Effective Clustering of MicroRNA Sequences by N-grams and Feature Weighting

Yuan Yi, Jihong Guan
Department of Computer Science and Technology
Tongji University
Shanghai 201804, China
E-mail: {1020080246, jhguan}@tongji.edu.cn

Shuigeng Zhou
Shanghai Key Lab of Intelligent Information Processing, and
School of Computer Science
Fudan University, Shanghai 200433, China
E-mail: sgzhou@fudan.edu.cn

***Abstract*—MicroRNA (miRNA in short) is a kind of small RNAs that acts as an important post-transcriptional regulator with the Argonaute family of proteins to regulate target mRNAs in animals and plants etc. Since its first recognition as a distinct class of small RNA molecules in the early 1990s, tens of thousands of miRNAs have been identified experimentally or computationally. Currently, the focus of miRNAs study is on single-miRNA functions that usually result in gene silencing and repression. With the rapid increase of miRNAs, biologists have manually organized these miRNAs into biologically meaningful families to facilitate further study. As the members in the same family tend to share similar biochemical functions, a high quality family organization will shed lights on the functions of unknown miRNAs. However, manually grouping large amounts of miRNAs is not only time-consuming but also expensive. In this paper, we employ a clustering method with N-grams and feature weighting to automatically group miRNAs into separate clusters (families). Our method is evaluated with datasets constructed from the online miRNA database miRBase. Experimental results show that the clustering method can successfully distinguishes most miRNA families, and outperforms the traditional K-means clustering algorithm and the average-link clustering approach.**

***Keywords*—miRNAs; clustering; N-grams; feature weighting.**

I. Introduction

MicroRNAs (miRNAs in short) are an important category of endogenous small noncoding RNAs (20-30 nt) existing in eukaryotic cells and viruses, and play important gene-regulatory roles by pairing to the messenger RNAs (mRNAs in short) to direct their post-transcriptional repression [1]. They serve as specificity factors that direct bound effector proteins, of which the core component is a member of the Argonaute protein superfamily, to target nucleic acid molecules via base-pairing interactions [2]. The first two miRNAs, expressed by *lin-4* and *let-7* genes in the worm *Caenorhabditis elegans*, were discovered in 1993 [3] and 2000 [4], respectively. Since then, tens of thousands of miRNAs were found in plants, animals, metazoans and viruses etc. For example, in the most authoritative online miRNA database miRBase [5], [6], [7], [8], there are now 18,226 entries representing hairpin precursors, which express 21,643 mature miRNAs distributed over 168 species (release 18[1]). By far, over 75% miRNAs registered in miRBase have been grouped into families, and members in the same family may have similar biological functions. To construct miRNA families, current semi-automated methods have been found difficult to keep up with the pace of miRNA discovery. In order to overcome this problem, machine learning based methods are introduced recently. Ding et al. proposed an automatic alignment free method for miRNA classification — *miRFam* [9], which was based on N-gram [10] represented miRNA sequences and a multiclass SVM. This method is both effective and efficient. However, since it is a supervised method, predefined family structure and labeled miRNA sequences are required for training the classifier.

In this study, we employ an unsupervised learning method to automatically group large amounts of miRNAs. The basic idea of this method is an improved K-means clustering approach with an adaptive feature weighting mechanism. Experiments over several datasets demonstrate the effectiveness and efficiency of the proposed approach, which significantly outperforms the traditional K-means clustering approach and the average-link clustering approach.

[1]available at: http://www.mirbase.org/

II. Results

We implemented our approach based on the SKWIC clustering algorithm [11], which is an improved K-means algorithm with simultaneous keyword identification and clustering for documents clustering, to unsupervisedly classify miRNAs in the miRNA database miRBase.

The overall procedure of our approach is as follows. Firstly, we transform the miRNA primary sequences into vectors, each dimension of the vectors corresponds to one feature of the sequences. This phase is called *feature extraction*. Here, N-grams [10] are used for extracting features from the miRNA primary sequences. After extracting N-grams from each miRNA sequence, a weighting method called *concentration factor* [9] is introduced to weight these N-grams. The detail of feature extraction will be presented in the *Materials and Methods* section. Secondly, we adopt the SKWIC algorithm to cluster these transformed miRNA vectors. SKWIC is a variant of the classic K-means clustering algorithm and has the ability to simultaneously weight the features of each cluster on the fly and assign each data point to the nearest cluster according to the weighted distances between the point and the centroids of all clusters. SKWIC was initially developed for document

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

clustering analysis. Here we employ it for miRNA sequences clustering. The details of SKWIC algorithm are also given later in the *Materials and Methods* section. Finally, we evaluate the proposed approach with several datasets from miRBase, and compare the clustering results with the families of miRBase.

Here, we conduct two experiments. The first experiment is to test our approach over a dataset extracted from miRBase16 and evaluate its clustering performance; The second experiment is to apply our approach to real unclassified miRNA sequences (those are not assigned to any predefined family) extracted from miRBase18 to validate the prediction capability of our approach. For convenience, the dataset extracted from miRBase16 is denoted as R1, and another dataset extracted from miRBase18 is denoted as R2. The details of these two datasets are shown in Table I.

A. Clustering Result on Dataset R1

Here, we use dataset R1 constructed from miRBase16 as the base test dataset to evaluate the clustering performance of our approach. In miRBase16, there are over 17,000 distinct mature miRNA sequences, all of which belong to four biological categories — animals, plants, viruses and chromalveolata. In this study, we choose miRNAs in the families that have no less than five members. The families with less than five members are too small and tend to misleading the clustering algorithm, hence are discarded. Since there are only 33 families of viruses and 1 family of chromalveolata in miRbase16, and 31 of the 32 families of viruses and the only one chromalveolata miRNA family have less than five members, all of these two species' miRNAs are excluded in our experiment. Of course, those miRNAs without family information are also ignored. As a result, we consider only miRNAs in animals and plants organisms in miRBase16. The final dataset R1 contains 9,225 mature miRNA primary sequences. These 9,225 miRNAs belong to 394 families, among which 319 families are of animals. The largest three families are *let-7* , *mir-17* and *mir-154* , which are all animal miRNA families and have 195, 175 and 169 members, respectively. Because the families of all of these miRNAs are already known, we can effectively evaluate the clustering performance of our approach.

Each mature miRNA sequence is transformed to a vector of 340 dimensions. The first 4 dimensions represent 4 unigrams *A*, *G*, *C* and *U*, followed by 16 bigrams *AA*, *AG*, *AC*, *AU*, *GA*, *GG*, *GC*, *GU*, etc., then 64 trigrams *AAA*, *AAG*, *AAC*, *AAU*, etc., and finally 256 tetragrams *AAAA*, *AAAG*, *AAAC*, *AAAU*, etc. Note that all of these 340 dimensions have been weighted by concentration factors.

Once we have the collection of miRNA vectors, the SKWIC clustering method is applied to these miRNA sequence vectors. Since SKWIC is based on the K-means method, the number K of clusters should be determined beforehand. It is reasonable to group the miRNAs into the same number of clusters as the number of families, i.e., 394 clusters. However, there are many small families, and miRNAs in these small families are likely to be merged into other larger families. This might cause the degradation of clustering results. So, we also choose some other numbers of clusters larger than 394, that is, 472, 550, 628, 706 and 784. The last number is nearly twice as the number of families. On the other hand, due to the randomness of SKWIC's initial cluster centroids, we perform the experiment 20 times for each cluster number to eliminate this randomness.

To evaluate our approach, we adopt two performance measurements — the *vote strategy* and the *Davies-Bouldin index* . The *vote strategy* is a validation measurement based on the clustering accuracy. It is calculated from the confusion matrix of the clustering result. It counts the largest family in each cluster, and the second largest family if the number of miRNAs of the second largest family in this cluster is no less than $1/3$ of the number of miRNAs in the largest family. The reason for taking the second largest family in a cluster as correctly clustered is that some families of miRNAs are very alike so that sequences in these families tend to be classified into the same cluster. Examples of such families include *mir169_1* and *mir169_2* of plant miRNAs.

Fig. 1 shows the average voted accuracies of our approach using *Manhattan distance* and *cosine similarity* respectively, the basic K-means method, and the average-link approach with MSA score as similarity measure between sequences. It can be seen from Fig. 1 that our approach using Manhattan distance has the best performance. When K=706, the average accuracy is approximately 92.5%, which means that on average about 8,533 out of the 9,225 miRNAs in this dataset are considered to be correctly clustered. By contrast, the maximum average accuracies of our approach with cosine similarity, K-means using Manhattan distance and Euclidean distance, average-link with MSA similarity are about 91.1% (706 clusters), 88.6% (628 clusters), 90.1% (628 clusters) and 92.2% (628 clusters), respectively. Fig. 1 also suggests that it is not the case that the more clusters the better clustering performance. The results of these methods tend to be stable when the cluster number is larger than 600.

The Davies-Bouldin index (DBI) [12] is an internal evaluation metric involving quantities of the dataset itself. It measures the inherent quality of partitions of a clustering algorithm without any apriori knowledge about the structure of the dataset. It is based on the intra-cluster similarity and the inter-cluster dissimilarity. The detail about this measurement is described in Section *Materials and Methods*. Fig. 2 shows the DBI values on dataset R1. Since DBI measures the structure quality of clustering based on distance metric, we compare only our approach with Manhattan distance and the basic K-means algorithm with Manhattan distance. Again, our approach outperforms the basic K-means clustering method. This means that the structure of clustering result of our approach is better than that of the simple K-means algorithm, which confirms the effectiveness of our approach.

Table II and Table III present some details of a clustering result. Here, the cluster number is set to 628, the voted accuracy and the Davies-Bouldin index are about 92.8% and 1.03, respectively. Table II is miRNA distribution over different clusters. We consider only miRNA families whose sizes are

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

TABLE I
THE DATASETS USED IN THIS STUDY

Dataset	Size	Source	Family	Description
R1	9,225(0)	mirBase16	394	From animal and plant families that contain no less than 5 members.
R2	17,205(5,777)	mirBase18	451+unclassified	From animal and plant families that contain no less than 5 members, plus unclassified sequences

The "Size" column indicates the number of miRNAs in each dataset, with the number of unclassified miRNAs enclosed in the parentheses.

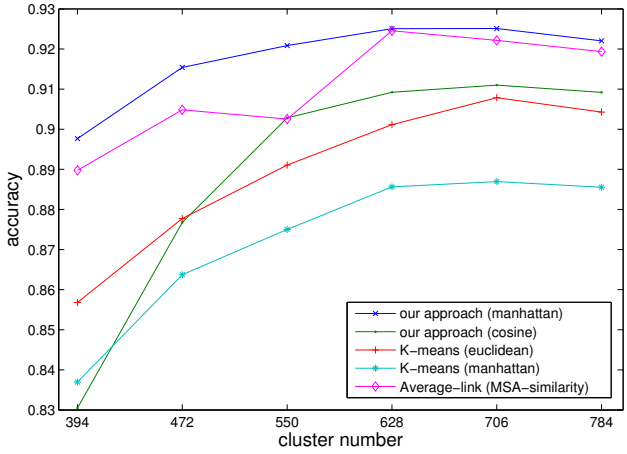

Fig. 1. The average voted accuracy results on dataset R1. There are two versions of our approach (with Manhattan distance and cosine similarity), the basic K-means approach (with Manhattan distance and Euclidean idstance), and the average-link clustering method with MSA similarity. It can be seen that our approach with Manhattan distance outperforms the other methods with regard to all selected numbers of clusters.

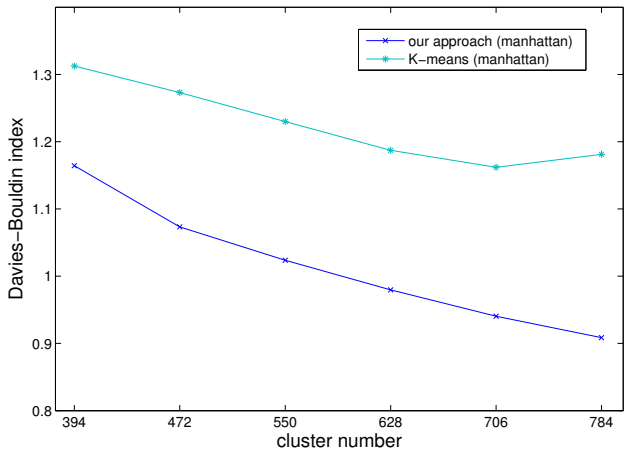

Fig. 2. The average DBI values on dataset R1. The lower the DBI value is, the better the performance is. The results show that the structure of clustering result of our approach is better than that of the basic K-means method. Since DBI measures the structure quality of clustering based on distance metric, we compare only our approach with the basic K-means algorithm, which all use the Manhattan distance.

more than 100, and the top 3 largest clusters. Note that the cluster number is set to 628, which is larger than the number of families, it is inevitable that miRNAs from one family may distribute across several different clusters. This does not matter because if most miRNAs of a family distribute among several much pure clusters, these clusters can be taken as all belonging to that family and postprocessing such as multiple sequence alignment (MSA) could be done to merge these clusters. Take the largest family *let-7* for example. We use Clustal [13] to analyze clusters 161, 512, 273 and find out that these clusters are all pure enough to indicate their own families. And then, we randomly choose several representative sequences from each cluster and align them together. From the result of MSA, we can merge these clusters to represent the family *let-7*. This procedure can be performed for every cluster to do possible merging. In Table II, most families have more than half of their miRNAs distributing among the top 3 largest clusters, except for *mir-154*. Even if we consider only the largest cluster for each family, there are nearly half of these families that have more than 50% of their miRNAs are assigned to one cluster. For example, the family *MIR166*, which has 141 miRNAs in total, 134 of them are in cluster #606, which accounts for more than 95% of that family. Table III is miRNAs distribution of families that have only 5 members, the minimum size of families in this experiment. There are totally 34 families that have only 5 members, Table III shows only 16 of them. Therein, 8 out of 16 families have all their members clustered into one cluster. In all 34 families of size 5, 20 of them (59%) have their members clustered into one cluster.

Fig. 3 shows the family size distribution of dataset R1 and cluster size distribution of the clustering result when R1 is grouped into 628 clusters. Table IV presents the family constituent of the largest 20 clusters. For example, the members of the biggest cluster, i.e. cluster #606, belong to only one family, *MIR166*. Obviously, this cluster is considered to be the cluster of this family. There are also 7 other pure clusters like this one. Although other clusters are not so pure, they contain members of at most 3 families and there is a dominant family in each cluster. For example, the second largest cluster #594 has 118 members, 117 of which belong to the family *MIR395* and only 1 of which belongs to the family *mir-30*. Of course, cluster #594 is also considered as the cluster of family *MIR395*. Note that it can also be seen from Table II that this cluster is the largest one for family *MIR395*, the rest member of this cluster, which belongs to *mir-30*, is obscured by the dominant family.

B. Clustering Results on Dataset R2 with Unclassified miRNAs

We also apply our approach to cluster those miRNAs whose family labels are not known in miRBase18. There are 21,643 mature miRNA sequences in miRBase18. The dataset R2 contains 17,205 sequences, where 11,428 sequences explicitly belong to 451 families and 5,777 sequences are not

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

TABLE II
THE MIRNA DISTRIBUTION OVER CLUSTERS FOR MIRNA FAMILIES IN R1 THAT HAVE MORE THAN 100 MEMBERS

Family	Size	1st cluster	2nd cluster	3rd cluster	Ratio1	Ratio2
let-7	195	77(161)	19(512)	18(273)	39.5%	58.5%
mir-17	175	44(255)	38(311)	27(517)	25.1%	62.3%
mir-154	169	16(289)	14(404)	14(534)	9.5%	26.0%
mir-515	146	34(9)	32(111)	24(298)	23.3%	61.6%
MIR166	141	134(606)	7(169)	-	95.0%	100%
MIR156	140	79(344)	42(388)	13(564)	56.4%	95.7%
mir-9	120	57(460)	14(50)	14(553)	47.5%	70.8%
MIR395	119	117(594)	1(222)	1(521)	98.3%	100%
mir-25	115	63(159)	18(36)	16(392)	54.5%	84.3%
mir-2	110	49(549)	37(532)	12(571)	44.5%	89.1%
MIR171_1	109	70(618)	35(416)	2(358)	64.2%	98.2%
mir-30	109	27(200)	21(361)	16(422)	24.8%	58.7%
mir-8	109	29(54)	19(207)	17(464)	26.6%	59.6%
MIR159	107	56(338)	48(368)	1(138)	52.3%	98.1%
MIR399	107	55(434)	37(261)	9(199)	51.4%	94.4%
mir-15	102	31(485)	28(397)	24(529)	30.4%	81.4%

Only the largest 3 clusters are shown. The number in parentheses is the cluster tag. For each family, "1st cluster", "2nd cluster" and "3rd cluster" mean the largest 3 clusters. "Ratio1" is the ratio of the largest cluster size over the family size, "Ratio2" is the ratio of the overall size of the largest 3 clusters over the family size.

TABLE III
THE MIRNA DISTRIBUTION OVER CLUSTERS FOR MIRNA FAMILIES IN R1 THAT HAVE ONLY 5 MEMBERS

Family	1st cluster	Family	1st cluster	Family	1st cluster	Family	1st cluster
mir-2808	2(286)	mir-584	5(220)	mir-762	4(181)	mir-92	3(159)
mir-3065	3(22)	mir-589	5(41)	mir-84	2(155)	mir-935	5(578)
mir-492	5(97)	mir-676	3(299)	mir-883	3(431)	mir-980	5(537)
mir-562	4(50)	mir-74	5(250)	mir-889	5(144)	mir-996	5(337)

Only the largest clusters are shown. The number in parentheses is the cluster tag.

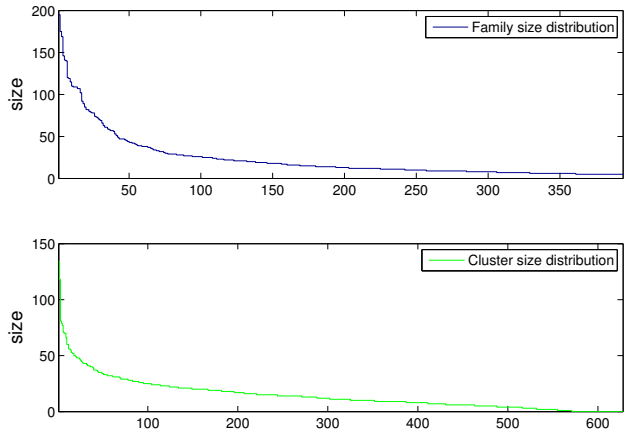

Fig. 3. The family size distribution of dataset R1 and the cluster size distribution of the clustering result when grouping dataset R1 into 628 clusters.

TABLE IV
THE FAMILY CONSTITUENT OF THE LARGEST 20 CLUSTERS

Cluster	Size	1st family	2nd family	3rd family
cluster #606	134	MIR166(134)	-	-
cluster #594	118	MIR395(117)	mir-30(1)	-
cluster #344	81	MIR156(79)	mir-63(1)	mir-2808(1)
cluster #259	79	MIR169_2(75)	MIR169_1(4)	-
cluster #161	77	let-7(77)	-	-
cluster #325	71	MIR160(69)	mir-675(2)	-
cluster #209	70	MIR169_1(67)	mir-344(2)	mir-1420(1)
cluster #618	70	MIR171_1(70)	-	-
cluster #159	66	mir-25(63)	mir-92(3)	-
cluster #460	60	mir-9(57)	MIR394(2)	mir-208(1)
cluster #622	60	mir-7(59)	mir-1422(1)	-
cluster #221	56	mir-1(55)	MIR2629(1)	-
cluster #338	56	MIR159(56)	-	-
cluster #434	55	MIR399(55)	-	-
cluster #60	53	mir-34(51)	mir-449(2)	-
cluster #48	52	mir-29(52)	-	-
cluster #383	52	MIR164(47)	mir-515(4)	MIR160(1)
cluster #40	50	mir-10(50)	-	-
cluster #549	50	mir-2(49)	mir-1419(1)	-
cluster #374	49	mir-125(49)	-	-

The number in parentheses is the number of miRNAs that belong to the family. Note that for the top 20 clusters, at most 3 families are covered.

assigned. Note that more than one third of these miRNAs are unclassified, which is a larger portion compared to that of the early versions of miRBase. The cluster number is set to 800 and 1,200.

Table V shows the clustering result on R2. According to the result, clusters containing unclassified miRNAs fall into 3 categories. The first category correspond to new families, where most miRNAs are unclassified. Here, if the number of unclassified miRNAs in a cluster is more than 3 times of that of classified miRNAs, these unclassified miRNAs in such a cluster will possibly form one or several new families (depending a more detailed analysis). The second category covers the mixed clusters where the unclassified miRNAs make up of the first or second largest part and the ratio of the numbers of

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

TABLE V
CLUSTERING RESULTS ON DATASET R2 WITH UNCLASSIFIED MIRNAS

Cluster number	New clusters	Mixed clusters	Others
800	199(1831+256)	378(2837+4459)	175(1109+6684)
1200	361(2276+199)	513(2547+4123)	235(954+6952)

The resulting clusters are subsumed to three categories: new families, mixed clusters and the others. The first numbers in the three (2nd $\sim$ 4th) columns are the number of clusters, and the numbers in the parentheses are the number of unclassified miRNAs and the number of classified miRNAs in the corresponding clusters.

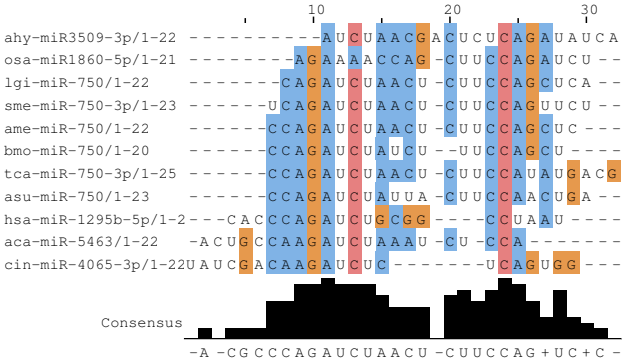

Fig. 4. The MSA result of a new cluster. All sequences in this cluster are unclassified. The cluster number is 800.

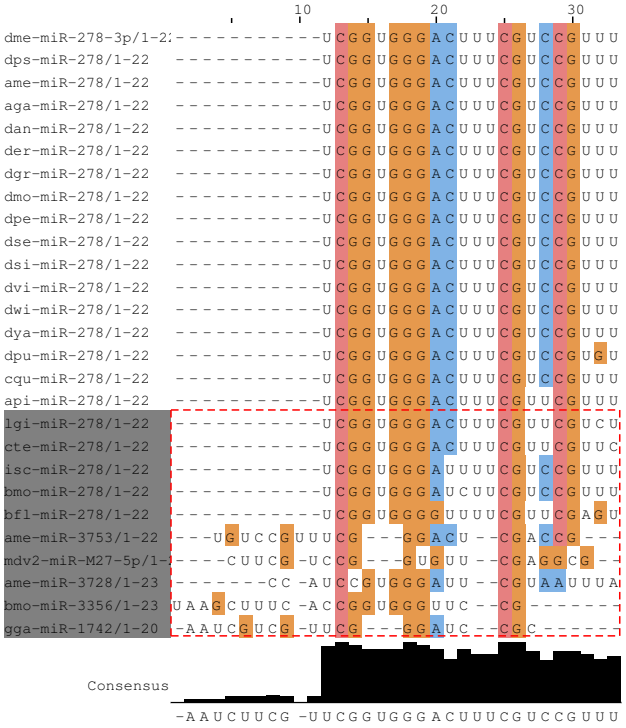

Fig. 5. The MSA result of a mixed cluster. The first 17 sequences belong to the family *mir-278* and the last 10 sequences (shaded) are unclassified in miRBase18. The cluster number is 1200.

miRNAs between the first largest part and the second largest part is no more than 3 times. In this case, we assign these unclassified miRNAs to the largest family in the cluster. The other clusters that contain unclassified miRNAs consist of the third category. The remaining clusters are dealt with as in the previous experiment, where all members belong to some known families. Table V shows the clustering results when the number of clusters is set to 800 and 1,200. When the dataset is clustered into 800 groups, we obtain 199 new clusters that contain in total 1,831 unclassified miRNAs, 378 mixed clusters that cover in total 2,837 unclassified and 4,459 classified miRNAs. In addition, 175 clusters contain relatively fewer unclassified miRNAs. When the dataset is clustered into 1,200 clusters, the numbers of all three categories of clusters increase up to 361, 513 and 235, respectively. More new families generated indicates that more unclassified miRNAs previously assigned to existing families are now distinguishable.

To further illustrate the clustering result, we also use multiple sequence alignment (MSA) to align sequences of two clusters. Fig. 4 is the alignment of unclassified sequences of a new cluster while the cluster number is 800. Though this is not an ideal cluster, some sequences likely belong to a new family. Fig. 5 is the alignment of a mixed cluster while the cluster number is 1200. The first 17 sequences belong to the family *mir-278* and the last 10 sequences (shaded) are unclassified in miRBase18. We can see that there are 5 unclassified sequences in this cluster that very likely belong to the dominant family. The other 5 unclassified sequences are different from the dominant family, which may belong to an unknown tiny family that has less than 5 members. This may be an indication of drawback of using short N-gram, since it considers only adjacent N locations. Nevertheless, if the sequences are similar enough, it is very likely for them to be clustered into the same cluster, just as shown in the experiment on miRBase16, and a few outliers can be detected by postprocessing such as manually inspection or MSA.

III. DISCUSSION

With the increase of un-annotated miRNAs in miRBase, traditional semi-automated annotation methods will not be sufficient. And if there are new unknown families hidden in these un-annotated miRNAs, then even supervised classification methods will not work. In such cases, the clustering methods, which can automatically find out cluster structures in miRNA datasets without any prior knowledge about the families, can be a useful tool to identify new families and classify miRNAs into the existing or the newfound families. The approach proposed in this study using N-gram representation and the SKWIC clustering algorithm to classify miRNA sequences is shown an effective and efficient method.

To use this approach to unsupervisedly classify miRNAs, there are two important parameters to specify, i.e., the number of features and the number of clusters. The number of features is determined by the number of N-grams used. In this study, we adopt up to 4-grams to represent miRNA sequences, and the number of features is 340. If longer N-grams are used, the number of features will increase exponentially. For instance, when 5-grams are used, the number of features will rise to

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

1,364, and then 5,460 for 6-grams, and so forth. The computational cost on vector operation will also increase exponentially. Hence, there is no much room for tuning this parameter. Another parameter, the cluster number, can be changed more freely, which depends on the number of existing families. It can be seen from our first experiment with miRBase16, where the families of all miRNAs are known, that a slightly larger number of clusters would result in a better accuracy. In our test of unclassified data in miRBase18, the cluster number is set to 800 and 1,200.

Despite the intrinsic randomness of the K-means based method, the experiment on classified miRNAs in miRBase16 shows that our method is quite stable, that is, no matter what cluster number is set with a reasonable range, the worst resulting voted accuracy is still always more than the average voted accuracy minus less than 1%. So, even if we run this method only one time, the clustering result will also be good enough. Furthermore, the efficiency of our approach is another advantage. With appropriate parameter setting, our approach using the Manhattan distance could almost always generate reasonable clusters in a few minutes.

Of course, short N-grams are less expressive than some other sophisticated presentation schemes, because it considers only N adjacent positions in a sequence. However, one of the biggest strength of our approach is its efficiency. Only several minutes are needed to do the clustering on a PC. The effectiveness can also be seen in the experiments. For the future work, other features such as the secondary structure units can be considered to increase the power of miRNA sequence presentation and hence to enhance the performance of clustering.

Another concern about the N-gram representation of miRNAs is the sparseness of the vectors. Since miRNAs are short, most of the features in a vector are 0. Actually, the weighting scheme in our approach can effectively and adaptively deal with data sparseness.

IV. Materials and Methods

A. Datasets

We use miRBase [5], [6], [7], [8] to test our approach. This is a searchable online repository for all miRNA sequences and annotations [5]. In its latest version of release 18, there are 18,226 hairpin precursors, expressing 21,643 maturities in 168 species. In this study, we choose the release 16 and 18 as our test datasets. The release 16 contains over 15,000 MIR gene loci in over 140 species, and 17,341 mature miRNA sequences [5]. Two datasets, R_1 and R_2, are constructed for experiments from miRBase 16 and 18, respectively. Dataset R_1 is based on miRBase 16 by choosing miRNAs in families that belong to plants and animals, and contain no less than 5 members. Dataset R_2 is constructed from miRBase 18, it includes all miRNAs in plant and animal families with no less than 5 members, along with the unclassified sequences. The details are summarized in Table I.

B. Feature extraction

We represent each miRNA sequence as a vector of N-grams [10]. An N-gram is a subsequence consisting of N spatially consecutive items extracted from a given sequence. In the context of miRNAs, items include A, G, C and U, which are the four base nucleotides constituting RNAs. For instance, given an miRNA sequence UCCAG, there are 1-grams A, G, C, U, 2-grams UC, CC, CA, AG, 3-grams UCC, CCA, CAG, and so forth. Here, we consider only up to 4-grams. In an miRNA vector, the first 4 dimensions are features representing 4 unigrams *A*, *G*, *C*and *U*, followed by 16 bigrams *AA*, *AG*, *AC*, *AU*, *GA*, *GG*, *GC*, *GU* etc., and then 64 trigrams *AAA*, *AAG*, *AAC*, *AAU* etc., and finally 256 tetragrams *AAAA*, *AAAG*, *AAAC*, *AAAU* etc. After the collection of miRNA vectors is obtained, each dimension is further weighted by the concentration factor [9]. The concentration factor is devised to reflect the importance of different types of N-grams. Longer grams are considered more important than shorter ones.

C. Clustering method

Clustering is the process of automatically grouping a set of data objects into different groups (i.e. clusters), without any prior knowledge of which group a data object belongs to. The target is to make sure that data objects in the same cluster are more similar with each other than with those in different clusters, according to some specified measurement. Here, the data objects refer to miRNAs and the task is to assign them into different clusters without knowing the true family of each of them.

There are a wide variety of clustering methods proposed to solve different kinds of problems. Typically, clustering methods can be divided into the following categories: partitioning-based methods, hierarchical methods, density-based methods and spectral clustering etc. In this study, we use the SKWIC clustering method [11] to perform the unsupervised classification of miRNA sequences. The SKWIC algorithm, or Simultaneous KeyWord Identification and Clustering, was originally proposed for clustering text documents, and is a variant of the Simultaneous Clustering and Attribute Discrimination (SCAD) method [14], both of which are based on the K-means method. The K-means method is a classic partitioning-based clustering method. The basic algorithm of K-means is as follows: first, specify the number k of clusters to be obtained and select k initial centroids (the centers of clusters); After that, iteratively distribute data objects to clusters and update the centroids according to data assignments until the centroids do not change or the amount of changes is under a specified threshold. This is an efficient and effective method to automatically group a set of data objects into clusters. It has the advantage of fast convergence to a local optimum.

The basic K-means algorithm treats each dimension or feature as equally relevant to every cluster. However, it is obvious that in many circumstances, different clusters differ largely in their best feature sets, and the relationships between clusters and their respective feature sets need to be discovered simultaneously [11]. The advantage of the SKWIC algorithm

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

over the basic K-means algorithm described above is that the former considers the weight of each feature in clusters simultaneously when clustering, and it uses the cosine similarity to measure the proximity of data objects in a high dimensional vector space, in which the Euclidean distance measurement is not applicable.

Like the K-means algorithm, the mathematical background behind the SKWIC algorithm [11] is to minimize an objective function as follows:

$$J(\mathbf{C},\mathbf{V};\chi)=\sum_{\mathrm{i}=1}^{\mathrm{C}}\sum_{\mathrm{x_j}\in\chi_\mathrm{i}}\sum_{\mathrm{k}=1}^{\mathrm{n}}\mathrm{v_{ik}}\mathrm{D}^{\mathrm{k}}_{\mathrm{wc_{ij}}}+\sum_{\mathrm{i}=1}^{\mathrm{C}}\delta_\mathrm{i}\sum_{\mathrm{k}=1}^{\mathrm{n}}\mathrm{v}^2_{\mathrm{ik}}, \quad (1)$$

subject to

$$v_{ik}\in[0,1]\ \forall i,k;\quad \text{and}\quad \sum_{k=1}^{n}v_{ik}=1,\ \forall i \quad (2)$$

where C is the number of clusters, n the number of dimensions, χ_i the cluster i, v_{ik} the weight of cluster i in dimension k, $D^k_{wc_{ij}}$ the distance along individual dimension k which will be discussed later. This objective function has some differences from that of classic K-means. More precisely, the first component is very much like the objective function of classic K-means except that its distances along individual dimensions are weighted with a positive value. Dimensions with high weights are more relevant to that cluster than those with low weights. And there is another component in the objective function to control the weights, that is, a weighted sum of squares of weights.

The first component in (1), that is, the sum of weighted distances between data points and their corresponding cluster centroids, is intended to obtain compact clusters. It is minimized when only one dimension in a cluster is totally relevant and all the other dimensions are irrelevant. The second component in (1), which is the weighted sum of squares of weights, is used to control the weights v_{ik}. It is minimized when all dimensions are equally weighted. By combining these two components, with appropriate parameter δ_i, the resulting clusters will have their within-cluster weighted distances minimized, while the feature weights for each cluster are optimized.

Given a set of centroids and a partition, we can adopt the Lagrange multiplier method to solve the constrained optimization problem about J with respect to dimension weight v_{ik}. We turn the objective function (1) and the constraint (2) into the following form, which is called the Lagrange function:

$$\begin{aligned}J(\mathbf{\Lambda},\mathbf{V})=&\sum_{\mathrm{i}=1}^{\mathrm{C}}\sum_{\mathrm{x_j}\in\chi_\mathrm{i}}\sum_{\mathrm{k}=1}^{\mathrm{n}}\mathrm{v_{ik}}\mathrm{D}^{\mathrm{k}}_{\mathrm{wc_{ij}}}+\sum_{\mathrm{i}=1}^{\mathrm{C}}\delta_\mathrm{i}\sum_{\mathrm{k}=1}^{\mathrm{n}}\mathrm{v}^2_{\mathrm{ik}}\\&-\sum_{i=1}^{C}\lambda_i(\sum_{k=1}^{n}v_{ik}-1),\end{aligned} \quad (3)$$

where $\Lambda=[\lambda_1,\lambda_2,...,\lambda_C]$ is the Lagrange multipliers. To find out the stationary point of equation (3), the gradient of J is set to zero and obtain

$$\begin{cases}\frac{\partial J(\mathbf{\Lambda},\mathbf{V})}{\partial \mathrm{v_{ik}}}=\sum_{x_j\in\chi_i}D^k_{wc_{ij}}+2\delta_i v_{ik}-\lambda_i=0\\ \frac{\partial J(\mathbf{\Lambda},\mathbf{V})}{\partial \lambda_\mathrm{i}}=(\sum_{k=1}^{n}v_{ik}-1)=0.\end{cases} \quad (4)$$

Solving the above simultaneous system of equations for v_{ik}, we obtain

$$v_{ik}=\frac{1}{n}+\frac{1}{2\delta_i}\sum_{x_j\in\chi_i}[\frac{\sum_{l=1}^{n}D^l_{wc_{ij}}}{n}-D^k_{wc_{ij}}]. \quad (5)$$

Through this equation, the dimension weights of clusters can be updated, given a set of centroids and a partition according to that centroids and weights, to reflect the current dimension relevance of clusters. The first part of equation (5) is $1/n$, which is the default weight if all dimensions are treated equally in a cluster. The second part, which is the sum of differences between the average of individual dimension distances and the individual distances of dimension k, is the bias that takes into account the differences between dimensions. This part can either be positive or negative. A positive value will increase that weight, which means that the corresponding dimension is associated with the cluster more closely, for the sum of individual distances of dimension k is less than the sum of the average of all individual distances. Similarly, a negative value of that part means less relevant to the cluster for a dimension.

The parameters δ_i in the above equations are important because it is used to weight the relative importance of the second component in equation(1). If δ_i is too small, then the contribution of the second part in (1) will be negligible, and one dimension in cluster χ_i will have relative high weight respect to other dimensions, which would have quite small weight or even zero weight. On the other hand, if δ_i is chosen too large, then almost all dimensions in cluster χ_i will be equally weighted with values $1/n$ approximately [11]. Consequently, δ_i will be updated iteratively as follows:

$$\delta_i^{(t)}=K_\delta\frac{\sum_{x_j\in\chi_i^{(t-1)}}\sum_{k=1}^{n}v_{ik}^{(t-1)}D^{k^{(t-1)}}_{wc_{ij}}}{\sum_{k=1}^{n}(v_{ik}^{(t-1)})^2}, \quad (6)$$

where the superscripts (t) and $(t-1)$ mean that their values are in the current iteration t and in the previous iteration $t-1$, respectively, and K_δ is a constant. Due to δ_i, the weights v_{ik} are often out of the range $[0,1]$. If this occurs frequently, then the constant K_δ should be increased. If it occurs occasionally, then we could re-tune the weights of the corresponding cluster as follows:

$$v_{ik}\leftarrow v_{ik}+|min_{k=1}^{n}\{v_{ik}\}| \quad \text{if}\ \ v_{ik}<0. \quad (7)$$

In the original SKWIC algorithm [11], the distance measurement of data points is based on the cosine similarity. However, in this study, the experiment results show that, when the distance is measured by the Manhattan distance (a.k.a. city-block distance), more miRNAs are correctly clustered. Hence, the distance along individual dimension $D^k_{wc_{ij}}$ is defined as follows:

$$D^k_{wc_{ij}}=|x_{jk}-c_{ik}| \quad (8)$$

978-1-4673-4398-5/12/$31.00 ©2012 IEEE

.

When partitioning, each data point is assigned to the nearest cluster centroid, measured by the distance defined as above in (8). Except for that the distance measurement is weighted, there is nothing particular here.

After each step of partitioning, a centroid updating step is carried out, as in the classic K-means. In SKWIC, this is done through the following equation:

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0, \\ \frac{\sum_{x_j \in \chi_i} x_{jk}}{|\chi_i|} & \text{if } v_{ik} > 0. \end{cases} \quad (9)$$

The whole clustering process of SKWIC is executed as follows: first, specify the number k of clusters to be obtained, select k initial centroids randomly, and initialize the partition with equal dimension weights $1/n$; after that, iteratively update dimension weights v_{ik} using (5), assign data objects to clusters, update centroids according to that assignment and update δ_i by (6), until the centroids do not change or the amount of changes is under a specified threshold.

In addition to the implementation of SKWIC algorithm, we also use other clustering methods, including simple K-means algorithm, EM algorithm, DBSCAN algorithm etc. Source codes of all these methods are from *Weka 3* [15], an open source data mining software repository (available at: http://www.cs.waikato.ac.nz/ml/weka/) written in Java.

D. Evaluation methods

To evaluate the quality of the clustering result, here we adopt an accuracy measurement based a *vote strategy*, which is based on the confusion matrix constructed from the clustering result and reflects the family/cluster relationship implied in the clustering result. While evaluating the voted accuracy, miRNAs of the largest family in a cluster, together with those in the second largest family which has no less than $1/3$ miRNAs of the largest family, are considered as correctly clustered. However, if the total number of miRNAs in a cluster is less than 5, then this cluster is thought to be invalid cluster and is ignored. The final value of the voted accuracy is obtained by dividing the number of correctly clustered miRNAs by the total number of miRNAs in the data set, and hence its range is [0, 1]. When all miRNAs in the families are correctly clustered into their corresponding clusters, the accuracy is 1.

We also use another measurement called *Davies-Bouldin index* (DBI) [16], which is an internal evaluation metric, to validate the clustering result. An internal evaluation metric uses only information of the tested dataset. The DBI measures both the intra-cluster similarity and the inter-cluster dissimilarity. Let s_i be the intra-cluster dissimilarity of cluster i, and d_{ij} be the inter-cluster dissimilarity between the centroids of cluster i and j. The dissimilarity R_{ij} between a pair of clusters i and j is defined as follows:

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}. \quad (10)$$

Then, the DBI is defined as

$$DB = \frac{1}{C} \sum_{i=1}^{C} \max_{j=1,\ldots,C; i \neq j} R_{ij}, \quad (11)$$

where C is the number of clusters.

ACKNOWLEGEMENTS

This work was supported by China 863 Program under grant No. 2012AA020403 and NSFC under grant No. 61173118. JG was also supported by the Shuguang Scholar Program of Shanghai Education Foundation and the Open Research Program of Shanghai Key Lab of Intelligent Information Processing. The authors would like to thank Ms. Linxia Wan for providing the prepared datasets, and Dr. Jiandong Ding for suggestions and comments on the draft.

REFERENCES

[1] D. Bartel, "Microrna: target recognition and regulatory functions," *Cell*, vol. 136, pp. 215–33, 2009.

[2] R. Carthew and E. Sontheimer, "Origins and mechanisms of mirnas and sirnas," *Cell*, vol. 136, pp. 642–655, 2009.

[3] R. Lee, R. Feinbaum, and V. Ambros, "The c. elegans heterochronic gene lin-4 encodes small rnas with antisense complementarity to lin-14," *Cell*, vol. 75, pp. 843–54, 1993.

[4] B. Reinhart, F. Slack, M. Basson, A. Pasquinelli, J. Bettinger, A. Rougvie, H. Horvitz, and G. Ruvkun, "The 21-nucleotide let-7 rna regulates developmental timing in caenorhabditis elegans," *Nature*, vol. 403, pp. 901–6, 2000.

[5] A. Kozomara and S. Griffiths-Jones, "mirbase: integrating microrna annotation and deep-sequencing data," *Nucleic acids research*, vol. 39(Database issue), pp. 152–157, 2011.

[6] S. Griffiths-Jones, H. Saini, S. van Dongen, and A. Enright, "mirbase: tools for microrna genomics," *Nucleic acids research*, vol. 36(Database issue), pp. 154–158, 2008.

[7] S. Griffiths-Jones, R. Grocock, S. van Dongen, A. Bateman, and A. Enright, "mirbase: microrna sequences, targets and gene nomenclature," *Nucleic acids research*, vol. 34(Database issue), pp. 140–144, 2006.

[8] S. Griffiths-Jones, "The microrna registry," *Nucleic acids research*, vol. 32(Database issue), pp. 109–111, 2004.

[9] J. Ding, S. Zhou, and J. Guan, "mirfam: an effective automatic mirna classification method based on n-grams and a multiclass svm," *BMC Bioinformatics*, vol. 12, p. 216, 2011.

[10] C. Suen, "n-gram statistics for natural language understanding and text processing," *IEEE transactions on pattern analysis and machine intelligence*, vol. 1, pp. 164–172, 1979.

[11] H. Frigui and O. Nasraoui, "Simultaneous clustering and dynamic keyword weighting for text documents," in *Survey of Text Mining*, M. Berry, Ed. Heidelberg: Springer, 2004, pp. 45–70.

[12] D. Davies and D. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 224–227, 1979.

[13] F. Sievers, A. Wilm, D. Dineen, T. Gibson, K. Karplus, W. Li, R. Lopez, H. McWilliam, M.Remmert, J. Soding, J. Thompson, and D. Higgins, "Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega," *Molecular Systems Biology*, vol. 7, no. 539, 2011.

[14] H. Frigui and O. Nasraoui, "Simultaneous clustering and attribute discrimination," San Antonio, Texas, 2000, pp. 158–163.

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, "The weka data mining software: an update," *SIGKDD Explorations*, vol. 11, pp. 10–18, 2009.

[16] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, vol. 17, pp. 107–145, 2001.

978-1-4673-4398-5/12/$31.00 ©2012 IEEE