Journal of King Saud University – Computer and Information Sciences 31 (2019) 35–51

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Swarm intelligence-based approach for educational data classification

Anwar Ali Yahya *

Faculty of Computer Science and Information Systems, Thamar University, Thamar, Yemen
Faculty of Computer Science and Information Systems, Najran University, Najran, Saudi Arabia

A R T I C L E I N F O

Article history:
Received 4 June 2017
Revised 1 August 2017
Accepted 15 August 2017
Available online 24 August 2017

Keywords:
Particle swarm classification
Rocchio Algorithm
Educational data mining
Questions classification
Bloom's taxonomy

A B S T R A C T

This paper explores the effectiveness of Particle Swarm Classification (PSC) for a classification task in the field of educational data mining. More specifically, it proposes PSC to design a classification model capable of classifying questions into the six cognitive levels of Bloom's taxonomy. To this end, this paper proposes a novel specialized initialization mechanism based on Rocchio Algorithm (RA) to mitigate the adverse effects of the curse of dimensionality on the PSC performance. Furthermore, in the design of the RA-based PSC model of questions classification, several feature selection approaches are investigated. In doing so, a dataset of teachers' classroom questions was collected, annotated manually with Bloom's cognitive levels, and transformed into a vector space representation. Using this dataset, several experiments are conducted, and the results show a poor performance of the standard PSC due to the curse of dimensionality. However, when the proposed RA-based initialization mechanism is used, a significant improvement in the average performance, from 0.243 to 0.663, is obtained. In addition, the results indicate that the feature selection approaches play a role in the performance of the RA-based PSC (average performance ranges from 0.535 to 0.708). Finally, a comparison between the performance of RA-based PSC (average performance = 0.663) and seven machine learning approaches (best average performance = 0.646) confirms the effectiveness of the proposed RA-based PSC approach.

1. Introduction

Particle Swarm Optimization (PSO) is a swarm-based optimization approach inspired by social behaviour of bird flocks and fish schools (Poli et al., 2007). Simply, PSO maintains a swarm of particles, where each particle is defined by a position in a search space that represents a potential solution to the optimization problem at hand. In PSO process, each particle flies across the search space and adjusts its position in favour of better one by combining some aspects of the history of its own current and best position with those of one or more members of the swarm. Over a number of iterations, the swarm as a whole, like a flock of birds collectively foraging for food, moves closer to the optimum position (Engelbrecht, 2006). Conventionally, PSO has been applied successfully to function optimization problems, however, in the recent years a remarkable growth in its application to problems in other areas has been reported. Data mining is one of those areas, where PSO is being applied to problems such as clustering, classification, feature selection, and outlier detection (Grosan et al., 2006; Martens et al., 2011). For classification, PSO has only recently gained increasing interest through a specific PSO variant, called Particle Swarm Classification (Nouaouria and Boukadoum, 2010; Nouaouria et al., 2013), though its first application to classification dates back to 2004 (Sousa et al., 2004). The cumulative evidence since then suggests that PSC is a suitable and competitive technique, which can be applied efficiently to demanding data classification problems, especially when accurate, yet comprehensible classifiers are required (Abraham et al., 2007).

Educational Data Mining (EDM) is an emerging data mining field that focuses on the development of methods for exploring unique types of educational data arising from an educational system or process (Romero and Ventura, 2013). In EDM, as well as other data mining fields, classification is a major task which appears in different contexts, and to which different techniques have been applied. A thorough review of EDM classification works, reported in five key EDM surveys (Baker and Yacef, 2009; Peña, 2014; Pena et al., 2009; Romero and Ventura, 2007, 2010) reveals

* Address at: Faculty of Computer Science and Information Systems, Najran University, Najran, Saudi Arabia.
E-mail address: aaesmail@nu.edu.sa

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2017.08.002
1319-1578/

that a particular offshoot of PSO specialized for data classification, called PSC, has not been investigated for a classification of educational data. Therefore, this research proposes PSC to build a classification component in teaching effectiveness model, which is capable of classifying teachers' classroom questions into the six cognitive levels of Bloom's taxonomy.

Furthermore, to mitigate the adverse effects of curse of dimensionality, an inherent problem of high dimensional dataset, this paper proposes a novel specialized initialization mechanism that is based on a particular information retrieval algorithm, called Rocchio Algorithm. The rationale behind proposing RA-based initialization mechanism can be expressed as follows: In many previous studies, the key role of initialization mechanisms for evolutionary algorithms, e.g. PSO, has been emphasized (Jabeen et al., 2009; Omran and Al-Sharhan, 2008; Pant et al., 2008; Parsopoulos and Vrahatis, 2002; Uy et al., 2007; Xue et al., 2014; Zhang et al., 2009). Nonetheless, some researches have started to question this role in high dimensional space (Kazimipour et al., 2014; Xue et al., 2014). More specifically, it has been reported that the advanced generic initialization mechanisms perform poorly in high dimensional space, and hence the performance of many evolutionary algorithms deteriorate significantly. Alternatively, a new trend has started to focus on developing specialized initialization mechanisms to improve the performance of evolutionary algorithms in high dimensional space (Ma and Vandenbosch, 2012; Xue et al., 2014). The specialized initialization mechanism exploits problem-specific knowledge to identify a promising region in the search space at which the evolution can occur. A particular region, which has been proven to be promising for the initialization of evolutionary algorithms, has been introduced in centre-based sampling theory (Rahnamayan and Wang, 2009b, 2009a). According to this theory, the centre region of the search space contains points with higher probability to be closer to the unknown optimal solution. On this basis, RA is proposed to identify the centre region of the search space for data classification, and thus ensure a promising initialization of the PSC particles.

In addition, since feature selection plays a crucial role in data classification, the effect of feature selection approaches on the proposed RA-based PSC is investigated. More specifically, the performance of RA-based PSC is investigated with the following four feature selection approaches: Term Frequency (TF) (Xu and Chen, 2010), Mutual information (MI) (Yang and Pedersen, 1997), Chi Square ($\chi 2$) (Galavotti et al., 2000), and Information Gain (IG) (Mladenic, 1998). Finally, the proposed RA-based PSC approach is validated by comparing its results with the results of the following Machine Learning (ML) approaches (Mitchell, 1997): k-nearest neighbor (kNN), Naïve Bayes (NB), Support Vector Machine (SVM), decision tree algorithm (J48), a rule based ML algorithm (RIPPER, JRip), Adaptive Boosting method (AdaBoost), and Bayesian Networks (ByesNet), which have been applied to the same data set under the same settings.

The remainder of this paper is organized as follows: the next section describes PSC in details. The subsequent section introduces the proposed RA-based PSC technique. Then, in Section 4, the recent literature on PSC, EDM, and the specific task of questions classification is reviewed. Thereafter, Section 5 describes how PSC is applied to the questions classification task. Afterwards, the experimental results are presented in Section 6 and discussed in Section 7. Finally, in Section 8, the drawn conclusions are presented.

2. Particle swarm classification

In many fields, PSO is one of the most widely used technique for solving complex optimization problems (Engelbrecht, 2006). Basically, PSO maintains a swarm of particles that represent potential solution to the given optimization problem. Associated with each particle are two values representing particle's position and velocity. Furthermore, each particle has a learning component that merges two types of information: cognitive information (particle's own experience) and social information (entire swarm's experience). While the cognitive information represents the best solution a particle has ever achieved in its history, the social information is the best position the swarm has ever achieved. Together, the cognitive and the social information are used to calculate the velocity of particles and then their next positions. Typically, PSO starts with a random initialization of particles' velocities and positions. Then, the particles move from one position to another to find a better solution

The Standard PSO Algorithm

begin
  for each particle i
    randomly initialize particle's position p_i and velocity v_i
  while ($t < T_{max}$)
    for each particle i
      determine the particle fitness value, f_i
      if f_i is better than f_{bi} // f_{bi} is the fitness of the current local best position
        then $b_i = p_i$
      $f_{bi} = f_i$ // bi is current local best position
    if f_i is better than f_g // f_g is the fitness of the current global best position
      then $p_g = p_i$
        $f_g = f_i$ // p_g is current global best position
  end for
  for each particle i
    calculate particle velocity, v_i // Eq. 1
    update particle position, p_i // Eq. 2
  end for
  end while
end

More formally, as demonstrated in the standard PSO algorithm above, the PSO maintains a swarm of M particles, where each particle is composed of three N-dimensional vectors (N is the dimensionality of the search space): the current position (p_i), the previous best position (b_i), and the velocity (v_i). Each particle moves iteratively with an adaptable velocity within the search space and retains in memory the best position it ever reached. The objective of PSO is to keep finding better positions and updating p_i and b_i by adding v_i coordinates to p_i and adjusting v_i from one iteration to the next as follows: for a particle i, velocity at time $t+1$ (v_i^{t+1}) is a linear combination of its velocity at time t (v_i^t), the difference between the position of the best solution found by the particle up to time t and its current position ($b_i^t - p_i^t$), and the difference between the best position ever found by the total swarm and the particle's current position ($b_g^t - p_i^t$).

$$v_i^{t+1} = w.v_i^t(t) + c_1.U(0,1) \otimes (b_i^t - p_i^t) + c_2.U(0,1) \otimes (b_g^t - p_i^t) \quad (1)$$

where $\otimes$ denotes point–wise vector multiplication; $U(0,1)$ is a function that returns a vector whose positions are randomly generated by a uniform distribution in the range [0,1]; c_1 is the cognitive parameter; c_2 is the social parameter; and w is the inertia factor whose range is [0, 1]. The velocity values must be within a range defined by two parameters: v_{min} and v_{max}. The position of each particle i in the next step is then computed by summing its current position and its velocity:

$$p_i^{t+1} = p_i^t + v_i^{t+1} \tag{2}$$

These operations are repeated for T_{max} iterations or until some other stopping criterion is met, for instance the minimal error with respect to the optimal solution.

Since its inception in 1995, PSO has been applied to various problems, and for many of which the problem-specific characteristics have motivated the development of a PSO variant (Hasanzadeh et al., 2013; Li et al., 2012). For instance, PSC is basically a PSO variant (Nouaouria et al., 2013) developed to tackle data classification tasks. The main idea underlying PSC is the transformation of a given classification problem into an optimization problem and using PSO to find the optimal classifiers according to some pre-specified measures. For this purpose, two types of PSC variants have been developed: rule-based PSC and nearest neighbor-based PSC (Martens et al., 2011). In the rule-based PSC, the classifiers are a set of rules represented in a form of 'IF-THEN' rules. Based on the type of data, categorical or continuous, two methods of rule representation have been developed (Sousa et al., 2004). For the categorical data, binary representation is used, whereas for the continuous data, representation that is more complicated is used. As for the nearest neighbour-based PSC, the classifier is a set of prototype vectors, in N-dimensional space, representing the centroids of classes, and the role of PSO is to find the optimal representative centroid for each class (Nouaouria and Boukadoum, 2010; Nouaouria et al., 2013). The classification on a new data instance is performed by measuring its distance to the previously found prototype vectors.

3. Rocchio Algorithm-based particle swarm classification

In this section, the proposed RA-based PSC is described in details. It is, simply, a nearest neighbour-based PSC with specialized initialization mechanism proposed to overcome the adverse effects of curse of dimensionality on the performance of PSC. From classification systems point of view, the RA-based PSC system is a set of binary classifiers, where a classifier represents an optimal centroid for a given class. The development of the RA-based PSC system is based on the generic method, depicted in Fig. 1. It starts with a preprocessing step of the training data, i.e. feature selection and extraction, followed by a conversion of the training data into a suitable representation. Afterwards, PSO is applied to find the optimal classifiers capable of classifying the testing data effectively according to pre-specified evaluation measures. In doing so, PSC maintains a swarm of M particles, whose coordinates are different tentative centroids for the given class. In other words, each particle has $2{\cdot}N$ components corresponding to the N-dimensional candidate centroid position and velocity. At training stage, PSO iteratively refines the positions to find the best classifier that represents the centroid of the class. In the subsequent testing stage, the discovered centroid is evaluated using class instances in the testing set. In the following the motivations, rationale, and the details of the proposed RA-based initialization mechanism are described.

Typically, in the evolutionary algorithms, the individuals of the initial population are initialized randomly using uniform random distribution. There is strong intuition that better initialization of individuals help the algorithm to achieve better results. Based on this, many advanced initialization mechanisms have been investigated to improve the performance of the evolutionary algorithms, e.g. PSO (Jabeen et al., 2009; Omran and Al-Sharhan, 2008; Pant et al., 2008; Parsopoulos and Vrahatis, 2002; Uy et al., 2007; Xue et al., 2014).

Recently, some researches on the scalability of initialization mechanisms to high dimensional domains have been published (Kazimipour et al., 2014, 2013; Xue et al., 2014). They investigates whether or not the claimed advantages of using the generic initialization mechanisms are still significant when the dimensionality of the data is beyond hundred variables. In these researches, it has been shown that the uniformity of population drops dramatically as the data dimensionality grows. This confirms that uniformity loss is encountered in high dimensional spaces, regardless of the type of evolutionary algorithm, initializer, or problem (Kazimipour et al., 2014). In light of these findings, some researchers have started considering the design of task-specific initialization mechanisms as a viable alternative to generic initialization

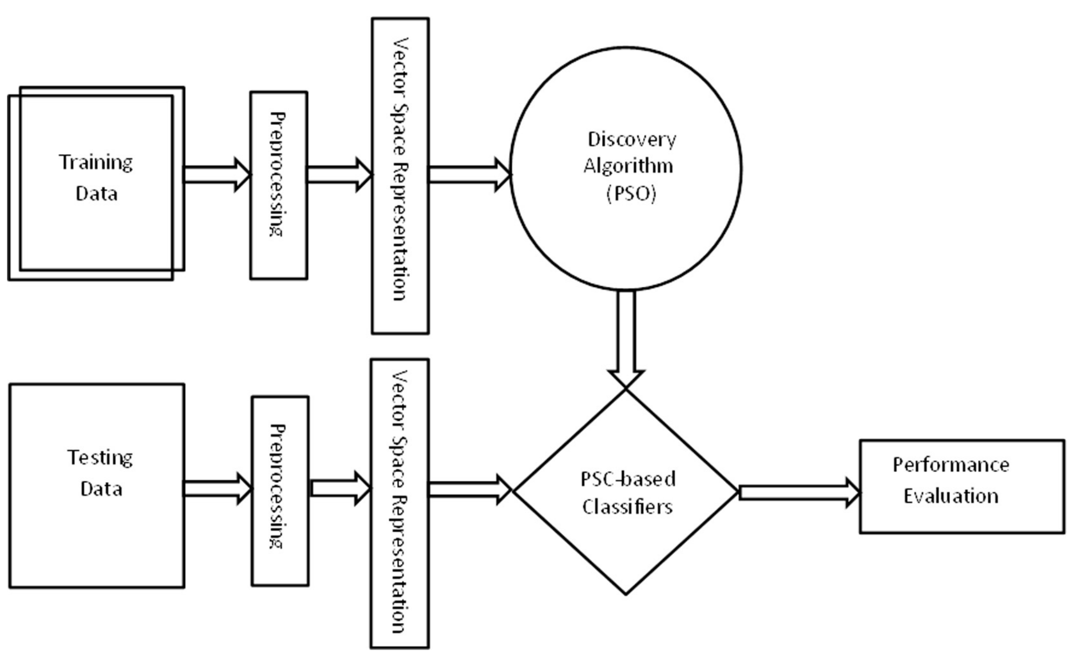

Fig. 1. The development process of PSC.

mechanisms in order to improve the performance of the PSO when it is applied to high dimensional domains (Ma and Vandenbosch, 2012; Xue et al., 2014). These mechanisms based essentially on identifying promising region in the search space to initialize the evolutionary algorithm. A particular region in the search space that has been proven to be a promising region for initializing evolutionary algorithms is introduced in Rahnamayan and Wang (2009b). According to this study, the centre region of the search space is a promising region, because it contains points with higher probability to be closer to the potential solution. Moreover, it has been proven that the probability of closeness to the potential solution increases directly with the dimensionality of the search space. In other words, the average distance of the points which are closer to the centre from the unknown solution is lower and similarly decreases sharply for the higher dimensions. On this basis, when the initialization occurs closer to the centre of the search space, they have a higher chance to be closer to an unknown solution, and also in average, their distance from solution is lower as well. Furthermore, for higher dimensions the mentioned advantageous increases. Given the advantageous of the center region of the search space, it has been utilized to develop generic initialization mechanisms for evolutionary algorithm (Rahnamayan and Wang, 2009a, 2009b).

Motivated by the aforementioned advancements, in this research a novel specialized initialization mechanism is proposed to improve the performance of PSC in high dimensional data classification. The proposed mechanism is based on an information retrieval algorithm, called Rocchio Algorithm, which provide a specific identification of the centre region of data classification search space. RA is an efficient information retrieval algorithm, which was originally designed to use relevance feedback in querying full-text databases (Rocchio, 1971). When it is used for classification, it generates for each class c a prototype vector, which is the average vector overall training set vectors that belong to the class c. To classify a new data instance, it calculates similarity between the vector of the new data instance and each of prototype vectors and assigns it to the class with maximum similarity. More formally, the centroid of a class c is computed as the vector average or centre of mass of its members as follows:

$$\vec{\mu}^c = \frac{1}{|D_c|}\sum_{d\in D_c}\vec{v}(d) \tag{3}$$

where Dc is the set of data instances in the dataset whose class is c. When it is used as an initialization mechanism for PSC particles, the centroid for a given class c is used to initialize each particle i as follows:

$$\vec{p}_i^c = \vec{\mu}^c + \vec{\alpha}_i \tag{4}$$

where $\vec{\mu}^c$ is a vector of mean values $<\mu_1, \ldots, \mu_j, \ldots, \mu_N>$, such that μ_j is the mean value of dimension j over all data instances of in Dc and is a random vector $<\alpha_1, \ldots, \alpha_j, \ldots, \alpha_N>$, such that αj is a small random value generated independently for dimension j in the interval [−R, R], such that the generated particle falls in the range $[\mu_j - R, \mu_j + R]$ cantered at μ_j. It is worth to mention that RA-based initialization mechanism is not computationally expensive, because the vector is calculated once and then used to generate the initial vector of each particle.

4. Literature review

In the recent years, the explosive growth of educational data, that come from educational sources (systems and processes), is a major challenge for educational institutions. EDM is an emerging data mining field that is concerned with developing methods for exploring the educational data. It builds on data mining in other domains such as commerce and biology (Romero and Ventura, 2013). Similar to other data mining fields, classification is a fundamental task in EDM. It represents a substantive part of the works published so far because in education teachers and instructors are classifying their students according to their behaviour and levels of knowledge and motivation. An overview of the typical educational data classification problems and methods are presented in Hämäläinen and Vinni (2010). In this work, several examples of EDM classification tasks are identified, such as classification of student's academic success at the university level, classification of student's success in specific course, predicting student's success in the next task, given her/his answers to previous tasks, and classification of metacognitive skills and other factors which affect learning.

With regard to the specific EDM classification task, questions classification, it has been addressed in many previous works. In one of these works, (Fei et al., 2003), an artificial neural network, namely back-propagation neural network, is used to classify question into three difficulty levels that are easy, medium, and hard. The obtained results in term of F-measure value is nearly 78%. A decision tree is used in Cheng et al. (2005), to construct an automatic classification of questions of E-learning examination pool according to its difficulty to choose questions that are suitable for each learner according to individual background. In Chein and Liau (2004), an automatic classifier for Chinese questions of particular keywords is designed. In Ince (2008), three artificial intelligence methods: artificial neural network, support vector machine, and adaptive network-based fuzzy inference system are proposed as a means to achieve accurate question level diagnosis, intelligent question classification and updates of the question model in intelligent learning environments such as E-learning or distance education platforms. It reports the investigation of the effectiveness and performances of these methods within a web-based E-learning environment in the testing part of an undergraduate course to observe their questions classification abilities depending on the item responses of students, item difficulties of questions and question levels that are determined by putting the item difficulties to Gaussian normal curve. The comparative test performance analysis conducted using the classification correctness revealed that the adaptive network-based fuzzy inference system yielded better performances than the artificial neural network and support vector machine. In Karahoca et al. (2009), a questions classification model is developed using rapidly-exploring random tree algorithm to determine the item difficulties of tests that are used on a computer adaptive testing system. The effect of the size of item pool on the question classification is also investigated and found important for question classification. An interesting work on question classification is presented in Nuntiyagul et al. (2008). In this work, an adaptable learning assistant tool for managing question bank is presented. The tool is able to automatically assist educational users to classify the question into predefined classes and correctly retrieve the questions by specifying the class and/or the difficulty levels. The system is tested and evaluated in terms of accuracy and user satisfaction. A recent work that embeds question classification within intelligent tutoring system is presented in Kavitha et al. (2012). In this work a norm referencing is used to classify questions based on item difficulty. The question classification have recently addressed in the online question and answer forums such as Stack Exchange and Quora, an increasingly popular resource for education. For such systems, a multi-label classification system that automatically tags users' questions using a linear support vector machine and a carefully chosen subset of the entire feature set to enhance user experience is proposed in Eric et al. (2014).

Although in most of the above reviewed works, the classification of questions is based on the levels of difficulty, more recently, the questions classification based on Bloom's taxonomy cognitive level has been reported in several works. In Haris and Omar (2013, 2015), a rule-based approach is used to develop questions classification model that classify written examination questions of computer programming course. The aim is to provide lecturers with a tool to assess the student's cognitive levels from the written examination questions. The preliminary results show that it is a viable approach to help categorize the questions automatically according to Bloom's Taxonomy. Another rule-based questions classification model of the exam questions based on Bloom's taxonomy is presented in Omar et al. (2012), Fattah et al. (2007), in which a question classification model, in the context of developing a web application system to design a good examination questions, is developed. The system involves question authoring module, question retrieval module, question analysis module, and exam paper generation module. The question analysis module determines the difficulty level of each question based on Bloom's Taxonomy in the question paper using the keyword/s found in the question. In Abduljabbar and Omar (2015), the question classification into Bloom's cognitive levels is tackled using support vector machine, naïve bayes, and k-nearest neighbour with or without feature selection methods, namely chi-square, mutual information and odd ratio. A combination algorithm is used to integrate the overall strength of the three classifiers. The classification model achieves highest result when applying mutual information, which proved to be promising and comparable to other similar models. In Jayakodi et al. (2016), WordNet with Cosine similarity algorithm are used to classify a given exam question according to Bloom's taxonomy learning levels. The question classification model consists of tag pattern generation module, grammar generation module, parser generation and cosine similarity checking module. The data is a set of exam questions taken from courses at the Department of Computing and Information Systems at Wayamba University. The results show that the performance of the classification model is consistent with those provided by domain experts on approximately 71% of occasions. In Yusof and Hui (2010), a question classification model to classify items in examinations based on Bloom's taxonomy is proposed. The model applies the artificial neural network approach, which is trained using the scaled conjugate gradient learning algorithm. Several data preprocessing techniques and feature reduction methods are investigated. The experimental results indicate that the model can enhance the convergence speed and document frequency is the most effective feature reduction method. In Yahya et al. (2013), Yahya and El Bashir (2014), the effectiveness of three machine learning approaches, namely, k-nearest neighbours, naïve Bayes, and support vector machines with different term selection approaches are investigated on the task of classifying teachers' classroom questions into different cognitive levels identified in Bloom's taxonomy. A dataset of questions is collected and annotated manually with Bloom's cognitive levels and several steps of pre-processing have been applied to convert these questions into a representation suitable for machine learning approaches. The results show that machine learning approaches have a superior performance over rule-based approach and the support vector machine shows a superior performance over k-nearest neighbour and naïve Bayes. In Dubey and Goyal (2016), a question classification model to classify questions asked on stack overflow, a popular social networks source of technical questions and answers useful for the education domain, according to Bloom's taxonomy is developed. LDA, a three-level hierarchical Bayesian model, i is applied to reduce the dimensions of each item and then use k-means algorithm on a collection having unlabelled and labelled items to get the result. Initially, an accuracy of 30.2% is obtained with this approach and with further augment other features like score, answer count and view count an accuracy of 56.33% is obtained.

From the forgoing works, it is obvious that the interest in questions classification task based on Bloom's taxonomy cognitive levels is growing due to its implications for educational systems and e-learning platforms. As can be observed, in most of the works, the question classification model is an embedded component within a certain educational system that is developed to emulate an educational practice, such as intelligent tutoring systems, item bank question systems, automatic test generation systems, adaptive test systems, question answering systems, etc. Definitely, the growing interest in integrating Bloom's taxonomy in educational systems is a reflection of its importance for many educational practices.

As the objective of this research is to develop a questions classification system using the recent state of the art approaches, a comprehensive review of the data classification in EDM is useful. To this end, five key surveys have been adopted as a source of information (Baker and Yacef, 2009; Peña, 2014; Pena et al., 2009; Romero and Ventura, 2007, 2010). Each survey reviews EDM works published on specific time interval and categorizes them according to a pre-specified taxonomy. The surveys have been analysed with particular attention paid to the classification works and the applied classification method. As shown in Fig. 2. Bayesian networks, decision trees, neural networks, and regression methods dominate other classification methods. The domination of these methods is a direct result of their ability to capture the dependencies between the class attributes and to avoid overfitting (Hämäläinen and Vinni, 2010). Furthermore, the analysis of these surveys reveals that there are still many methods that have not yet been explored for classification in EDM. Swarm intelligence-based methods are prominent examples of these methods, which in spite of their proven success for tackling data mining tasks in different domains, have not yet been explored for EDM classification (Grosan et al., 2006; Yahya et al., 2014). This can be attributed to the relatively nascent stage of the EDM field and to the popularity of using swarm intelligence methods, e.g., PSO and Ant Colony Optimization, for tackling optimization tasks rather than classification (Nouaouria et al., 2013).

On the contrary to the lack of PSC applications in EDM classification, the interest in applying PSC to data classification in other domains has grown considerably and resulted in two types of PSC models: rule-based PSC and nearest neighbour-based PSC (Martens et al., 2011). As the interest of this research lay in the nearest neighbour PSC, the following review is confined to the nearest neighbour-based PSC works, regardless of their domains. Furthermore, for the sake of organization, the reviewed works are grouped into three categories: the PSC models that depend on the standard PSO, the PSC models that depend on modified PSO, and the PSC models that are applied for high dimensional data classification.

In the first category, the standard PSO has been used in many works. For example, in Tewolde and Hanna (2007), the standard PSO is used to develop a PSC for the implementation of single and multi-surface based data separation methods for classification of breast cancer data from University California, Irvine (UCI) machine learning repository. The experimental results showed excellent classification accuracies, ranging from 97% to 100%, by the two approaches. In Tsai and Yeh (2008), a PSC approach is developed for inventory classification problems, where inventory items are classified based on a specific objective or multiple objectives, such as minimizing costs, maximizing inventory turnover ratios, and maximizing inventory correlation. The experimental results of applying the developed PSC on inventory data collected from a printed circuit board manufacturer with 1101 items and 47 suppliers show that PSC performs comparatively well with respect to those schemes that are commonly used in practice. In Ng et al. (2009), a PSC model that is based on the standard PSO

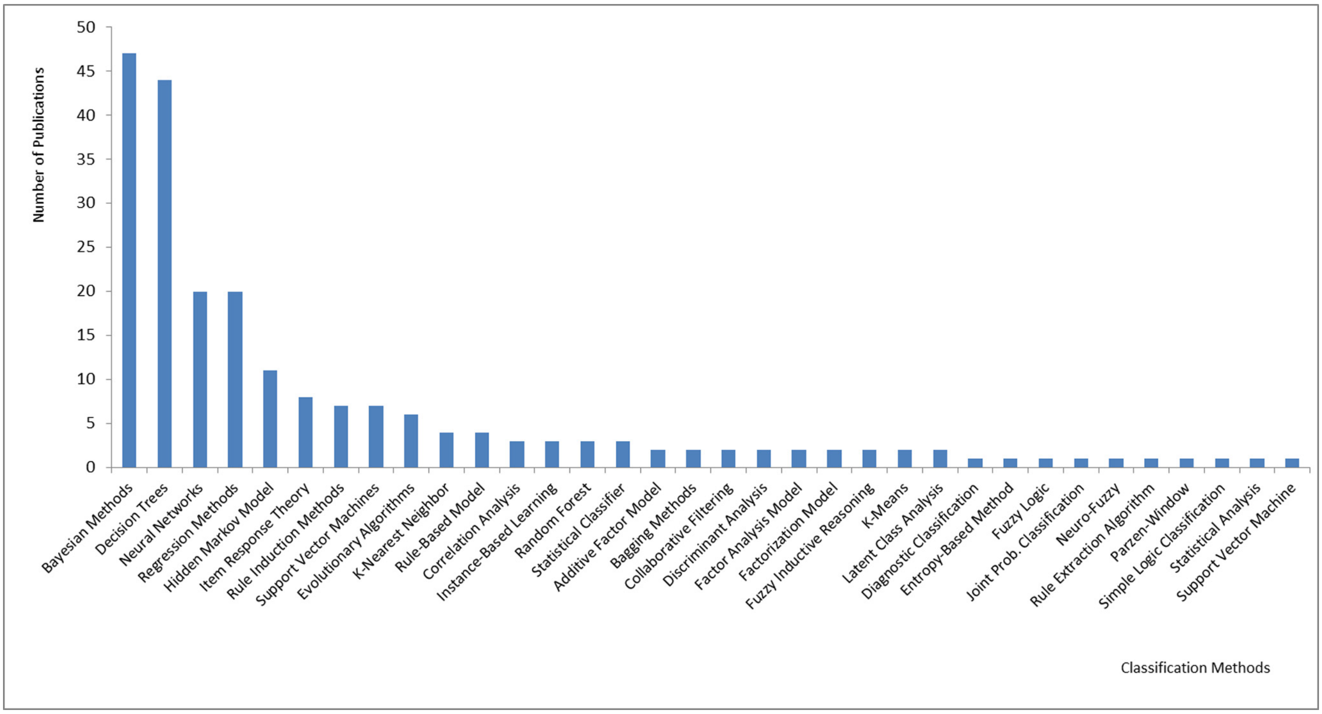

Fig. 2. Statistics of the Previous Classification Methods in EDM literature.

is developed and used to perform unsupervised clustering of multi-class images. The experimental results drawn from separate multi-class image data sets revealed that the PSC produced better and more consistent results in terms of cluster characteristics and subsequent recognition than the k-means algorithm. In Liu et al. (2010), a PSC model was developed based on the standard PSO for data clustering. In this PSC model, four cluster validity indices (Euclidean distance-based PBM, the kernel function-induced CS, Point Symmetry, and Manifold Distance) are used as fitness functions. Results of experimentation on 31 data sets (artificial, synthesized data and UCI data) uncovered that the PSC with the Manifold Distance index outperforms the PSC with the other indices in terms of clustering accuracy and robustness. In Nebti and Boukerram (2010), PSO, Bees Algorithm, Artificial Bees Colony Optimization, Multilayer Perceptron, and a Hybrid Multilayer Perceptron and Bees Algorithm are applied to the problem of hand-written Arabic digits recognition (selected from the MNIST database). The experimental results disclosed that performance of the PSC is better than those of the Artificial Bees Colony Optimization and Multilayer Perceptron but worse than those of the Bees Algorithm and the Hybrid Multilayer Perceptron and Bees Algorithm, which produces the best results. In Kalyani and Swarup (2011), the standard PSO is used to develop a PSC for static security assessment of large-scale power systems. Results of experiments on data sets of the IEEE 14 Bus, IEEE 57 Bus, and IEEE 118 Bus systems points out that the PSC produces fairly-high classification accuracy and a low misclassification rate. In Wang and Ma (2014), a novel PSC based on the standard PSO and the k-nearest neighbour algorithm is proposed for diagnosing faults in a power plant thermal system under two operating points (600-MW and 480-MW). The experimental results derived from processing five UCI data sets (Diabetes, Glass, Heart, Iris, and Wisconson) demonstrate the validity of the proposed approach.

It should be mentioned that the standard PSC is not always effective as nearest neighbour-based classification model, and consequently different variants of PSO have been developed and employed in the development of the nearest neighbour PSC. In Owechko et al. (2004), a PSC that is based on PSO extended with sequential equations is used to classify a set of infrared images for human detection using Haar wavelet and edge-symmetry features. The results show that the proposed PSC is effective, very fast and can robustly detect multiple objects in the scene. In O'Neill and Brabazon (2006), a PSC model that is based on a novel self-organizing particle swarm (SOSwarm) is proposed and tested on four UCI ML repository data: Breast cancer, Pima Indians diabetes, New Thyroid, and Glass. The proposed PSC outperforms or accords with the best-reported results on all four data sets. In Omran and Al-Sharhan (2007), A new variant of PSO is proposed in which no parameters' tuning is required. It is used in a PSC model to classify synthetic, MRI, and satellite images. The experimental results demonstrate that the proposed PSC outperforms the state-of-the-art algorithms. In Cervantes et al. (2009), a PSC model that is based on Adaptive Michigan PSO (AMPSO) is developed, in which each particle in the swarm represents a single prototype in the solution space and the swarm evolves using modified PSO equations with both particle competition and cooperation. The experiments on seven UCI data sets (Balance Scale, Bupa, Diabetes, Glass, Iris, Thyroid, and Wisconsin) show that AMPSO could always find a better solution than the standard PSO. In addition, the AMPSO is competitive with some of the most commonly-used classification algorithms. A Quantum-based PSO (QPSO) is proposed in Chen et al. (2008) and used to develop a PSC for gene expression data analysis. In the QPSO, the update rule is a combination of the mean of personal best positions among the particles and a contraction-expansion coefficient. Experiments on four gene expression data sets (rat CNS data set, the GAL data set, and two yeast cell data sets) show that the QPSO-based PSC is always able to obtain partitions with outstanding effect and that it is a promising tool for gene expression data clustering.

As the interest in using PSC for classification grows, several concerns about its effectiveness have been raised. A major concern that has been receiving considerable attention is its effectiveness in high dimensional data classification. This concern was initially addressed in De Falco et al. (2005, 2007) where the PSC is applied to nine data sets and the obtained results are compared with the results of nine classical classification algorithms. In these studies, the effects of the number of classes and data size and dimensionality on the PSC effectiveness are investigated. Experiments are conducted on nine UCI data sets: Card, Diabetes, Glass, Heart, Horse, Iris, Wdbc, Wdbc-I, and Wine. In these data sets, the number of classes ranges from 2 to 6, the dimensions range from 4 to 58, and the sizes range from 150 to 768. The conclusion is that the PSC classification accuracy tends to decrease with increasing values of classes in the data set and with the value of the product of data set size by dimensionality. This conclusion is questioned in Nouaouria and Boukadoum (2009, 2010) by investigating the performance of the PSC on a more complex data set (Fluorescence measurements on substance data set with 19 classes, 2103 data instances, and 64 space dimensions). The results show positive performance of PSO when using a mechanism of confinement of the data values. Further investigation is carried out in order to evaluate the extent to which the generalization of the previous conclusions holds for three additional data sets with sizes of 2103, 16000, and 3823; dimensions of 64, 16, and 64; and class numbers of 19, 26, and 10. The results bring to surface that PSC with additional coping mechanism has good potential as a classification tool, even for high dimensioned problem spaces with a large number of instances and multiple classes.

In summary, the above review reveals a couple of findings that motivate the current work. First, it indicates that questions classification based on Bloom's taxonomy is a task of crucial importance for e-learning systems and platforms. Second, in spite of wide range of PSC application domains, no works on PSC application to EDM filed has been reported. Third, the review indicates that the performance of the standard PSC tends to decrease as data dimension increases, and when coping mechanisms are used, the performance of the PSC improves, however, none of the previous works has investigated the effects of initialization mechanisms on the performance of PSC for high dimensional data classification.

5. RA-based PSC for educational questions classification

In education, teaching effectiveness is a multi-dimensional concept that is defined as the ability of a teacher to inculcate knowledge and skills in students as well as to change their behaviours. Although it is influenced by a combination of teacher skills, it is widely acknowledged that teacher's questioning skill is one of the key indicators of teaching effectiveness. Hamilton, an early adopter of the importance of acquisition and development of questioning skills, was quoted as saying that questions are the core of effective teaching (Ramsey et al., 1990). Presently, it is quite a common fact that the essence of effective teaching is related to good questioning (Ornstein, 1987). In educational practices, questioning is the most frequently used instructional intervention in classroom as teachers ask many questions in a lecture for a variety of purposes (e.g., developing interest and critical thinking skills) (Levin and Long, 1981). Taxonomy-based analysis is one of the most commonly used methods for analysing teacher's questions (Hackbarth, 2005). Within this context, many classification systems have been developed (Bloom et al., 1984), and Bloom's Taxonomy is the most salient example. It was developed by Benjamin Bloom, who identified six levels of learning in cognitive domain known as Bloom's cognitive levels (BCLs), and organized them on the basis of hierarchy as depicted in Fig. 3.

It follows from the above that the automatic classification of the questions of the teacher in the classroom is a key component for modelling teaching effectiveness. Besides that, as described in Section 3, the automatic question classification is a key component in many computer-based educational systems and e-learning platforms, such as intelligent tutoring systems, automatic test generation systems, item bank management systems, adaptive testing systems, question answering systems, etc. On account of this, the RA-based PSC is proposed to automatically assign a BCL to a given teacher's classroom questions as described in the following subsections.

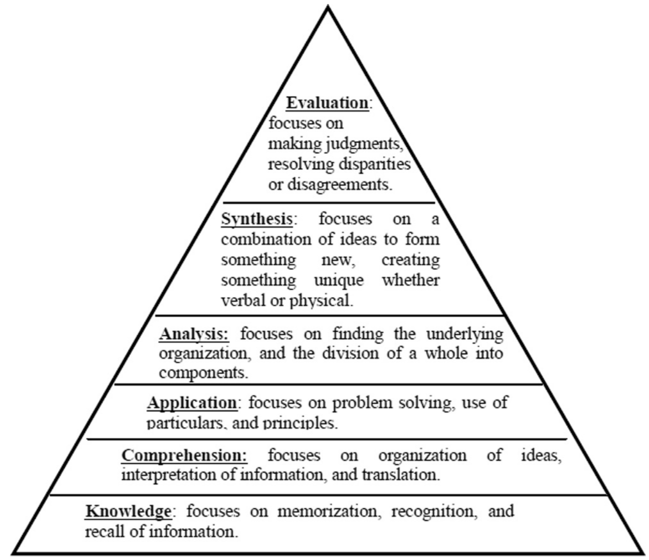

Fig. 3. Bloom's Taxonomy Cognitive Levels.

5.1. Collection of educational questions data

The questions data set is a set of questions collected over a period of three semesters of the academic years 2011 and 2012, from lecturers of computer science courses at Najran University. The procedure of data collection was based on lecturers who were requested to keep records of the questions they direct to their students in the classrooms. All lecturers were informed about Bloom's taxonomy and the objectives of this work. It should be mentioned that due to nascence of the EDM field, the availability of benchmark data sets is still a challenge (Woolf et al., 2013) despite some current efforts such as the Datashop from Pittsburgh Science of Learning Center (Koedinger et al., 2008), which provides many educational data sets and facilitates data analysis. With regard to the collected questions data set, which has been used in this work, it was used in several previous studies (Yahya et al., 2013; Yahya and El Bashir, 2014), and currently is available for benchmarking. Table 1 shows a sample of questions and their corresponding BCL class.

5.2. Questions preprocessing

Since the textual data are not suitable for most classifier learning algorithms, a transformation of the questions data set into a

Table 1
Examples of Questions Dataset.

BCL	Question Example
Knowledge	what are the main components of a finite automata
Comprehension	describe the four clauses in the syntax of a simple SQL retrieval query
Application	if you apply quick select to find the median of the list of numbers what would be the results ?
Analysis	compare right-linear grammar with left-linear grammar.
Synthesis	How would you model this problem by a graph?
Evaluation	Which grammar do you think should be used?

suitable representation (e.g., vector space representation) was performed. Typically, data preprocessing involves steps of linguistic, morphological, and statistical analysis such as tokenization, stemming, and feature selection. The following subsections describe the preprocessing steps applied to the questions data set.

5.2.1. Tokenization (Term Extraction)

Tokenization involves breaking the text stream into meaningful tokens, also called terms, such as symbols, phrases, and words. Typically, tokenization starts by defining the meaning of token, usually in a form of regular expression. For the questions data set, the term is defined as a maximal sequence of non-blank characters where all letters are in the lower case form. Therefore, tokenization involves reducing the question text to lower case characters and generating its terms set.

5.2.1.1. Unuseful term removal. In English language, there are groups of words that are not informative for text classification, called stop words (the most frequently used words) such as pronouns, prepositions, and conjunctions. The stop words in English include about 400–500 words such as 'the', 'of', 'to', ... etc. It has been proven that the removal of the stop words is very useful for text classification (Silva and Ribeiro, 2003). In this work, the stop words as defined in Salton (1989) have been removed. Besides the stop words, the following three groups have been also removed:

- Punctuation marks. All punctuation marks.
- Numbers. Terms consisting purely of digits.
- Low-frequency terms. Terms with frequency less than three.

5.2.2. Stemming

In linguistic morphology, stemming is the process of reducing the inflected words to their roots or base forms, known as stems, which may not be the same as the morphological root of that word. Stemming is usually done by removing any attached suffixes and prefixes from terms to improve the classification accuracy. Although many stemming algorithms have been developed, a porter stemmer has been used for the questions data set (Porter, 1980). This stemmer is based on the idea that the suffixes in the English (approximately 1200) are mostly made up of a combination of smaller and simpler suffixes. It proceeds in steps, and in each step, rules are applied until one of them is satisfied. If a rule is accepted, the suffix is removed accordingly, and the next step is performed. Eventually, the resultant stem at the end of the fifth step is returned.

5.2.3. Term selection

In this step, a feature selection approach is applied to select a sub-set from the original terms set (a set of all the terms in the questions under consideration) such that only the most representative terms are used. A computationally easier term selection approach is the filter approach (Yahya et al., 2011), which selects a sub-set of terms that receive the highest score according to a function of term importance for the classification task. In this work, a filter approach based on Term Frequency (TF) has been applied due to its ability to take into account the multiple appearances of a term in the questions (Xu and Chen, 2010).

5.2.4. Term weighting

Term weighting is the task of assigning a numerical value to each term based on its contribution to characterization of classes. In its simplest form, a binary weighting can be used, where 1 denotes presence and 0 absence of the term. However, non-binary weight forms are most often used. In this work, a non-binary weighting in the form of term frequency inverse document frequency (*tfidf*) has been applied (Sebastiani, 2002). First, the *tfidf* of each term t_k in a question q_j is computed as follows:

$$tfidf(t_k, q_j) = tf(t_k, q_j) \times \log\left(\frac{N(Tr)}{N\left(q_{t_k}, Tr\right)}\right) \tag{5}$$

where $tf(t_k, q_j)$ denotes the number of times t_k appears in q_j, $N(Tr)$ represents the number of questions in the training set Tr, and $N(q_{t_k}, Tr)$ expresses the number of questions, in Tr in which the q_{t_k} term, t_k is encountered. The term weight is then computed as follows

$$w(t_k, q_j) = \frac{tfidf(t_k, q_j)}{\sqrt{\sum_{k=1}^{T}\left(tfidf(t_k, q_j)\right)^2}} \tag{6}$$

where T is the number of unique terms in Tr.

5.2.5. Vector space representation

In this step, each question q_j is represented as a vector of term weights $<w_{1j}, \ldots, w_{Tj}>$, where $0 \le w_{kj} \le 1$ represents the weight of term t_k in q_j.

5.3. Classifiers learning

In this phase, RA-based PSC is applied to learn a classifier for each *BCL* class from the training set. The main idea is that given a training set divided into instances labelled with the given *BCL* (D_{BCL}) and instances labelled with other *BCLs* ($D_{\overline{BCL}}$) represented in *N*-dimensional vector space, the RA-based PSC searches for the optimal centroids of that *BCL* and $\overline{BCL}$ in the *N*-dimensional space. By so doing, RA-based PSC starts with a swarm of particles initialized as described in Section III, where each particle represents potential centroids of that *BCL* and $\overline{BCL}$. Then, the RA-based PSC iterates through a search process to obtain the best centroids. The *i*th particle in the swarm is encoded as a concatenation of four vectors as follows:

$$\left\{\vec{p}_i^{BCL}, \vec{p}_i^{\overline{BCL}}, \vec{v}_i^{BCL}, \vec{v}_i^{\overline{BCL}}\right\} \tag{7}$$

In this representation, $\vec{p}_i^{BCL}$ and $\vec{p}_i^{\overline{BCL}}$ of particle i represent the candidate centroid of the given BCL and $\overline{BCL}$ as follows:

$$\vec{p}_i^{BCL} = \{p_{i1}^{BCL}, \ldots, p_{iN}^{BCL}\} \tag{8}$$

$$\vec{p}_i^{\overline{BCL}} = \left\{p_{i1}^{\overline{BCL}}, \ldots p_{iN}^{\overline{BCL}}\right\} \tag{9}$$

Similarly, $\vec{v}_i^{BCL}$, $\vec{v}_i^{\overline{BCL}}$ of particle i express the velocity components of the particle as follows:

$$\vec{v}_i^{BCL} = \{v_{i1}^{BCL}, \ldots, v_{iN}^{BCL}\} \tag{10}$$

$$\vec{v}_i^{\overline{BCL}} = \left\{v_{i1}^{\overline{BCL}}, \ldots v_{iN}^{\overline{BCL}}\right\} \tag{11}$$

During the search, each particle i is evaluated in order to assess its fitness by a fitness function ψ computed over all training set instances, D_{BCL} and $D_{\overline{BCL}}$, as the sum of two components: the summation of the Euclidean distance between $\vec{x}_j$ and $\vec{p}_i^{BCL}$ divided by the number of data instances in D_{BCL}, and the summation of the Euclidean distance between $\vec{x}_j$ and $\vec{p}_i^{\overline{BCL}}$ divided by the number of data instances in $D_{\overline{BCL}}$ as follows:

$$\psi(i) = \frac{1}{D_{BCL}}\sum_{j=1}^{D_{BCL}} d(\vec{x}_j, \vec{p}_i^{BCL}) + \frac{1}{D_{\overline{BCL}}}\sum_{j=1}^{D\overline{BCL}} d(\vec{x}_j, \vec{p}_i^{BCL}) \tag{12}$$

As such, the problem is a typical minimization problem to which RA-based PSC can be applied. At the end of this phase, the best particle found is used as the centroid representing the BCL and $\overline{BCL}$ classes in the subsequent evaluation phase.

5.4. Evaluation of the classifiers

The performance of the RA-based PSC classifiers depends essentially on computing a contingency table obtained from the classification of a sub-set of questions referred to as the testing set. For a given *BCL* classifier, the contingency table consists of the following

- True Positive (*TP*): number of questions a classifier correctly assigns to the right *BCL* classes.
- False Positive (*FP*): number of questions a classifier incorrectly assigns to *BCL* classes.
- False Negative (*FN*): number of questions that belong to the class but which the classifier incorrectly assigns to other *BCL* class(es).
- True Negative (*TN*): number of questions a classifier does not assign to inappropriate *BCL* class(es).

The following are the common measures used to evaluate the performance of a given *BCL* classifier:

- Precision (*P*): probability that if a random question is classified under *BCL*, then this classification is correct. That is,

$$P = \frac{TP}{TP + FP} \tag{13}$$

- Recall (R): probability that if a random question ought to be classified under ci, then this classification is done. That is

$$R = \frac{TP}{TP + FN} \tag{14}$$

Normally, the P and R measures are combined into a single F_β measure (harmonic mean), which is defined for $\beta = 1.0$, as follows:

$$F_{1.0} = \frac{2RP}{R + p} \tag{15}$$

Based on the above measure, the performance across a set of BCLs classifiers can be measured by Macro-Average (unweighted mean of performance across all classes) and Micro-Average (performance computed from the sum of per-class contingency tables). In this work, the Macro-Average of F_1 has been used.

6. Experimental results

This section presents the results obtained from sets of experiments conducted to evaluate the performance of the proposed RA-based PSC approach for the classification of the questions data set. First, the standard PSC with several generic initialization mechanisms is applied. Then, the proposed RA-based PSC is applied to evaluate the role of RA-based initialization mechanisms in the classification process. Thereafter, the effect of four feature selection approaches on the performance of the RA-based PSC is investigated. Finally, to validate the proposed RA-based PSC approach, its results are compared with the results of seven ML approaches experimented on the same dataset.

Concerning the questions data set, it has been decided to have an equal distribution of questions among BCLs such that each BCL will have 1000 questions, so as to avoid the potential effect of data skewness. Moreover, the questions data set has been divided into a training set (70%) and a testing set (30%) and each approach has been repetitively applied to 50 experimental cases. In each case, the number of terms, *k*, used for question representation varies as follows: $k = 10, 20, \ldots, 500$. On the other hand, the PSC control parameters were set as recommended by a previous work on application of the PSC to a high dimensional data classification problem (Nouaouria and Boukadoum, 2010). Thus, the PSO parameter values are as follows: $T_{max} = 1000$, $v_{max} = 0.5$, $v_{min} = -0.5$, $c1 = 2.0$, $c2 = 2.0$, $v_{op} = -1$, $v_{su} = 1$, and $M = 100$ particles. Furthermore, the parameters of the proposed RA-based initialization mechanism were set to $r_{min}=-0.1$ and $r_{max}=0.1$. In the experiments of ML approaches, Weka tool (Hall et al., 2009) is used.

In all experiments, evaluation of the classifier's performance is based on the following three F_1-based criteria: the average performance over all experimental cases, the best F_1 of each BCL classifier obtained using a specific number of terms, and the best F_1 of each BCL classifier obtained using different numbers of terms for each classifier. In addition to that and for comparison between approaches, the number of experimental cases in which the particular approach outperformed the others was used.

6.1. The standard PSC

In this experiment, the standard PSC is applied to classify questions of the testing set into BCLs, as described above, and the results are presented in Table 2. It is clear that the performance of the standard PSC is poor, in all aspects of evaluation mentioned above. The poor performance of the PSC is consistent with the findings of previous works (S. Chen et al., 2015; Clerc and Kennedy, 2002; Hatanaka et al., 2009; Kazimipour et al., 2014; Li and Yao, 2009; Vesterstrom and Thomsen, 2004), which applied PSO to function optimization in high dimensional spaces.

6.2. Rocchio Algorithm-based PSC approach

The results of applying the proposed RA-based PSC to induce classifiers for BCLs are shown graphically in Fig. 4. The general pattern of the RA-based PSC performance is characterized by a gradual improvement as the number of terms increases. However, when the number of terms reaches a certain value, the performance tends to be stable with a slight tendency to improvement. Fig. 4 also discloses that the ability of the RA-based PSC to find classifiers for BCLs varies. It is the highest for the Analysis BCL and the lowest

Table 2
Performance of the Standard PSC.

BCL	Average F_1 over all experimental Cases	Best F_1 using 410 terms	Best F_1 using any number of terms	
			Best F_1	*No. of terms*
Knowledge	0.241	0.286	0.299	480
Comprehension	0.239	0.272	0.286	450
Application	0.248	0.375	0.375	370
Analysis	0.253	0.286	0.286	20
Synthesis	0.251	0.192	0.338	400
Evaluation	0.234	0.264	0.264	250
Average	0.243	0.279	0.308	328

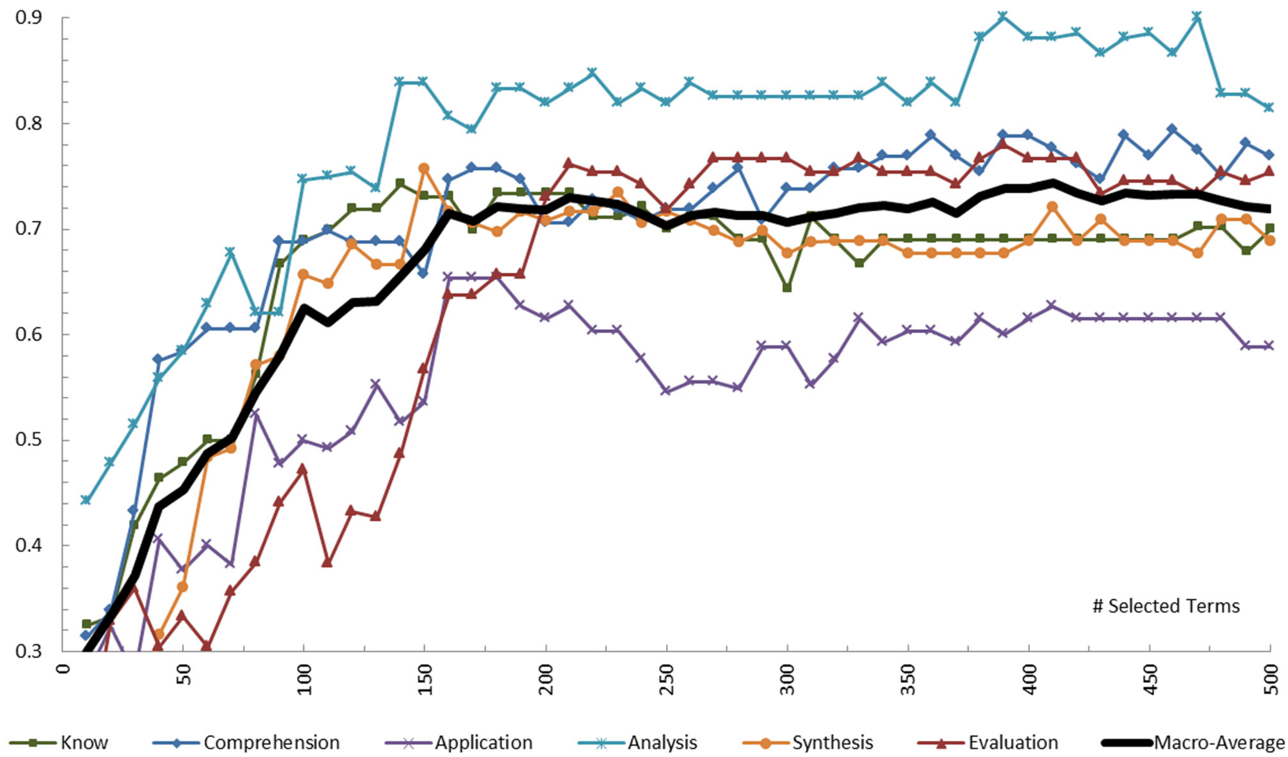

Fig. 4. Performance of the RA-based PSC.

Table 3
Performance of the RA-based PSC.

BCL	Average F_1 over all experimental Cases	Best F_1 using 410	Best F_1 using any No. of terms	
			Best F_1	*No. of terms*
Knowledge	0.660	0.690	0.742	140
Comprehension	0.704	0.776	0.794	460
Application	0.553	0.627	0.654	160
Analysis	0.785	0.881	0.900	390
Synthesis	0.639	0.721	0.758	150
Evaluation	0.634	0.767	0.779	390
Macro Avg-F_1	0.663	0.744	0.771	282

for the Application BCL, however, it is comparable for the Comprehension, Evaluation, Synthesis, and Knowledge BCLs.

Besides, Table 3 summarizes the results of the RA-based PSC performance in terms of the evaluation criteria specified earlier. The average values in the last row of the table can be interpreted as follows; the average ability of the RA-based PSC to build a BCLs classification system, regardless of the number of terms used for each BCL, is 0.663; the Macro-Average F1 of the best classification system composed of BCLs classifiers obtained using 410 terms is 0.744; and the Macro-Average F1 of the classification system composed of the best BCLs classifiers obtained using different number of terms is 0.771.

The comparison between the RA-based PSC and the PSC with the generic initialization mechanisms provides evidence on the key role of the RA-based initialization mechanisms for improving the performance of PSC when applied to high dimensional data classification. As discussed above, the generic initialization mechanisms fail in the high dimensional space to initialize the PSC particles with good starting positions and, consequently, they converge prematurely to a local solution far away from the global one. By initializing particles using RA-based initialization mechanism, the particles are positioned at a promising region of the search space and thus, the convergence of the PSC is likely speed-up.

6.3. RA-based PSC with feature selection approaches

In the previous experiments, the TF approach was used to select k terms for each experimental case. In the current set of experiments the objective is to investigate the impact of feature selection approaches on the performance of the RA-based PSC. Therefore, the following term selection approaches are examined: Mutual information (MI), Chi Square ($\chi 2$), and Information Gain (IG), and consequently the corresponding RA-based PSC variants are abbreviated as follows: RA-based PSC_{TF}, RA-based PSC_{MI}, RA-based $PSC_{\chi 2}$, and RA-based PSC_{IG}.

The results of these experiments, shown in Table 4, illustrate the outperformance of the RA-based PSC_{IG} in terms of the average performance over all experimental cases, outperformance of the RA-based $PSC\chi 2$ in terms of the best F_1 of the BCLs classifiers using a specific number of terms, the comparable performance of the RA-based PSC_{TF}, RA-based $PSC\chi 2$, and the RA-based PSC_{IG} in terms of the best F_1 for each BCL obtained using different numbers of terms.

Table 4
Performances of the RA-based PSC Variants.

BCL	RA-based PSC_{TF}				RA-based PSC_{MI}			
	Average F_1 over all experimental Cases	Best F_1 using 410 terms	Best F_1 using any No. of terms		Average F_1 over all experimental Cases	Best F_1 using 490 terms	Best F_1 using any No. of terms	
			Best F_1	*No. of terms*			*Best F_1*	*No. of terms*
Knowledge	0.660	0.690	0.742	140	0.465	0.623	0.623	500
Comprehension	0.704	0.776	0.794	460	0.547	0.719	0.730	480
Application	0.553	0.627	0.654	160	0.400	0.571	0.571	500
Analysis	0.785	0.881	0.900	390	0.680	0.933	0.933	470
Synthesis	0.639	0.721	0.758	150	0.543	0.676	0.727	190
Evaluation	0.634	0.767	0.779	390	0.576	0.706	0.746	480
Average	0.663	0.744	0.771	282	0.535	0.705	0.722	436
BCL	RA-based $PSC_{\chi 2}$				RA-based PSC_{IG}			
	Average F_1 over all experimental Cases	Best F_1 using 500 terms	Best F_1 using any No. of terms		Average F_1 over all experimental Cases	Best F_1 using 430 terms	Best F_1 using any No. of terms	
			Best F_1	No. of terms			Best F_1	No. of terms
Knowledge	0.601	0.667	0.719	250	0.633	0.698	0.700	380
Comprehension	0.688	0.716	0.746	500	0.696	0.735	0.750	490
Application	0.626	0.741	0.741	460	0.678	0.755	0.755	430
Analysis	0.856	0.918	0.918	470	0.869	0.900	0.929	100
Synthesis	0.69	0.730	0.750	420	0.692	0.697	0.758	320
Evaluation	0.681	0.75	0.765	40	0.680	0.704	0.732	360
Average	0.69	0.754	0.773	357	0.708	0.748	0.771	347

Moreover, the results (Table 4) reveal a low performance of the RA-based PSC_{MI} in all aspects of evaluation criteria.

Finally, Fig. 5 graphically represents the performances of the four RA-based PSC variants in all the experimental cases. A comparative look at the performance patterns indicates that the RA-based PSC_{TF} and the RA-based PSC_{MI} have a similar pattern that is characterized by a gradual improvement with the increase in the number of terms, followed by a long period of wavy pattern with a slight tendency to improvement. Conversely, the performance patterns of the RA-based $PSC_{\chi 2}$ and the RA-based PSC_{IG} are characterized by a high and wavy curvature, particularly when the number of selected terms is lower than 100 terms. Moreover, the performance of the RA-based PSC_{MI} is the lowest, whereas the performance of the RA-based PSC_{IG} is the highest, particularly when the number of selected terms is less than 150.

6.4. RA-based PSC approach versus ML approaches

In order to evaluate the performances of RA-based PSC approaches, the following ML techniques have been adopted:

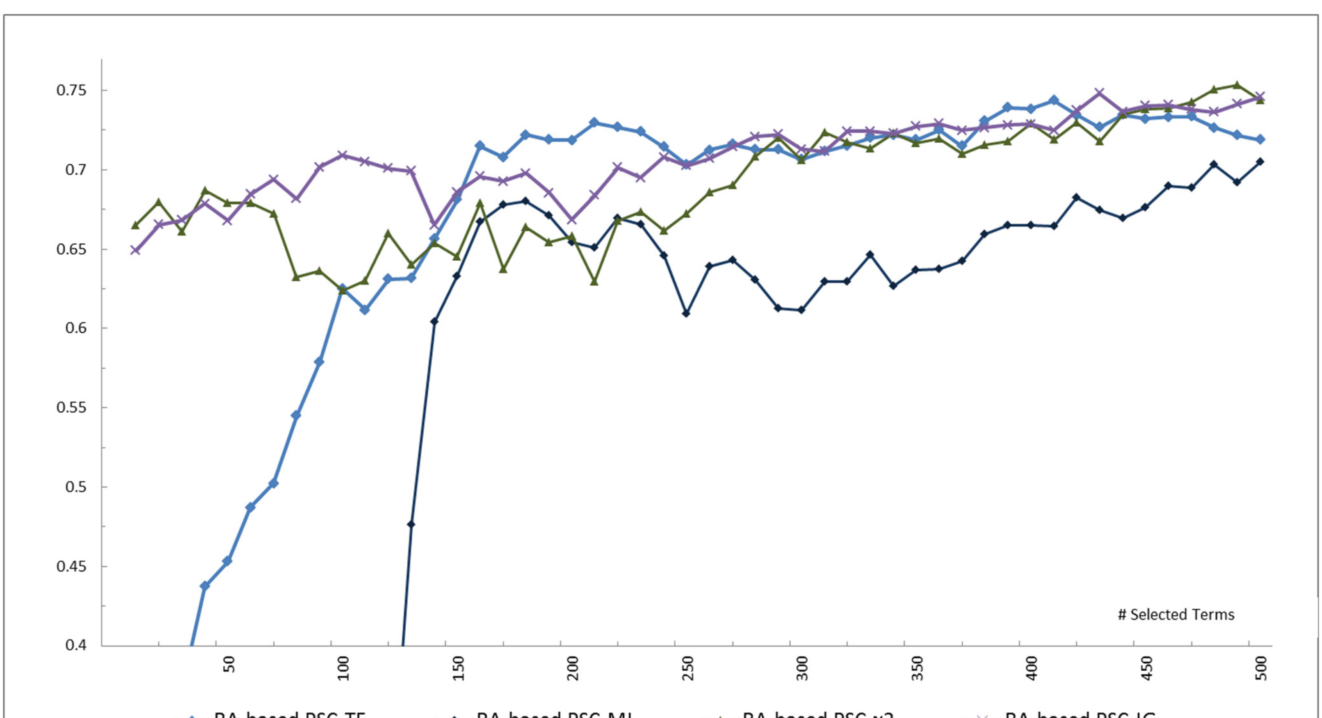

Fig. 5. Performances of the RA-Based PSC Variants.

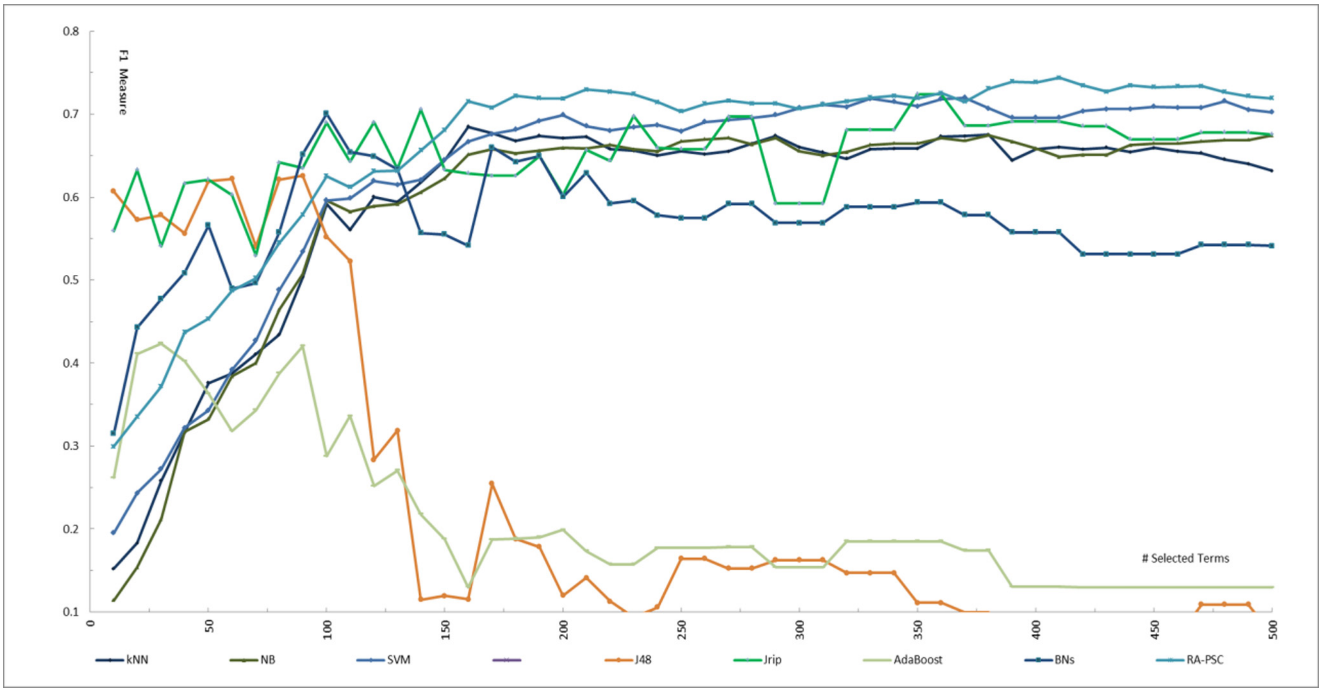

Fig. 6. Performances of the PSC-based approaches in comparison with those of the ML approaches.

k-nearest neighbor (kNN), Naïve Bayes (NB), Support Vector Machine (SVM), decision tree algorithm (J48), a rule based ML algorithm, (RIPPER, JRip), Adaptive Boosting method (AdaBoost), and Bayesian Networks (ByesNet). Fig. 6 presents the performances of RA-based PSC approach and the ML approaches in terms of Macro-Average F1 for all experimental cases. It can be seen that most of the ML approaches except J48, AdaBoost, and JRip have similar performance patterns. It is also clear that the performance of J48 and AdaBoost deteriorates sharply as the number of selected terms increases, whereas, the performance of JRip tends to be unstable. Furthermore, the performances of SVM is comparable with the RA-based PSC when the number of selected terms falls in the range of 260–370, whereas, for most of the remaining experimental cases, RA-based PSC approach outperforms SVM. On the other hand, the performance of NB and kNN are comparable and lower than SVM, yet higher than the performance of BayesNet.

Table 5 presents a comparison between RA-based PSC and ML approaches in terms of Average F_1 over all experimental cases for each BCL. The values of the Macro-Average F_1 given in the last row indicate that the average ability of RA-based PSC to build a question classification system is higher than those of ML techniques. Among the ML techniques, the J48 and AdaBoost have the lowest abilities to build a question classification system. The abilities of kNN, NB, and BayesNet are higher than those of the J48 and AdaBoost and the abilities of kNN and NB are comparable. In addition, the ability of the SVM is higher than those of kNN and NB.

On the other hand, Table 6 presents a comparison between RA-based PSC approach and the ML approaches in terms of the best F_1 obtained using specific number of terms for all classifiers. The results demonstrate that the RA-based PSC approach outperforms ML approaches. Among ML approaches, SVM and JRip are comparable while performance of the BayesNet is poorer. Meantime, the performances of the NB and kNN are comparable and fall in between those of the BayesNet and J48. Additionally, the AdaBoost has the lowest performance.

Finally, Table 7 summarizes a comparison between RA-based PSC approach and the ML approaches in terms of the best F_1 obtained using different numbers of terms for each classifier. The comparison revealed that BayesNet performs slightly better than PA-based PSC and JRip, whose performances are comparable. SVM has higher performance than kNN and NB, which themselves have comparable performance. J48 technique exhibited lower performance than kNN and NB, yet better than AdaBoost, which show the lowest performance.

7. Discussion

In this section several aspects of the RA-based PSC model of question classification are discussed in light of the above results.

Table 5
Average Performance of the RA-based PSC approaches vs. the ML approaches.

BCL	Classifier							
	ML Approaches							RA-Based PSC
	kNN	NB	SVM	J48	Jrip	AdaBoost	BayesNet	
Knowledge	0.590	0.461	0.563	0.270	0.734	0.227	0.485	0.660
Comprehension	0.679	0.649	0.696	0.296	0.704	0.317	0.646	0.704
Application	0.455	0.462	0.481	0.244	0.508	0.218	0.586	0.553
Analysis	0.730	0.797	0.803	0.433	0.746	0.279	0.740	0.785
Synthesis	0.552	0.578	0.662	0.177	0.595	0.230	0.583	0.639
Evaluation	0.559	0.609	0.561	0.257	0.587	0.124	0.379	0.634
Macro- Average F_1	0.594	0.593	0.628	0.280	0.646	0.233	0.570	0.663

Table 6
The Best Macro-Average F1 for RA-based PSC and ML approaches.

BCL	Classifier							
	kNN using 160 terms	NB using 380 terms	SVM using 370 terms	J48 using 90 terms	JRip using 360 terms	AdaBoost using 30 terms	BayesNet using 100 terms	RA-based PSC using 410 terms
Knowledge	0.667	0.489	0.588	0.786	0.800	0.513	0.776	0.690
Comprehension	0.733	0.719	0.778	0.678	0.780	0.643	0.742	0.776
Application	0.511	0.558	0.591	0.579	0.700	0.244	0.595	0.627
Analysis	0.868	0.852	0.909	0.607	0.769	0.417	0.821	0.881
Synthesis	0.717	0.653	0.745	0.577	0.642	0.341	0.667	0.721
Evaluation	0.612	0.778	0.708	0.528	0.655	0.383	0.604	0.767
Average F_1	0.685	0.675	0.720	0.626	0.724	0.423	0.701	0.744

Table 7
Comparison of the Best Classifier for each BCL between RA-based PSC and ML approaches.

Classifier	ML Approach														RA-Based PSC	
	kNN		NB		SVM		J48		JRip		AdaBoost		BayesNet			
BCL	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms	Best F_1	No. of Terms
Know	0.667	150	0.640	140	0.692	120	0.786	90	0.836	140	0.588	20	0.776	90	0.742	140
Comp	0.786	390	0.742	420	0.808	460	0.852	30	0.780	40	0.807	20	0.813	30	0.794	460
Appl	0.578	340	0.591	330	0.591	330	0.651	10	0.732	340	0.564	10	0.744	340	0.654	160
Anls	0.873	190	0.909	260	0.909	320	0.758	100	0.833	100	0.586	10	0.862	210	0.9	390
Synt	0.737	130	0.704	170	0.769	210	0.618	80	0.737	110	0.390	80	0.704	140	0.758	150
Eval	0.750	290	0.807	270	0.750	440	0.642	10	0.714	140	0.704	90	0.759	150	0.779	390
Average	0.732	248	0.732	256	0.753	313	0.718	53	0.772	145	0.607	38	0.776	160	0.771	282

In the first place, the obtained results indicates that the standard PSC in not effective for high dimensional data classification, and therefore, a specialized mechanisms such as RA-based initialization mechanism, need to be integrated in the PSC model of data classification. In fact, there are four factors that might have contributed exclusively or in combination to the poor performance of the standard PSC in the current high dimensional space: The first factor is the sparseness of the high dimensional search space, which make the distance measures between the PSC particles in the whole dimension space meaningless (Richards and Ventura, 2004). Although increasing the swarm size was proposed to help PSC to overcome the sparseness problem, it has been reported in Nouaouria and Boukadoum (2009) that increasing the size of the swarm cannot remedy the problem when the computational budget is fixed. The second factor is the tendency of the particles to converge to non-optimal solution as the space dimensions increases, which is attributed to the position updating mechanism which takes place in all dimensions simultaneously (Hendtlass, 2009). The third factor is the high sensitivity of the PSC to its control parameters settings such as inertia weight, w, and learning factors, c1 and c2, where slightly different settings may lead to substantially different performance (Rezaee Jordehi and Jasni, 2013). Two issues that exacerbate the effects of this factor is the absence of a formal analytical model to fine-tune the control parameters values and the unsuitability of the parameters that work well for the low dimensional space for the high dimensional spaces (Wang, 2011). The fourth factor is the heavy dependence of the PSC performance on the initial swarm (Clerc, 2008; Richards and Ventura, 2004). Although many generic initialization mechanisms have been proposed and shown impressive results, recently it has been reported that the generic initialization mechanisms perform poorly in high dimensional spaces (Kazimipour et al., 2014, 2013). Although in this research, a specialized initialization mechanism is proposed to enable the PSC to mitigate the adverse effects of curse of dimensionality, other ideas can be also investigated such as controlling the search behaviour of the PSC in a way that guide it towards promising region of the search space.

Secondly, the results of the RA-based PSC with the four feature selection approaches, Table 4, shed light on the role of these approaches for the proposed RA-based PSC model of questions classification. The variations in the performances of the four RA-based PSC variants are a direct result of the variation in the employed term selection mechanisms. In specific, each approach uses a different metric to determine the relationship between a given term and BCL, and, therefore, different ranked lists of terms are generated. In each experimental case, the top k terms from each ranked lists are used to represent data instances in the subsequent PSC learning stage, and consequently, different PSC performances are obtained

Thirdly, the performance of RA-based model of questions classification can be considered promising and competitive as compared to the best ML approach. More precisely, the performance can be evaluated based on the classification accuracy and the computation time. Concerning the classification accuracy in terms of F_1-measure, three forms of F_1-measure are adopted, that are: the average F_1 performance over all experimental cases, the best F_1 obtained using certain number of terms for all classifier, and the best F_1 performance obtained using different number of terms for each classifier. The results indicate that the PSC outperforms the ML approaches in the first and second form of F_1-measure (Table 5,6) and also in the third form (Table 7) with exception to Bayesian Network (BayesNet) and JRip classifiers where PSC performs competitively with them.

In order to confirm the above conclusions statistically, a test of the statistical significance, a t-test (Two Paired Samples, 2-tails) using 95% confidence interval, has been conducted between RA-based PSC approach and each ML approaches. In each test, the null hypothesis is that there is no differences between the two approaches in the Macro-Average F_1 over all experimental cases. The tests results are presented in Table 8. For each test case, the mean and standard deviation of the Macro-Average F_1 data of each approach and the differences between them are given, followed by the results of the test, which are found in the values of "t", "t-crit", and "P value", and finally the result of the test is given in "sig" in

Table 8
t-Test of statistical Significance.

Pair of Approaches	Group 1		Group 2		Difference			t-test Results			
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev	Std Err	t	p-value	t-crit	Sig
RA-based PSC & kNN	0.594	0.133	0.663	0.113	−0.068	0.028	0.004	−17.501	1.01E-22	2.010	Yes
RA-based PSC & NB	0.593	0.142	0.663	0.113	−0.070	0.034	0.005	−14.698	1.37E-19	2.010	Yes
RA-based PSC & SVM	0.628	0.139	0.663	0.113	−0.035	0.030	0.004	−8.234	8.4E-11	2.010	Yes
RA-based PSC & J48	0.23	0.197	0.663	0.113	−0.433	0.302	0.0428	−10.115	1.4E-13	2.010	Yes
RA-based PSC & Jrip	0.656	0.043	0.663	0.113	−0.021	0.072	0.010	−2.030	0.048	2.012	Yes
RA-based PSC &AdaBoost	0.209	0.090	0.663	0.113	−0.454	0.196	0.028	−16.396	1.6E-21	2.010	Yes
RA-based PSC & BN	0.568	0.062	0.663	0.113	−0.095	0.095	0.013	−7.080	5.0E-09	2.010	Yes

the last column. As shown in table 8, in all t-test cases, there is statistical significance between the two approaches, because P-value is less than 0.05 and the t value is less than t-crit. These results suggest the rejection of the null hypothesis, which means that there are differences between these approaches.

The above comparison between RA-based PSC approach and ML approaches provides experimental evidences on the effectiveness of RA-based PSC approach in solving the questions classification task. Obviously, the proposed initialization mechanism plays a critical role in improving the performance of PSC for high dimensional data classification. As regards ML approaches, the variations in their performances are ascribed to the underlying mechanism used by each approach to overcome the curse of dimensionality problem. The performance of the k-NN approach, which classifies a new object by examining the class values of the k most similar data instances, is affected by two main factors: the distance concentration and hubness of the search space. The distance concentration problem refers to the tendency of distances between all pairs of points in high dimensional space to become almost equal and, therefore, the meaningfulness of finding the nearest neighbours in high dimensional spaces (Aggarwal et al., 2001). Hubness of the search space refers to the skewness of k-occurrence of the number of times a point appears among the k nearest neighbours of other points in a data set (Radovanović et al., 2009). As the dimensionality increases, this distribution becomes considerably skewed and hub points emerge (points with very high k-occurrences). It is an inherent property of high dimensional vector spaces, and its influence on applications based on measuring distances in vector spaces is highly passive.

NB approach mitigates the effects of dimensionality by making a conditional independence assumption that dramatically reduces the number of parameters to be estimated, however, its performance is negatively affected by this assumption, because it seldom holds in practice. In fact, it has been shown that this assumption of NB is only a sufficient but not a necessary condition for the optimality of the NB (Domingos and Pazzani, 1997). Although, BayesNet relaxes the conditional independence assumption, its performance in high dimensional applications is affected by the requirement of an initial knowledge of many probabilities (Mittal and Cheong, 2004). With regard to the SVM, although it can bypass the curse of dimensionality by providing a way of controlling model complexity independent of the dimensionality, increasing the dimensionality affects its performance in many practical cases. For example, the characteristics of the data set (i.e. if the number of dimensions is much greater than the number of data samples) and the selection of SVM parameters (kernel function and its parameters, and the margin parameter, C) are very serious problem in high dimensional data (Vapnik, 2013).

On the other hand, the poor performance of AdaBoost in the questions classification problem is expected, because AdaBoost was previously reported to perform poorly with high dimensional data (Lusa, 2015; Dudoit et al., 2002; Schapire, 2003). As reported in Lusa (2015), when it is easy to overfit the training data with the base classifier, AdaBoost performs exactly as its base classifier, which can explain its poor performance in high dimensional data classification. The poor performance of J48 approach is ascribed to the strict hierarchical partitioning of the data used by J48 as a decision tree algorithm, which imparts disproportionate importance to some features and a corresponding inability to effectively leverage all the available features (Aggarwal and Reddy 2013).

Another aspect of comparison between the RA-based PSC approach and ML approaches is the computation time efficiency (learning time and classification time). In terms of classification time they are comparable, whereas, in terms of learning time, Table 9 reports the learning times for several experimental cases (No. of selected terms = 100, 200, 300, 400, and 500) as measured in C++ clock function of time. The very long learning time required by the RA-based PSC approach compared with ML approaches is noticeable. However, in most practical cases, the learning time is not of crucial, and it may last up to hours for most applications, because it is not achieved in real time. Nonetheless, for some real-time applications, it is of key importance and in these cases, explicit parallelization of the inherent distributed system may overcome the challenge.

Finally, an important aspect of models in educational context is the model interpretability, where the model should be transparent to the learner (O'Shea et al., 1984). In this respect, the all ML approaches, except SVM, offer more or less understandable models, especially NB, which has a comprehensive visual representation, whereas, RA-based PSC model is classified as black-box approach, which means that it has low interpretability. Nonetheless, the results of the RA-based PSC model of questions classification raises several interesting insights that deserves further investigations by practitioners from education perspective. In particular, it indicates that the performance of the classification model is highly dependent on the questions domain for which the model is developed. It is because the classification of questions depends on superficial linguistic analysis of the questions, which induce a set of keywords

Table 9
Execution Time for some Experimental Cases.

No. of Terms	kNN	NB	SVM	J48	JRip	Ada Boost	Bayes Net	RA-Based PSC
100	0.141	0.109	0.114	0.48	1.56	0.59	0.2	283.62
200	0.234	0.171	0.177	0.58	3.79	1.17	0.2	560.76
300	0.28	0.156	0.187	0.86	5.15	1.25	0.34	853.42
400	0.328	0.187	0.234	1.44	6.3	1.64	0.47	1139.47
500	0.609	0.234	0.338	1.2	6.76	1.62	0.39	1420.23

Table 10
Performance of Rule-based Approach.

BCL	Best F_1
Knowledge	0.466
Comprehension	0.627
Application	0.554
Analysis	0.509
Synthesis	0.558
Evaluation	0.552
Average	0.544

for each BCL class from the dataset in the training phase of the model. This is in contrast to the common thought that the Bloom's taxonomy provides a general set of keywords and action verbs for each cognitive level, which can be used to classify a question regardless of its domain. In order to verify this insight, additional experiments have been conducted to automatically classify the question data set using a set of rules in a form of *if (t1* v *t2* ... v *tn) then BCLi*,where the t_1, t_2, ... t_n are a set of action verbs recommended by Bloom's taxonomy for *BCLi*. The results of this experiments, shown in Table 10, suggest that the development of the question classification system based on Bloom's taxonomy should be context-dependent. This particularly emphasize the importance of using automatic methods to discover the discriminative terms and action verbs for each BCL, instead of depending on pre-specified context-independent sets.

Another insight is drawn from the experimental results of RA-based PSC, Fig. 4, is the variation in the classification accuracy among different BCLs. It is the highest for the Analysis BCL, and the lowest for the Application BCL, however, it is comparable for the pairs Comprehension-Evaluation, and Knowledge-Synthesis BCLs. Experimentally, this variations is attributed to the variation in the characteristics of the questions instances for each BCL. In other words, the availability of the discriminative terms for a given BCL affects the ability of the RA-based PSC to find its optimal classifier. However, from education point of view this may add a new dimension, by which the BCLs can be interrelated, in addition to the hieratical interrelation proposed by Bloom's Taxonomy.

8. Conclusion

This research proposes a swarm intelligence-based approach, namely particle swarm classification, to design a classification model for classifying educational questions into the six cognitive levels of Bloom's taxonomy. Based on a deep analysis of several aspects of the high dimensional search space and its effects on the performance of the PSC, it proposes a Rocchio Algorithm-based as an initialization mechanism to mitigate the curse of dimensionality effects by initializing the PSC particles at a promising region of the search space. The experimental results confirm the key role of the RA-based initialization mechanism for improving the performance of PSC for high dimensional data classification. The results also demonstrate that feature selection approaches play a role in the performance of the PSC. Moreover, a comparison between PSC and seven ML approaches has led to the following interesting conclusions:- First, the RA-based PSC approach is more effective than ML techniques for the high dimensional questions classification. Second, although the learning time of RA-based PSC is much longer than ML techniques, they are comparable in classification time. Finally, although the RA-based model of questions classification corresponds to lower model interpretability, it has raised questions concerning the set of keywords of each Bloom's cognitive levels and the interrelation among them, which need further investigation from educational perspective.

References

Abduljabbar, D.A., Omar, N., 2015. Exam Questions Classification Based on Bloom's Taxonomy Cognitive Level Using Classifiers Combination. J. Theor. Appl. Inf. Technol. 78 (3), 447.
Abraham, A., Grosan, C., Ramos, V., 2007. Swarm Intelligence in Data Mining, Vol. 34. Springer.
Aggarwal, C.C., Reddy, C.K., 2013. Data Clustering: Algorithms and Applications. Chapman and Hall/CRC.
Aggarwal, C.C., Hinneburg, A., Keim, D.A., 2001. On the surprising behavior of distance metrics in high dimensional space. In: International Conference on Database Theory. Springer, pp. 420–434.
Baker, R.S., Yacef, K., 2009. The state of educational data mining in 2009: A review and future visions. JEDM J. Educational Data Min. 1 (1), 3–17.
Bloom, B., Krathwohl, D., Masia, B., 1984. Bloom taxonomy of educational objectives. Allyn and Bacon, Boston, MA. Online at http://www.coun.uvic.ca/learn/program/hndouts/bloom.html.
Cervantes, A., Galván, I.M., Isasi, P., 2009. AMPSO: a new particle swarm method for nearest neighborhood classification. IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) 39 (5), 1082–1091.
Chein, B., Liau, S., 2004. An automatic classifier for chinese items categorization. In: the National Conference on Artificial Intelligence and Its Application, Taiwan.
Chen, W., Sun, J., Ding, Y., Fang, W., Xu, W., 2008. Clustering of gene expression data with quantum-behaved particle swarm optimization. New Front. Appl. Artif. Intell., 388–396
Chen, S., Montgomery, J., Bolufé-Röhler, A., 2015. Measuring the curse of dimensionality and its effects on particle swarm optimization and differential evolution. Appl. Intell. 42 (3), 514–526.
Cheng, S.-C., Huang, Y.-M., Chen, J.-N., Lin, Y.-T., 2005. Automatic leveling system for e-learning examination pool using entropy-based decision tree. In: International Conference on Web-Based Learning. Springer, pp. 273–278.
Clerc, M., 2008. Initialisations for particle swarm optimisation. Online at http://clerc.maurice.free.fr/pso.
Clerc, M., Kennedy, J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. 6 (1), 58–73.
De Falco, I., Della Cioppa, A., Tarantino, E., 2005. Evaluation of particle swarm optimization effectiveness in classification. In: International Workshop on Fuzzy Logic and Applications. Springer, pp. 164–171.
De Falco, I., Della Cioppa, A., Tarantino, E., 2007. Facing classification problems with particle swarm optimization. Appl. Soft Comput. 7 (3), 652–658.
Domingos, P., Pazzani, M., 1997. On the optimality of the simple Bayesian classifier under zero-one loss. Mach. Learn. 29 (2), 103–130.
Dubey, M., Goyal, V., 2016. Classifying stack overflow questions based on bloom's taxonomy Msc. thesis. Indraprastha Institute of Information Technology Delhi, India..
Dudoit, S., Fridlyand, J., Speed, T.P., 2002. Comparison of discrimination methods for the classification of tumors using gene expression data. J. Am. Stat. Assoc. 97 (457), 77–87.
Engelbrecht, A.P., 2006. Fundamentals of Computational Swarm Intelligence. John Wiley and Sons.
Eric, M., Klimovic, A., Zhong, V., 2014. #ML #NLP: Autonomous Tagging of Stack Overflow Questions. Stanford University, Online at. ~meric/files/cs229.pdf" xlink:type="simple" id="ir015">http://stanford.edu/~meric/files/cs229.pdf.
Fattah, S., Tanalol, S.H., Mamat, M., 2007. Classification of Examination Questions Difficulty Level Based on Bloom's Taxonomy. In: Regional Conference on Computational Science and Technology, Kota Kinabalu, Sabah, (pp. 204–207).
Fei, T., Heng, W.J., Toh, K.C., Qi, T., 2003. Question classification for e-learning by artificial neural network. Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia, Vol. 3. IEEE, pp. 1757–1761.
Galavotti, L., Sebastiani, F., Simi, M., 2000. Experiments on the use of feature selection and negative evidence in automated text categorization. In: ECDL-00, 4th European Conference on Research and Advanced Technology for Digital Libraries, Lisbon, Portugal, 2000, (PP. 59–68).
Grosan, C., Abraham, A., Chis, M., 2006. Swarm intelligence in data mining. Swarm Intell. Data Min., 1–20
Hackbarth, A.J., 2005. An Examination of Methods for Analyzing Teacher Classroom Questioning Practices. J. Res. Teach. Educ. 2005, 34–51.
Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H., 2009. The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter 11 (1), 10–18.
Hämäläinen, W., Vinni, M., 2010. Classifiers for educational data mining. Handbook Educational Data Min., 57–74
Haris, S.S., Omar, N., 2013. Determining Cognitive Category of Programming Question with Rule-based Approach. Int. J. Inf. Process. Manage. 4 (3), 86–95.
Haris, S.S., Omar, N., 2015. Bloom's Taxonomy Question Categorization using Rules And N-Gram Approach. J. Theor. Appl. Inf. Technol. 76 (3).
Hasanzadeh, M., Meybodi, M.R., Ebadzadeh, M.M., 2013. Adaptive cooperative particle swarm optimizer. Appl. Intell. 39 (2), 397–420.
Hatanaka, T., Korenaga, T., Kondo, N., Uosaki, K., 2009. Search performance improvement for pso in high dimensional space. In: Particle Swarm Optimization. Aleksandar Lazinica (Ed.).

Hendtlass, T., 2009. Particle swarm optimisation and high dimensional problem spaces. In: Evolutionary Computation, 2009. CEC'09. IEEE Congress on, (pp. 1988–1994).

Ince, I., 2008. Intelligent Question Classification for E-Learning Environments by Data Mining Techniques Master Thesis. Institute of Science, Computer Engineering, Bahcesehir University, Istanbul, Turkey.

Jabeen, H., Jalil, Z., Baig, A.R., 2009. Opposition based initialization in particle swarm optimization. In: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers, (pp. 2047–2052).

Jayakodi, K., Bandara, M., Meedeniya, D., 2016. An automatic classifier for exam questions with WordNet and Cosine similarity. In: Moratuwa Engineering Research Conference (MERCon), 2016, (pp. 12-17).

Kalyani, S., Swarup, K.S., 2011. Classifier design for static security assessment using particle swarm optimization. Appl. Soft Comput. 11 (1), 658–666.

Karahoca, D., Karahoca, A., Erdogdu, B., Uzunboylu, H., Güngör, A., 2009. Computer Based Testing for E-learning: Evaluation of Question Classification for Computer Adaptive Testing. In: the 5th E-learning Conference, Berlin, Germany.

Kavitha, R., Vijaya, A., Saraswathi, D., 2012. A two-phase item assigning in adaptive testing using norm referencing and bayesian classification. In: Advances in Computer Science, Engineering and Applications (pp. 809–816).

Kazimipour, B., Li, X., Qin, A.K., 2013. Initialization methods for large scale global optimization. In: Evolutionary Computation (CEC), 2013 IEEE Congress on, 2013 (pp. 2750–2757).

Kazimipour, B., Li, X., Qin, A.K., 2014. Why advanced population initialization techniques perform poorly in high dimension? In: SEAL, (pp. 479–490).

Koedinger, K., Cunningham, K., Skogsholm, A., Leber, B., 2008. An open repository and analysis tools for fine-grained, longitudinal learner data. In: The 1st International Conference on Educational Data Mining, Montreal, pp. 157–166.

Levin, T., Long, R., 1981. Effective Instruction. ASCD, Washington, DC.

Li, X., Yao, X., 2009. Tackling high dimensional nonseparable optimization problems by cooperatively coevolving particle swarms. In Evolutionary Computation, 2009. CEC'09. In: IEEE Congress on, (pp. 1546–1553).

Li, Y., Xiang, R., Jiao, L., Liu, R., 2012. An improved cooperative quantum-behaved particle swarm optimization. Soft. Comput. 16 (6), 1061–1069.

Liu, R., Sun, X., Jiao, L., 2010. Particle swarm optimization based clustering: a comparison of different cluster validity indices. Life Syst. Model. Intell. Comput., 66–72

Lusa, L., 2015. Boosting for high-dimensional two-class prediction. BMC Bioinformatics 16 (1), 300.

Ma, Z., Vandenbosch, G.A., 2012. Impact of random number generators on the performance of particle swarm optimization in antenna design. In: Antennas and Propagation (EUCAP), 2012 6th European Conference on, (pp. 925–929).

Martens, D., Baesens, B., Fawcett, T., 2011. Editorial survey: swarm intelligence for data mining. Mach. Learn. 82 (1), 1–42.

Mitchell, T.M., 1997. Machine Learning. McGraw Hill.

Mittal, A., Cheong, L.-H., 2004. Addressing the problems of Bayesian network classification of video using high-dimensional features. IEEE Trans. Knowledge Data Eng. 16 (2), 230–244.

Mladenic, D., 1998. Feature subset selection in text learning. In ECML-98. In: 10th European Conference on Machine Learning. Chemnitz, Germany, (pp. 95–100).

Nebti, S., Boukerram, A., 2010. Handwritten digits recognition based on swarm optimization methods. Networked Digital Technol., 45–54

Ng, E., Lim, M., Maul, T., Lai, W., 2009. Investigations into particle swarm optimization for multi-class shape recognition. Adv. Neuro-Inf. Process., 599–606

Nouaouria, N., Boukadoum, M., 2009. A particle swarm optimization approach for substance identification. In: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, (pp. 1753–1754).

Nouaouria, N., Boukadoum, M., 2010. Particle swarm classification for high dimensional data sets. In Tools with Artificial Intelligence (ICTAI). In: 2010 22nd IEEE International Conference on, (Vol. 1, pp. 87–93).

Nouaouria, N., Boukadoum, M., Proulx, R., 2013. Particle swarm classification: a survey and positioning. Pattern Recognition 46 (7), 2028–2044.

Nuntiyagul, A., Naruedomkul, K., Cercone, N., Wongsawang, D., 2008. Adaptable learning assistant for item bank management. Comput. Education 50 (1), 357–370.

Omar, N., Haris, S.S., Hassan, R., Arshad, H., Rahmat, M., Zainal, N.F.A., et al., 2012. Automated analysis of exam questions according to Bloom's taxonomy. Proc. Soc. Behavioral. Sci. 59, 297–303.

Omran, M., Al-Sharhan, S., 2007. Barebones particle swarm methods for unsupervised image classification. In: Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, (pp. 3247–3252).

Omran, M.G., Al-Sharhan, S., 2008. Using opposition-based learning to improve the performance of particle swarm optimization. In Swarm Intelligence Symposium, 2008. SIS 2008. IEEE, (pp. 1–6).

O'Neill, M., Brabazon, A., 2006. Self-organizing swarm (SOSwarm): a particle swarm algorithm for unsupervised learning. In: Evolutionary Computation, 2006. CEC 2006. IEEE Congress on, (pp. 634–639).

Ornstein, A.C., 1987. Questioning: The essence of good teaching. NASSP Bull. 71 (499), 71–79.

O'Shea, T., Bornat, R., du Boulay, B., Eisenstadt, M., Page, I., 1984. Tools for creating intelligent computer tutors. In Proc. of the international NATO symposium on Artificial and human intelligence, (pp. 181–199).

Owechko, Y., Medasani, S., Srinivasa, N., 2004. Classifier swarms for human detection in infrared imagery. In: Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on, (pp. 121–121).

Pant, M., Thangaraj, R., Grosan, C., Abraham, A., 2008. Improved particle swarm optimization with low-discrepancy sequences. In Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on, (pp. 3011–3018).

Parsopoulos, K., Vrahatis, M., 2002. Initializing the particle swarm optimizer using the nonlinear simplex method. Adv. Intell. Syst. Fuzzy Syst. Evol. Comput. 216, 1–6.

Peña, A., 2014. Educational data mining: A survey and a data mining-based analysis of recent works. Expert Syst. Appl. 41 (4), 1432–1462.

Pena, A., Domínguez, R., Medel, J., 2009. Educational data mining: a sample of review and study case. World J. Educational Technol. 1 (2), 118–139.

Poli, R., Kennedy, J., Blackwell, T., 2007. Particle swarm optimization. Swarm Intell. 1 (1), 33–57.

Porter, M.F., 1980. An algorithm for suffix stripping. Program 14 (3), 130–137.

Radovanović, M., Nanopoulos, A., Ivanović, M., 2009. Nearest neighbors in high-dimensional data: The emergence and influence of hubs. In: Proceedings of the 26th Annual International Conference on Machine Learning, (pp. 865–872).

Rahnamayan, S., Wang, G.G., 2009a. Center-based initialization for large-scale black-box problems. In: The 8th WSEAS international conference on Artificial intelligence, knowledge engineering and data bases, (pp. 531–541).

Rahnamayan, S., Wang, G.G., 2009b. Center-based sampling for population-based algorithms. In: Evolutionary Computation, 2009. CEC'09. IEEE Congress, (pp. 933–938).

Ramsey, I., Gabbard, C., Clawson, K., Lee, L., Henson, K.T., 1990. Questioning: An effective teaching method. The Clearing House 63 (9), 420–422.

Rezaee Jordehi, A., Jasni, J., 2013. Parameter selection in particle swarm optimisation: a survey. J. Exp. Theor. Artif. Intell. 25 (4), 527–542.

Richards, M., Ventura, D., 2004. Choosing a starting configuration for particle swarm optimization. In: IEEE International Joint Conference of Neural, (Vol. 3, pp. 2309–2312).

Rocchio, J.J., 1971. Relevance feedback in information retrieval. The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall, Cambridge, pp. 313–323.

Romero, C., Ventura, S., 2007. Educational data mining: A survey from 1995 to 2005. Exp. Syst. Appl. 33 (1), 135–146.

Romero, C., Ventura, S., 2010. Educational data mining: a review of the state of the art. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) 40 (6), 601–618.

Romero, C., Ventura, S., 2013. Data mining in education. Wiley Interdiscip. Rev. Data Min. Knowledge Discovery 3 (1), 12–27.

Salton, G., 1989. Automatic text processing: The transformation, analysis, and retrieval of. Reading: Addison-Wesley.

Schapire, R.E., 2003. The boosting approach to machine learning: An overview. In: Nonlinear estimation and classification (pp. 149–171).

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surv. (CSUR) 34 (1), 1–47.

Silva, C., Ribeiro, B., 2003. The importance of stop word removal on recall values in text categorization. In: Neural Networks, 2003. Proceedings of the International Joint Conference on, (Vol. 3, pp. 1661–1666).

Sousa, T., Silva, A., Neves, A., 2004. Particle swarm based data mining algorithms for classification tasks. Parallel Comput. 30 (5), 767–783.

Tewolde, G.S., Hanna, D.M., 2007 Particle swarm optimization for classification of breast cancer data using single and multisurface methods of data separation. In: Electro/Information Technology, 2007 IEEE International Conference on, (pp. 443–446).

Tsai, C.-Y., Yeh, S.-W., 2008. A multiple objective particle swarm optimization approach for inventory classification. Int. J. Prod. Econ. 114 (2), 656–666.

Uy, N.Q., Hoai, N.X., McKay, R.I., Tuan, P.M., 2007. Initialising PSO with randomised low-discrepancy sequences: the comparative results. In: Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, (pp. 1985–1992).

Vapnik, V., 2013. The Nature of Statistical Learning Theory. Springer science and business media.

Vesterstrom, J., Thomsen, R., 2004. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In: Evolutionary Computation, 2004. CEC2004. Congress on, (Vol. 2, pp. 1980–1987).

Wang, J., 2011. Particle swarm optimization with adaptive parameter control and opposition. J. Comput. Inf. Syst. 7 (12), 4463–4470.

Wang, X.-X., Ma, L.-Y., 2014. A compact K nearest neighbor classification for power plant fault diagnosis. J. Inf. Hiding Multimed. Signal. Process. 5 (3), 508–517.

Woolf, B.P., Lane, H.C., Chaudhri, V.K., Kolodner, J.L., 2013. AI Grand Challenges for Education. AI Magazine 34 (4), 38–66.

Xu, Y., Chen, L., 2010. Term-frequency based feature selection methods for text categorization. In: Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on, (pp. 280–283).

Xue, B., Zhang, M., Browne, W.N., 2014. Particle swarm optimisation for feature selection in classification: novel initialisation and updating mechanisms. Appl. Soft Comput. 18, 261–276.

Yahya, A.A., El Bashir, M.S., 2014. Applying machine learning to analyse teachers' instructional questions. Int. J. Adv. Intell. Paradigms 6 (4), 312–327.

Yahya, A.A., Osman, A., Ramli, A.R., Balola, A., 2011. Feature selection for high dimensional data: an evolutionary filter approach.

Yahya, A.A., Osman, A., Alattab, A.A., 2013. Educational data mining: A case study of teacher's classroom questions. In Intelligent Systems Design and Applications (ISDA). In: 2013 13th International Conference on, (pp. 92–97).
Yahya, A.A., Osman, A., Taleb, A., 2014. Swarm intelligence in educational data mining. In: The Machine Learning and Data Analytics Symposium MLDAS'2014.
Yang, Y. Pedersen, J.O., 1997. A comparative study on feature selection in text categorization. In: ICML-97, 14th International Conference on Machine Learning, (pp. 412–420).
Yusof, N., Hui, C.J., 2010. Determination of Bloom's cognitive level of question items using artificial neural network. In: Intelligent Systems Design and Applications (ISDA), 2010 10th International Conference on, (pp. 866–870).
Zhang, C., Ni, Z., Wu, Z., Gu, L., 2009. A novel swarm model with quasi-oppositional particle. In Information Technology and Applications, 2009. IFITA'09. International Forum on, (Vol. 1, pp. 325–330).