Improving an AI-Based Algorithm to Automatically Generate Concept Maps

Sara Alomari[1] & Salha Abdullah[1]

[1] Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

Correspondence: Sara Alomari, Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.

Abstract

Concept maps have been used to assist learners as an effective learning method in identifying relationships between information, especially when teaching materials have many topics or concepts. However, making a manual concept map is a long and tedious task. It is time-consuming and demands an intensive effort in reading the full content and reasoning the relationships among concepts. Due to this inefficiency, many studies are carried out to develop intelligent algorithms using several data mining techniques. In this research, the authors aim at improving Text Analysis-Association Rules Mining (TA-ARM) algorithm using the weighted K-nearest neighbors (KNN) algorithm instead of the traditional KNN. The weighted KNN is expected to optimize the classification accuracy, which will, eventually, enhance the quality of the generated concept map.

Keywords: data mining, concept map, weighted KNN, TA-ARM

1. Introduction

Concept maps have been used to assist learners as one of the effective learning methods in identifying relationships between information. More precisely, when teaching materials have many topics or concepts, it would be easier for learners to study from an organized reference in order to receive organized knowledge. Moreover, it assists educators by showing the structure between concepts and how they are related. Concept maps can provide them with guidance on how they should explain the information due to the complexities. As a result, they can ensure that learners can interpret their explanation correctly and construct accurate knowledge hierarchies in their minds (Atapattu, Falkner, & Falkner, 2017).

However, constructing a manual concept map is deemed to be a tedious task. It is time-consuming, and it demands an extensive effort in reading the full content and reasoning the relationships among concepts. This effort presents the academics with a heavy workload. They may abandon the benefits of concepts maps when comparing them to the effort they should put at generating one concept map. With this regard, the researchers have developed many algorithms to attain the automation of generating concept maps from a given text using data mining methods and Natural Language Processing (NLP) (Shao, Li, Wang, Zhao, & Guo, 2018). The generated concept maps have been evaluated against manually generated ones by experts and by using different accuracy measurements.

Shao et al. (2018) had generated the concept maps using an algorithm called TA-ARM. TA-ARM uses the traditional KNN algorithm to classify the dataset into concepts, used the Apriori principle to find the associations and generate the concept map. In this research, the authors aim to improve the TA-ARM algorithm by using the weighted KNN instead of traditional KNN to optimize the classification accuracy and thus, generate quality concept maps.

2. Research Terminology

2.1 Concept Maps

Concept maps were first introduced by Joe Novak in 1972 during his research at Cornell University to assess the enhancement in the learners' understanding of Science (Cañas & Carvalho, 2004). The work solved the problem of identifying the change of understanding level of learners by asking them to draw a concept map in different periods of times. It observed how the learners linked the concepts at each time to evaluate their understanding of

the progress.

A concept map is a graphical scheme composed of a set of nodes; each represents a concept and labelled lines to define the relations between a pair of nodes (Svanström, Sjöblom, Segalàs, & Fröling, 2018). It represents knowledge in a hierarchical structure where the most inclusive concepts are at the top, and then they are divided into more detailed concepts (Marqués & Pelta, 2017). Molinari (2017) defined concept maps as a knowledge representation technique that graphically visualizes the relationships among different concepts which helps learners to structure their knowledge. Thus, it enables them to identify their strengths and weaknesses and makes them aware of the gaps in their knowledge.

According to Novak & Cañas (2006), the word 'concept' represents an expected regularity in objects or events. The concept also records objects or events that are designed by the label. Concepts are linked by connecting them by lines or edges, which enable the learner to read a meaningful statement, as shown in Figure 1. The major components of concept maps are concepts (nodes), relations (linking lines), and labels that describe the relations between the connected concepts. The composition of the three components makes up a meaningful statement or a proposition (Taie, 2014).
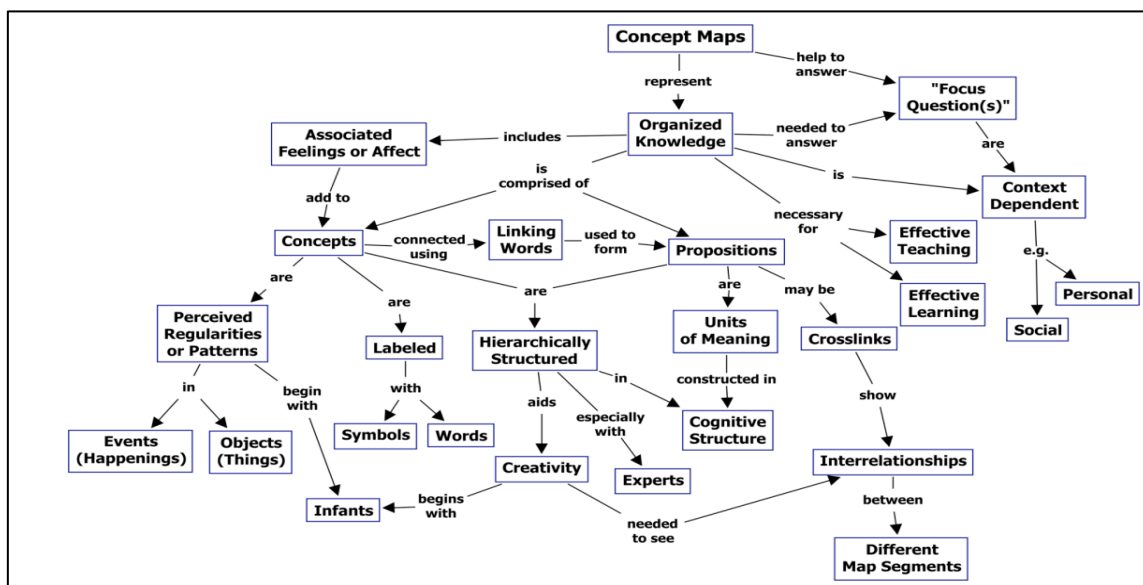

Figure 1. A concept map by Joseph Novak

The construction of concept maps differs according to the situation in which the knowledge domain is used. Therefore, it is recommended to create a set of questions for the concept maps. As stated by Novak & Cañas (2006) and Falke (2019), these questions are known as *focus questions*; they are intended to address what the learners are trying to figure out correctly. Therefore, the concept map will be appertained to a certain context by answering these questions in an organized presentation of knowledge.

Initially, concept maps were created by Novak to help the educators to assess the learners' understanding level's progress (Cañas & Carvalho, 2004). Nowadays, they have been widely used for teaching and learning objectives by enabling educators to provide organized knowledge to learners that prepare them for success (Roessger, Daley, & Hafez, 2018). The strength of concept maps is that they can be used for learning purposes and can have permanent records for the learners over the years, specifically when the learners themselves construct them. In this way, they can observe the changes in their understanding level of a particular topic over time (Sundberg, 2002).

On the other hand, the process of constructing concept maps is considered to be a tedious task as a reasonable amount of time and effort are consumed in analyzing the different associations among the concepts to generate a concept map of good quality (Khrais & Saleh, 2017). Due to the limitations, some researches such as Atapattu et al. (2017), Shao et al. (2018), Huang, Yang, & Lawrence (2015), Shao, Li, Wang, Zhao, & Guo (2018), Zubrinic, Kalpic, & Milicevic (2012) and Chen & Sue (2013), have proposed intelligent tools and algorithms to automate this process and generate accurate maps in comparatively lesser time and efforts.

2.2 Classification

In classification, the dataset is split into training examples and testing examples. The rules of the classification are generated by the training examples to be applied to the testing ones (Kuhkan, 2016). It is a supervised learning task where the classifier learns from the training examples to predict the class of testing examples. This learning process continues until the classifier achieves an acceptable level of performance. The testing item's attributes will be examined by the classifier to consequently assign the testing item to one of the predefined classes (Chen & Liu, 2004). The most popular classification techniques are decision-trees (Islam, 2010), Naive Bayes (Patil & Sherekar, 2013), and KNN algorithms (Deng, Cheng, & Zhang, 2016).

2.2.1 KNN Algorithm

KNN algorithm is a non-parametric algorithm used in the data mining applications. This algorithm is used to classify different objects according to the closest K training examples in the dataset, where 'K' is a small positive integer that represents the number of neighbors. Let us say, if K=1, then the testing example in testing data will be allotted to the class of single nearest neighbor. As stated by Kaghyan & Sarukhanyan (2012), the KNN algorithm is a category of learning in which the generalization is considered beyond the training data delay until or unless a query is made. However, it must be noted that it is the simplest classification algorithm in which little to none-prior knowledge is needed for it to work (Imandoust & Bolandraftar, 2013).

KNN algorithm identifies the K nearest neighbor by calculating the distance between the testing example and all the training examples to recognize the similarities between the examples. Then, the testing example will be assigned to the class that includes the nearest neighbor, which has the shortest distance. However, if the value of K was more than 1, and with different classes, then the testing example will be assigned to the majority class (Mulak & Talhar, 2015). Figure 2 illustrates the KNN algorithm workflow.
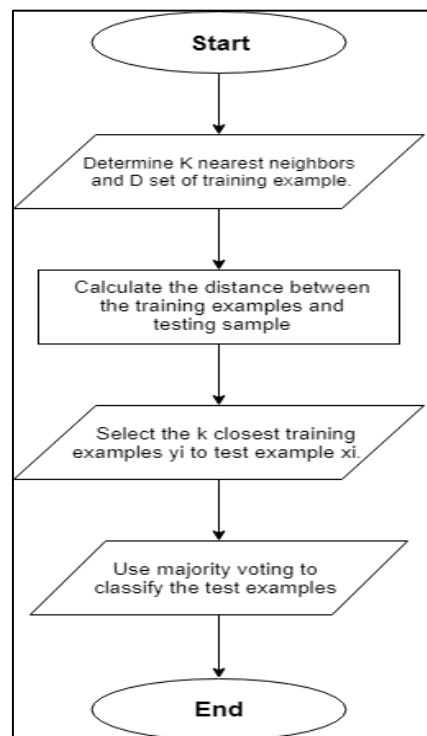

Figure 2. KNN algorithm workflow

2.2.2 Weighted KNN Algorithm

Using distance functions to measure the similarity prompts the KNN algorithm to evaluate all the features equally regardless of their actual importance. This, in turn, may result in less accurate classifications, because the model can be easily affected by the value of K. If the K value were too small, the model would be outlier sensitive. On the other hand, if the K value was too large, then the neighbors include too many examples from other classes. Therefore, assigning the testing example to the class of the majority vote may not be reliable

anymore since the examples of the major class may be more distant than other examples to the testing point (Weighted KNN, n.d).

Such issues can be avoided by considering the actual importance of features while classifying and prediction the KNN algorithm and the weights of these features (Dietterich, 1994). In order to improve the accuracy of KNN regardless of the impact of K value, it has been proposed to add weights to the features used in distance calculation (Dietterich, 1994). These weights help to achieve better accuracy since they help in discarding the impact of the least important features on the classification process (Kuhkan, 2016). They can be calculated for each K nearest neighbor as the inverse of its distance from the testing example to be classified.

3. Literature Review

Huang et al. (2015) have addressed the concept map's accuracy and complexity in order to generate more accurate maps that better support the learning process. The dataset has come from test questions and students' answers. Apriori principle (Yuan, 2017) is used as a tool of association rules mining and considered all the potential possibilities of the question. Then, they constructed a set of association rules and computed their confidence. They omitted the rules that have less than the minimum confidence. Finally, they examined the relationship between questions and concepts by calculating the relevance degree. For the evaluation, they built a simulation model and used an actual dataset for students' test records. They used four test questions from four concepts of algebra subject that varied in the difficulty level. C1 had the lowest difficulty, while C4 had the highest. Then, they obtained the answers from five students to define the relationships among the questions. Based on relevance degree results, they proved that their algorithm could generate robust concept maps.

Atapattu et al. (2017) have proposed an algorithm that generates concept maps from the lecture slides. The aim was to support learners and facilitate studying for CS courses by providing them with concept maps. Atapattu et al. (2017) have employed the concept of NLP that enables the extraction of concept-relation-concept triple to construct a concept map. These triples are ranked based on their significance and the layout of the lecture slides. The slides were collected as PPT files written in English. The algorithm was implemented in Java language. With the application of NLP, the algorithm extracted the concepts exist in all the input lecture slides. In addition, they also received academics' assistance who extracted a different number of concepts. This is because of their different judgment on how important a concept is. Then, they utilized the layout of slides to define the relationships among the extracted concepts and ranked them according to their importance. Finally, they visually represented the results by generating an XML-based file. The evaluation was done by the academics' judgment that judged the generated map based on multiple factors that are accuracy, coverage, and suitability. The results were promising, and the algorithm was recommended to be used in many applications and knowledge representation fields.

Shao et al. (2018) have proposed Text Clustering and Association Rules Mining (TC-ARM) algorithm to generate concept maps. TC-ARM goes through two stages: test questions clustering and the generation of association rules among the generated clusters. Firstly, the test questions were clustered, where each cluster consists of test questions with similar text features, and these clusters are abstracted to a set of concepts. Secondly, the association rules were generated among the test questions clustered in the first stage using the students' answers record, which will map the questions according to the concepts' relevance. The dataset consists of authentic 82 test questions and answers of 298,070 students for different computer science courses at the college level. They implemented the algorithm using C# language, and the result has shown the ability of TC-ARM of generating highly accurate concept maps without any intervention of experts.

Chen & Sue (2013) have proposed an algorithm for the automatic construction of concepts maps to be used for adaptive learning systems. First, it computes the consistency between any two questions using students' answers. Then, it generates association rules using the Apriori principle according to the consistency value and finally finds the relevance degree between two concepts in order to construct the concept map. The proposed algorithm has proved its effectiveness at generating concept and also outperformed some previous algorithms used data mining for the same purpose.

Shao et al. (2018) have proposed Text Analysis and Association Rules Mining (TA-ARM) algorithm to generate concept maps. The algorithm classifies test questions into concepts through the use of traditional KNN algorithm. It then uses the student answers to establish associations among the concepts using the Apriori principle. Finally, the established association rules are used to form concept maps. For the experiment, they used 3001 test questions as the training sample, 90 test questions as the testing sample, and 617,940 answers records. The questions cover nine concepts from the Computer Culture Foundation book studied by undergraduate students at a college in China. The algorithm implemented by Python and the experiment showed that TA-ARM is a robust

generator of highly accurate concept maps according to human judgment results and the execution time. Figure 3 summarizes the algorithm of TA-ARM.
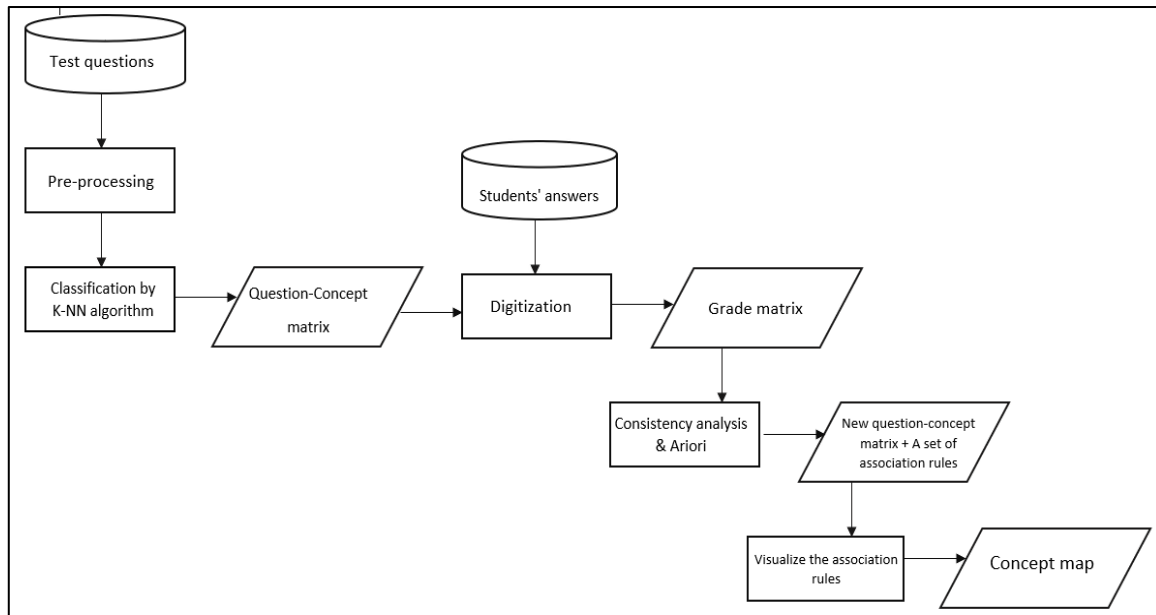

Figure 3. TA-ARM algorithm to generate concept maps

Zubrinic et al. (2012) have proposed an algorithm to generate concept maps out of documents to facilitate the process of summarization. They alleged that concept maps could significantly help readers to navigate smoothly through documents and find the part of interest faster. This is because of the nature of concept maps that can view a large amount of information in a concise and highly readable shape. However, to use concept maps for such purpose, it would be required to create a concept map for each document needed to be summarized, that in turn will cost time and effort especially for a large set of documents.

Valerio et al. (2012) have developed an algorithm based on text mining and natural language processing techniques to generate concept maps from human subjects' documents. The algorithm identifies the concepts throughout the document and then finds the associations among them depending on the syntactic structure of sentences. The algorithm was evaluated by comparing the effect of studying the subject from the automatically generated concept map, the manually created concept map, and the subject's raw materials. The results have shown that studying for the automatic concept map has positively affected the speed of answering questions compared to the other two methods. Nevertheless, it retained the same level or less of the answers' correctness.

Lee, Park, & Yoon (2015) built an approach based on the words' bursts and the associations among these bursts. They aimed to facilitate the learners' understanding and comprehension of subjects' materials. The approach detects the words that are topical and finds the associations among them. It was less sensitive to the highly frequented words, which assure the inclusion of more meaningful associations in the concept map. The approach has been implemented in Java programming language and evaluated by comparing its concept map to a human-made concept map. The results have shown that the proposed approach performed better as told by the recall measure.

Aguiar, Cury, & Zouaq (2016) have proposed an algorithm to generate concept maps to facilitate the learners' understanding of subjects' materials. The algorithm proceeds through 12 steps starting with the preparation and ending by the ranking and summarization. It was implemented in Java programming language. It used academic text, particularly the introduction part of a research paper, as a dataset. The algorithm has yielded acceptable results regarding the identification of the concept map's major elements.

A simple algorithm to generate concept maps was suggested by Kumar (2018). First, it identifies the stop words and removes them to keep only the keywords. Second, it identifies the relations among the keywords using the P.O.S technique that determines the verbs. As a final step, the algorithm pairs the keywords identified in step 1, which represent the concepts and then calculate the correlation among the pairs to construct the concept map. It

was implemented in Java, and the results have shown a few errors. However, Kumar (2018) claimed that the algorithm is promising but needs to be improved to enhance accuracy.

4. Research Scope

As evident in the previous sections, many algorithms for the automation of concept maps generation have been proposed. In this research, the effectiveness of the KNN classification model used in TA-ARM algorithm, which was proposed initially by Shao et al. (2018), will be evaluated on small-sized data. This research will improve the classification by using weighted KNN instead of traditional KNN.

Accordingly, the dataset of this research will be multiple choice and open-ended test questions taken from the faculty level. The test questions in Shao et al. (2018) were taken from the final examinations of Computer Culture Foundation book. Notably 3001 questions were taken as the training sample and 90 questions as the testing sample. The questions cover nine concepts. However, in this work, the researcher will work on 107 test questions taken from final examinations of E-Commerce course that comprise four concepts.

The weighted KNN will be implemented in Python since it has built-in libraries for the preprocessing techniques and machine learning algorithms. Then, the classification performance of weighted KNN will be evaluated in terms of the same metrics used in Shao et al. (2018), which are accuracy, precision, recall, and F1-score. Figure 4 illustrates the improvement scope of this research.
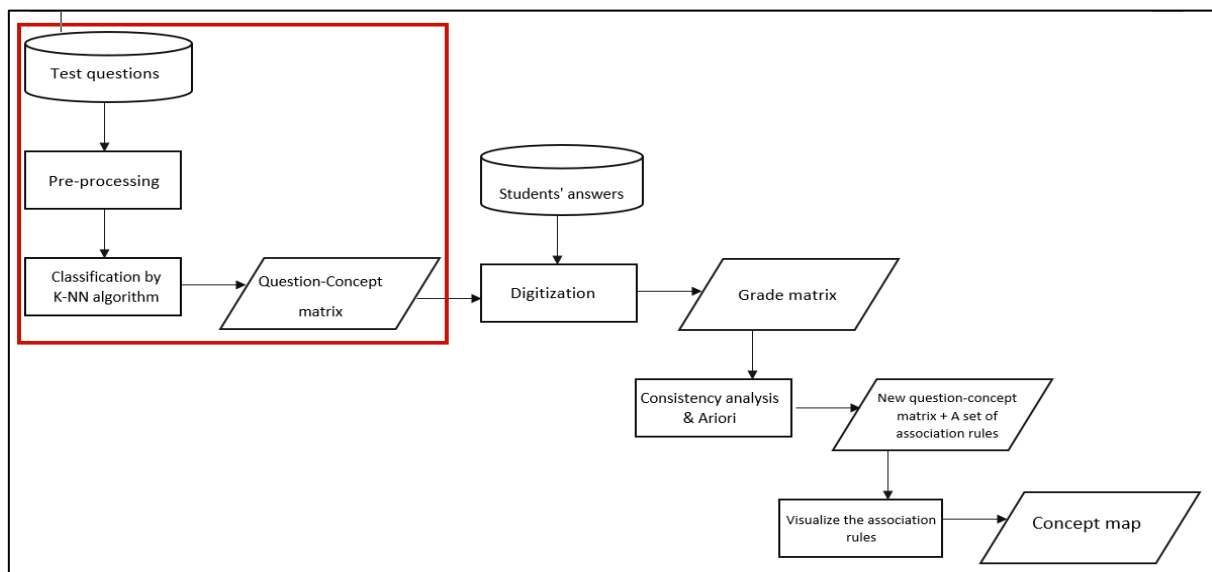

Figure 4. The research scope represented in the red box

5. The Proposed Algorithm

In this research, the author is working to improve the classification results in the TA-ARM algorithm by using weighted KNN instead of the traditional KNN. Thus, the classification approach, as shown in Figure 5, will proceed through a sequence of steps that are described in the subsequent sections.

5.1 Text Pre-Processing

This step will be performed on the test questions before being classified into concepts, as shown in Figure 5. The goal of this step is to transform the unstructured text of test questions into numbers which the KNN algorithm better understands. It is worth mentioning that this is not the only goal of text preprocessing. A preprocessed data means a reduction of the file size since the stop words represent about 20-30% of the whole text file. The stemming, as well, can reduce the file size and leads to better data indexing. Another goal of preprocessing is to enable the data mining technique to achieve better performance and improves its efficiency (Vijayarani, Ilamathi, Nithya, 2015).

There are many different preprocessing methods in data mining, and they have a direct and significant impact on data mining applications. Preprocessing is primarily composed of the following (Vijayarani et al., 2015):

- **Tokenization:** It represents the process of breaking up a stream of strings into keywords, phrases, symbols, or other meaning elements called tokens. The goal of this method is to disclose the essential keywords within the sentence in the text file.

- **Stop words elimination:** Stops words should be eliminated from the text data file since they occupy a lot of the file size while they have the least importance. Pronouns, prepositions, and articles are common examples of stop words. The classic way to eliminate the stop words is to remove them based on a pre-compiled list that includes the most common stop words used in that text language.

- **Stemming:** The process of identifying the root (stem) of the words while considering that different words HAVING the same meaning should be matched to the same stem. As a result, the number of words will be reduced, which can save more time and memory space.
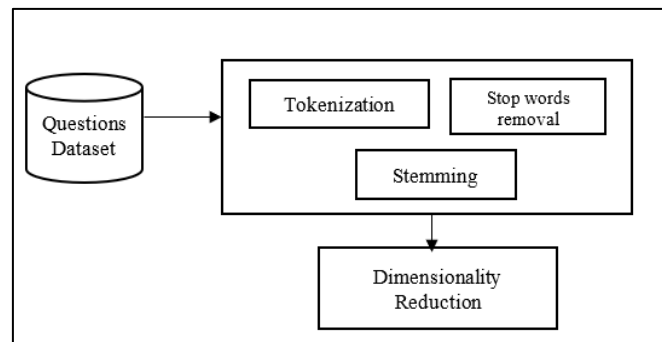

Figure 5. Dataset pre-processing steps

5.2 Text Features Extraction

This step digitizes the preprocessed textual data so they can be understood by the computer (Shao et al., 2018). In other words, it quantifies the textual data into features to be used in statistical models and ideally for the training model. The followings are the different techniques to extract features from text data.

- **Bag of words:** It is considered as the simplest technique to extract features. It simply tokenizes the text and counts the frequency of every token. The set of tokens essentially forms the bag. In every single document in a given corpus, it counts every instance of every token. This produces a big matrix where the columns represent the tokens, and the rows represent the documents. The values in this matrix would be 0's and integer numbers that represent the number of times that a given token (word) appears in that document (Waykole & Thakare, 2018).

- **TF-IDF:** It is a very common technique in the field of natural language processing (NLP) (Trstenjak, Mikac, & Donko, 2014). It extracts the features of a text 'W' by determining the relevance of the word in the entire document. Mathematically, 'W' of a given word is the frequency of the word in a particular document multiplied by the inverse of the frequency of the same word in the entire document corpus. Intuitively, words with high TF-IDF value are more common and thus, have higher importance (Ramos, 2003).

- **Word2vec:** It is a two-layer neural network that takes a text corpus as input to produce word embeddings. It usually shows better results because unlike the previous techniques, it considers the context where words with similar contexts have similar embeddings or word vectors (Waykole &Thakare, 2018).

- **GloVe:** Global Vector is for words representation. Both word2vec and GloVe are the same and use the same methodology. However, GloVe is trained on a larger corpus of words, and it is more common than word2vec (Waykole & Thakare, 2018).

In summary, each technique extracts the features according to a different aspect. The Bag of words weighs the words according to the frequency, while TF-IDF measures the relevance based on the word frequency. The word2vec considers the context and semantics. The choice between these techniques does not depend on the dataset size. Word2ve is used usually with deep learning models, for instance, Convolution Neural Network (CNN), Long Short-Term Memory (LSTM) or Recurrent Neural Networks (RNN). TF-IDF and Bag of words are better with the traditional models such as KNN and Support Vector Machine (SVM) (Ved, 2018).

5.3 Test Questions Classification

After extracting the text features, test questions will be classified by the weighted KNN algorithm into concepts. First, it will be split into training and testing samples. Secondly, the training sample will be classified into pre-determined concepts to train the weighted KNN algorithm. It will classify the testing sample of test questions into a set of the pre-determined concepts. A matrix of questions-concepts will be produced where 0 means that the question does not belong to the concept, and 1 means vice versa. At this point, the classification algorithm has accomplished the intended purpose.

6. Weighted KNN Implementation

The data source is the final exam questions of E-Commerce course collected from the Faculty of Computing and Information Technology at King Abdulaziz University. The dataset size is 107 of MCQ and open-ended questions that belong to four different concepts as shown in Table 1.

Table 1. Concepts of MCQ and open-ended questions

Concept label	Actual content
1	E-commerce applications
2	E-commerce support services
3	E-commerce business strategy
4	E-marketplace models and structures

The weighted KNN algorithm is implemented three times. Each with a different feature extraction technique to determine which one is the most suitable for the dataset. First, the dataset was preprocessed by applying tokenization, stop words elimination and stemming processes. Then, text features were extracted by the three techniques: BoW, TF-IDF and GloVe. Then, the dataset was split into 70% as the training sample and 30% as the testing sample. K value is set to 5 as the default value of the toolkit. Finally, the traditional KNN and the weighted KNN algorithms are developed to classify the text features into the four concepts. All the implementation results are represented in Figures 6-8.

$$QC = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 6. Questions-Concept Matrix (BoW)

$$QC = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Figure 7. Questions-Concept Matrix (TF-IDF)

$$QC = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Figure 8. Question-Concept Matrix (GloVe)

7. Weighted KNN Evaluation

Based on the evaluation method performed by the studies of Taneja et. al (2014), Sun et. al (2010) and Tan (2018) that proposed improvements on the traditional KNN, the evaluation of the weighted KNN algorithm will be achieved by comparing its performance to the traditional KNN that is used in TA-ARM algorithm. For that, the evaluation metrics will be the same metrics used to evaluate the classification performance in the TA-ARM, which are accuracy, precision, recall, and F1-score.

7.1 Evaluation Results

The results in Figures 9-11 show that weighted KNN algorithm has achieved better results in all metrics using all feature extraction techniques. However, the best classification performance resulted from using TF-IDF and the weighted KNN algorithm where accuracy was 72%, precision, recall and F1-score were 71%.
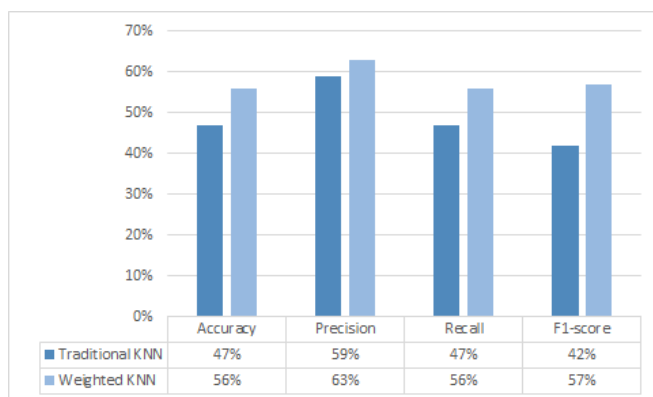
	Accuracy	Precision	Recall	F1-score
Traditional KNN	47%	59%	47%	42%
Weighted KNN	56%	63%	56%	57%

Figure 9. Evaluation results (BoW)

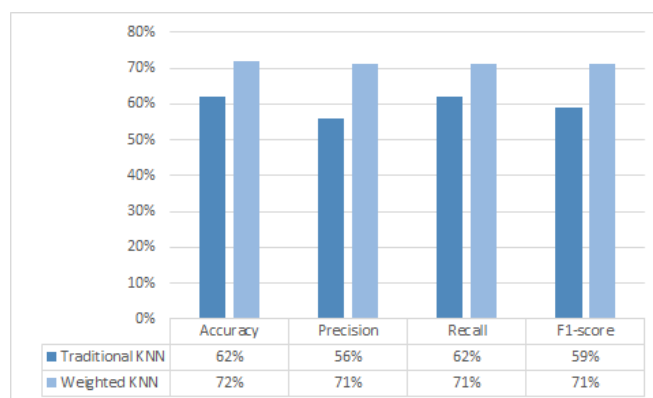|	Accuracy	Precision	Recall	F1-score
Traditional KNN	62%	56%	62%	59%
Weighted KNN	72%	71%	71%	71%

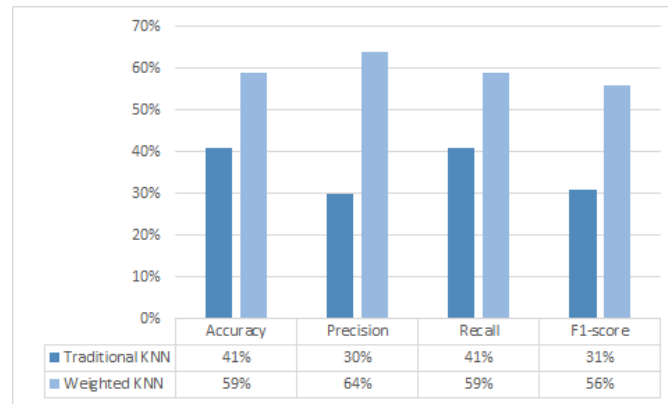Figure 10. Evaluation results (TF-IDF)

Figure 11. Evaluation results (Glove)

8. Conclusion and Future Work

Concept maps have become one of the important knowledge representation's tools that prove their effectiveness as a learning method. They facilitate the study process for the learners, especially when the subject materials are loaded with different information about different topics that are related to each other. Concept maps help learners to spot the relationships and associations between different information and topics. Concept maps though have many benefits; they still have some shortcomings. Making concepts maps is a non-trivial process. It requires extra effort from a specialized person who has knowledge in the field. Therefore, many different algorithms have been proposed to automate this process and relieve the need for human intervention.

This paper proposes different ways of enhancing one of the algorithms which is TA-ARM. TA-ARM can generate concept maps automatically using the KNN algorithm and Apriori principle. The authors implemented the classification using the weighted KNN algorithm instead of the traditional KNN to optimizing the classification accuracy. The evaluation results showed that the weighted KNN algorithm outperformed the traditional KNN used in TA-ARM in terms of the accuracy, F1 score, precision, and recall.

This research and its applications have promising future works. Future work involves creating better concept maps via Naïve Bayes algorithm instead of KNN in TA-ARM algorithm. Moreover, this technique can be applied to a big dataset to measure the algorithm's ability to handle it. The future works should also involve experts during the evaluation process as the human judgment evaluation factor.

References

Aguiar, C. Z., Cury, D., & Zouaq, A. (2016). Automatic Construction of Concept Maps from Texts. In *Conference on Concept Mapping-CMC. Tallinn.* Retrieved from http://cmc.ihmc.us/cmc2016papers/cmc2016-p90.pdf

Atapattu, T., Falkner, K., & Falkner, N. (2017). A comprehensive text analysis of lecture slides to generate concept maps. *Computers & Education*, *115*, 96-113. https://doi.org/10.1016/j.compedu.2017.08.001

Cañas, A. J., & Carvalho, M. (2004, November). Concept Maps and AI: an Unlikely Marriage? In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 1, No. 1, pp. 1-10).

Chen, S. Y., & Liu, X. (2004). The contribution of data mining to information science. *Journal of Information Science*, *30*(6), 550-558. https://doi.org/10.1177/0165551504047928

Chen, S., & Sue, P. (2013). Constructing concept maps for adaptive learning systems based on data mining techniques. *Expert Systems with Applications*, *40*(7), 2746-2755. https://doi.org/10.1016/j.eswa.2012.11.018

Deng, Z., Zhu, X., Cheng, D., Zong, M., & Zhang, S. (2016). Efficient kNN classification algorithm for big data. *Neurocomputing*, *195*, 143-148. https://doi.org/10.1016/j.neucom.2015.08.112

Falke, T. (2019). *Automatic Structured Text Summarization with Concept Maps* (Doctoral dissertation, Technische Universität).

Huang, X., Yang, K., & Lawrence, V. B. (2015, July). An efficient data mining approach to concept map generation for adaptive learning. In *Industrial conference on data mining* (pp. 247-260). Springer, Cham. https://doi.org/10.1007/978-3-319-20910-4_18

Imandoust, S. B., & Bolandraftar, M. (2013). Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background. *International Journal of Engineering Research and Applications*, *3*(5), 605-610.

Islam, M. Z. (2010, June). EXPLORE: a novel decision tree classification algorithm. In *British National Conference on Databases* (pp. 55-71). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-25704-9_7

Kaghyan, S., & Sarukhanyan, H. (2012). Activity recognition using k-nearest neighbor algorithm on smartphone with tri-axial accelerometer. *International Journal of Informatics Models and Analysis (IJIMA), ITHEA International Scientific Society, Bulgaria*, *1*, 146-156.

Khrais, H., & Saleh, A. (2017). The Outcomes of Integrating Concept Mapping in Nursing Education: An Integrative Review. *Open Journal of Nursing*, *7*(11), 1335-1347. https://doi.org/10.4236/ojn.2017.711096.

Kuhkan, M. (2016). A method to improve the accuracy of k-nearest neighbor algorithm. *International Journal of Computer Engineering and Information Technology*, *8*(6), 90-95.

Kumar, R. (2018). Automation of Concept Map Generation using Correlation and Stop Word Identification. *International Journal of Advances in Computer and Electronics Engineering*, *3*(6), 1-5.

Lee, S., Park, Y., & Yoon, W. C. (2015). Burst analysis for automatic concept map creation with a single document. *Expert Systems with Applications*, *42*(22), 8817-8829. https://doi.org/10.1016/j.eswa.2015.07.034

Marqués, J., & Pelta, C. (2017). Concept maps and simulations in a computer system for learning Psychology. *European Journal of Education and Psychology*, *10*(1), 33-39. https://doi.org/10.1016/j.ejeps.2016.07.002

Molinari, G. (2017). From learners' concept maps of their similar or complementary prior knowledge to collaborative concept map: Dual eye-tracking and concept map analyses. *Psychologie Française*, *62*(3), 293-311. https://doi.org/10.1016/j.psfr.2015.11.001

Mulak, P., & Talhar, N. (2015). Analysis of Distance Measures Using K-Nearest Neighbor Algorithm on KDD Dataset. *International Journal of Science and Research*, *4*(4), 2101-2104.

Novak, J. D., & Cañas, A. J. (2006). The theory underlying concept maps and how to construct them. *Florida Institute for Human and Machine Cognition*, *1*, 2006-2001. https://doi.org/10.1.1.100.8995

Patil, T. R., & Sherekar, S. S. (2013). Performance analysis of Naive Bayes and J48 classification algorithm for data classification. *International journal of computer science and applications*, *6*(2), 256-261.

Ramos, J. (2003, December). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning* (Vol. 242, pp. 133-142).

Roessger, K. M., Daley, B. J., & Hafez, D. A. (2018). Effects of teaching concept mapping using practice, feedback, and relational framing. *Learning and Instruction*, *54*, 11-21. https://doi.org/0.1016/j.learninstruc.2018.01.011

Shao, Z., Li, Y., Wang, X., Zhao, X., & Guo, Y. (2018). Research on a new automatic generation algorithm of concept map based on text analysis and association rules mining. *Journal of Ambient Intelligence and Humanized Computing,* 1-13. https://doi.org/10.1007/s12652-018-0934-9

Shao, Z., Li, Y., Wang, X., Zhao, X., & Guo, Y. (2018, August). Research on a New Automatic Generation Algorithm of Concept Map Based on Text Clustering and Association Rules Mining. In *International Conference on Intelligent Computing* (pp. 479-490). Springer, Cham. https://doi.org/10.1007/978-3-319-95930-6_44

Sun, B., Du, J., & Gao, T. (2009, November). Study on the improvement of K-nearest-neighbor algorithm. In *2009 International Conference on Artificial Intelligence and Computational Intelligence* (Vol. 4, pp. 390-393). IEEE. https://doi.org/10.1109/AICI.2009.312

Sundberg, M. D. (2002). Assessing Student Learning. *Cell Biology Education*, *1*(1), 11-15. https://doi.org/10.1187/cbe.02-03-0007

Svanström, M., Sjöblom, J., Segalàs, J., & Fröling, M. (2018). Improving engineering education for sustainable development using concept maps and multivariate data analysis. *Journal of Cleaner Production*, *198,* 530-540. https://doi.org/10.1016/j.jclepro.2018.07.064

Taie, E. S. (2014). Concept Mapping as an Innovative Teaching Strategy to Enhance Cognitive Learning in Nursing Administration Course. *International Journal for Innovation Education and Research*, *2*(7), 12-25.

Tan, Y. (2018, August). An improved KNN text classification algorithm based on K-medoids and rough set. In *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)* (Vol. 1, pp. 109-113). IEEE. https://doi.org/10.1109/ihmsc.2018.00032

Taneja, S., Gupta, C., Goyal, K., & Gureja, D. (2014, February). An enhanced k-nearest neighbor algorithm using information gain and clustering. In *2014 Fourth International Conference on Advanced Computing & Communication Technologies* (pp. 325-329). IEEE.https://doi.org/10.1109/ACCT.2014.22

Trstenjak, B., Mikac, S., & Donko, D. (2014). KNN with TF-IDF based framework for text categorization. *Procedia Engineering*, *69*, 1356-1364. https://doi.org/10.1016/j.proeng.2014.03.129

Valerio, A., Leake, D., & Cañas, A. (2012). USING AUTOMATICALLY GENERATED CONCEPT MAPS FOR DOCUMENT UNDERSTANDING: A HUMAN SUBJECTS EXPERIMENT. In *Proc. of the Fifth Int. Conference on Concept Mapping*.

Ved, M. (2018). Feature Selection and Feature Extraction in Machine Learning: An Overview. *Medium*, Retrieved September 22, 2019, from https://medium.com/@mehulved1503/feature-selection-and-feature-extraction-in-machine-learning-an-over view-57891c595e96

Vijayarani, S., Ilamathi, M. J., & Nithya, M. (2015). Preprocessing Techniques for Text Mining - An Overview. *International Journal of Computer Science & Communication Networks*, *5*(1), 7-16. https://doi.org/10.5121/ijcga.2015.5105

Waykole, R., & Thakare, A. (2018). A Review of Feature Extraction Methods for Text Classification. *International Journal of Advance Engineering and Research Development*, *5*(4), 351-354.

Weighted KNN. (n.d). In *GeeksforGeeks*. Retrieved September 22, 2019, from https://www.geeksforgeeks.org/weighted-k-nn/

Wettschereck, D. (1994). A study of distance-based machine learning algorithms.

Yuan, X. (2017). An improved Apriori principle for mining association rules. In *AIP conference proceedings* (Vol. 1820, No. 1, p. 080005). AIP Publishing. https://doi.org/10.1063/1.4977361

Zubrinic, K., Kalpic, D., & Milicevic, M. (2012). The automatic creation of concept maps from documents written using morphologically rich languages. *Expert Systems with Applications*, *39*(16), 12709-12718. https://doi.org/10.1016/j.eswa.2012.04.065

Copyrights