter), IDF (global parameter), and ICF (category global parameter). The $TF * IDF * ICF$ equation is visualized in:

$$W_{TF*IDF*ICF(t_i,d_j,c_k)} = tf_{t_i,d_j} \times \left(\log \frac{D}{df_{(t_i)}} \right) \times \left(\log \frac{c}{cf_{(t_i)}} \right) \quad (7)$$

where $W_{TF*IDF*ICF(t_i,d_j,c_k)}$ is the weight of term i in document j, $tf_{ti,dj}$ is the number of terms i in document j, D is the total number of documents, $df_{(t_i)}$ is the total number of documents containing the term i, C is the total number of categories, and $cf_{(t_i)}$ is the total number of categories containing the term i.

3) Inverse Class Space Density Frequency (ICSδF)

ICSδF calculates the density of documents in the category space based on each term. Since the ICF function gives the lowest score to terms that appear in multiple categories without concern about the category space, the ICSδF calculation is proposed. ICSδF begins by calculating the class density C_δ by counting documents that contain terms in a particular category c_k as:

$$C_\delta(t_i) = \frac{n_{c_k(t_i)}}{N_{c_k}} \quad (8)$$

where $C_\delta(t_i)$ is the class density for term i, $n_{c_k(ti)}$ is the number of documents in category c_k containing the term i, and N_{c_k} is the total number of documents in category c_k.

Then, the density of the category space is calculated, which is the sum of the densities of all existing categories CS_δ:

$$CS_\delta(t_i) = \sum_{c_k} C_\delta(t_i) \quad (9)$$

where $CS_\delta(t_i)$ is the class space density for the term i, $C_\delta(t_i)$ is the category density for the term i, and c_k is the category $(k = 1, 2, \ldots, n)$. Then, the result of the category space density $CS_\delta(t_i)$ is inversed according to the concept in the previous TF-IDF-ICF as:

$$ICS_\delta F(t_i) = \log \left(\frac{c}{CS_\delta(t_i)} \right) \quad (10)$$

where $ICS_\delta F(t_i)$ is the inverse class space density frequency for term i, C is the total number of categories, and $CS_\delta(t_i)$ is the class space density for term i.

The next step is to multiply the result of the inverse density of the category space against term i ($ICS_\delta F(t_i)$) with TF-IDF:

$$W_{TF*IDF*ICS_\delta F(t_i,d_j,c_k)} = tf_{t_i,d_j} \times \left(\log \frac{D}{df_{(t_i)}} \right) \times \left(\log \frac{c}{CS_\delta(t_i)} \right) \quad (11)$$

where $W_{TF*IDF*ICS_\delta F(t_i,d_j,c_k)}$ is the weight of term i in document j in category k, tf_{t_i,d_j} is the number of terms i in

document j, D is the total number of documents, $df_{(t_i)}$ is the number of documents containing the term i, C is the total number of categories, and $CS_\delta(t_i)$ is the space density of categories for term i.

4) Inverse Bloom Space Density Frequency (IBSδF)

If ICSδF pays attention to the category space density of a term, IBSδF pays more attention to the cognitive BT space density. The density of the BT cognitive space is calculated to determine how much a term's weight is if the term's rarity from the entire BT cognitive space is also calculated. The rarer the term appears in the BT cognitive, the more the term has a high inverse value. Then, the first process in calculating IBSδF is to calculate the density of the cognitive BT first, as

$$B_\delta(t_i) = \frac{n_{b_l}(t_i)}{N_{b_l}} \quad (12)$$

where $B_\delta(t_i)$ expresses the density of BT's cognitive space for the term t_i, $n_{b_l}(t_i)$ is the number of occurrences of the term t_i in the l^{th} BT cognitive question document, N_{b_l} is the total number of all terms in the l^{th} BT cognitive question document, and b_l is the BT cognitive question document ($l = 1, 2, 3, \dots, n$).

In addition, the results of the density of the BT cognitive space against term i are calculated inversely to determine the level of scarcity of terms against the BT cognitive space, as:

$$BS_\delta(t_i) = \sum_{b_l} B_\delta(t_i) \quad (13)$$

where $BS_\delta(t_i)$ is the inverse of BT's cognitive space density for term t_i, measuring how rarely the term appears in various BT cognitive question documents. B represents the total number of BT cognitive question documents, while $BS_\delta(t_i)$ is the density of the BT cognitive space for term t_i, calculated based on the distribution of the term in various question documents. The less frequently a term appears in all BT cognitive question documents, the higher its inverse value, indicating that it has a more significant weight in BT-based text analysis.

Furthermore, the hybrid model obtained from the inverse results is multiplied by $TF * IDF * ICS\delta F$ to determine the term weight that considers the density of classes and the BT cognitive spaces as:

$$W_{hybrid} = tf_{t_i,d_j} \times \left(\log \frac{D}{df_{(t_i)}}\right) \times \left(\log \frac{C}{CS_\delta(t_i)}\right) \times \left(\log\left(\frac{B}{BS_\delta(t_i)}\right)\right) \quad (14)$$

where W_{hybrid} is the weight of term i in BT j cognitive in category k in BT l cognitive question document, tf_{t_i,d_j} is the number of terms i in cognitive BT j, D is the total number of BT cognitive question documents, $df_{(t_i)}$ is the number of BT cognitive question documents containing the term i, C is the total number of categories, $CS_\delta(t_i)$ is the density of category space for term i, B is the total number of BT cognitive document collections, and $BS_\delta(t_i)$ is the space density of the BT cognitive question documents against the term i.

D. Classification Method and Evaluation

This study used SVM and Naïve Bayes ML classification models. Both models use standard parameters, with SVM using SVC (kernel='linear') and NB using MultinomialNB(). Data were split in 80:20 for training and testing. The evaluation matrix uses accuracy and F1-score. Choosing the F1-score and accuracy gives a good idea of the quality of the model in terms of overall performance and the balance between positive and negative predictions.

$$Accuracy = \frac{\sum_{i=1}^{n}(TP_i + TN_i)}{\sum_{i=1}^{n}(TP_i + TN_i + FP_i + FN_i)} \quad (15)$$

$$Precision = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FP_i)} \quad (16)$$

$$Recall = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FN_i)} \quad (17)$$

$$F1 - score = 2 \times \frac{\sum_{i=1}^{n} Precision_i \times Recall_i}{\sum_{i=1}^{n}(Precision_i + Recall_i)} \quad (18)$$

where TP denotes true positives, TN denotes true negatives, FP denotes false positives, and FN denotes false negatives.

III. RESULTS AND DISCUSSION

SVM and NB were employed to process and classify the data. The evaluation focuses on key performance metrics, specifically accuracy and F1-score, to measure the effectiveness of each approach. Comparing the proposed method with existing text weighting techniques aims to demonstrate the impact of incorporating space density-based thematic weighting on classification performance. The experiments were conducted on four distinct datasets, ensuring a comprehensive analysis of the model's ability to generalize across different data distributions.

The comparative analysis includes four text data processing methods: ETF-IDF, ETFPOS-IDF, TF-ICSδF, and HTF-IBSδF, as presented in Tables II and III. These methods were evaluated for their ability to enhance feature representation and improve classification outcomes in BT question categorization. The results highlight variations in performance across datasets, emphasizing the strengths and limitations of each technique. The inclusion of space density-based weighting in the HTF-IBSδF approach shows a noticeable improvement over traditional methods, particularly in handling semantic relationships within the textual data. This analysis further reinforces the importance of context-aware weighting schemes in optimizing classification tasks and improving the reliability of automated educational assessments.

TABLE II. EXPERIMENTAL RESULTS OF SVM

Dataset	ETF-IDF		ETFPOS-IDF		TF-ICSδF		HTF-IBSδF	
	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy
1	0.469	0.517	0.610	0.540	0.560	0.660	**0.620**	**0.690**
2	0.741	0.733	0.790	0.780	0.790	0.790	**0.810**	**0.810**
3	0.946	0.946	0.950	0.950	0.950	0.950	**0.960**	**0.960**
4	0.978	0.984	0.970	0.980	**0.980**	0.980	**0.980**	**0.990**

TABLE III. EXPERIMENTAL RESULTS OF NB

Dataset	ETF-IDF		ETFPOS-IDF		TF-ICSδF		HTF-IBSδF	
	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy
1	0.585	0.586	0.630	**0.660**	0.640	**0.660**	**0.660**	**0.660**
2	0.725	0.725	0.700	0.700	0.700	0.700	0.770	0.770
3	0.875	0.876	**0.880**	**0.880**	0.870	0.870	**0.880**	**0.880**
4	0.927	0.950	0.890	0.930	**0.930**	0.950	**0.930**	**0.950**

TABLE IV. AVERAGE RESULTS PER DATASET

Dataset	ETF-IDF		ETFPOS-IDF		TF-ICSδF		HTF-IBSδF	
	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy
1	0.527	0.552	0.620	0.600	0.600	0.660	**0.640**	**0.675**
2	0.733	0.729	0.745	0.740	0.745	0.745	**0.790**	**0.790**
3	0.911	0.911	0.915	0.915	0.910	0.910	**0.920**	**0.920**
4	0.953	0.967	0.930	0.955	**0.955**	0.965	**0.955**	**0.970**

Table II shows the experimental results using SVM. These results show that HTF-IBSδF consistently produces the highest F1-score and accuracy in all datasets, reflecting its superiority in capturing data patterns and characteristics. On the smallest dataset, Dataset 1, HTF-IBSδF has an F1-score of 0.620 and an accuracy of 0.690, which is superior to other methods. This superiority is even more evident on the larger dataset, Dataset 4, where HTF-IBSδF reaches a maximum F1-score of 0.980 and 0.990 for accuracy, indicating its reliability on more complex or structured datasets.

Meanwhile, the TF-ICSδF method performs close to HTF-IBSδF, especially in datasets 3 and 4, with identical F1-score and accuracy. This indicates that TF-ICSδF is also an effective method, although not as strong as HTF-IBSδF in some cases. On the other hand, ETF-IDF and ETFPOS-IDF show improved performance as the complexity of the dataset increases but cannot surpass the performance of TF-ICSδF and HTF-IBSδF. Table I underlines the importance of choosing the proper text processing method to maximize the analysis results. HTF-IBSδF is the best choice based on the test results.

Table III shows the experimental results using the NB algorithm, based on F1-score and accuracy on the same four datasets as in Table II. However, the results in Table III tend to be more consistent and show a more balanced pattern among the methods tested. In Dataset 1, all methods performed similarly, with the best values achieved by ETFPOS-IDF, TF-ICSδF, and HTF-IBSδF. Interestingly, ETF-IDF had significantly improved performance on subsequent datasets, although it remained below the performance of ETFPOS-IDF, TF-ICSδF, and HTF-IBSδF.

Compared with Table II, it can be seen that the difference between the methods in Table III is smaller, indicating the possibility of adjustment or normalization in the evaluation. The HTF-IBSδF method, which was dominant in the first table, performs similarly to TF-ICSδF in the second one, especially in datasets 3 and 4, with the highest F1-score and accuracy of 0.930 and 0.950, respectively. This change shows the importance of the influence of the data structure and evaluation parameters on the test results. Table III reinforces the conclusion that more complex methods, such as HTF-IBSδF, can perform better on larger or more complex datasets. In contrast, the ETF-IDF, ETFPOS-IDF, and TF-ICSδF methods remain competitive in specific scenarios.

Table IV shows the average performance results of the ETF-IDF, ETFPOS-IDF, TF-ICSδF, and HTF-IBSδF methods on the four datasets. HTF-IBSδF again stands out with the highest average value, especially on datasets 1 and 2. On the smallest dataset, Dataset 1, HTF-IBSδF achieved an F1-score of 0.640 and an accuracy of 0.675, which is higher than the other methods. This pattern shows that HTF-IBSδF is superior at capturing BT thematic patterns even on more straightforward datasets. This superiority is maintained on more complex datasets, such as Dataset 4, where HTF-IBSδF recorded an F1-score of 0.955 and an accuracy of 0.970, equivalent to the highest performance of the TF-ICSδF method but still showing better consistency on the previous dataset.

The superiority of HTF-IBSδF can be attributed to using the space density-based BT thematic weighting scheme in the hybrid model. This approach allows the method to comprehensively understand information distribution by integrating spatial and semantic factors in word weighting. This is especially important in datasets that contain hidden patterns or uneven data distribution. Compared to other methods, such as ETF-IDF, that rely more on simple weighting, HTF-IBSδF can optimize the retrieval of relevant information with higher precision. In addition, the hybrid model used by HTF-IBSδF helps reduce data bias that often affects the results of conventional weighting.

The consistent and superior results of HTF-IBSδF demonstrate its potential to be applied to complex text-based

applications, such as sentiment analysis, text classification, or information extraction. In a scientific context, this thematic weighting-based hybrid model not only strengthens the validity of the method but also broadens the scope of its use. By combining density space and thematic scheme, HTF-IBSδF proves its ability to produce a more robust and relevant analysis than other methods, such as ETF-IDF, ETFPOS-IDF, and TF-ICSδF. This makes HTF-IBSδF a highly recommended approach for high-complexity NLP tasks.

The results in Figure 2 demonstrate the superiority of the HTF-IBSδF text weighting scheme in classifying BT questions, achieving the highest accuracy (0.826) and F1-score (0.839) across four datasets using two ML models. This consistent performance indicates that HTF-IBSδF effectively enhances text classification by leveraging a space density-based thematic weighting approach, which captures deeper semantic relationships between words within the BT context.

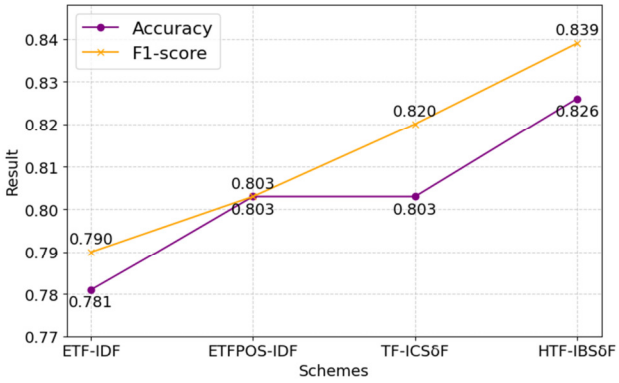

Fig. 2. Average results of each scheme on all datasets.

The primary novelty of this study lies in integrating space density-based thematic weighting into a hybrid machine-learning model for BT classification. Unlike conventional weighting schemes such as ETF-IDF and ETPOS-IDF that focus primarily on weighting verbs, or TF-ICSδF that only applies general space density weighting, HTF-IBSδF introduces a more dynamic mechanism that incorporates both space density and thematic relevance. This allows the model to distinguish cognitive levels with higher precision by enhancing the representation of semantically significant terms in educational text classification.

This work significantly advances text classification in educational settings by demonstrating that context-aware weighting strategies outperform traditional frequency-based approaches. Unlike previous studies that rely on standard TF-IDF variations, this provides empirical evidence that incorporating space density and semantic correlations leads to measurable improvements in classification accuracy. A key strength of this study is that the experiments were carried out using the same datasets for all weighting schemes, ensuring a fair and direct performance comparison. By achieving state-of-the-art results, this research contributes to the development of more effective automated assessment systems, adaptive learning platforms, and intelligent tutoring systems, ultimately enhancing technology-based learning environments.

IV. CONCLUSION

This study presented a space density-based thematic weighting scheme with a hybrid model (HTF-IBSδF). The effectiveness of the proposed scheme was examined using two ML models and four datasets. The results of the proposed HTF-IBSδF were compared with previous methods, using two commonly used classification evaluation metrics, accuracy and F1-score. The classification results show that the proposed scheme had an accuracy of 0.826 and an F1-score of 0.839. These results show that the hybrid model with keyword thematic space density significantly improves the accuracy in classifying the overall question dataset. Furthermore, the research found ideal weight differences between the specifications of the six BT categories using ML methods.

ACKNOWLEDGMENT

The authors would like to thank the Directorate General of Higher Education, Research and Technology, Ministry of Education, Culture, Research and Technology of the Republic of Indonesia, for providing financial support for this research through the 2024 Doctoral Dissertation Research Scheme 0667/E5/AL.04 /2024.

REFERENCES

[1] M. Seaman, "Bloom's taxonomy," in *Curriculum and Teaching Dialogue*, vol. 13, American Association for Teaching & Curriculum, 2012.

[2] M. Chindukuri and S. Sivanesan, "Transfer learning for Bloom's taxonomy-based question classification," *Neural Computing and Applications*, vol. 36, no. 31, pp. 19915–19937, Nov. 2024, https://doi.org/10.1007/s00521-024-10241-y.

[3] L. H. Waite, J. F. Zupec, D. H. Quinn, and C. Y. Poon, "Revised Bloom's taxonomy as a mentoring framework for successful promotion," *Currents in Pharmacy Teaching and Learning*, vol. 12, no. 11, pp. 1379–1382, Nov. 2020, https://doi.org/10.1016/j.cptl.2020.06.009.

[4] P. Sahu, M. Cogswell, A. Divakaran, and S. Rutherford-Quach, "Comprehension Based Question Answering using Bloom`s Taxonomy," in *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, Dec. 2021, pp. 20–28, https://doi.org/10.18653/v1/2021.repl4nlp-1.3.

[5] A. S. Alammary, "Arabic Questions Classification Using Modified TF-IDF," *IEEE Access*, vol. 9, pp. 95109–95122, 2021, https://doi.org/10.1109/ACCESS.2021.3094115.

[6] M. Sudarma and J. Sulaksono, "Implementation of TF-IDF Algorithm to detect Human Eye Factors Affecting the Health Service System," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 4, no. 1, pp. 123–130, Feb. 2020, https://doi.org/10.29407/intensif.v4i1.13858.

[7] M. Liang and T. Niu, "Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs," *Procedia Computer Science*, vol. 208, pp. 460–470, Jan. 2022, https://doi.org/10.1016/j.procs.2022.10.064.

[8] A. Aninditya, M. A. Hasibuan, and E. Sutoyo, "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)*, Bali, Indonesia, Nov. 2019, pp. 112–117, https://doi.org/10.1109/IoTaIS47347.2019.8980428.

[9] T. Zhang and S. S. Ge, "An Improved TF-IDF Algorithm Based on Class Discriminative Strength for Text Categorization on Desensitized Data," in *Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence*, Suzhou, China, Mar. 2019, pp. 39–44, https://doi.org/10.1145/3319921.3319924.

[10] K. Chen, Z. Zhang, J. Long, and H. Zhang, "Turning from TF-IDF to TF-IGM for term weighting in text classification," *Expert Systems with Applications*, vol. 66, pp. 245–260, Dec. 2016, https://doi.org/10.1016/j.eswa.2016.09.009.

[11] Z. Tang, W. Li, and Y. Li, "An improved supervised term weighting scheme for text representation and classification," *Expert Systems with Applications*, vol. 189, Mar. 2022, Art. no. 115985, https://doi.org/10.1016/j.eswa.2021.115985.

[12] M. O. Gani, R. K. Ayyasamy, A. Sangodiah, and Y. T. Fui, "USTW Vs. STW: A Comparative Analysis for Exam Question Classification based on Bloom's Taxonomy," *MENDEL*, vol. 28, no. 2, pp. 25–40, Dec. 2022, https://doi.org/10.13164/mendel.2022.2.025.

[13] A. A. Yahya, Z. Toukal, and A. Osman, "Bloom's Taxonomy–Based Classification for Item Bank Questions Using Support Vector Machines," in *Modern Advances in Intelligent Systems and Tools*, 2012, pp. 135–140, https://doi.org/10.1007/978-3-642-30732-4_17.

[14] M. Mohammed and N. Omar, "Question Classification Based on Bloom's Taxonomy Using Enhanced TF-IDF," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 4–2, pp. 1679–1685, Sep. 2018, https://doi.org/10.18517/ijaseit.8.4-2.6835.

[15] M. Mohammed and N. Omar, "Question classification based on Bloom's taxonomy cognitive domain using modified TF-IDF and word2vec," *PLOS ONE*, vol. 15, no. 3, 2020, Art. no. e0230442, https://doi.org/10.1371/journal.pone.0230442.

[16] D. Sheelam, "Blooms data set." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/dineshsheelam/blooms-data-set.

[17] P. Lu *et al.*, "ScienceQA: Science Question Answering Dataset." https://scienceqa.github.io/#dataset.

[18] Supriyono, A. P. Wibawa, Suyono, and F. Kurniawan, "Advancements in natural language processing: Implications, challenges, and future directions," *Telematics and Informatics Reports*, vol. 16, Dec. 2024, Art. no. 100173, https://doi.org/10.1016/j.teler.2024.100173.

[19] D. D. Prasetya, A. P. Wibawa, and T. Hirashima, "The performance of text similarity algorithms," *International Journal of Advances in Intelligent Informatics*, vol. 4, no. 1, pp. 63–69, Mar. 2018, https://doi.org/10.26555/ijain.v4i1.152.

[20] F. A. Ahda, A. P. Wibawa, D. D. Prasetya, and D. A. Sulistyo, "Comparison of Adam Optimization and RMS prop in Minangkabau-Indonesian Bidirectional Translation with Neural Machine Translation," *JOIV : International Journal on Informatics Visualization*, vol. 8, no. 1, pp. 231–238, Mar. 2024, https://doi.org/10.62527/joiv.8.1.1818.

[21] S. Sucipto, D. D. Prasetya, and T. Widiyaningtyas, "A review questions classification based on Bloom taxonomy using a data mining approach," *ITEGAM-JETIA*, vol. 10, no. 48, pp. 161–170, Aug. 2024, https://doi.org/10.5935/jetia.v10i48.1204.

[22] A. Alqahtani, H. Alhakami, T. Alsubait, and A. Baz, "A Survey of Text Matching Techniques," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6656–6661, Feb. 2021, https://doi.org/10.48084/etasr.3968.

[23] T. T. Goh, N. A. A. Jamaludin, H. Mohamed, M. N. Ismail, and H. Chua, "Semantic Similarity Analysis for Examination Questions Classification Using WordNet," *Applied Sciences*, vol. 13, no. 14, Jan. 2023, Art. no. 8323, https://doi.org/10.3390/app13148323.

[24] K. Jayakodi, M. Bandara, I. Perera, and D. Meedeniya, "WordNet and Cosine Similarity based Classifier of Exam Questions using Bloom's Taxonomy," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 11, no. 04, Apr. 2016, Art. no. 142, https://doi.org/10.3991/ijet.v11i04.5654.

[25] V. A. Silva, I. I. Bittencourt, and J. C. Maldonado, "Automatic Question Classifiers: A Systematic Review," *IEEE Transactions on Learning Technologies*, vol. 12, no. 4, pp. 485–502, Jul. 2019, https://doi.org/10.1109/TLT.2018.2878447.

[26] P. Sharma and N. Joshi, "Knowledge-Based Method for Word Sense Disambiguation by Using Hindi WordNet," *Engineering, Technology & Applied Science Research*, vol. 9, no. 2, pp. 3985–3989, Apr. 2019, https://doi.org/10.48084/etasr.2596.

[27] T. Wang, Y. Cai, H. Leung, R. Y. K. Lau, H. Xie, and Q. Li, "On entropy-based term weighting schemes for text categorization," *Knowledge and Information Systems*, vol. 63, no. 9, pp. 2313–2346, Sep. 2021, https://doi.org/10.1007/s10115-021-01581-5.

[28] A. Sangodiah, T. J. San, Y. T. Fui, L. E. Heng, R. K. Ayyasamy, and N. A. Jalil, "Identifying Optimal Baseline Variant of Unsupervised Term Weighting in Question Classification Based on Bloom Taxonomy," *MENDEL*, vol. 28, no. 1, pp. 8–22, Jun. 2022, https://doi.org/10.13164/mendel.2022.1.008.

[29] D. E. Cahyani and I. Patasik, "Performance comparison of TF-IDF and Word2Vec models for emotion text classification," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2780–2788, Oct. 2021, https://doi.org/10.11591/eei.v10i5.3157.