IJICT International Journal of Information & Communication Technology

ITRC

Volume 3- Number 2- March 2011 (35-45)

Customizing Feature Decision Fusion Model Using Information Gain, Chi-Square and Ordered Weighted Averaging for Text Classification

Mohammad Ali Ghaderi
Control & Intelligent Processing Center of Excellence, School of ECE
University of Tehran
Tehran, Iran
mohghaderi@gmail.com

Behzad Moshiri
Control & Intelligent Processing Center of Excellence, School of ECE
University of Tehran
Tehran, Iran
moshiri@ut.ac.ir

Nasser Yazdani
Control & Intelligent Processing Center of Excellence, School of ECE
University of Tehran
Tehran, Iran
yazdani@ut.ac.ir

Maryam Tayefeh Mahmoudi[1,2]
[1] Knowledge Management & E-Organization Group, IT Research Faculty, Research Institute for ICT
[2] School of ECE, University of Tehran
Tehran, Iran
mahmodi@itrc.ac.ir

Received: December 13, 2010 - Accepted: Feburary 15, 2010

Abstract—**Automatic classification of text data has been one of important research topics during recent decades. In this research, a new model based on data fusion techniques is introduced which is used for improving text classification effectiveness. This model has two major components, namely feature fusion and decision fusion; therefore, it is called Feature Decision Fusion (FDF) model. In the feature fusion component, two well-known text feature selection algorithms, Chi-Square (X^2) and Information Gain (IG) were used; this component applied Ordered Weighted Averaging (OWA) operator in order to make better feature selection. The second component, Decision fusion component, combined two kinds of results using the Majority Voting (MV) algorithm. The results were obtained with feature fusion and without feature fusion. To evaluate the proposed model, K-Nearest Neighbor (KNN), Decision Tree and Perceptron Neural Network algorithms were used for classifying Rueters-21578 dataset documents. Experiments showed that this model can improve effectiveness of text classification in accordance to both Micro-averaged F1 and Macro-averaged F1 measures.**

Keywords- Text classification; text categorization; document classification; document categorization; text feature selection; data fusion

I. INTRODUCTION

As a large amount of data is still in the text format, classification of the existing data can be very helpful in processing the data and improving information retrieval applications such as search engines. Many classification algorithms have been developed for classifying data automatically; although many of these algorithms are acceptably effective, none of them have 100% accuracy. This means that there are always some documents that a classifier cannot assign to proper classes. False classification refers to different issues such as inappropriate feature selection, weakness of classifiers and scarcity of enough training data; this has a direct impact on the performance of a whole system. For example, a text classification algorithm in a retrieval system can decrease the accuracy of retrieved data and fixing wrong classified documents is very difficult and sometimes impossible.

In this paper, a new model based on data fusion techniques, called Feature Decision Fusion Model (FDF), is presented in order to provide better classification results for the existing classifiers. This model combines not only text features for having a better feature selection but also classification results for having better results; thus, it is advantageous in terms of both having better features and combining results from different situations. This leads to the improvement of classification effectiveness without modifying the algorithms.

To evaluate the proposed model, three known classification algorithms, "K-Nearest Neighbor" (KNN), Perceptron Neural Network and Decision Tree, are applied on a well-known standard dataset called Rueters-21578 [8]. In this respect, four different issues are investigated in this research: effectiveness of Ordered Weighted Averaging (OWA) operator for feature fusion, impacts of different weights of OWA on the above-mentioned algorithms, effects of FDF on a single classifier and, finally, effectiveness of FDF model with Decision Fusion (DF) of mentioned classifiers. Macro-averaged F1 (Macro-F1) and Micro-averaged F1 (Micro-F1) are the two measures used in the evaluation process.

Experimental results show that using OWA for feature fusion can outperform Averaging (Avg) and Maximum (Max) operators in many cases; it should, however, be noted that choosing weights for the OWA operator has a great influence on its performance. In addition, the FDF model can increase the effectiveness of a single classifier to approximately 2% using Macro-F1 and Micro-F1 measures. Also, this model can outperform both feature fusion (FF) operators and decision fusion techniques when used by different classifiers.

The rest of this paper is organized as follows: In Section 2, related and previous researches on text feature and decision fusion are briefly discussed. In Section 3, basic definitions and algorithms and formal definition of the problem with respect to the proposed model are presented. The FDF model is also explained in Section 3. The evaluation process of the proposed model and discussion of the experimental results are presented in Section 4. Section 5 states two possible practical usage of the FDF model in addition to their pseudo codes. Finally, the paper concludes in Section 6 by explaining main achievements, limitations and future plans.

II. RELATED WORKS TO TEXT FEATURE SELECTION AND DECISION FUSION

Text classification has been one of the most active research fields since early 60's and has many applications such as automatic indexing for Boolean information retrieval systems, document organization, text filtering, word sense disambiguation and hierarchical categorization of web pages [8].

Since the number of features in the text classification can easily exceed thousands of features, especially in Vector Space Model, researchers have tried to find methods which reduce the number of features while increasing classification quality [1]; it has been shown that feature subset selection not only helps the classifier to avoid over-fitting but also can lead to increased effectiveness [8] [1]. Several methods for feature selection have been proposed so far which include Information Gain(IG) , Chi-square (X^2), Document Frequency (DF), Term Strange (TS) and Mutual Information (MI) algorithms [9] [1] [8]; moreover, there exist some bio-inspired algorithms which can be used for feature selection like ant-colony that optimizes the selection of features [6]. There are some other new research activities with respect to feature selection such as variance-mean-based filtering method [10], Two-stage Feature Selection Method [19], Multi-class Odds Ratio (MOR) and Class Discriminating Measure (CDM) for Naïve bayse classifier [20]. Although there are some feature selectors that outperform others because of their attention to semantic concepts of features and domain of context [11], there are still some opportunities available to obtain better feature selection using the combination of feature selection algorithms. Combining feature selectors can improve their effectiveness to a large extent. The most frequently used feature selection algorithms for combination are IG, X^2 and DF algorithms [4] [5]. As little attention has been given to using data fusion techniques for feature selection, in this paper, there is a focus on OWA as a data fusion operator for this purpose.

Effectiveness of classification depends on not only the features which are selected but also the classification method which is applied in a special content. Classifiers have a great influence on those applications which are dependent on their underlying classification parts [8]. Many classification algorithms have been used in the context of text classification such as Decision Tree (DT), K-Nearest Neighbor (KNN), Naïve Bayse (NB) and Neural Networks, to name few, but none of these algorithms provide a through solution to the problem of classification; therefore, there has been an inclination to combine their results and obtain better results. The idea that each classifier can be like an expert and its results can be considered as the opinion of that expert leads to the creation of a committee of classifiers in order to achieve better effectiveness for the whole classification system. Several methods exist for combining the classification results; the major ones

are: boosting in [13] and [14], OWA operator and Decision Template in [10] and Majority Voting (MV) in [12] and [8].

Previous studies have shown that data fusion methods have positive impacts on the effectiveness of text classification; that is why a model based on data fusion techniques is presented in this paper.

III. THE PROPOSED APPROACH

A. Basics

In this section, definitions of important concepts and algorithms used in this paper are briefly introduced. Most of references used in this section are [1], [2], [8] and [16].

Vector Space Model (VSM) is one of the most famous text representation models; this model is used in this paper as the base model of representing documents. In this model, each document is represented by a vector of its terms. There are several ways to assign weights to these terms (features) [8] like the following one which is also used in this paper:

$$(1+\log(TF))*(N-\log(IDF))$$

N denotes the number of documents, TF denotes frequency of the term and IDF denotes inverse document frequency of that term. Due to different lengths of documents, normalization of the vectors is a usual task in text classification.

Clearly, in a large set of documents, there can be thousands of words; therefore, vectors can have large dimensionality (number of features). Selection between features not only simplifies classification task but also avoids over-fitting of models [8][1]. Information Gain (IG) and Chi-Square (X^2) are two well-known algorithms that are used for feature selection in this paper. IG score of term t can be calculated using the following formula [1]:

$$G(t) = -\sum_{i=1}^{|c|}\Pr(c_i)\log(\Pr(c_i)) + \Pr(t)\sum_{i=1}^{|c|}\Pr(c_i \mid t)\log(\Pr(c_i \mid t)) + \Pr(\bar{t})\sum_{i=1}^{|c|}\Pr(c_i \mid \bar{t})\log(\Pr(c_i \mid \bar{t}))$$

where |c| is the number of classes, *Pr* is a probability function, Pr (c_i | t) calculates the probability of existence of a word in each class, Pr (c_i) is the probability of each class, Pr (t) is the probability of existence of the term and Pr (t□) is the probability of its not existence. These probabilities can be calculated simply by counting the number of documents and classes.

For calculating X^2, at first, it is required that a two-way contingency table be calculated [1]. This table has four squares: A, B, C and D. A is the number of co-occurrence of term t and class c. B is equal to the number of times t seen without c. C is the occurrence of t without c. And, finally, D is the number of times c and t does not occur. The following formula can be used for calculating X^2 score of each term for a specific category:

$$x^2(t,c) = \frac{N\times(AD-CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)}$$

Final X^2 score for each term can be calculated using these formulas:

$$x^2_{avg}(t) = \sum_{i=1}^{|c|}\Pr(c_i)x^2(t,c_i) \quad \text{(Average value)}$$

$$x^2_{\max}(t) = \max_{i=1}^{|c|}\{x^2(t,c_i)\} \quad \text{(Maximum value)}$$

where |c| denotes the number of classes. In this paper, the combination of features was made by combining IG score and maximum value of X^2 score. Furthermore, the values were normalized between 0 and 1 in order to make them suitable for the OWA operator.

Mentioned feature selection algorithms assign scores to the features. It is possible to combine these scores using combination operators like Max operator that selects maximum score and Avg operator that calculates average score.

In this paper, OWA operator was used to combine these scores; OWA operator is a well-known combination operator which is introduced by Yager [2]. For combining data a_1 to a_n (normalized between 0 and 1) using OWA, function F can be identified as follows:

$$F(a_1,a_2,...,a_n) = W_1b_1 + W_2b_2 + ... + W_nb_n$$

$$\sum_i W_i = 1 \quad (0<W_i<1)$$

where W_i is combination weights and b_1 to b_n are descending sorted values of a_1 to a_n. OWA weights vector used in this paper is selected from the set W=[a={1,0.9,0.8,…,0},1-a], for example [0.3, 0.7].

Classification problem can be shown in a formal definition. Assume that set of documents are represented by D={d_1,d_2,…d_N} and set of classes by C={c_1,c_2,…,$c_{|c|}$} where N is the number of documents and |c| is number of classes; assigning all pairs of (d_i,c_i) □ D×C to a Boolean value is a solution of the classification problem. In other words, function Φ: D×C → {True, False} is a classifier function which means that this function identifies whether document j belongs to class i or not. In Single-Label classification, each document is assigned to only one class of C [8]. In Multi-Label classification, each document can belong to more than one class of C [8]. In this paper, the proposed model is evaluated on a Multi-Label class dataset.

B. Feature Decision Fusion Model

In this section, a model from Sensor Data Fusion literature [18] "Feature Decision Fusion Model" (FDF) is proposed to improve text classification effectiveness. The FDF model has been used in different applications such as object identification from different sensors for military purposes where precision and sensitivity are so important.

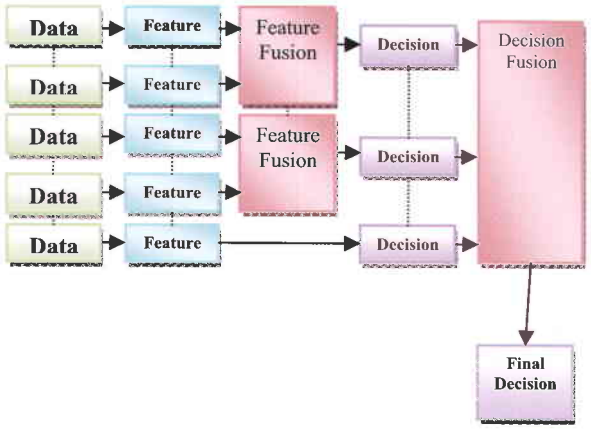

Figure 1. Feature Decision Fusion General Model

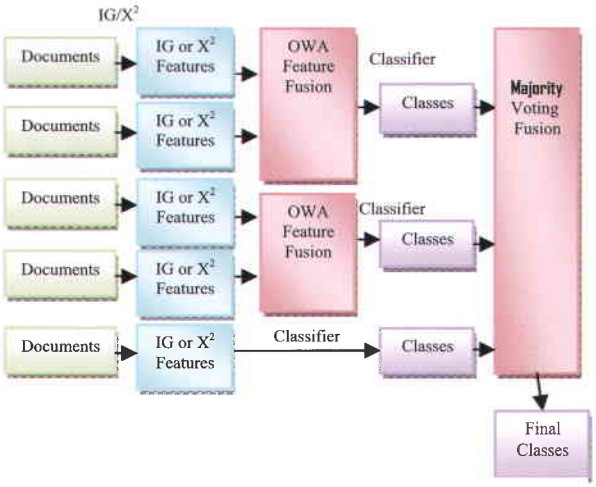

Figure 2. Feature Decision Fusion (FDF) Model for text classification

Figure 1 illustrates the general view of the FDF model that can be customized for many applications as well as text classification.

This model includes two main parts:

- Feature Fusion (FF) which is based on the combination of feature scores; it uses feature selection algorithms like IG and X^2 for selecting text features and a fusion algorithm like Ordered-Weighted-Averaging (OWA) for combining those features.
- Decision Fusion (DF) which is based on the combination of results obtained from different circumstances of the first part using a fusion algorithm like the Majority Voting (MV), OWA and Decision Template algorithm.

In the above schema, processes like feature selection and classification are not illustrated as separate boxes; arrows show these processes.

C. *Customizing Feature Decision Fusion Model for Text Classification*

To customize the FDF model for the purpose of text classification problem, "Data" can be replaced with text documents; "Features" can be replaced with the terms extracted from the text; selecting between features can be done using IG, X^2, Mutual Information (MI) or any other feature selection algorithms; finally, "Decision" can be replaced with classification results.

To select features, IG score and the maximum value of X^2 were used in the present approach. These algorithms were selected because of their wide usage and strong capability in text classification. In addition, combination of these algorithms using Max and Avg operators has been investigated in previous studies, which seems to improve the classification precision [4]. This provides an opportunity for the present researchers to compare their results with those of some previous attempts of feature fusion in this area. Nevertheless, it is still possible to customize the FDF model using other feature selection algorithms.

In "Feature Fusion" component of the proposed model, Ordered-Weighted-Averaging (OWA) operator is used as an operator for combining the feature selection algorithms. OWA operator is a simple and well-known operator that has an acceptable performance in many applications. That is why it was chosen for this part of the FDF model. As there are lots of features in text documents and most of these features are similar to each other, text classification problem differs from machine learning classification; this makes it possible to get completely different feature sets with the same performance in classification; therefore, combining IG and X^2 using the OWA operator despite their different results may lead to a better feature set and classification effectiveness. In addition, previous studies have shown that IG and X^2 cannot outperform each other in all cases; thus, better results are expected from combining them using the OWA operator, which performs well in combining the same level of experts.

In order to evaluate the above model, OWA combination of features was evaluated first to ensure that performance of the whole model was not just the result of the OWA operator. Effectiveness of this operator was investigated in comparison with that of Max and Avg operators, which have been used in previous researches. Also, the impacts of different weights vector for the OWA operator were studied in this research. This study also made it possible to clarify the roles of the OWA operator in final results.

KNN with K=10 (KNN10), Perceptron Neural Network and Decision Tree (DT) algorithms were used as classifiers for identifying document classes. Although different values of K were possible for KNN, the results were produced using K=10 and the performance of KNN using different K values below 10 did not show any considerable change or improvement. To produce results, these classifiers were used with different feature sets selected from the feature selection algorithms like IG and their combinations calculated in the "Feature Fusion" component.

For "Decision Fusion" component, the Majority Voting (MV) algorithm was implemented. This algorithm simply counted the number of voted classes by each classifier. A class should get more than half of classifiers' votes in order to be selected. There were many other operators that might have outperformed Majority Voting (MV) in this application, but the simplicity of this algorithm ensured that the final

outcome would be a result of the FDF model itself and not a complex combination operator in "Decision Fusion" component. This was the main reason of MV selection over other combination methods. In addition, if a simple operator works well in a situation, it is expectable to get even better results using a more complex one.

In order to increase the performance of the MV, it was modified by choosing a class with the highest probability for unclassified documents. By processing obtained results using the MV algorithm, several documents remained unclassified as they did not have gotten the majority of votes in any class. The hypothesis that choosing a class with more votes than others can increase the effectiveness of the MV algorithm motivated the researchers to create MV2. Thus, the modified version of MV is proposed. For this purpose, the most probable class was selected for unclassified documents. Due to pre-assumptions as discussed above (similarity of feature selection algorithms) for a single classifier, the probability of selecting each class was equal and the first one was selected heuristically. To improve MV2, a probability function for selecting the most probable class can be considered in future.

It should be noted that such a customization of the FDF model for all text classification is not optimized. However, this can satisfy the purpose of evaluating the proposed text classification method based on the FDF model. As optimization of this model can vary from one application to another, further studies are required to investigate the optimization of this model.

D. Evaluation Process

The FDF model customized for text classification is illustrated in Figure 2. To show the capability of the FDF model for text classification, it was applied on Rueters-21578 dataset which is a standard and well-known dataset for text classification. A simple tokenizer method was applied to tokenize dataset documents by common separator characters such as space, dot and comma. Porter Stemmer method [16] was used for stemming tokens and extracting features.

After evaluating the OWA operator in feature fusion component, the FDF model was implemented on a single classifier (like KNN10) with different

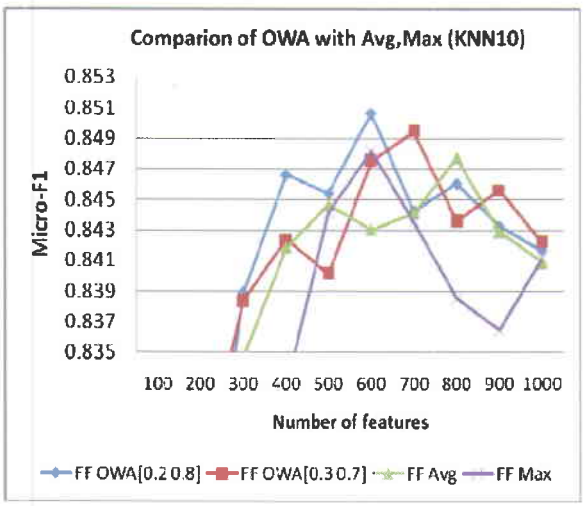

Figure 3. Comparison of OWA with Avg and Max operator using KNN-10 classifier (Micro-F1).

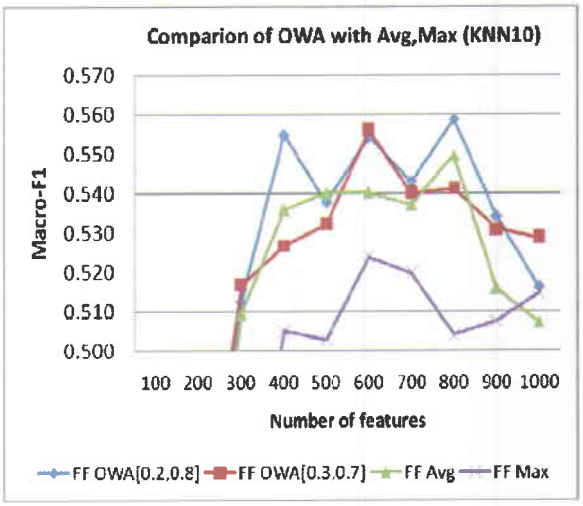

Figure 4. Comparison of OWA with Avg and Max operator using KNN-10 classifier (Macro-F1).

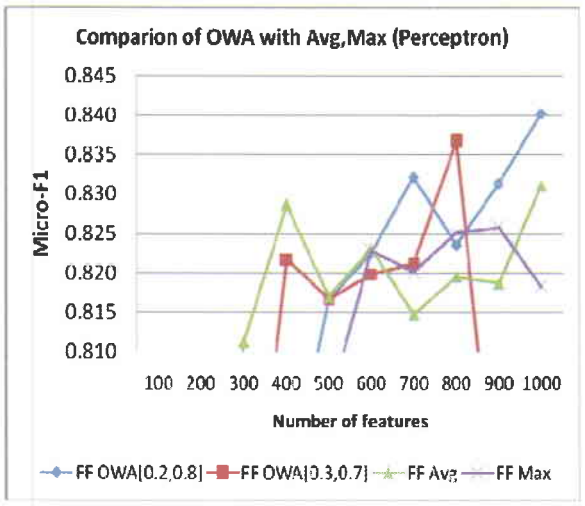

Figure 5. Comparison of OWA with Avg and Max operator using Perceptron classifier (Micro-F1)

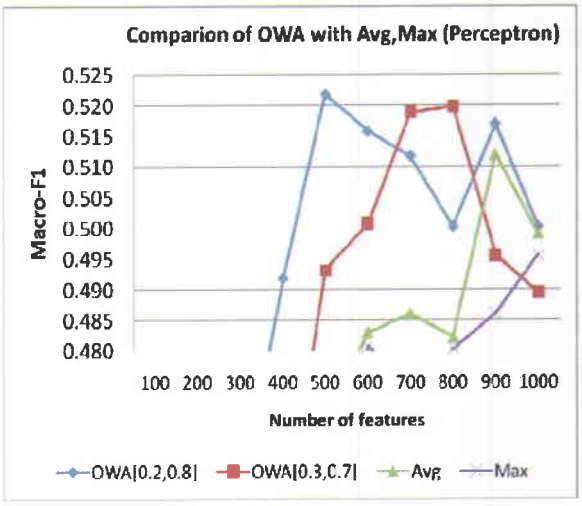

Figure 6. Comparison of OWA with Avg and Max operator using Perceptron classifier (Macro-F1)

feature sets and it was compared with the results obtained from this classifier without using this model. Afterwards, the FDF model was examined with the combination of results obtained from different classifiers.

Finally, Micro-F1 and Macro-F1 measures were applied to evaluate the obtained results. Implementation was done using an open-source text mining library called Java-Bag-Of-Words Library [17].

IV. Analysis of Experimental Results

Evaluation results of the OWA operator for feature fusion in comparison with those of Max, Avg operators are illustrated in Figures 3-6. To show the usability of the FDF model on a single classifier, it was examined using the OWA operator for feature fusion using three classifiers of KNN10 (K-Nearest Neighbor with K=10), Perceptron Neural Network and Decision Tree. The obtained results for Micro-F1 and Macro-F1 measures are illustrated in Figures 7-12. It is followed by examining the FDF model using different classifiers in Figures 13-14. As the evaluation process is based on Micro-F1 and Macro-F1 measure, each result set is depicted in two diagrams: one for Micro-F1 measure and one for Macro-F1 measure.

In the following figures, OWA [x,y] means fusion of X^2 and IG scores using the OWA operator with weights W1=x and W2=y. For instance, "OWA [0.8,0.2]" means feature fusion of IG and X^2Max using the OWA operator with the weights vector [0.8,0.2].

A. OWA Feature Fusion

Results of the OWA operator for feature fusion (FF) compared with those of Avg and Max operators are illustrated in Figures 3-6. Figures 3 and 4 show that the OWA can outperform Max and Avg operators for the KNN classifier in Micro-F1 and Macro-F1 measures. Figures 5 and 6 show that the OWA combination of features can increase both Macro-F1 and Micro-F1 measures for the Perceptron algorithm, too. In addition, these results reveal that the number of features is an important criterion that should be considered for optimizing the overall effectiveness. Moreover, it is clear from these diagrams that different OWA weights can generate different results. More information about the OWA operator for text feature fusion can be found in [21].

In summary, although Max and Avg are special cases of the OWA operator, both Micro-F1 and Macro-F1 measures show that the OWA operator can outperform Max and Avg operators in most cases. It means that this method performs well not only in common classes but also in rare classes.

B. FDF Model using Single Classifier

Results of the FDF model performance on a single classifier are depicted in Figures 7-12. To have a better evaluation, different possible combinations were tried. To reduce the number of possible combinations, combinations of classification results obtained by unequal numbers of features were ignored. For instance, the combination of the results obtained by 200 numbers of IG features with the results achieved by 600 numbers of OWA[0.2,0.8] features were ignored. Therefore, if there are n numbers of feature sets and d numbers of them are considered for combination, there would be $\binom{n}{d}$ different possible combinations. In this paper, combination of results obtained from three different feature sets selected from 11 OWA weights plus IG and X^2 were evaluated; thus, $\binom{13}{3}$ numbers of combination were tested and one of the best combinations was used for depicting the results (similar to the previous work in [22]):

1- X^2Max (without using "Feature Fusion")

2- IG (without using "Feature Fusion")

3- OWA [0.3,0.7] in "Feature Fusion"

Figures 7-12 show that the FDF with the MV2 can outperform feature fusion methods. Also, the MV2 provides better combination results than the MV algorithm in most cases.

The following figures (7 and 8) illustrate the performance of the proposed FDF model on the KNN classifier with K=10.

Figures 9 and 10 illustrate the performance of the FDF model using Perceptron algorithm obtained by the same parameters mentioned for the KNN10 in Figures 7 and 8. As can be seen in Figures 9 and 10, the FDF model can increase the effectiveness of the Perceptron classifier. Figure 10 can clearly reveal the advantage of MV2 over MV algorithm for rare classes (Macro-F1 measure). However, experience has shown that the performance of classification using the MV algorithm is similar to that of the MV2 in Micro-F1 measure. In this measure the MV algorithm even has better performance than the MV2 in some cases.

By replacing KNN10 algorithm with Decision Tree classifier for the evaluation process, Figures 11 and 12 are depicted. These figures show a significant increase of classification effectiveness obtained by applying the FDF model on the Decision Tree algorithm.

Figures 7-12 clearly show that applying the FDF model to a single classifier can provide better classification effectiveness over feature fusion using the OWA operator. In other words, although "Feature Fusion" component of the customized FDF model uses the OWA operator, the achieved performance of this model is much higher than that of the combination of features using that operator.

Effectiveness of data fusion techniques depends highly on the diversity of classification results. Using different features as well as using the best ones can provide appropriate diversity for having a proper fusion via the FDF model.

Although FDF can increase effectiveness of a single classifier, this increment is not very significant for algorithms like KNN and perceptron. There are two main reasons for this issue: first, underlying algorithms have poor performance in some classes; so, their combinations cannot lead to better overall

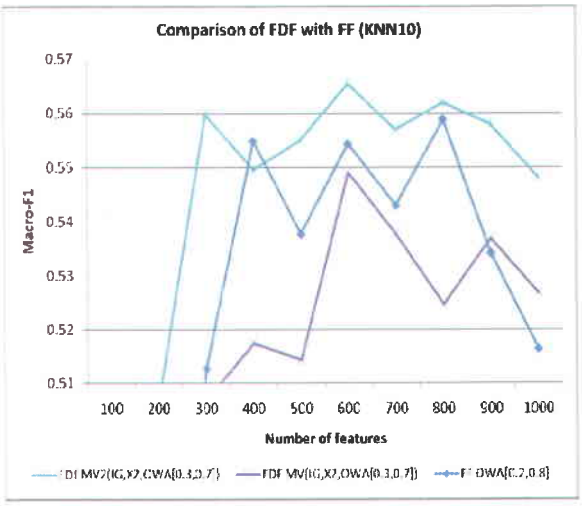

Figure 7. Comparison of the customized FDF model with feature fusion (FF). (Macro-F1)

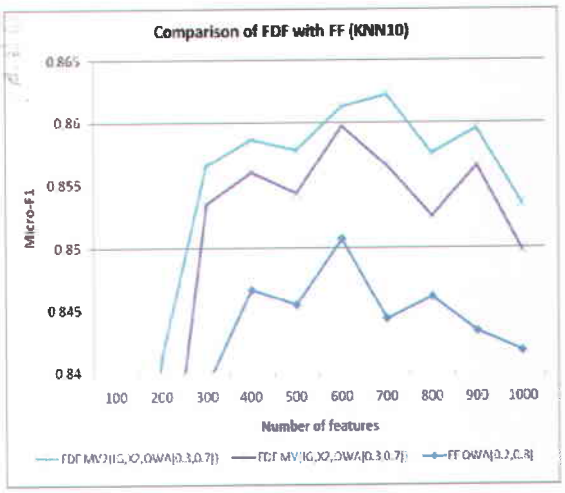

Figure 8. Comparison of the customized FDF model with feature fusion (FF). (Micro-F1)

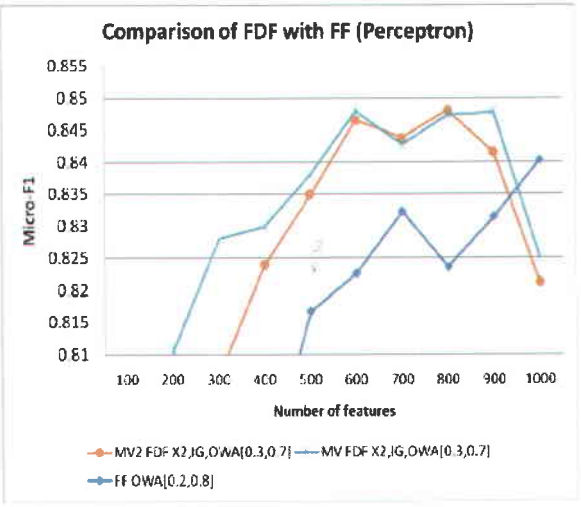

Figure 9. Comparison of the customized FDF model with feature fusion (FF). (Micro-F1)

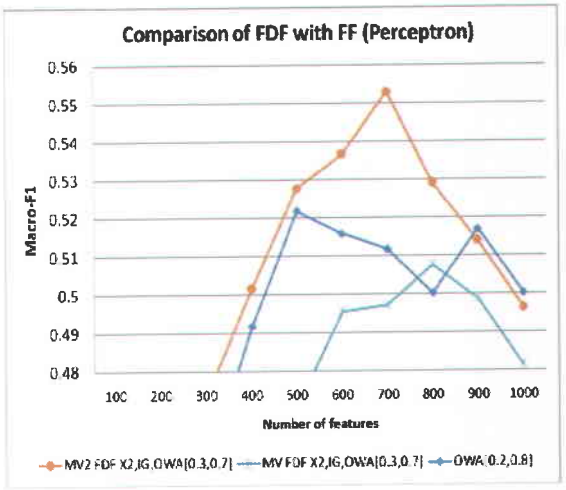

Figure 10. Comparison of the customized FDF model with feature fusion (FF). (Micro-F1)

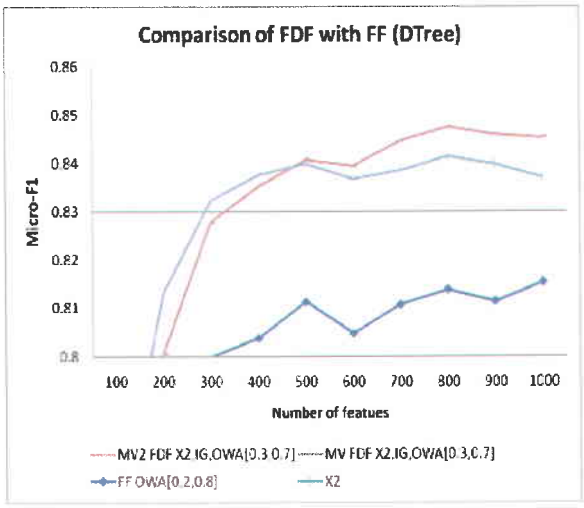

Figure 11. Comparison of the customized FDF model with feature fusion (FF). (Micro-F1)

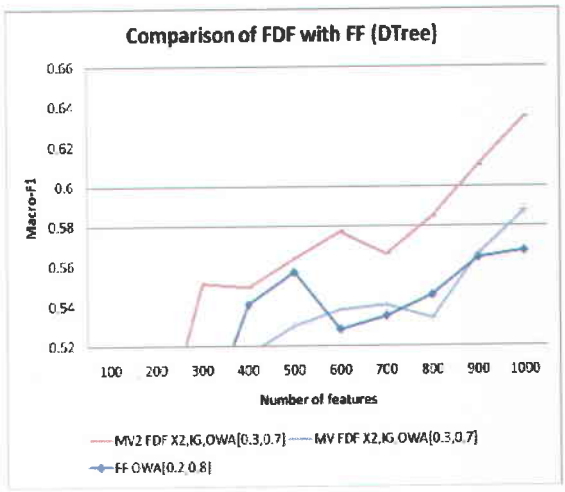

Figure 12. Comparison of the customized FDF model with feature fusion (FF). (Micro-F1)

performance. Second, diversity of their results is not enough for fusion. Although different features can result in diversity of decisions, they cannot hide the role of classification algorithm. For example, for those items which KNN cannot classify in true classes, fusion of KNN results has nothing to say. Therefore, it is logical to expect that not all combinations in the FDF can lead to better classification effectiveness.

What has been discussed in this section was applying the FDF model to a single classifier. Now, it is time to investigate the effectiveness of the FDF model by combining results of different classifiers.

C. *FDF Model using Different Classifiers*

By using the FDF model with a single classifier, the whole classification results depend highly on that classifier. That is why the FDF model was evaluated with different classifiers in this section. As the number of possible combinations here were really high, only the same feature sets (created by feature combination or without feature combination) were considered for all of the three above mentioned algorithms.

"KPT" expression in labels of figures stands for the combination of results of KNN10, Perceptron and Decision Tree. "DF" stands for decision fusion without using the combination of features. For instance, "DF MV2 KPT IG" means using decision fusion of three mentioned algorithms by the modified MV with IG feature selection algorithm.

It is clear that there could be many kinds of combinations for evaluating the FDF model. Although not all combinations have such effectiveness, the above samples reveal that it is possible to get better classification results using this model.

In comparison with older methods, the FDF model provides a better solution. A significant improvement in Macro-F1 measure (about 3%) as well as acceptable improvement in Micro-F1 measure (about 1%) can be seen in Figures 13 and 14.

An important point that should be considered while interpreting results of the FDF model on multiple classifiers is that for those applications that classification of both rare and common classes are important, using the FDF model can significantly increase classification effectiveness. Although it seems that achieved 1% performance in Micro-F1 measure (Figure 13) is not very significant, comparing the performance of the best obtained results without FDF, "DF MV2 KPT IG", in Figure 13, with its performance in Figure 14 discloses that the FDF model can significantly increase classification effectiveness as a whole.

In summary, it was no surprise that using the FDF model on multiple classifiers can lead to better results since it benefits from both feature fusion algorithms and decision fusion ones. That is why even a simple fusion algorithm like MV can perform well in the "Decision Fusion" component.

V. PRACTICAL USAGE

In this paper, a model was proposed based on data fusion techniques and its performance in certain conditions was investigated. Results based on the experiments were illustrated and discussed in the previous section. As mentioned in Section 3, the number of combination situations is high and it is not practical to test all possible situations for building a suitable model. In this section, two practical ways for using the FDF model for both a single classifier and multiple classifiers were proposed.

To use the FDF model in practice, at first, parameters of this model should be found by dividing the train dataset using sampling methods. Percentage of deviation depends on the dataset, but it is obvious that there should be enough samples for this purpose. Sampling method should not remove all instances of a class or make a bias toward one class. In the following approaches, parameters are tried to be found in step one and, in the next step, the final FDF model is built. It is clear that building the final model using updatable classifiers is faster compared with using non-updatable models. In these approaches, however, updatable algorithms and optimization process are not considered for reducing the training time.

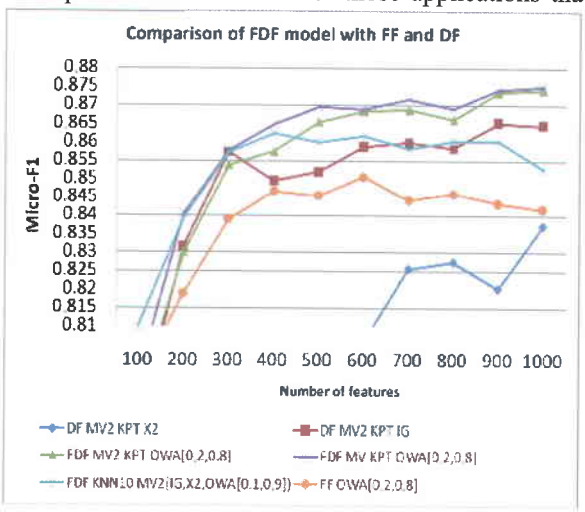

Figure 13. Comparison of customized FDF model with feature fusion (FF). (Micro-F1)

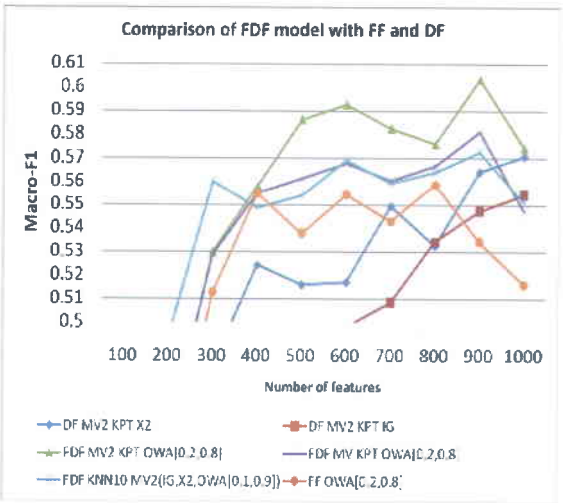

Figure 14. Comparison of customized FDF model with feature fusion (FF). (Macro-F1)

A. Practical Usage of the FDF Model for Single Classifier

Based on obtained results of the FDF model using a single classifier, it was found that the following combinations can outperform others in most of the cases:

1- X^2Max (without using "Feature Fusion")
2- IG (without using "Feature Fusion")
3- OWA with [0.1,0.9], [0.2,0.8] or [0.3,0.7] weights in "Feature Fusion"

Thus, for the practical usage of the FDF model, it is not required to evaluate all combinations of different OWA feature sets; instead, the combination of obtained results using features provided by X^2, IG and the best OWA combination of them is applied. For example, the following provided instance is not required to be tested:

1- OWA [0.1,0.9] in "Feature Fusion"
2- OWA [0.2,0.8] in "Feature Fusion"
3- IG without using "Feature Fusion"

The following pseudo-code is provided to show a possible practical usage of the FDF model for classifying text documents by a single classifier:

```
Step 1: finding parameters
Calculate X^2 and IG values of all features of dataset.
Use suitable number of features for training 13
different classifiers on divided training set.
BestVector = [0,0];
BestMeasureValue = 0;
For (x1 = 0; x1 < 1; x1 += 0.1) {
    Train classifier with OWA [x1, 1-x1] features
    MeasureValue =Evaluate FDF of (X^2,IG and
OWA [x1, 1-x1])
    If (MeasureValue > BestMeasureValue) {
        BestVector = [x1, 1-x1];
        BestMeasureValue = MeasureValue;
        If (MeasureValue is satisfying)
            break;
    }
}
Step 2: Building final model
Use 3 Trained classifiers by these feature sets X^2, IG
and BestVector
Combine results of these classifiers using the FDF
model with a combination algorithm like MV2
```

"Suitable number of features" is different for any dataset; however, it has been found previously that about 3% of features are suitable for most text datasets [4]. In addition, it is possible to investigate this value by training a classifier using different numbers of features like 1%, 2%, …, 5%. Calculation of this value is out of the scope of this research.

Furthermore, because of different usage of classification in different applications, "Measure" can be any measure like Macro-F1 or Micro-F1. To the best knowledge of researchers, the FDF model can significantly increase the effectiveness of those algorithms that are mostly dependent on their feature sets like Decision Tree. Simply, "Measure" is the measure that is going to be achieved.

B. Practical Usage of the FDF Model for Multiple Classifiers

As mentioned before, the number of possible combinations is high when using multiple classifiers. Therefore, the practical usage of the FDF model for multiple classifiers is similar to the way proposed for obtaining experimental results.

The following pseudo-code is provided for showing a possible solution of the FDF model usage for classifying text documents by multiple classifiers:

```
Step 1: finding parameters
Calculate X^2 and IG values of all features of dataset.
Use suitable number of features for training your
classifiers on divided training set.
Select a classifier which has a better performance and
fast enough to build different models.
BestVector = [0,0];
BestMeasureValue = 0;
For (x1 = 0 ; x1 < 1; x1 += 0.1) {
    Train classifier with OWA [x1, 1-x1] features
    MeasureValue =Evaluate FDF of (X2,IG and
OWA [x1, 1-x1]
    If (MeasureValue > BestMeasureValue) {
        BestVector = [x1, 1-x1];
        BestMeasureValue = MeasureValue;
        If (MeasureValue is satisfying)
            break;
    }
}
Step2: Building final model
Train all of your classifiers with BestVector.
Combine results of all classifiers using the FDF
model with a combination algorithm like MV2
```

This pseudo-code calculates the best OWA vector and uses this vector to train classifiers. Finally, results of classifiers are combined with an algorithm like MV. The heuristic idea behind this approach is that OWA provides better feature sets; therefore, each classifier can provide better classification effectiveness and, as a result, their combination increases effectiveness to the point close to the optimum.

Although results obtained from this approach are not optimum, due to what discussed in Section 4, the FDF model has better performance in comparison with simple combination of these classifier results.

Complexity analysis of the FDF model is not rational in general. This model can be customized with different parameters and can utilize different algorithms. As a result, different classifiers and different feature selection algorithms have different complexity. Also, it is possible to customize algorithms and optimize them in order to decrease time complexity of the final implemented FDF model. It is noteworthy to say that creating the FDF model using training data can take longer time compared with a single classifier; however, the final model can work as fast as a normal classifier.

VI. CONCLUSION AND FUTURE WORKS

Achieving higher effectiveness in classification has a strong influence on systems with a classification module. Data fusion techniques can help to have more effective results. In this paper, a data-fusion-based model was proposed for providing a better solution for text classification. This model performs well not only for a single classifier but also for multiple classifiers. Although the FDF model is not a complete model, the results show that it can increase the effectiveness of classifiers like Decision Tree. OWA combination of features obtained by IG and X^2 was also studied in this research. Experiments showed that OWA was a better combination algorithm in most cases compared with Max and Avg.

For future works, an algorithmic solution is under study for choosing OWA weights using machine learning approaches and selection of results for the decision fusion. As this model was tested on Rueters-21578 dataset using three classification algorithms, it can be implemented by other classifiers like SVM and on other standard datasets, too. Moreover, the analysis of time complexity as a factor of performance can be considered. Also, more powerful fusion operators for decision fusion component such as OWA, Decision Template and Fuzzy Integral can be studied later.

There are some limitations about this research. The proposed practical approach was not based on machine learning; therefore, it increased training time. We did not consider time complexity as a factor of performance because it just influenced the training time. Nevertheless, after finding a good combination of results for a dataset and classifiers, it is not required to repeat this process again.

REFERENCES

[1] Y. Yang, and J. O. Pedersen. "A Comparative Study on Feature Selection in Text Categorization". *In Proceedings of the Fourteenth international Conference on Machine Learning (July 08 - 12, 1997)*. D. H. Fisher, Ed. Morgan Kaufmann Publishers, San Francisco, CA, pp. 412-420, 1997.

[2] R. R. Yager. On ordered weighted averaging aggregation operators in multi-criteria decision-making. IEEE Trans. Syst. Man Cybern. 18, 1, pp. 183-190, 1988.

[3] S. Li, R. Xia, C. Zong, C.-R., Huang. "A framework of feature selection methods for text categorization". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Vol.2, pp. 692-700, 2009.

[4] M. Rogati, Y. Yang. "High-performing feature selection for text classification". In *Proceedings of the Eleventh international Conference on information and Knowledge Management* (McLean, Virginia, USA, November 04 - 09, 2002). CIKM '02. ACM, New York, NY, pp. 659-661, 2002. DOI= http://doi.acm.org/10.1145/584792.584911.

[5] J. S. Olsson and D. W. Oard. "Combining feature selectors for text classification". In *Proceedings of the 15th ACM international Conference on information and Knowledge Management* (Arlington, Virginia, USA, November 06 - 11, 2006). CIKM '06. ACM, New York, NY, pp. 798-799, 2006. DOI= http://doi.acm.org/10.1145/1183614.1183736

[6] M. H. Aghdam, N. Ghasem-Aghaee, M. E. Basiri. Text feature selection using ant colony optimization". *Expert Syst. Appl.* 36, 3 (Apr. 2009), pp. 6843-6853, 2009. DOI= http://dx.doi.org/10.1016/j.eswa.2008.08.022 .

[7] G. Nunzio. "A bidimensional view of documents for text categorization". In *Proceedings of the 26th European Conference on IR Research (ECIR'04)*, Sun-derland, United Kingdom, Apr. 5–7, pp. 112–126, 2004.

[8] F. Sebastiani. "Machine learning in automated text categorization". *ACM Comput. Surv.* 34, 1 (Mar. 2002), pp. 1-47, 2002. DOI= http://doi.acm.org/10.1145/505282.505283.

[9] R. Battiti, "Using mutual information for selecting features in supervised neural net learning". IEEE Transactions on Neural Networks 5, pp. 537-550, 1994.

[10] M. Srinivas, K. P. Supreethi, E. V. Prasad, S. A. Kumari. "Efficient Text Classification Using Best Feature Selection and Combination of Methods". In *Proceedings of the Symposium on Human interface 2009 on Conferenceuniversal Access in Human-Computer interaction. Part I: Held As Part of HCI international 2009* (San Diego, CA, July 19 - 24, 2009). M. J. Smith and G. Salvendy, Eds. Lecture Notes In Computer Science, vol. 5617. Springer-Verlag, Berlin, Heidelberg, pp. 437-446, 2009. DOI= http://dx.doi.org/10.1007/978-3-642-02556-3_50

[11] A. Khan, B. Baharudin, K. Khan. "Efficient feature selection and domain relevance term weighting method for document classification". Computer Engineering and Applications, International Conference on 2, pp. 398-403, 2010. DOI=http://doi.ieeecomputersociety.org/10.1109/ICCEA.2010.228

[12] R. Liere, P. Tadepall. "Active learning with committees for text categorization". In *Proceedings of AAAI-97, 14th Conference of the American Association for Artificial Intelligence* (Providence, RI, 1997), pp. 591–596, 1997.

[13] R. E. Schapire, Y. Singer. "BoosTexter: a boosting-based system for text categorization". Machine Learning 39, 2/3, pp. 135–168, 2000.

[14] R. E. Schapire, Y. Singer, A. Singhal. "Boosting and Rocchio applied to text filtering". In *Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval* (Melbourne, Australia, August 24 - 28, 1998). SIGIR '98. ACM, New York, NY, pp. 215-223, 1998. DOI= http://doi.acm.org/10.1145/290941.290996

[15] J. Novovicova, A. Malik. "Information-theoretic feature selection algorithms for text classification". *IEEE International Joint Conference on Neural Networks, IJCNN 5*, pp. 3272-3277, 2005.

[16] W. Frakes, R. Baeza-Yates. Information Retrieval Data Structures & Algorithms. Prentice Hall, 1992.

[17] P. Bednar, P. Butka, J. Paralic. "Java library for support of text mining and retrieval". In *Proceedings of Znalosti 2005*, (Stara Lesna), pp. 162–169, 2005.

[18] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications". Proceedings of the IEEE 85, 1, pp. 24-38, 1997.

[19] Li Xi; D. Hang; W. Mingwen. "Two-Stage Feature Selection Method for Text Classification," *Multimedia Information Networking and Security, 2009. MINES '09. International Conference on 18-20 Nov.* (1) pp. 234-238, 2009.

[20] J. Chen, H. Huang, S. Tian, Y. Qu, "Feature selection for text classification with Naïve Bayes". *Expert Syst. Appl.* (Apr. 2009) 36(3) pp. 5432-5435, 2009.

[21] M.A. Ghaderi, N. Yazdani, B. Moshiri and M.Mahmoudi, "OWA combination of feature set selection algorithms for text classification," *2010 5th International Symposium on Telecommunications (IST'2010), Tehran, Iran*, pp. 579-583, 2010.

[22] M.A. Ghaderi, N. Yazdani, and B. Moshiri, "A method for increasing text classification effectiveness", *2010 International Conference on Computer and Computational Intelligence (ICCCI 2010), China*, vol.2, pp.399-403, 2010.

Mohammad Ali Ghaderi received his B.Sc. degree in applied-scientific software development form Shiraz Islamic Azad University in 2008 and his M.Sc. degree in Information Technology from University of Tehran in 2011. His fields of research areas include multi-agent systems, information systems and simulation in general, and information retrieval, text mining, and databases in particular. He is co-author of several research papers in these areas.

Behzad Moshiri received his B.Sc. degree in mechanical engineering from Iran University of Science and Technology (IUST) in 1984 and M.Sc. and Ph.D. degrees in control systems engineering from the University of Manchester, Institute of Science and Technology (UMIST), U.K. in 1987 and 1991 respectively. He joined the school of electrical and computer engineering, university of Tehran in 1992 and is currently professor of control systems engineering. He was the member of ISA (Canada Branch) in 1991-1992. He has been the member of ISIF since 2002 and Senior member of IEEE since 2006. He has been the head of Machine Intelligence & Robotics division at school of ECE. He has also been the president and vice-president of Iranian Society of Instrument & Control Engineers. He was one of the founders of both "Control & Intelligent Processing, Center of Excellence" and "Iranian Society of Mechatronics Engineers". He is the author/co-author of more than 280 articles including 70 journal papers and 20 book chapters. Professor Moshiri's fields of research include advanced industrial control design, advanced instrumentation design, applications of information and data fusion in information technology, mechatronics, robotics, process control, bioinformatics and intelligent transportation systems (ITS).

Dr. Naser Yazdani is an associate professor in the Department of Electrical and Computer Engineering in Tehran University currently. He got his B.Sc. in Computer Engineering from Sharif University of Technology, Tehran, Iran. He worked in Iran Telecommunication Research Center (ITRC) as a consultant, researcher and developer for few years. To pursue his education, he entered Case Western Reserve Univ, Cleveland, Ohio, USA, later and graduated as a PhD in computer science and engineering. Then, he worked in different companies and research institutes in USA. He joined the ECE Dept. of Univ. of Tehran, Tehran, Iran, at Sep. 2000. Dr. Yazdani has initiated different research projects and labs in high speed networking and systems. His research interests include networking, packet switching, access methods, operating systems and database systems.

Maryam Tayefeh Mahmoudi is a Ph.D. candidate at department of machine intelligence in the University of Tehran majoring in artificial intelligence with emphasis on intelligent organization of educational contents. Within the past years, she has been involved in a variety of research works at Knowledge Management & e-Organization Research Group of IT Research Faculty, Research Institute for ICT (ex ITRC), working on issues like automatic generation of ideas and contents, decision support systems for research & education purposes, as well as conceptualization of IT research projects. She is a co-author of many research papers in different journals and proceedings of conferences.