ISSN: 1803-3814 (Printed), 2571-3701 (Online)
https://doi.org/10.13164/mendel.2022.2.025

USTW Vs. STW: A Comparative Analysis for Exam Question Classification based on Bloom's Taxonomy

Mohammed Osman Gani[1], Ramesh Kumar Ayyasamy[2,✉], Anbuselvan Sangodiah[3], Yong Tien Fui[2]

[1]Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Kampar, Malaysia
[2]Department of Information Systems, Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Kampar, Malaysia
[3]School of Computing, Faculty of Computing and Engineering, Quest International University Perak, Ipoh, Malaysia
osman.gani@1utar.my, rameshkumar@utar.edu.my✉, anbuselvan.sangodiah@qiu.edu.my, yongtf@utar.edu.my

Abstract

Bloom's Taxonomy (BT) is widely used in educational institutions to produce high-quality exam papers to evaluate students' knowledge at different cognitive levels. However, manual question labeling takes a long time, and not all evaluators are familiar with BT. The researchers worked to automate the exam question classification process based on BT as a solution. Enhancement in term weighting is one of the ways to increase classification accuracy while working with text data. However, all the past work on the term weighting in exam question classification focused on unsupervised term weighting (USTW) schemes. The supervised term weighting (STW) schemes showed effectiveness in text classification but were not addressed in past studies of exam question classification. As a result, this study focused on the effectiveness of STW in classifying exam questions using BT. Hence, this research performed a comparative analysis between the USTW schemes and STW for exam question classification. The STW schemes used in this study are TF-ICF, TF-IDF-ICF, and TF-IDF-ICSDF, whereas the USTW schemes used for comparison are TF-IDF, ETF-IDF, and TFPOS-IDF. This study used Support Vector Machines, Naïve Bayes, and Multilayer Perceptron to train the models. Accuracy and F1 score were used in this study to evaluate the classification results. The experiment results showed that overall, the STW scheme TF-ICF outperformed all the other schemes, followed by the USTW scheme ETF-IDF. Both the ETF-IDF and TFPOS-IDF outperformed standard TF-IDF. The outcome of this study indicates the future research direction where the combination of STW and USTW schemes may increase the accuracy of BT-based exam question classification.

Keywords: *Bloom's Taxonomy, Exam Question Classification, Supervised, Term Weighting, TF-IDF, TF-ICF, Unsupervised.*

Received: 17 October 2022
Accepted: 17 November 2022
Online: 27 November 2022
Published: 20 December 2022

1 Introduction

The written examination is still a traditional way of assessing students' knowledge in today's educational institutions. Questioning is an effective method for evaluating students' knowledge. So, a high-quality question paper needs to produce to assess the students' knowledge at different cognitive levels. However, producing a high-quality question paper containing questions from different cognitive levels is difficult for the evaluators [1]. Therefore, to produce high-quality exam papers, many academicians tend to use a framework called Bloom's Taxonomy (BT) [16]. Its advantage for academicians is that it covers multiple cognitive levels.

The cognitive domain of the BT consists of six different levels, which are ordered in Fig. 1 according to their increasing level of complexity. For every cognitive level of the cognitive domain, there is a set of verb lists associated with it. These verbs are also known as BT keywords. Academicians may classify questions into different cognitive levels based on these keyword lists. Still, there are some issues academicians face while classifying questions based on BT, such as manual classification of exam questions is very time-consuming and tedious. The manual process of classification is not practical when it comes to the amount of time it takes. Other than time, [30] mentioned that not every academician has the ability to classify the cognitive level of questions which may lead to high chances of misclassification. That is why it is significant to automate the classification of exam questions using BT.

There is a lot of past research to automate the process of BT-based exam question classification. To improve classification accuracy, researchers have put a lot

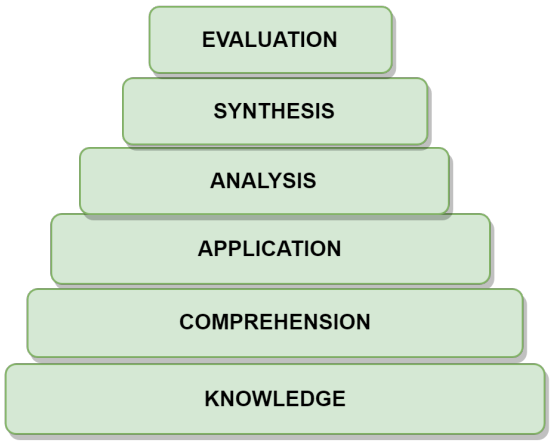

Figure 1: Each level of the cognitive domain of BT.

of effort into BT-based exam question classification. One of the ways to achieve better classification accuracy is term weighting. Term weighting is a method of indicating the importance and significance of a term in a particular document by assigning a numerical weight for that term. Many past studies [16, 18, 11, 17] worked on the enhancement of the term weighting in exam question classification. Term weighting can be both supervised term weighting (STW) and unsupervised term weighting (USTW).

All previous work on term weighting [16, 18, 11, 17] in exam question classification focused on USTW schemes. No past work has ever used the STW approach nor investigated whether it is effective for exam question classification, to the best of our knowledge. However, there is a likelihood that the STW schemes may work well for exam question classification due to their effectiveness in text classification. So, it is significant to investigate the effectiveness of STW schemes for BT-based exam question classification to improve classification accuracy. For that reason, this study aims to conduct a comparative analysis between STW and USTW schemes for BT-based exam question classification.

This paper consists of five sections. The first section is the introduction, where the background of the BT-based exam question classification is discussed. The next section of the paper is related work. The related work section discussed previous studies on term weighting schemes in BT-based exam question classification in addition to the STW and USTW schemes in text classification. The third section is the discussion of the methodology used in this research. Section four covered the experiment results and the discussion of the findings. The last section discussed future research and the conclusion of the paper.

2 Related Work

Term weighting refers to the text's vector representation in the field of text classification. In other words, term weighting schemes assign a score to all the terms present in the document during the vector representation [2]. Machine learning (ML) based classifiers are unable to interpret the raw text. As a result, term weighting is essential, as raw text cannot be directly input into the ML classifier. Indicating a term's importance and significance in a particular document is another purpose of term weighting. So, assigning a numerical weight for all the terms present in the document is essential for ML-based text classification. Besides feature selection and feature extraction, term weighting is one of the areas where researchers worked extensively to improve text classification accuracy. In the same way, the researchers worked on term weighting [16, 18, 11, 17] in classifying BT-based exam questions to improve classification accuracy. The above-stated past studies on term weighting for BT-based exam question classification used USTW schemes. STW schemes are very much established in text classification and show good effectiveness. Several earlier studies [15, 21, 14] reported better performance of some STW schemes in some cases compared to USTW schemes in text classification. Similarly, using the STW scheme might improve the Accuracy of BT-based exam question classification. So, this study aims to investigate the effectiveness of STW schemes for BT-based exam question classification.

2.1 USTW Schemes in Text Classification

The USTW schemes are based on the distribution of terms in a corpus. USTW schemes do not rely on the class information of the documents during term weighting. Not considering document distribution is the major drawback of the USTW schemes [2]. The advantage of the USTW schemes is their applicability for binary as well as multi-class classification problems. Binary term weighting is one of the earliest and most basic term weighting schemes, also called term presence (TP) [10]. Binary term weighting comes under USTW since, in binary term weighting, there is no class or category distribution involvement. In binary term weighting, if a term appears in the document, then the assigned weight for that term is 1, else 0. The binary weighting scheme fails in text classification because it cannot distinguish whether a term appears just once or multiple times [13]. Later, researchers used another USTW scheme called term frequency (TF), in which the weight of a term depends on the frequency of that term in the document. According to [13], TF fails in text classification because it gives higher weight to terms that appear more often, yet they have no discriminative power. To overcome the limitation of TF, researchers introduced inverse document frequency (IDF) in text classification [6]. IDF of a term is obtainable by dividing the total count of documents by the total count of documents containing that term in the corpus. IDF discriminates between the terms that occur frequently and rarely in the corpus. A novel USTW scheme called TF-IDF is formed by combining TF and IDF. TF-IDF is the most

frequently used USTW scheme in text classification [9].

2.2 STW Schemes in Text Classification

Text classification is a supervised learning task, so class information is available for every document. Unlike USTW, STW considers the distribution of the documents to the classes or categories present in a dataset. STW schemes often use local weighting schemes such as TP, TF, and augmented term frequency (ATF) in addition to document distribution. In other words, the IDF portion of the USTW scheme TF-IDF is replaced with the supervised component, as shown in Table 1.

[7] proposed three statistics-based STW schemes: TF- information gain, TF-chi-square, and TF-gain ratio, which are also famous as feature selection methods in ML. The experiment result of [7] showed the overall superiority of the gain ratio over the other two statistical approaches, chi-square and information gain. This paper further mentioned that the STW schemes did not show consistent superiority over the USTW scheme TF-IDF. Many studies [15, 21, 9] widely compared these popular statistics-based term weighting methods with USTW schemes such as TF and TF-IDF in text classification. [15] reported that, in some cases, these STW schemes outperformed USTW schemes, but the consistent superiority of STW schemes over the USTW schemes is not proven. [10] discussed an issue regarding the popular statistics-based STW schemes. They mentioned that statistics-based STW schemes have limitations when it comes to multi-class classification problems. These statistics-based STW schemes were designed for binary classification problems by default. Statistics-based STW schemes can be used for multi-class problems by the one vs. rest classification method but are computationally expensive.

There are a lot of schemes proposed by the researchers applicable for multi-class text classification, as shown in Table 1. [21] reported an issue regarding the TF. They stated that exam questions usually contain fewer terms. For that reason, the TF of each term is usually 1. They further mentioned that a term in a question might have multiple occurrences, but it is tough to claim that the multiple occurrences of the terms should be more significant than the terms with TF 1. So they proposed QF*ICF and IQF*QF*ICF, where QF and IQF are question frequency and inverse question frequency. These two STW schemes are supervised variants of the famous USTW scheme TF-IDF and can be used for multi-class text classification. [26] proposed term-frequency-inverse category frequency (TF-ICF). TF-ICF differs from the QF*ICF since it uses TF instead of question frequency. Their experiment result showed that TF-ICF outperformed TF and TF-IDF consistently in all the experiments. [22] combined the TF-ICF with the IDF and introduced TF-IDF-ICF. They highlighted that both TF-IDF and TF-ICF favor rare terms. To create positive discrimination between rare and frequent terms, they modified the TF-IDF-ICF and proposed another

Table 1: Previous research work on STW in text classification.

Research Work	Binary or Multi-class?	Proposed scheme
[7]	Binary	TF-Chi-square (X2)
[7]	Binary	TF-Information Gain (IG)
[7]	Binary	TF-Gain Ratio (GR)
[21]	Multi-class	QF*ICF
[21]	Multi-class	IQF*QF*ICF
[26]	Multi-class	TF-ICF
[22]	Multi-class	TF-IDF-ICF
[22]	Multi-class	TF-IDF-ICSDF
[4]	Multi-class	TF-IGM, RTF-IGM
[10]	Multi-class	ITE
[8]	Multi-class	TF-IGMimp, RTF-IGMimp
[5]	Multi-class	TF-DFS, TF-MDFS

STW scheme called TF-IDF-ICSDF, where ICSDF is the short form of inverse class space density frequency. [4] mentioned that standard TF-IDF is not fully effective in text classification. So, they proposed a new STW scheme called TF-IGM and its variant RTF-IGM. IGM stands for inverse gravity moment, and RTF is the square root of TF. Their experiment result showed that the TF-IGM and RTF-IGM outperformed the TF-IDF and the STW schemes such as TF-CHI, TF-Prob, TF-IDF-ICSDF, and TF-RF. [10] proposed an STW scheme inverse term entropy (ITE), which is applicable for multi-class classification and is computationally less expensive than the statistics-based STW schemes. [8] proposed an improved version of TF-IGM and RTF-IGM. Their experiment result showed that the proposed STW schemes TF-IGMimp and RTF-IGMimp outperformed the standard TF-IGM and RTF-IGM. A novel STW scheme called TF-DFS was proposed by [5] by adapting the distinguishing feature selector, a famous feature selection method. In the same paper, they also proposed a modified version of TF-DFS named TF-MDFS to solve the defects found in TF-DFS. The overall experiment result showed that the TF-MDFS outperformed the advanced weighting schemes.

Exam question classification based on BT is a multi-class text classification problem. Statistics-based STW schemes proposed by [7] were devised mainly for binary classification and are computationally expensive if we want to use them for multi-class classification [10]. The STW schemes TF-ICF, TF-IDF-ICF, and TF-IDF-ICSDF are the base or standard schemes for the multi-class text classification. The newly proposed term weighting schemes such as TF-IGM, RTF-IGM, ITE, TF-IGMimp, RTF-IGMimp, TF-DFS, and TF-MDFS can be used for multi-class classification problems. But this research aims to investigate whether the STW schemes are effective or not for BT-based exam question classification. For that reason, the base schemes can be used to investigate the effectiveness of the STW schemes for exam question classification based on BT instead of the most recently proposed schemes. So, this study used standard STW schemes TF-ICF [26], TF-IDF-ICF [22], and TF-IDF-ICSDF [22] in this comparative study.

Table 2: Past work on term weighting in BT-based exam question classification.

Research Work	Approach	Scheme
[18]	Rule-based	Category Weighting
[28]	ML-based	TF-IDF
[11]	Rule-based	Category Weighting
[23]	ML-based	Binary
[16]	ML-based	ETF-IDF
[17]	ML-based	TFPOS-IDF

2.3 Related Work in Exam Question Classification

Table 2 presents the works on term weighting by the previous researchers in exam question classification. In previous work, we found two approaches to classify exam questions: rule-based and ML-based. As shown in Table 2, all the works regarding the term weighting in ML-based exam classification are USTW. From Table 2, we can see that both studies [18, 11] used a rule-based approach to classify the exam questions based on BT. [18] highlighted that the BT levels have overlapping keywords. Because of the overlapping keyword problem, a question might fall into more than one cognitive level. So, they introduced category weighting to overcome the keyword overlapping problems. The weight was calculated based on the question's category from subject matter experts (SMEs). [11] used a different way of calculating the category weighting. Instead of calculating from SMEs as done by [18], they used wordnet similarity and the cosine similarity values to assign the weight for each question category. To identify the most suitable algorithms for exam question classification based on BT, [11] also compared several wordnet similarity algorithms. Path similarity was found to be the best algorithm among the wordnet similarity algorithms. Their experimental results showed that among 45 questions, 32 questions were classified correctly by using the ruleset they generated.

Some past studies [28, 19] used TF-IDF for the term weighting in ML-based exam question classification. They used the standard TF-IDF to represent the questions in vector format since the raw text cannot be fitted directly into ML algorithms for the training. After computing the TF-IDF, length normalization was applied to get the normalized weight for the terms. According to [23], TF and TF-IDF perform best in a situation where a specific term frequently occurs in a question. But in a question, multiple occurrences of terms are rare since the question contains very few words, unlike the documents. So, instead of using TF and TF-IDF, [23] used the binary term weighting scheme to weight the terms. [16] mentioned that verbs have a higher impact than any other parts of speech (POS) while determining the cognitive levels of exam questions. Other than verbs, nouns and adjectives have a higher impact than the rest of the POS. But the traditional TF-IDF does not consider POS while calculating the weight. So, they modified the traditional TF-IDF and proposed an enhanced TF-IDF weighting scheme. The Stanford tagger (version 3.9.1) was used to tag the terms of the questions and assign a weight based on POS. For the experiment, several classifiers were used such as Support Vector Machines (SVM), Naïve Bayes (NB), and K-Nearest Neighbors (KNN). The proposed method of [16] showed improvement in the exam question classification. [16] conducted another study [17] in later years. They proposed another term weighting scheme based on POS in the latter study. In addition, they attempted to address the issue of BT keyword overlap by integrating the word embedding approach Word2Vec with their proposed TFPOS-IDF. For the experiment, several classifiers were used, such as KNN, Logistic Regression (LR), and SVM. The proposed method was tested with multiple datasets. The experiment result showed that combining enhanced TFPOS-IDF and Word2Vec improved classification accuracy.

For this comparative task, both category weighting schemes shown in Table 2 are not applicable in this work since these schemes were devised for rule-based exam question classification. Term weighting schemes such as Binary, TF-IDF, ETF-IDF, and TFPOS-IDF are usable for ML-based classification. Binary term weighting is the most basic USTW scheme. So, this study implemented TF-IDF, ETF-IDF, and TFPOS-IDF to compare the results with the STW schemes.

2.4 Research Gap Analysis

All the works related to term weighting in ML-based exam question classification are USTW schemes, as shown in Table 2 and the preceding discussion. Despite the success of STW schemes in text classification, no past work tested the effectiveness of STW schemes for BT-based exam question classification. The use of STW schemes in exam question classification may reduce misclassification. So, this comparative study tested the effectiveness of STW schemes for BT-based exam question classification.

3 Research Method

Fig. 2 illustrates all the phases involved in the methodology used in this study. These phases are preprocessing, Feature extraction, classification, and model evaluation. All these phases are discussed in detail in this section.

3.1 Dataset

This study used three different datasets. All three datasets are from past research. The purpose of using multiple datasets is to investigate the effectiveness of STW schemes for questions from a single domain as well as for the multi-domain. The first dataset [23] comprises a total of 181 questions, all of which are from the business domain. The second dataset is also from the same study and consists of 415 questions. These questions were collected from several universities in

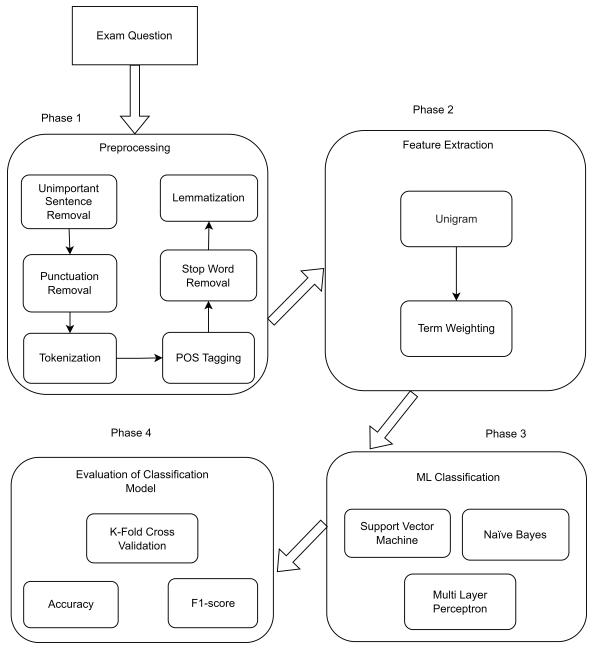

Figure 2: Phases involved in exam question classification model.

Table 3: Class distribution of all datasets.

Cognitive Level	single-domain	multi-domain-1	multi-domain-2
Knowledge	23	50	56
Comprehension	37	135	92
Application	29	72	62
Analysis	30	56	45
Synthesis	29	45	66
Evaluation	33	57	66
Total	181	415	387

Malaysia. The second dataset contains questions from various areas or fields, such as Science, Computing, Multimedia, Mathematics, Social Science, Programming, and Business. The third dataset used in this study contained 600 questions and was introduced by [29]. The authors of [29] collected these questions from the item bank accessible over the internet. Among 600 questions, many questions don't contain any BT keywords. After removing these two types of questions, 387 questions remain in this dataset. Every question of each dataset was already labeled according to the cognitive domain of BT. Table 3 shows the distributions of questions at each level of BT for all the datasets, and some sample questions from the first dataset are shown in Table 4.

3.2 Preprocessing

The raw text data cannot be directly input into the ML algorithms. So, preprocessing the data is essential. Lowercase conversion, tokenization, punctuation removal, stop word removal, lemmatization, and POS tagging are the steps involved in preprocessing. These methods are the standard way of preprocessing exam questions, which have been widely used in past studies [1, 16, 11, 19, 29]. The famous python library Natural Language Toolkit (NLTK, version 3.6.1) was used for lemmatization and Stop word removal. NLTK has been widely used in past studies [16, 17] for preprocessing exam questions. A question can contain multiple sentences. Only sentences containing the BT keywords were preprocessed. All terms in each question were transformed to lowercase, and unnecessary sentences were discarded from the dataset. Tokenization and punctuation removal were performed through regular expression. The Stanford tagger (version 4.2.0) was used to obtain the POS tagging of each term after tokenization and punctuation removal. Stanford tagger was also used in past studies [17, 24]. Following that, the stop words in the question were removed using NLTK's standard stop word list. Finally, lemmatization was used to derive the root word of each word present in the question. The WordNetLemmatizer of the NLTK was used to perform the lemmatization. The output of each process involved in preprocessing is elaborated below in Table 5 with the help of a question taken from the single-domain dataset.

Sample question used in Table 5. "Davenport (2000) discusses four emerging trends to be considered in designing future enterprise systems. Analyse each emerging trend, its relevance and importance to the designing of future enterprise systems."

3.3 Feature Extraction

Following preprocessing, a feature set needs to extract for training the classification model. In the feature extraction process, a feature set was extracted first, and after that, performed the term weighting. During model training, all the extracted features from the initial dataset were considered independent variables or features.

Unigram: Unigram is a simple feature extraction method that creates a set of all unique terms from the dataset. Several Past studies [16, 17, 24] used unigram to obtain the feature set from the questions. Other than unigram, as reported by [24], there are many more techniques to obtain the feature set, such as bigram, trigram, POS tagging, headword, and so forth. The basic feature extraction approach unigram was used since this study focused on term weighting for exam question classification. Terms with low frequencies may be significant in classifying exam questions. As a result of feature selection, we may lose those important features. So, no feature selection was performed in this study after extracting the feature set. [23] also avoided feature selection with the concern of losing important features that rarely occur in question. Other than [23], numerous past studies [16, 17, 24] on term weighting did not perform feature selection. Another reason for not performing feature selection in this study is the small dataset, since creating a large dataset of exam questions is very difficult, time-consuming, and challenging.

Table 4: Sample questions from the single-domain dataset for each cognitive level.

Cognitive Level	Sample Question
Knowledge	"State TWO (2) advantages of continuous bioreactors over batch bioreactors." "Name a major culture collection."
Comprehension	"Describe the term 'urbanization' and discuss how urbanization influences development of marketing and advertising strategies." "Construct a confusion matrix for the data and estimate the Apparent Error Rate."
Application	"Solve the following optimization problem by using the golden section search method and terminate the computation when the length of the interval." "Use Lagrange multiplier to solve the following nonlinear programming problem."
Analysis	"Discuss in detail the criteria used to assess the success or failure of a newly released transgenic crop with improved tolerance to stress." "Examine the connection and connectionless protocols using appropriate examples."
Synthesis	"Suggest a possible Java Web architecture(s) for the proposed application based on the case study." "Discuss TWO (2) web application frameworks that are suitable to be used in the proposed application."
Evaluation	"Differentiate between classification and prediction. Justify your answer using an example for each." "Based on your opinion, discuss critically, how far you agree that race relations have become unhealthy in our country."

Table 5: The output of each process involved in preprocessing.

Process	Output	Remarks
Lowercase Conversion	"davenport (2000) discusses four emerging trends to be considered in designing future enterprise systems. analyse each emerging trend, its relevance and importance to the designing of future enterprise systems."	Lowercase conversion converted all the terms present in the question to lowercase.
Removing unimportant sentences	"analyse each emerging trend,its relevance and importance to the designing of future enterprise systems."	Removed the first sentence from the question since no BT keyword is present in that sentence.
Tokenization	['analyse', 'each', 'emerging', 'trend,' , 'its', 'relevance', 'and', 'importance', 'to', 'the', 'designing', 'of', 'future', 'enterprise', 'systems.']	Question tokenized into words.
Punctuation removal	['analyse', 'each', 'emerging', 'trend' , 'its', 'relevance', 'and', 'importance', 'to', 'the', 'designing', 'of', 'future', 'enterprise', 'systems']	Removed the comma from the end of the word 'trend' and removed the full stop from 'systems.'
POS Tagging	[('analyse', VB), ('each', DT), ('emerging', VBG), ('trend', NN), ('its', PRP$), ('relevance', NN), ('and', CC), ('importance', NN), ('to', IN), ('the', DT), ('designing', NN), ('of', IN), ('future', JJ), ('enterprise', NN), ('systems', NNS)]	Where NN = Noun (singular), DT = Determiner, VB = Verb, JJ = Adjective, VBG = Verb (gerund) , PRP$ = Possessive Pronoun, CC = Coordinating Conjunction, IN = Preposition and NNS = Noun (plural).
Stop word removal	[('analyse', VB), ('emerging', VBG), ('trend', NN), ('relevance', NN), ('importance', NN), ('designing', NN), ('future', JJ), ('enterprise', NN), ('systems', NNS)]	Stop words: 'each,' 'its,' 'and,' 'to,' and 'of' removed from the question.
lemmatization	[('analyse,' VB), ('emerge,' VBG), ('trend,' NN), ('relevance,' NN), ('importance,' NN), ('design,' NN), ('future,' NN), ('enterprise,' NN), ('system,' NNS)]	The words 'emerging,' 'designing,' and 'systems' are converted to their root form 'emerge,' 'design,' and 'system,' respectively.

Term Weighting: This study implemented the six-term weighting schemes selected in the literature review. Among them, three are USTW schemes: TF-IDF [28, 19], ETF-IDF [16], TFPOS-IDF [17], and the rest of the three are STW schemes: TF-ICF [26], TF-IDF-ICF [22], TF-IDF-ICSDF [22].

USTW Schemes. This section discussed the USTW schemes used in this comparative study.

TF-IDF: TF-IDF is the acronym for the term frequency-inverse document frequency. There are a lot of variants of TF-IDF present in text classification and exam question classification, as reported by [25]. In exam question classification, the variant used by [17] is not the same as the one used by [28]. Among the variants of TF-IDF, [25] found the variant used by [17] as the most optimal one. Hence this study used this variant of TF-IDF. The formulas for TF and IDF are given in Eq. (1) and Eq. (2), respectively.

$$TF(t,d) = \frac{C(t_d)}{T_d} \tag{1}$$

Where $C(t_d)$ is the number of times t appears in document d, and T_d indicates the total number of terms in document d.

$$IDF(t) = 1 + log\left(\frac{D}{d_t}\right) \tag{2}$$

Where D is the total number of documents in the corpus, and d_t is the number of documents containing the term t.

Finally, TF.IDF (t, d) is the multiplication of the TF and IDF, as shown in Eq. (3).

$$TF - IDF(t,d) = TF(t,d).IDF(t) \tag{3}$$

ETF-IDF: [16] enhanced the traditional TF-IDF by introducing the impact factor (IF). Eq. (4) shows the formula for calculating the impact factor. The impact factor was assigned to the terms based on the POS.

$$IF(t) = \left\{ \begin{array}{c} X(t) + 3, \text{ if } t \text{ is } VB \\ X(t) + 1, \text{ if } t \text{ is } NN \text{ or } ADJ \\ 1, \text{ otherwise} \end{array} \right\} \tag{4}$$

Where X(t) is,

$$X(t) = \sqrt{\frac{1}{C}\sum_{i=1}^{C}\left(eq\,(t,c_i) - \frac{1}{C}\right)^2} \tag{5}$$

In Eq. (5), C refers to the total number of classes in the dataset. The equation $eq\ (t, c_i)$ indicates the total number of documents that exist in class c_i and contains the term t, divided over the total number of documents in the corpus.

Finally, the enhanced TF-IDF was calculated by multiplying TF-IDF with the impact factor, as shown in Eq. (6).

$$E - TFIDF(t,d) = TF - IDF(t,d).IF(t) \tag{6}$$

TFPOS-IDF: The same authors of ETF-IDF proposed this scheme in a later study [17].

$$w_{pos}(t) = \left\{ \begin{array}{c} w1, \text{ if } t \text{ is } Verb \\ w2, \text{ if } t \text{ is } Noun \text{ or } Adjective \\ w3, \text{ otherwise} \end{array} \right\} \tag{7}$$

where weight value for w1 = 5, w2 = 3, and w3 = 1. The TFPOS was calculated with Eq. (8).

$$TFPOS(t,d) = \frac{C\,(t,d) * \ w_{pos}(t)}{\sum_i C(t_i,d) * \ w_{pos}(t_i)} \tag{8}$$

Where $C\,(t,d)$ is the number of times t appears in document d and $\sum_i C(t_i,d)$ is the total number of terms in document d.

Finally, the TFPOS-IDF was calculated with Eq. (9), where the newly calculated TFPOS was multiplied by the IDF described in Eq. (2).

$$TFPOS - IDF(t,d) = TFPOS(t,d).IDF(t) \tag{9}$$

STW Schemes. This section discussed the STW schemes used in this comparative study.

TF-ICF: The full form of TF-ICF is the term frequency-inverse category frequency, proposed by [26]. The formula of TF-ICF is given in Eq. (10). The findings of [25] reported that the variant of TF used in TF-ICF was the most optimal among all the variants of TF.

$$TF - ICF(t_i,d_j) = tf(t_i,d_j) * log(1 + \frac{|C|}{cf(t_i)}) \tag{10}$$

From Eq. (10), $tf(t_i,d_j)$ is the raw TF of term t_i in document d_j. $|C|$ denotes the total number of classes in the training corpus. $cf(t_i)$ denotes the count of classes where term t_i occurs.

TF-IDF-ICF: Term frequency-inverse document frequency-inverse category frequency is the full form of TF-IDF-ICF. This scheme is proposed by [22].

$$TF - IDF = tf(t_i,d_j) * (1 + log\frac{D}{d(t_i)}) \tag{11}$$

$$TF - IDF - ICF = TF - IDF * (1 + log\frac{C}{c(t_i)}) \tag{12}$$

From Eq. (11), $tf(t_i,d_j)$ is the raw TF of term t_i in document d_j. D denotes the total number of documents in the corpus. $d(t_i)$ denotes the total number of documents containing the term t_i. C denotes the total number of classes in the training corpus. $c(t_i)$ denotes the count of classes where the term t_i occurs.

TF-IDF-ICSDF: This STW scheme is also proposed by [22]. The full form of TF-IDF-ICSDF is the term frequency-inverse document frequency-inverse class space density frequency.

$$TF - IDF - ICSDF = TF - IDF * \left(1 + log\frac{C}{\sum_{k=1}^{C}\frac{d(t_i,\ c_k)}{d(c_k)}}\right) \tag{13}$$

In Eq. (13), the TF-IDF portion of TF-IDF-ICSDF remains the same as in Eq. (11). The total number of documents belonging to class c_k is denoted as $d(c_k)$. The count of documents belongs to class c_k and contains the term t_i is denoted as $d(t_i,c_k)$.

The authors who proposed the schemes: TFPOS-IDF, TF-ICF, TF-IDF-ICF, and TF-IDF-ICSDF normalized the term weighting values before training the classification model. So, all the term weighting schemes used in this study were normalized except TF-IDF and ETF-IDF by following the authors who proposed these schemes. The normalization step was not performed for the NB classifier since the performance of NB may drop with the normalization, as observed from the outcome of [25]. The L2 normalization was used in this study to normalize the term weighting values by following the earlier studies [17, 24]. The equation that represented l2 normalization is:

$$L^2 = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{14}$$

Where x is the weight of a term in the question, and n is the total number of terms in the question.

3.4 Classification

Three ML algorithms were used in this study to train the exam question classification model. They are SVM, NB, and Multi-Layer Perceptron (MLP). Following past studies [16, 17, 23] of exam question classification, a famous Python open-source library Scikit-learn (version 1.0.1), was used in this study to train and validate the classification model.

SVM. SVM was introduced by [12] in text classification, a supervised ML algorithm. It works by learning an optimal hyperplane (also known as linear decision boundary [1]) that best divides the two sets of data from each other [24]. In exam question classification, SVM is frequently used in past studies [1, 28, 29, 27] , and is known for higher text classification accuracy [23]. Exam question classification, like text classification, contains high-dimensional feature spaces. The advantage of using SVM is it generalizes well in high-dimensional feature spaces, as mentioned by [18]. Many previous studies [16, 24] used the linear kernel of SVM and adopted it in this study.

NB. The second classifier used in this study is NB, a supervised ML classifier. NB has been used frequently in several applications, including the text classification task, and achieves state-of-the-art results [1]. The multinomial variation of NB was used in this study, which was also used in past studies [16, 24] of exam question classification.

MLP. The third classifier used in this study is MLP, also known as the feed-forward artificial neural network (ANN). MLP classifier consists of three layers: input layer, hidden layer, and output layer. More than one hidden layer can be possible in MLP, but this research used one hidden layer as used by an earlier study [30] for question classification. Regarding the number of neurons or nodes in the hidden layer, this study used the default setting of the MLP classifier available in Scikit-learn.

3.5 Evaluation

This study used two evaluation metrics, such as Accuracy and F1 score, to evaluate the performance of the classifiers. The number of correctly predicted data points out of all the data points present in the dataset is known as Accuracy. On the other hand, the F1 score is the harmonic mean of precision and recall. These evaluation metrics have been used often in previous studies [28, 29, 27] of exam question classification.

To split the dataset for training and testing the ML classifiers, this study used k-fold cross-validation. This method has been widely used in past studies [16, 19, 24, 3] of BT-based exam question classification. Regarding the k-value, different studies used different values, such as [16] used 5-fold, [19], and [3] used 10-fold. Other than these studies, [24] introduced multiple k values for cross-validation starting from 3 until 10. They computed the average value of each k-value first (refer to Eq. (15)), followed by the average of all k-values (refer to Eq. (16)) obtained to evaluate the classifiers. This study adopted the multiple k-values technique since it provides a more realistic and reliable result, as reported by [24].

$$\bar{A}_k = \frac{\sum_{i=1}^{k} A_i}{k} \tag{15}$$

Where $\bar{A}_k$ is the average Accuracy/F1 score of each k-fold value. A_i is the Accuracy/F1 score of a particular fold.

$$\overline{A} = \frac{\sum_{k=3}^{10} \overline{A}_k}{\sum_{3}^{10} 1} \tag{16}$$

Where $\overline{A}$ refers to the average Accuracy/F1 score of all k values. $\bar{A}_k$ indicates the average Accuracy/F1 score of each k-fold value.

4 Results and Discussion

This study used six different term weighting schemes, three classifiers, and three datasets for comparison. The term weighting schemes used are TF-IDF, ETF-IDF, TFPOS-IDF, TF-ICF, TF-IDF-ICF, and TF-IDF-ICSDF. The first three are USTW schemes, and the rest of the three are STW schemes. The classifiers used in this study are SVM, NB, and MLP. This study used the Scikit-learn library to train and test the classifiers. The default setting was used for NB. The linear kernel was used with the one-vs-one multi-class strategy for SVM, as exam question classification based on BT is a multi-class text classification task. As for the MLP classifier, the default setting was used except for the solver. The 'lbfgs' was used as a solver since it converged faster than the default 'adam' with the small dataset, according to the documentation of Scikit-learn [20]. This study used Accuracy and F1 score to evaluate the performance of the classifiers. Stratified cross-validation was used in this study to split the datasets for training and testing. The random state of cross-validation produces the same result across a different run. According to the Scikit-learn documentation [20], a random state can be any integer number. So, this study used 42 as a random state. Regarding the k value of the cross-validation, this study used the k-fold values ranged 3 to 10 for more dependable results.

4.1 Term Weighting Results

The term weighting values for two questions are tabulated in tables 6 and 7. These two questions were selected randomly from the single-domain dataset. The first question contains one verb, which is also a BT keyword. Apart from the BT keyword, the second question contains two non-BT verbs: 'implement' and 'reinforce.' The verb 'implement' is not a BT keyword

Table 6: Term weighting values of each term weighting scheme for a question.

Scheme type	Terms/ TW	briefly	describe (V)	general	type	factor	production
USTW	TF-IDF	0.3762	0.4132	0.5429	0.3924	0.4264	0.4927
	ETF-IDF	0.2161	0.7503	0.3635	0.2613	0.2846	0.3297
	TFPOS-IDF	0.1074	0.5898	0.4649	0.3360	0.3651	0.4219
STW	TF-ICF	0.3509	0.3509	0.6215	0.2926	0.2926	0.4428
	TF-IDF-ICF	0.3124	0.3431	0.6161	0.2945	0.3200	0.4645
	TF-IDF-ICSDF	0.2671	0.3246	0.6222	0.2872	0.3505	0.4802

Table 7: Term weighting values of each term weighting scheme for a question.

Scheme type	Terms/ TW	discuss (V)	how	implement (V)	organization	reinforce (V)	financial	stand
USTW	TF-IDF	0.2396	0.2380	0.3655	0.3225	0.4223	0.4223	0.4223
	ETF-IDF	0.3595	0.1139	0.5530	0.1786	0.6396	0.2354	0.2354
	TFPOS-IDF	0.3269	0.0649	0.4985	0.2639	0.5760	0.3456	0.3456
STW	TF-ICF	0.2337	0.2011	0.3536	0.2011	0.4964	0.4964	0.4964
	TF-IDF-ICF	0.1879	0.1712	0.3599	0.2320	0.5006	0.5006	0.5006
	TF-IDF-ICSDF	0.1360	0.1324	0.3588	0.2713	0.5038	0.5038	0.5038

here since this verb is the root form of the word 'implemented' after the lemmatization. However, the BT keyword 'describe' is present in the first question, while the BT keyword 'discuss' is present in the second.

From Table 6, we can observe that in both ETF-IDF and TFPOS-IDF, the difference in weight between the verb and the rest of the POS is higher than in TF-IDF. That is because the ETF-IDF and TFPOS-IDF assigned a higher weight to the verbs than the other POS, which is not the case for TF-IDF. The same case we can observe in Table 7 also. As for the STW schemes, overall, the difference in weight between the verb and other POS is higher with TF-ICF than with the other STW schemes. The IDF could be the reason behind reducing the weight of the verbs. Since in both TF-IDF-ICF and TF-IDF-ICSDF, IDF is present but not in TF-ICF. In TF-IDF-ICSDF, the weights of verbs even decreased compared to the TF-IDF-ICF. The decrement in verb weight could result from ICSDF since ICF was replaced with ICSDF in TF-IDF-ICSDF.

4.2 Results of SVM

This section discussed the results of SVM with all term weighting schemes and datasets used in this study. From Fig. 3, Among the three USTW schemes, overall, TFPOS-IDF outperformed ETF-IDF with a difference of 1% and 0.8% in Accuracy and F1 score, respectively. Though in multi-domain-1, ETF-IDF outperformed TFPOS-IDF with a slight difference of 0.4% and 0.7% in Accuracy and F1 score, as can be seen from Table 8. However, in the other two datasets, TFPOS-IDF outperformed ETF-IDF. Regarding the standard TF-IDF, it performed the least satisfactorily among the three USTW schemes in all three datasets. In other words, ETF-IDF and TFPOS-IDF consistently outperformed TF-IDF in all the datasets with the SVM classifier. In the single-domain dataset, the difference in performance between TF-IDF and ETF-IDF is nearly identical. The difference in performance between TF-IDF and ETF-IDF in the single-domain dataset is approximately 0.3% and 0.9% in terms of Accuracy and F1 score, respectively. On the other hand, the difference between TF-IDF and TFPOS-IDF is 1.8% and 2.5% in the Accuracy and F1 score, respectively, in the single-domain dataset. In the other two datasets, the difference is higher.

These results showed the impact of POS-based weighting in the classification process. Among the POS, verbs were given a higher weight in ETF-IDF and TFPOS-IDF compared to the other POS. The term 'describe,' which is a verb, got a higher weight in both ETF-IDF and TFPOS-IDF among the terms in the question, as shown in Table 6. As a result, it created better discrimination between the verbs and other POS. But in the case of TF-IDF, the weight for the verb 'describe' is lower than the weight of other terms such as 'general,' 'factor,' and 'production.' A similar effect can be observed in Table 7 also. So, ETF-IDF and TFPOS-IDF created better discrimination than the TF-IDF between the verb and other POS. That could be the possible reason for the better performance of ETF-IDF and TFPOS-IDF over standard TF-IDF.

Among the STW schemes, TF-ICF outperformed the other two schemes in all three datasets. TF-IDF-ICF performed very closely to TF-ICF in the single-domain dataset with an approximate difference of 1% and 1.2% in Accuracy and F1 score, respectively. But in other datasets, TF-ICF significantly outperformed TF-IDF-ICF in both Accuracy and F1 score. Both TF-ICF and TF-IDF-ICF outperformed TF-IDF-ICSDF across all datasets and evaluation metrics. The IDF is present in both TF-IDF-ICF and TF-IDF-ICSDF but not in TF-ICF. Because the only difference between TF-ICF and TF-IDF-ICF is the IDF, we may deduce that the performance of TF-IDF-ICF is lower than TF-ICF because of the IDF. Tables 6 and 7 show that overall, the difference in weight between the verb and other POS is decreased in both TF-IDF-ICF and TF-IDF-ICSDF compared to the TF-ICF. It is a clear impact of IDF, which reduced the weight of the verb,

Table 8: Results of SVM- Accuracy (Acc.) and F1 score (F1).

Dataset	USTW Schemes						STW Schemes					
	TF-IDF		ETF-IDF		TFPOS-IDF		TF-ICF		TF-IDF-ICF		TF-IDF-ICSDF	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
single-domain	0.703	0.694	0.706	0.703	0.721	0.719	0.764	0.763	0.754	0.751	0.673	0.668
multi-domain-1	0.639	0.620	0.696	0.689	0.692	0.682	0.704	0.697	0.660	0.648	0.581	0.550
multi-domain-2	0.726	0.724	0.786	0.785	0.803	0.802	0.812	0.809	0.772	0.770	0.704	0.697

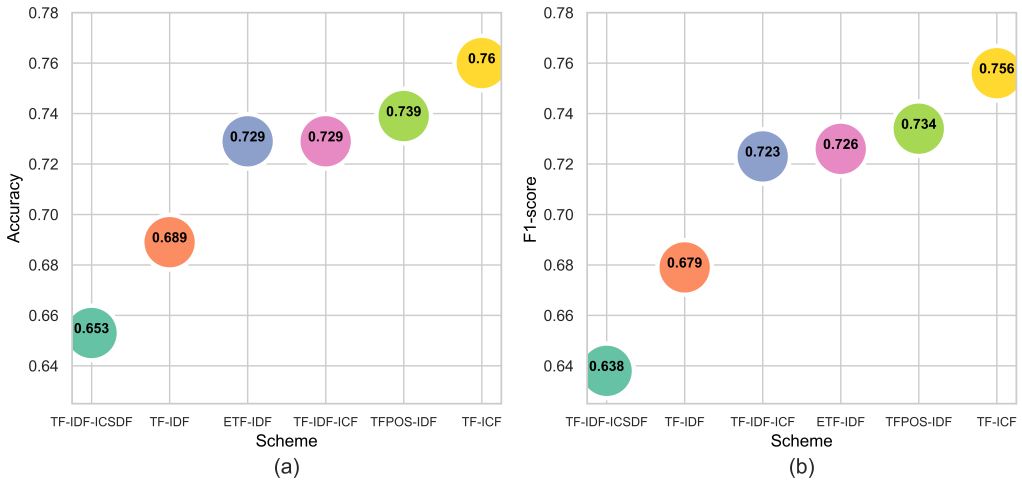

Figure 3: Average Accuracy (a) and average F1-score (b) of each term weighting scheme with SVM.

resulting in the reduction of discrimination power between the verb and other POS. The authors of [22], who proposed both TF-IDF-ICF and TF-IDF-ICSDF in text classification, reported better performance of TF-IDF-ICSDF compared to TF-IDF-ICF. However, text classification differs from exam question classification. Since, in exam question classification, BT keywords, verbs, and nouns are more important than any other terms present in the question, which is not the case in text classification [24]. If we compare the TF-IDF-ICF and TF-IDF-ICSDF, the ICF is replaced by ICSDF in TF-IDF-ICSDF. So, we can conclude that the ICSDF is the reason behind the least satisfactory result of TF-IDF-ICSDF. However, the result may differ with a large dataset since these schemes were tested with large datasets by the authors who proposed these schemes in text classification.

Comparison of the STW and USTW schemes with SVM. Fig. 3 shows the average performance of each term weighting scheme with the SVM classifier. The average performance of each scheme was computed by averaging the results of each dataset. Using SVM as a classifier, if we compare both STW and USTW schemes, the STW scheme TF-ICF outperformed all the other schemes used in this comparative study with an average Accuracy of 0.760 and an average F1 score of 0.756. Overall, USTW schemes ETF-IDF and TFPOS-IDF outperformed the STW scheme TF-IDF-ICF. On the other hand, the STW scheme TF-IDF-ICSDF performed the least satisfactorily among all the term weighting schemes used in this study. With the SVM, the consistent superiority of TF-ICF was found over the USTW schemes in this study, even though the higher weight was assigned to the verbs in ETF-IDF and TFPOS-IDF. The USTW scheme ETF-IDF and TFPOS-IDF contained IDF. When the IDF combined with TF-ICF, it reduced the performance, which we have seen in the case of TF-IDF-ICF. Thus, IDF could be a factor in the less satisfactory results of ETF-IDF and TFPOS-IDF. So, we can conclude that despite the higher weight assigned to the verbs in ETF-IDF and TFPOS-IDF, still, TF-ICF outperformed both ETF-IDF and TFPOS-IDF with the SVM classifier.

4.3 Results of NB

This section discussed the results of NB with all term weighting schemes and datasets used in this study. Overall, ETF-IDF outperformed TFPOS-IDF and standard TF-IDF among the USTW schemes with the NB, as shown in Fig. 4. Although in the single-domain dataset, TFPOS-IDF performed slightly better than ETF-IDF, as demonstrated in Table 9. However, in other datasets, ETF-IDF performed significantly better than TFPOS-IDF. The performance of ETF-IDF in terms of Accuracy and F1 score improved dramatically in multi-domain datasets when using the NB classifier. If we compare ETF-IDF with TFPOS-IDF, the result showed an approximately 15% and 18% increase in multi-domain-1 in Accuracy and F1 score, respectively. In multi-domain-2, the increment is approximately 3.7% in Accuracy and 3.5% in F1 score. Thus, it can be concluded that the larger and multi-

Table 9: Results of NB- Accuracy (Acc.) and F1 score (F1).

Dataset	USTW Schemes						STW Schemes					
	TF-IDF		ETF-IDF		TFPOS-IDF		TF-ICF		TF-IDF-ICF		TF-IDF-ICSDF	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
single-domain	0.684	0.675	0.699	0.692	0.706	0.702	0.712	0.698	0.653	0.643	0.549	0.541
multi-domain-1	0.503	0.452	0.705	0.700	0.559	0.525	0.667	0.648	0.624	0.626	0.528	0.525
multi-domain-2	0.691	0.686	0.796	0.793	0.759	0.758	0.760	0.758	0.675	0.673	0.524	0.520

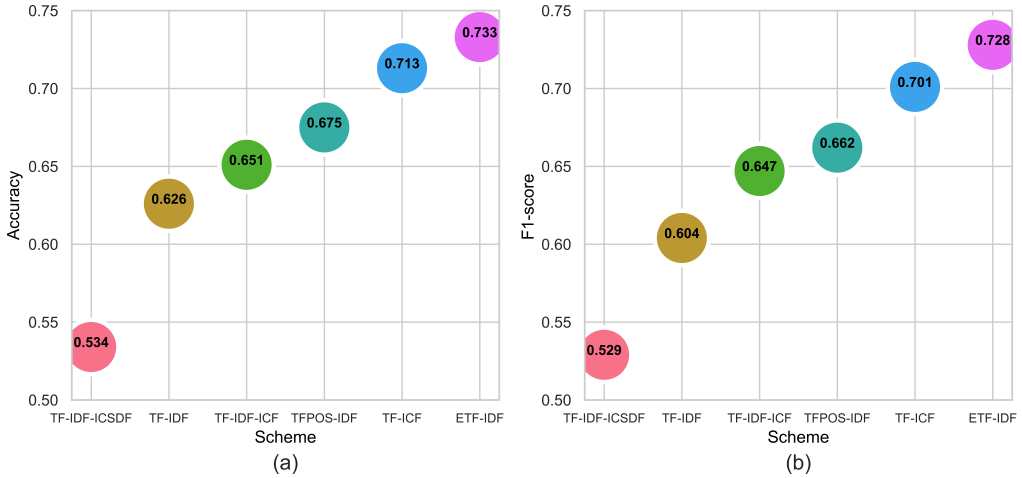

Figure 4: Average Accuracy (a) and average F1-score (b) of each term weighting scheme with NB.

domain dataset favored the ETF-IDF with the NB classifier. [24] reported that the type of classifier impacts the pattern of term weighting schemes, which we observed in the case of ETF-IDF and NB with the larger datasets. Another notable finding is that both ETF-IDF and TFPOS-IDF outperformed standard TF-IDF in all the datasets, which demonstrated the impact of POS-based weighting, where verbs received higher weight.

The results of NB with STW schemes are demonstrated in Table 9 and Fig. 4. We can observe from the figure that, like the SVM, TF-ICF outperformed all other STW schemes with the NB. The results are consistent with the SVM since IDF reduced the performance when it was included with TF-ICF and formed TF-IDF-ICF. From Table 9, The difference between TF-ICF and TF-IDF-ICF is approximately 6% in the single-domain dataset in both metrics, whereas 4.3% and 2.2% in the multi-domain-1 in Accuracy and F1 score, respectively. In multi-domain-2, the difference in performance is more significant, 8.5% in both evaluation metrics. The TF-IDF-ICSDF performed the least satisfactorily in the table, which was also the case with the SVM classifier.

Comparison of the STW and USTW schemes with NB. Fig. 4 shows the average performance of each term weighting scheme with the NB classifier. The average performance of each scheme was computed by averaging the results of each dataset. The ETF-IDF outperformed all the term weighting schemes with the NB classifier, achieving 0.733 and 0.728 in Accuracy and F1 score, respectively. The TF-ICF outperformed all the term weighting schemes in Accuracy and TFPOS-IDF in the F1 score in the single-domain dataset. In the other datasets, ETF-IDF outperformed all the schemes. The TF-ICF outperformed ETF-IDF with only 1.3% and 0.6% in terms of Accuracy and F1 score in the single-domain dataset. In the other two datasets, the ETF-IDF outperformed TF-ICF with a significant difference. In multi-domain-1, the difference between ETF-IDF and TF-ICF is approximately 4% and 5% in Accuracy and F1 score, respectively. In multi-domain-2, the difference is approximately 3.5% in both evaluation metrics. This result showed that the multi-domain datasets favored the ETF-IDF with the NB classifier. Overall, TF-ICF outperformed TFPOS-IDF with the NB classifier. As with SVM, the STW scheme TF-IDF-ICSDF performed the least satisfactorily with NB among all the term weighting schemes used in this study.

4.4 Results of MLP

This section discussed the results of the MLP classifier with all term weighting schemes and datasets used in this study. In all datasets, ETF-IDF outperformed the standard TF-IDF and TFPOS-IDF, as shown in Table 10 of the MLP classifier experiment results. However, the difference in performance is nearly identical in terms of Accuracy and F1 score between ETF-IDF and TFPOS-IDF in the single-domain dataset. In single-domain, the difference in performance between ETF-IDF and TFPOS-IDF is approximately 0.9% and 1.4% in the Accuracy and F1 score, respectively. In multi-

Table 10: Results of MLP - Accuracy (Acc.) and F1 score (F1)

Dataset	USTW Schemes						STW Schemes					
	TF-IDF		ETF-IDF		TFPOS-IDF		TF-ICF		TF-IDF-ICF		TF-IDF-ICSDF	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
single-domain	0.686	0.678	0.705	0.698	0.696	0.684	0.759	0.752	0.715	0.710	0.643	0.638
multi-domain-1	0.695	0.691	0.719	0.718	0.706	0.702	0.745	0.740	0.717	0.712	0.651	0.647
multi-domain-2	0.758	0.756	0.800	0.799	0.769	0.766	0.815	0.811	0.789	0.785	0.712	0.707

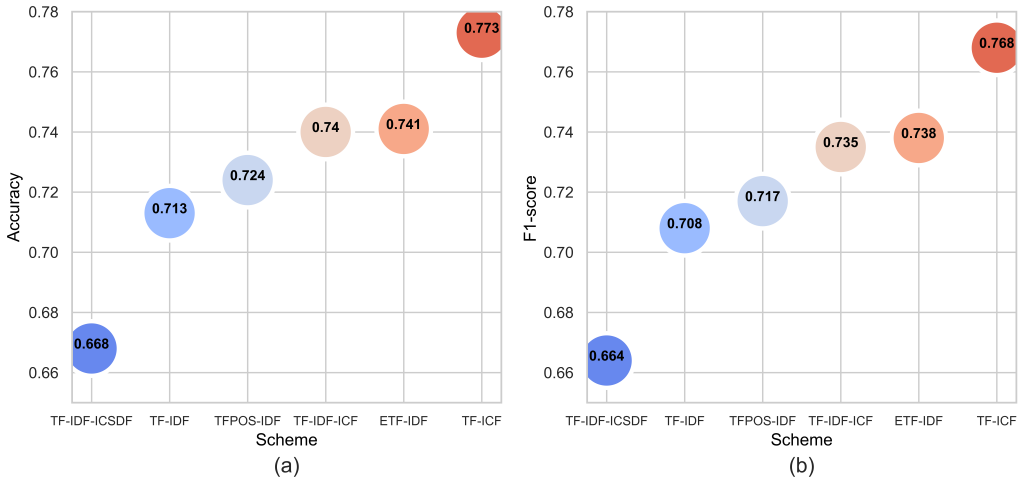

Figure 5: Average Accuracy (a) and average F1-score (b) of each term weighting scheme with MLP.

domain-1, the difference is 1.3% in Accuracy and 1.6% in the F1 score. The difference increased in multi-domain-2, approximately 3% in both evaluation metrics. Another outcome is that overall, TFPOS-IDF and ETF-IDF outperformed TF-IDF with the MLP classifier, as shown in Fig. 5. This outcome is consistent with the SVM and NB, where ETF-IDF and TFPOS-IDF outperformed TF-IDF. Like SVM, overall, the MLP classifier showed very consistent results for USTW schemes.

With the MLP classifier, TF-ICF outperformed other STW schemes in all datasets, which we already observed in SVM and NB. The difference in performance between TF-ICF and TF-IDF-ICF is approximately 4% in both evaluation metrics in the single-domain dataset. In multi-domain-1, the difference in performance is around 3% in both evaluation metrics and 2.5% in multi-domain-2. Another STW scheme TF-IDF-ICSDF performed the least satisfactorily. This outcome is also consistent with the results of SVM and NB classifiers. Overall, the MLP classifier showed very consistent results for STW schemes.

Comparison of the STW and USTW schemes with MLP. Fig. 5 shows the average performance of each term weighting scheme with the MLP classifier. The average performance of each scheme was computed by averaging the results of each dataset. Overall, with the MLP classifiers, the TF-ICF outperformed all the schemes, which we have already seen in the case of the SVM classifier. Overall, TF-ICF achieved an average Accuracy of 0.773 and an average F1 score of 0.768 with the MLP classifier. In all the datasets, the TF-ICF outperformed all other term weighting schemes in both Accuracy and F1 score. However, in multi-domain-2, the difference in performance between TF-ICF and ETF-IDF is very identical, 1.5% and 1.2% in Accuracy and F1 score, respectively. In the other two datasets, the difference in performance between TF-ICF and ETF-IDF is significant. Not only ETF-IDF, TF-ICF significantly outperformed another USTW scheme, TFPOS-IDF, in all the datasets. The STW scheme TF-IDF-ICSDF performed the least satisfactorily with the MLP classifier among all the term weighting schemes. This outcome is consistent with the results of the SVM and NB classifier.

4.5 Impact of the dataset on the performance

Table 11 shows the average result of each dataset for the USTW and STW schemes. The performance of each dataset for STW and USTW schemes was obtained by calculating the average performance of the classifiers in each dataset for both types of schemes. Table 11 shows that the single-domain dataset achieved higher Accuracy and F1 score than the multi-domain dataset multi-domain-1. However, another multi-domain dataset, multi-domain-2, achieved the highest Accuracy and F1 score among all three datasets. Among the three datasets, multi-domain-1 has the lowest percentage of verbs, and multi-domain-2 has the highest. The single-domain dataset has a higher percentage of verbs than multi-domain-1 but is lower than multi-domain-2. The percentage of verbs present in the dataset might be the reason for the superior results

Table 11: Average performance achieved with each Dataset – Accuracy (Acc.) and F1 score (F1).

Dataset	USTW schemes		STW schemes	
	Acc.	F1	Acc.	F1
single-domain	0.701	0.694	0.691	0.685
multi-domain-1	0.657	0.642	0.653	0.644
multi-domain-2	0.765	0.763	0.729	0.726

of multi-domain-2 and the least satisfactory results of multi-domain-1.

4.6 Summary of the Results

Fig. 6 shows the average performance of each term weighting scheme used in this study. Each scheme's average performance was computed by averaging the results of all classifiers and datasets used in this study. Dataset multi-domain-1 reported lower Accuracy and F1 score compared to other datasets for all the classifiers. However, multi-domain-2, a multi-domain dataset, delivered the best performance out of the three datasets. If we compare the classifiers used in this study, SVM and MLP are consistent overall, whereas NB is inconsistent. From Fig. 6, overall, the ETF-IDF outperformed TFPOS-IDF and TF-IDF among the USTW schemes. However, TFPOS-IDF outperformed ETF-IDF in some instances. Both the ETF-IDF and TFPOS-IDF significantly outperformed the standard TF-IDF. This outcome is consistent with the findings reported by [16] and [17], where these studies reported consistently better results for ETF-IDF and TFPOS-IDF, respectively, compared to the standard TF-IDF. As a result, this finding supports the previous study's [24] suggestion to use the ETF-IDF and TFPOS-IDF as baseline schemes instead of TF-IDF to compare the results for exam question classification. Regarding the STW schemes, TF-ICF outperformed TF-IDF-ICF and TF-IDF-ICSDF in all the datasets and the classifiers. If we compare the performance of all the schemes used in this study, Fig. 6 shows that the TF-ICF outperformed all with an average of 0.748 and 0.742 in Accuracy and F1 score, respectively. Another outcome is that the USTW schemes ETF-IDF and TFPOS-IDF outperformed the STW scheme TF-IDF-ICF. TF-IDF-ICSDF, an STW scheme, performed the least satisfactory of all the weighting schemes used in this study, with an average Accuracy of 0.618 and an average F1 score of 0.610. We may conclude from this discussion that overall, the STW scheme TF-ICF outperformed all the term weighting schemes used in this study.

4.7 Statistical test

This study performed the two-sample t-test to investigate whether the performance of a term weighting scheme is statistically different or not from another. A few past studies [17, 25] of exam question classification

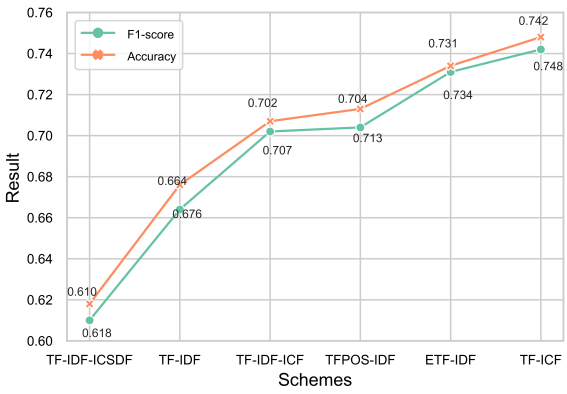

Figure 6: Average results of each term weighting scheme.

used the two-sample t-test to investigate the statistical difference between the performance of term weighting schemes. This statistical test was performed for some of the notable findings of this study. One of the findings of this study is that ETF-IDF and TFPOS-IDF outperformed the standard TF-IDF. Hence, the t-test was performed between ETF-IDF and TF-IDF to investigate whether the difference in performance between these two USTW schemes is statistically significant. The subsequent t-test was conducted between two USTW systems, TFPOS-IDF and TF-IDF. This study also found that the STW scheme TF-ICF outperformed the USTW schemes ETF-IDF and TFPOS-IDF. So, the t-test was conducted between TF-ICF and ETF-IDF and between TF-ICF and TFPOS-IDF.

This study used three datasets, three classifiers, and two evaluation metrics, so for each of the comparisons mentioned earlier, a total of 18 tests were conducted. Null and alternative hypotheses were set for each hypothesis test. The null hypothesis is that the mean difference between the schemes' performance is zero, whereas the alternative hypothesis is that the mean difference between the schemes' performance is not zero. The alpha value was set to 0.05 in this hypothesis test. The difference is statistically significant if the p-value is smaller than the alpha value; else, it is statistically insignificant. The null hypothesis is rejected if the result is statistically significant, otherwise retained. These tests are two-tailed because we are not concerned with a specific direction and investigating whether the performance is significantly different.

As per the comparison between USTW schemes from Table 12, the difference in performance between the ETF-IDF and TF-IDF is not statistically significant with the single-domain dataset, except with the MLP classifier. Regarding the other datasets, the difference in performance is statistically significant with all the classifiers. The difference in performance between the TFPOS-IDF and TF-IDF is also statistically insignificant with the single-domain dataset except with the NB classifier in F1 score, as shown in Table 13. However, in other datasets, the difference in performance is statistically significant except for the two cases:

Table 12: Statistical test result of ETF-IDF vs. TF-IDF

Dataset	classifier	P-value		Decision	
		Accuracy	F1 score	Accuracy	F1 score
single-domain	SVM	0.833439	0.520945	x [c]	x
single-domain	NB	0.208852	0.092941	x	x
single-domain	MLP	0.016632	0.005424	✓[a]	✓
multi-domain-1	SVM	<0.000001	<0.000001	✓	✓
multi-domain-1	NB	<0.000001	<0.000001	✓	✓
multi-domain-1	MLP	0.012627	0.005431	✓	✓
multi-domain-2	SVM	0.000016	0.000008	✓	✓
multi-domain-2	NB	<0.000001	<0.000001	✓	✓
multi-domain-2	MLP	0.000080	0.000053	✓	✓

[a] statistically significant
[c] statistically insignificant

Table 13: Statistical test result of TFPOS-IDF vs. TF-IDF

Dataset	classifier	P-value		Decision	
		Accuracy	F1 score	Accuracy	F1 score
single-domain	SVM	0.252930	0.078153	x [c]	x
single-domain	NB	0.111442	0.025819	x	✓[a]
single-domain	MLP	0.248097	0.402414	x	x
multi-domain-1	SVM	<0.000001	<0.000001	✓	✓
multi-domain-1	NB	0.000040	0.000005	✓	✓
multi-domain-1	MLP	0.256367	0.236227	x	x
multi-domain-2	SVM	<0.000001	<0.000001	✓	✓
multi-domain-2	NB	0.000004	0.000002	✓	✓
multi-domain-2	MLP	0.215473	0.241175	x	x

[a] statistically significant
[c] statistically insignificant

Table 14: Statistical test result of TF-ICF vs. ETF-IDF

Dataset	classifier	P-value		Decision	
		Accuracy	F1 score	Accuracy	F1 score
single-domain	SVM	0.000143	0.000020	✓[a]	✓
single-domain	NB	0.092215	0.421947	x [c]	x
single-domain	MLP	0.000010	0.000001	✓	✓
multi-domain-1	SVM	0.269070	0.259336	x	x
multi-domain-1	NB	0.001138	0.000101	✓	✓
multi-domain-1	MLP	0.008875	0.018961	✓	✓
multi-domain-2	SVM	0.004426	0.003453	✓	✓
multi-domain-2	NB	0.000085	0.000029	✓	✓
multi-domain-2	MLP	0.038331	0.051881	✓	x

[a] statistically significant
[c] statistically insignificant

multi-domain-1, and multi-domain-2, with the MLP classifier. These results showed that overall, ETF-IDF and TFPOS-IDF statistically significantly outperformed the standard TF-IDF, except with the single-domain dataset. The smaller size of the single-domain dataset could be the reason for this insignificant result.

Table 14 shows the statistical test results between the TF-ICF and ETF-IDF. From Table 14, we can see that the difference in performance in the single-domain dataset is statistically significant with the SVM and MLP classifiers. The results are statistically significant in other datasets except for multi-domain-1 with SVM in both metrics and multi-domain-2 with MLP in the F1 score. Regarding the statistical test results of TF-ICF vs. TFPOS-IDF, we can see in Table 15 that the difference in performance is statistically significant in the single-domain dataset with the SVM and MLP classifiers. The test results are not statistically significant with NB in the single-domain dataset. In multi-domain-1, the results are statistically significant with two classifiers: NB and MLP, whereas statistically insignificant with the SVM in Accuracy. About multi-

Table 15: Statistical test result of TF-ICF vs. TFPOS-IDF

Dataset	classifier	P-value		Decision	
		Accuracy	F1 score	Accuracy	F1 score
single-domain	SVM	0.001458	0.000139	✓[a]	✓
single-domain	NB	0.558082	0.639104	x [c]	x
single-domain	MLP	0.000010	0.000001	✓	✓
multi-domain-1	SVM	0.084398	0.034068	x	✓
multi-domain-1	NB	<0.000001	<0.000001	✓	✓
multi-domain-1	MLP	0.001020	0.001105	✓	✓
multi-domain-2	SVM	0.183014	0.225432	x	x
multi-domain-2	NB	0.976847	0.962520	x	x
multi-domain-2	MLP	0.000031	0.000021	✓	✓

[a] statistically significant
[c] statistically insignificant

domain-2, only the results of the MLP classifier are statistically significant for TF-ICF vs. TFPOS-IDF. We can conclude from this discussion that TF-ICF outperformed ETF-IDF and TFPOS-IDF statistically significantly in many cases. However, according to statistical test results, this study could not find the consistent superiority of TF-ICF over the ETF-IDF and TFPOS-IDF.

5 Conclusion

This study analyzed STW and USTW schemes for exam question classification based on BT. For a better comparison, this study used three classifiers and three datasets. As for the term weighting, this research used three STW and three USTW schemes. The SVM and MLP classifiers showed better consistent results than the NB. Overall, multi-domain-1, a multi-domain dataset, reported lower Accuracy and F1 score compared to the single-domain dataset. However, another multi-domain dataset, multi-domain-2, reported better results than the single-domain dataset. This outcome indicated that the STW approach is effective for single and multi-domain datasets. Among the STW schemes, TF-ICF significantly outperformed the other STW schemes, TF-IDF-ICF and TF-IDF-ICSDF. Regarding the USTW schemes, TFPOS-IDF and ETF-IDF significantly outperformed standard TF-IDF. This result reflected the impact of POS-based weighting, in which the verb was given a higher weight in both the ETF-IDF and the TFPOS-IDF. As a result of this finding, we can conclude that POS-based weighting is significant in exam question classification to improve classification accuracy. The USTW scheme ETF-IDF outperformed TFPOS-IDF in most of the scenarios. However, the STW scheme TF-ICF outperformed all the term weighting schemes used in this study. This finding suggests that while term weighting, both POS-based weighting and document distribution by class category are significant for exam question classification. As a result, combining POS-based weighting with document distribution by class category may improve exam question classification performance. In the future, we may work on hybridization of both the STW and USTW schemes to improve the BT-based exam question classification accuracy.

Acknowledgement: This work was supported by The UTAR Research Fund (UTARRF) under research grant number: IPSR/RMC/UTARRF/2020-C2/A01.

References

[1] Abduljabbar, D. A., and Omar, N. Exam questions classification based on bloom's taxonomy cognitive level using classifiers combination. *Journal of Theoretical and Applied Information Technology 78*, 3 (2015), 447.

[2] Alsaeedi, A. A survey of term weighting schemes for text classification. *International Journal of Data Mining, Modelling and Management 12*, 2 (2020), 237–254.

[3] Aninditya, A., Hasibuan, M. A., and Sutoyo, E. 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)* (2019), IEEE, pp. 112–117.

[4] Chen, K., Zhang, Z., Long, J., and Zhang, H. Turning from tf-idf to tf-igm for term weighting in text classification. *Expert Systems with Applications 66* (2016), 245–260.

[5] Chen, L., Jiang, L., and Li, C. Modified dfs-based term weighting scheme for text classification. *Expert Systems with Applications 168* (2021), 114438.

[6] Chen, L., Jiang, L., and Li, C. Using modified term frequency to improve term weighting for text classification. *Engineering Applications of Artificial Intelligence 101* (2021), 104215.

[7] Debole, F., and Sebastiani, F. Supervised term weighting for automated text categorization. In *Proceedings of the 2003 ACM symposium on Applied computing* (2003), pp. 784–788.

[8] Dogan, T., and Uysal, A. K. Improved inverse gravity moment term weighting for text classification. *Expert Systems with Applications 130* (2019), 45–59.

[9] Domeniconi, G., Moro, G., Pasolini, R., and Sartori, C. A study on term weighting for text categorization: A novel supervised variant of tf. idf. In *DATA* (2015), pp. 26–37.

[10] Gu, Y., and Gu, X. A supervised term weighting scheme for multi-class text categorization. In *International Conference on Intelligent Computing* (2017), Springer, pp. 436–447.

[11] Jayakodi, K., Bandara, M., Perera, I., and Meedeniya, D. Wordnet and cosine similarity based classifier of exam questions using bloom's taxonomy. *International Journal of Emerging Technologies in Learning 11*, 4 (2016).

[12] Joachims, T. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning* (1998), Springer, pp. 137–142.

[13] Kavadi, D. P., Ravikumar, P., and Srinivasa Rao, K. A new supervised term weight measure for text classification. *International Journal of Advanced Science and Technology 29*, 6 (2020), 3115–3128.

[14] Lan, M., Tan, C. L., Su, J., and Lu, Y. Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31*, 4 (2008), 721–735.

[15] Mazyad, A., Teytaud, F., and Fonlupt, C. A comparative study on term weighting schemes for text classification. In *International Workshop on Machine Learning, Optimization, and Big Data* (2017), Springer, pp. 100–108.

[16] Mohammed, M., and Omar, N. Question classification based on bloom's taxonomy using enhanced tf-idf. *International Journal on Advanced Science, Engineering and Information Technology 8* (2018), 1679–1685.

[17] Mohammed, M., and Omar, N. Question classification based on bloom's taxonomy cognitive domain using modified tf-idf and word2vec. *PLOS ONE 15*, 3 (2020), e0230442.

[18] Omar, N., Haris, S. S., Hassan, R., Arshad, H., Rahmat, M., Zainal, N. F. A., and Zulkifli, R. Automated analysis of exam questions according to bloom's taxonomy. *Procedia-Social and Behavioral Sciences 59* (2012), 297–303.

[19] Osman, A., and Yahya, A. Classifications of exam questions using linguistically-motivated features: a case study based on bloom's taxonomy. In *The Sixth International Arab Conference on Quality Assurance in Higher Education (IACQA'2016)* (2016), vol. 467, p. 474.

[20] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[21] Quan, X., Wenyin, L., and Qiu, B. Term weighting schemes for question categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence 33*, 5 (2010), 1009–1021.

[22] Ren, F., and Sohrab, M. G. Class-indexing-based term weighting for automatic text classification. *Information Sciences 236* (2013), 109–125.

[23] Sangodiah, A., Ahmad, R., and WAN AHMAD, W. F. Taxonomy based features in question classification using support vector machine. *Journal of Theoretical & Applied Information Technology 95*, 12 (2017).

[24] Sangodiah, A., Fui, Y. T., Heng, L. E., Jalil, N. A., Ayyasamy, R. K., and Meian, K. H. A comparative analysis on term weighting in exam question classification. In *2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* (2021), IEEE, pp. 199–206.

[25] Sangodiah, A., San, T. J., Fui, Y. T., Heng, L. E., Ayyasamy, R. K., and Jalil, N. A. Identifying optimal baseline variant of unsupervised term weighting in question classification based on bloom taxonomy. *MENDEL 28*, 1 (2022), 8–22.

[26] Wang, D., and Zhang, H. Inverse-category-frequency based supervised term weighting scheme for text categorization. *Journal of Information Science and Engineering 29* (2013), 209–225.

[27] Yahya, A. A., and Osman, A. Automatic classification of questions into bloom's cognitive levels using support vector machines. In *The International Arab Conference on Information Technology* (2011).

[28] Yahya, A. A., Osman, A., Taleb, A., and Alattab, A. A. Analyzing the cognitive level of classroom questions using machine learning techniques. *Procedia-Social and Behavioral Sciences 97* (2013), 587–595.

[29] Yahya, A. A., Toukal, Z., and Osman, A. Bloom's taxonomy–based classification for item bank questions using support vector machines. In *Modern Advances in Intelligent Systems and Tools*. Springer, 2012, pp. 135–140.

[30] Yusof, N., and Hui, C. J. Determination of bloom's cognitive level of question items using artificial neural network. In *2010 10th International Conference on Intelligent Systems Design and Applications* (2010), IEEE, pp. 866–870.