Finding Informative Genes from Multiple Microarray Experiments: A Graph-based Consensus Maximization Model

Liang Ge
The State University
of New York at Buffalo
201 Bell Hall, Buffalo, USA
liangge@buffalo.edu

Nan Du
The State University
of New York at Buffalo
201 Bell Hall, Buffalo, USA
nandu@buffalo.edu

Aidong Zhang
The State University
of New York at Buffalo
201 Bell Hall, Buffalo, USA
azhang@cse.buffalo.edu

***Abstract*—With the rapid advancement of biology technology, many microarray experiments are conducted towards the same problem of finding informative genes. Therefore, it is important to find a set of informative genes integrating multiple microarray experiments that achieves maximal *consensus*. Most previous researches formulated this problem as a rank aggregation problem. In this paper, we propose a novel Graph-based Consensus Maximization (GCM) model to estimate the conditional probability of each gene being informative, then the genes are ranked by this probability. The estimation of the probabilities is formulated as an optimization problem on a bipartite graph, where the criterion function favors the smoothness of the prediction over the graph and penalizes deviations from the initial input ranked lists from microarray experiments. We solve this problem through iterative propagation of probability estimates among neighboring nodes. In addition, when certain genes have already been identified to be informative, it has never been explored in the literature how to take advantage of such information to improve the consensus result. Our proposed GCM model can be naturally extended to incorporate such information, thus increasing the quality of the predicted result. In the experimental evaluation, we conducted experiments on the five prostate cancer microarray studies. The results showed that our model outperformed other baseline methods in finding informative genes. Furthermore, by adding only one piece of information that some gene is informative, our model yielded a significantly better result. The experimental evaluation demonstrates that the proposed GCM model is effective and superior in finding informative genes from multiple microarray experiments.**

I. Introduction

As the advancement of microarray analysis technology, many experiments are conducted to target the same biological or medical problem. In a typical cancer microarray experiment, by comparing gene expression levels between tumor and healthy samples, researchers can identify significantly and differentially expressed genes that are informative of the cancer. Since those experiments were carried out in different laboratories, with different materials and microarray platforms, there exists certain discrepancy between those results. Therefore, it is of great importance to integrate different experiment results into an outcome that achieves the maximal *consensus*. In this paper, we focus on the following problem:

Problem Description: The problem of Finding Informative Genes (FIG) from multiple microarray experiments can be formulated as: Given a pool of n genes $G = \{g_1, ..., g_n\}$, m different experiments are conducted to find the informative genes for the same task (e.g., the prostate cancer). Each experiment is performed on a subset of G and produces a ranking of the participating genes, in which the top-k genes are deemed to be informative. The goal of the FIG problem is to combine different experiment results and produce a ranking of genes that is better than any individual experiment result.

Because of this problem's high resemblance with rank aggregation or meta-search in text mining and information retrieval area, previous researches approach the FIG problem using unsupervised rank aggregation [1][7]. One typical way to formulate the problem is to minimize the distance between the optimal result and the weighted sum of the input experimental results. The distance metric can be Spearman distance [11] or Kendall tau distance [12]. This formulation is shown to be NP-hard [12], thus a number of optimization methods including cross entropy and genetic algorithm are proposed.

With the development of research and experiments, some expressed genes have already been identified to be informative for prostate cancer, including HPN [13], AMACR [14], FASN [15], GUCY1A3 [16], ANK3 [17], STRA13 [18], CCT2 [18], CANX [18] and TRAP1 [18]. Those information is usually referred as the side information in Semi-Supervised Learning. This motivates the problem of **Semi-FIG** where little work has been done: given some genes are known to be informative, how to produce a better result by utilizing this side information.

In this paper, we propose a novel Graph-based Consensus Maximization (GCM) model to tackle the problems of FIG and Semi-FIG. Instead of directly computing the rank of each gene as in rank aggregation methods [1][7], we first estimate the conditional probability of a gene being informative in the final result (i.e., being the *top-k* genes in the final result), then we rank the genes based on this probability. The estimation of the probabilities is formulated into an optimization problem on a bipartite graph, where the criterion function favors the smoothness of the prediction over the graph and penalizes

978-0-7695-4574-5/11 $26.00 © 2011 IEEE
DOI 10.1109/BIBM.2011.34

deviations from the initial input ranked lists from experiments. We solve this problem through iterative propagation of probability estimates among neighboring nodes. For the Semi-FIG problem, our model can be naturally extended to utilize the side information and produce significantly better results.

We conduct experiments on five microarray studies of prostate cancer denoted as Luo [3], Welsh [6], Dhana [2], True [5] and Singh [4]. Two sets of experiments for the FIG and Semi-FIG problems are conducted, respectively. In the experiments for the FIG problem, the GCM model is shown to outperform other rank aggregation baseline methods in that the prediction of the proposed model contains the most known informative genes and the average rank of those known informative genes is the highest. In the Semi-FIG experiments, we show that by providing the side information that one gene is informative, our model yields a significantly improved result over all the results in the FIG experiments. Those encouraging experimental results demonstrate that our model is effective and superior in handling the problem of finding informative genes from multiple microarray experiments.

The major contributions of this paper are

- We propose a novel Graph-based Consensus Maximization (GCM) model to solve the problem of Finding Informative Genes (FIG) from multiple microarray experiments.
- It is the first work to investigate the problem of Semi-FIG: finding informative genes from multiple microarray experiments with side information. Our GCM model is naturally extended to utilize the side information and produces significantly better results.
- In the experimental study of five microarray experiments for prostate cancer, our model is shown to outperform other baseline methods for the FIG problem and yields better results when the side information is utilized.

The organization of the paper is as follows: in Section 2 we describe our Graph-based Consensus Maximization model and propose an iterative algorithm to solve it. The proposed solution propagates information among neighboring nodes until stabilization. In Section 3, we discuss how our model can be naturally extended to utilize the side information. An extensive experimental study is reported in Section 4 where benefits of the approach are illustrated in integrating five microarray studies for prostate cancer. Section 5 discusses the related work and we conclude our work in Section 6.

II. Graph-based Consensus Maximization Model

Let's follow the problem description in Section 1: given a pool of *n* genes $G = \{g_1, ..., g_n\}$, *m* experiments are conducted to find the informative genes. Each experiment is performed on a subset of G and produces a ranking of the participating genes, in which the top-*k* genes are deemed to be informative. For each gene g_i and a given experiment m_j, there are only three possible cases: 1) g_i is deemed to be informative in the experiment m_j; 2) g_i is deemed to be uninformative in the experiment m_j; 3) g_i is not investigated in the experiment m_j. We can readily view each experiment as a classifier for the pool of genes. It classifies the whole pool of genes into three classes: Informative (I), Uninformative(U) and Not-experimented (N). Our ambition is to rank the genes based on their conditional probability of being informative, i.e., in class I. Therefore, the problem is turned into estimating the conditional probability for each gene given *m* experiments.

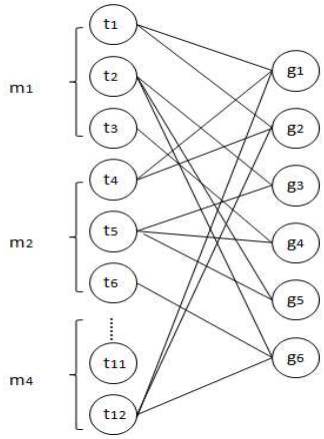

Fig. 1: The bipartite graph for the example

A. Model Formulation

Since each experiment classifies the genes into three classes, we can represent each experiment as three groups. There are *m* experiments, so there are *3m* groups in total. For each gene in the pool, given an experiment m_j, it must belong to one of the three groups representing m_j. This formulates a natural bipartite graph representation. We use the following toy example to illustrate the bipartite graph representation and the model formulation.

TABLE I: Toy Example

Exp.	#1	#2	#3	#4	#5
m_1	1	2	3	5	6
m_2	2	1	3	4	5
m_3	4	3	5	-	-
m_4	3	4	5	-	-

Suppose we have a gene pool of $\{1, 2, 3, 4, 5, 6\}$ and we conducted 4 experiments, the results are shown as in Table I. In each experiment, we dictate that the top-2 genes are the informative genes. Based on our model, the bipartite graph representation of these 4 experiments are shown in Figure 1. The group nodes are on the left side and gene nodes are on the right side. In this bipartite graph, t_1 represents the Informative class , t_2 the Uninformative class and t_3 the Not-experimented class in the experiment m_1. Gene g_2, for instance, belongs to group t_1, t_4, t_9, t_{12} because it is Informative in the experiments m_1 and m_2, Not-experimented in the experiments m_3 and m_4.

We begin to lay down some notation. The affinity matrix $A_{n\times v}$ of the bipartite graph is defined as $a_{ij} = 1$ if gene g_i is assigned to the group t_j and 0 otherwise, where $v = 3 \times m$ is the number of groups. Since we want to estimate the

conditional probability of g_i being informative, the conditional probability is denoted by $U_{n\times c}$. In our model, the conditional probability of each group is also involved, denoted by $Q_{v\times c}$. We have

$u_{iz} = Prob(g_i \text{ is class z}|g_i)$ and $q_{jz} = Prob(t_j \text{ is class z}|t_j)$,

Here c=3, class *z* denotes one of the three classes: Informative, Uninformative and Not-experimented. Also we define the initial class labels for the groups $Y_{v\times c}$ as $y_{jz} = 1$ if the group t_j's class is z and 0 otherwise.

The intuition behind our model is: a group t_j corresponds to class *z* if the majority of genes in this group belong to class *z*, meanwhile a gene corresponds to class *z* if the majority of the groups it belongs to correspond to class *z*. To model such intuition, we are about to optimize the following criterion:

$$\begin{aligned} &\min \sum_{j=1}^{v}\sum_{z=1}^{c}\sum_{i=1}^{n} a_{ij}(q_{jz}-u_{iz})^2 \\ &\text{s.t. } \sum_{z=1}^{c} u_{iz} = 1,\ \sum_{z=1}^{c} q_{jz} = 1 \\ &u_{iz} \in \{0,1\}, q_{jz} \in \{0,1\}, \end{aligned} \tag{1}$$

The criterion in Eq. 1 clearly penalizes the deviation between genes nodes and group nodes. The objective function will achieve its minimum when the gene nodes and group nodes achieve maximal consensus. However, optimization of the objective function in Eq. 1 is a typical binary integer programming problem. It is a notorious NP-hard problem as its decision problem is NP-complete.

To solve this hard problem, we make the following relaxations: first we relax u_{iz} and q_{jz} from binary integer to continuous values in $[0, 1]$. This will enable us to perform gradient operation. Secondly, as a common technique in the machine learning community, we add a regularizer to avoid overfitting. This regularizer also reflects the smoothness assumption that the final prediction for the groups should not deviate too much from their original classes. After the two relaxations, Eq. 1 is turned into the following equation:

$$\begin{aligned} \min J(U,Q) = &\min \sum_{j=1}^{v}\sum_{z=1}^{c}\sum_{i=1}^{n} a_{ij}(q_{jz}-u_{iz})^2 \\ &+ \alpha \sum_{j=1}^{v}\sum_{z=1}^{c}(q_{jz}-y_{jz})^2 \\ &\text{s.t. } \sum_{z=1}^{c} u_{iz} = 1,\ \sum_{z=1}^{c} q_{jz} = 1 \\ &u_{iz} \in [0,1], q_{jz} \in [0,1], \end{aligned} \tag{2}$$

where α is the parameter that expresses the confidence of our belief on the initial labels of the group nodes. $Q^*_{v\times c}$ and $U^*_{n\times c}$ that achieve the minimal value of Eq. 2 are our desired result. Since the first column of $U^*_{n\times c}$ is the estimated conditional probability of each gene being informative, we can directly rank the genes based on this probability.

B. Iterative Algorithm

Since all the variables in Eq. 2 are continues, we can apply gradient operation to obtain its optimal value. There are two matrices U and Q, we therefore propose to iteratively solve the problem. At iteration *t*, we fix U, then the Hessian matrix with regard to Q is a diagonal matrix and its diagonal elements are $\sum_{i=1}^{n} a_{ij} + \alpha > 0$, which means it is convex and $\bigtriangledown J(Q, U^{t-1}) = 0$ gives the global minimum of the objective function in terms of Q. Similarly, fixing Q at iteration *t*, we can verify that the corresponding Hessian matrix is also a diagonal matrix with entries $\sum_{i=1}^{n} a_{ij} > 0$, therefore we can get the global minimum of $J(Q, U)$ in terms of U. The following two equations show such derivations:

$$Q^t = (D_v + \alpha K_v)^{-1}(A^T U^{t-1} + \alpha K_v Y), \tag{3}$$

$$U^t = D^{-1} A Q^t, \tag{4}$$

where $Dv = diag\{(\sum_{i=1}^{n} a_{ij})\}_{v\times v}$, $D_n = diag\{(\sum_{j=1}^{v} a_{ij})\}_{n\times n}$, $Kv = diag\{(\sum_{z=1}^{c} y_{jz})\}_{v\times v}$ and *diag* means the diagonal elements of a matrix. D_v and D_n are the normalization factors. K_v acts as constraints for the group nodes. The whole algorithm is shown in Alg. 1:

Algorithm 1 The GCM algorithm
Input: $A_{n\times v}$, $Y_{v\times c}$, parameter α, ϵ
Output: Consensus matrix U

```
Initialize U^0, U^1 randomly;
t=1;
while || U^t − U^{t−1} ||> ε do
    Q^t = (D_v + αK_v)^{−1}(A^T U^{t−1} + αK_v Y);
    U^t = D^{−1} A Q^t;
    t=t+1;
end while
Output U^t
```

The algorithm depicts the picture that at each iteration, the conditional probability estimate of group node Q receives the information from its neighboring gene nodes while not deviating from its initial value Y. In return, the updated probability estimates of group nodes propagate the information back to its neighboring gene nodes. The propagation stops when the process converges. The process converges to a stationary point.

III. Incorporating Side Information

In recent machine learning and data mining research, side information, usually in the form of labels or constraints, are often leveraged to improve the performance of the original model. We have the similar side information in microarray experiments for prostate cancer. Over years, researchers have identified some expressed genes to be informative for prostate cancer. They are: HPN , AMACR, FASN, GUCY1A3, ANK3, STRA13, CCT2, CANX and TRAP1. Therefore, we focus on the Semi-FIG problem in this section and show that our model can be naturally extended to handle the Semi-FIG problem.

Let $n \times c$ matrix F denote side information, where $f_{iz} = 1$ if gene g_i is given class z and 0 otherwise. Class z could be Informative, Uninformative or Not-experimented. Since only class Informative is valuable in experiments, usually the input side information is that some gene(s) is informative. Then, we could modify Eq. 2 into the following:

$$\begin{aligned} \min J(U,Q) = \min & \sum_{j=1}^{v}\sum_{z=1}^{c}\sum_{i=1}^{n} a_{ij}(q_{jz}-u_{iz})^2 \\ & + \alpha \sum_{j=1}^{v}\sum_{z=1}^{c}(q_{jz}-y_{jz})^2 \\ & + \beta \sum_{i=1}^{n}\sum_{z=1}^{c} h_i(u_{iz}-f_{iz})^2 \\ \text{s.t. } & \sum_{z=1}^{c} u_{iz} = 1, \ \sum_{z=1}^{c} q_{jz} = 1 \\ & u_{iz} \in [0,1], q_{jz} \in [0,1], \end{aligned} \tag{5}$$

where $h_i = \sum_{z=1}^{c} f_{iz}$. Similar to parameter α, β expresses our confidence on the initial classes of the gene nodes. This updated criterion captures the side information by enforcing the constraint that the conditional probability of a gene should be close to its provided class with the parameter β.

After adding the side information constraint, the equation to compute the update of U is changed into the following:

$$U^t = (D_n + \beta H_n)^{-1}(AQ^t + \beta H_n F), \tag{6}$$

$H_n = diag\{(\sum_{z=1}^{c} f_{iz})\}_{n\times n}$. And it is easy to see that the equation to update Q remains unchanged.

IV. Experimental Evaluation

In this section, we experimentally evaluate our model on the FIG and Semi-FIG problems on five prostate cancer studies. We first introduce the five studies and then briefly describe the baseline methods that we compare. After presenting the results on the FIG problem, we show that, for the Semi-FIG problem, by adding small amount of side information, the final result is significantly improved.

A. Data Sets

We investigate the top-25 ranked genes that were found to be up-regulated in prostate tumors compared to normal prostate tissues from five studies: Luo [3], Welsh [6], Dhana [2], True [5] and Singh [4]. These five studies are very discrepant at what genes are deemed to be informative. The detailed data set descriptions can be found in [7].

B. Evaluation Metric

Since we already know that some genes are informative, it is straightforward to evaluate the result in terms of those *ground truth* genes. We propose to evaluate the result by the average rank of those *ground truth* genes and also their number of appearance in the result.

C. Baseline Methods

Borda Count [10]: Given *k* full lists $\tau_1, \tau_2, ..., \tau_k$, Borda's method can be thought of as assigning a *k*-element position vector to each candidate (the positions of the candidate in the *k* lists), and sort the candidates by the L_1 norm of these vectors. There are also other variants of Borda's count: sorting by L_p norm for $p > 1$, sorting by the median of the *k* values, sorting by geometric mean of the *k* values, etc. In this paper, we implemented four variants of Borda count: $Borda_1$ (L_1 norm), $Borda_2$ (median), $Borda_3$ (geometric mean), $Borda_4$ (L_2 norm).

Markov Chain [1][8]: Markov chains can also be used to generate the consensus results. The states of the chain correspond to the *n* genes to be ranked, the transition probabilities depend in some particular way on the given lists, and the stationary probability distribution will be used to sort the *n* candidates. We implemented three different Markov Chains.

Cross Entropy Monte Carlo (CEMC) [7]: CEMC is one representative of the typical ways of solving rank aggregation problem. [7] proposed to employ CEMC method to solve this optimization problem.

D. Experimental Results and Discussions

We applied all the above baseline methods to the five prostate cancer studies. Also we applied our GCM model with parameter $\alpha = 2$.

Table III shows the results evaluated by the metric we proposed in this section. If a *ground truth* gene doesn't appear in the result, we assign its rank to be *k+1* (26 in this case). Also we list the evaluation results for the five prostate studies. As we can see from the result, our GCM model not only contains the most *ground truth* genes (ties with $Borda_3$) but also achieves the best average rank. Borda count methods are better than Luo, True and Singh's result but worse than that of Welsh and Dhana, so do Markov Chain methods. CEMC does achieve good average rank comparing to Borda and Markov Chain methods and only inferior to Welsh's result. The goal of the FIG problem is to ensemble the inputs and produce the result that is better than any individual input list. Our proposed model is the only method to achieve this goal. An interesting finding is that for the CEMC method, the Spearman distance with unweighed case achieves the best performance of all CEMC methods, quite contradictory to the intuition that weighted case is normally better than unweighed case. We show the experimental results in details in Table II. We pick up the variant achieving the best performance from each baseline method. In this table, the *ground truth* is in bold. Borda denotes $Borda_3$, MC denotes MC_3 and CEMC denotes $CEMC_1$.

As seen from the results in Table II, Borda successfully contains 7 *ground truth* genes but comparing with our results, CANX, STRA13 and GUCY1A3 rank relatively low, making its result inferior to ours. In Markov Chain method, the ranking of CANX and STRA13 is quite close to the result of the GCM, yet it fails to include GUCY1A3, causing its average rank lower than ours. CEMC is very good at predicting the rank

TABLE II: Detail Experimental Output

Rank	Borda	MC	CEMC	GCM	Semi-GCM
1	**HPN**	**HPN**	**HPN**	**HPN**	**STRA13**
2	**AMACR**	**AMACR**	**AMACR**	GDF15	**HPN**
3	**FASN**	GDF15	**FASN**	**AMACR**	GDF15
4	GDF15	NME1	GDF15	NME1	**AMACR**
5	NME2	**FASN**	NME2	**FASN**	NME1
6	SLC25A6	EEF2	UAP1	KRT18	**FASN**
7	EEF2	KRT18	OACT2	EEF2	KRT18
8	OACT2	UAP1	SLC25A6	ALACM	EEF2
9	OGT	NME2	KRT18	SND1	OACT2
10	KRT18	SLC25A6	EEF2	OACT2	ALCAM
11	NEM1	OACT2	**STRA13**	**STRA13**	SND1
12	UAP1	**STRA13**	NME1	**CANX**	**CANX**
13	CCND2	**CANX**	**CANX**	GRP58	GRP58
14	CYP1B1	GRP58	ALCAM	TEEM4	TMEM4
15	CBX3	SND1	GRP58	**CCT2**	**CCT2**
16	SAT	MTHFD2	SND1	NME2	NME2
17	**CANX**	ALCAM	FMO5	SLC25A6	SLC25A6
18	BRCA1	MRPL3	TMEM4	CALR	CALR
19	GRP58	TMEM4	**CCT2**	EIF4A1	EIF4A1
20	MTHFD2	PPIB	PRKACA	**GUCY1A3**	**GUCY1A3**
21	**STRA13**	SLC19A1	MTHFD2	LMAN1	LMAN1
22	LGALS3	**CCT2**	PTPLB	OGT	OGT
23	**ANK3**	FMO5	PPIB	STAT6	STAT6
24	**GUCY1A3**	CYP1B1	MRPL3	TCEB3	TCEB3
25	LDHA	ATF5	SLC19A1	LDHA	**ANK3**

of top-ranked genes in input lists: HPN, AMACR and FASN. These three genes rank high in most experiments. And the ranking of STRA13 and CANX is close to that of the GCM model. However, it also fails to include GUCY1A3.

Another interesting discovery across all results is that there are several genes that appear in all the output lists. Those genes, though not deemed informative by now, worth further investigation. They genes are: NME2, GDF15, KRT18, EEF2, OACT2, SLC25A6 and GRP58. Also gene NME1 appears in four of the five output lists.

TABLE III: Experimental Results

Method	Appear #	Avg. Rank
Luo	3	19.4
Welsh	6	13.4
Dhana	5	14.1
True	3	19.8
Singh	4	18.4
$Borda_1$	5	15.2
$Borda_2$	4	17.4
$Borda_3$	**7**	15.8
$Borda_4$	5	15.2
MC_1	5	16.1
MC_2	6	15.5
MC_3	6	14.7
$CEMC_1$	6	14.1
$CEMC_2$	6	14.4
$CEMC_3$	5	14.8
$CEMC_4$	6	14.6
GCM	**7**	**13.2**

Semi FIG Problem: Next, we experimented our method on the Semi-FIG problem. Since we have 9 informative genes at hand, we picked up one gene each time, assigned it to be informative and applied our GCM model with such side information. We set the parameter $\alpha = 2$, $\beta = 8$. The results are shown in Table IV.

TABLE IV: Semi GCM Experimental Results

Gene	Appear #	Avg. Rank
HPN	7	13.2
AMACR	**8**	12.7
FASN	7	13.0
GUCY1A3	7	11.7
ANK3	7	11.3
STRA13	**8**	**12.3**
CCT2	6	13.5
CANX	7	13.2
TRAP1	7	11.6

As seen from the result, adding only one piece of side information significantly improves the quality of the final result. When gene AMACR is assigned to be informative, the GCM model includes 8 informative genes, beating all the previous results. Moreover, gene AMACR is a high-ranked gene in the input lists and therefore improving the result in this way is quite encouraging. We also notice that adding HPN as the side information doesn't increase the performance at all, mainly because HPN is predicted to be top-rank in most input lists. Adding such information will not help propagate the information in our model. Adding CCT2 into our model decreases the performance by both *ground truth* gene appearance times and average rank. This is because gene CCT2 only appears in list True and Singh, adding this information will raise the probabilities of these two lists and equivalently decrease the probabilities of the other three lists, therefore gene GUCY1A3 from Dhana experiment is eliminated from the final result. However, in general, adding side information will help improve the quality of the GCM model. The best results of the Semi GCM model is shown in the last column

in Table II. This result includes gene ANK3 that only appears in the best Borda count methods.

V. Related Work

Because the problem of FIG is of high resemblance with rank aggregation in information retrieval and web search areas, some researchers in bioinformatics tackle the problem by applying state-of-the-art methods from Information Retrieval areas [1][7]. Generally, the problem of rank aggregation is approached in two categories in Information Retrieval and Web Search areas: Unsupervised and Supervised. For unsupervised approaches, they usually express the problem as a distance minimization problem: Given *n* input lists $\tau_1, \tau_2, ..., \tau_k$, the optimal result τ^* satisfies: $\tau^* = argmin \sum_{i=1}^{k} w_i d(\tau_i, \tau^*)$ where w_i is the weight for each input list, d is the distance between two lists. Choices include Spearman distance [11] and Kendall-tau distance [12]. This optimization problem is shown to be NP-hard [12]. Those methods differ in how to solve this optimization. It is noted that the most recent work [7] that studies the same problem employed cross entropy Monte Carlo method to solve the optimization problem.

The rank aggregation can also be formalized as a supervised machine learning task in information retrieval, in which a variety of machine learning techniques have been actively investigated for solving the learning task. However, the supervised approaches are incompatible in solving the FIG problem in bioinfomatics because training set is needed for supervised methods while in bioinformatics the training set implies some genes with precise ranking of their being informative. Such information is infeasible, if not impossible, to get in biological experiments.

In addition, the problem in this paper is related to clustering ensemble or consensus clustering: given a number of clustering results, find a consensus result that is better than any single input clustering result. Jing. etal. [9] proposed a bipartite graph method to solve consensus maximization problem in which given a bunch of class labels and clustering results, find the consensus labels that achieves maximal consistency between them. Our work is similar to Jing's work in that we both enforce the information propagated between neighboring nodes in the bipartite graph.

VI. Conclusion

In this paper, we studied the problem of finding informative genes from multiple microarray experiments. Rather than directly computing the rank of each gene, we first estimated the conditional probability of a gene being informative in the final result, then we ranked the genes based on this probability. We proposed a Graph-based Consensus Maximization (GCM) model where the estimation of the probabilities is cast as an optimization problem on a bipartite graph. We solved this problem through iterative propagation of probability estimates among neighboring nodes. Furthermore, our GCM model can be naturally extended to incorporate the side information that some genes have already been identified to be informative. In the experimental study, we compared our model with three major baseline methods and their variants. The experimental evaluation showed that in the FIG problem, the GCM model outperformed the other baseline methods. In addition, by adding only one piece of side information, the quality of results of our model was greatly improved, showing that our proposed GCM model is effective and superior in finding informative genes from multiple microarray experiments.

References

[1] DeConde, R.P., Hawley, S., Falcon, S., Clegg, N., Knudsen, B., and Etzioni, R. Combining results of microarray experiments: a rank aggregation approach. *Statistical Applications in Gene. and Mole. Biology,* , 2001

[2] Dhanasekaran, S.M., Barrette, T.R., Ghosh, D., Shah, R., Varambally, S., Kurachi, K.,Pienta, K.J., Rubin, M.S., and Chinnaiyan,A.M. Delineation of prog-nostic biomarkers in prostate cancer. *Nature*, 412, 822-826, 2001

[3] Luo, J., Duggan, D.J., Chen, Y., Sauvageot, J., Ewing, .M., Bittner, M.L., Trent, J.M., and Isaacs, W.B. Human prostate cancer and benign prostatic hyperplasia: molecular dissection by gene expression profiling. *Cancer Research*, 2001

[4] Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., DAmico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantok, P.W., Golub, T.R., and Sellers, W.R. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 2002

[5] True, L., Coleman, I., Hawley, S., Huang, A., Gikord, D., Coleman, R., Beer, T., Gelman, E., Datta, M., Mostaghel, E., Knudsen, B., Lange, P., Vessella, R., Lin, D., Hood, L., and Nelson, P. A Molecular Correlate to the Gleason Grading System for Prostate Adenocarcinoma. *Proceedings of the NSA*, 2006

[6] Welsh, J.B., Sapinoso, L.M., Su, A.I., Kern, S.G., Wang-Rodriguez, J., Moskaluk, C.A., Frierson, H.F.Jr., and Hampton,G.M. Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. *Cancer Research*, 2001

[7] Shili, L., Jie, D. Integration of Ranked Lists via Cross Entropy Monte Carlo with Applications to mRNA and microRNA Studies, 2008, Technical Report

[8] Dwork, C., Kumar, R., Naor, M., and Sivakumar, D. Rank aggregation methods *www01*, 2001

[9] Jing, G., Feng, L., Wei, F., Yizhou, S., Jiawei, H. Graph-based Consensus Maximization among Multiple Supervised and Unsupervised Models,*NIPS*,2009

[10] J. C. Borda. Memoire sur les elections au scrutin. Histoire de l'Academie Royale des Sciences, 1781.

[11] P. Diaconis and R. Graham. Spearman's footrule as a measure of disarray. *J. of the Royal Statistical Society, Series*, 1977

[12] J. J. Bartholdi, C. A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 1989

[13] Klezovitch, O., Chevillet, J., Mirosevich, J., Roberts, R., Matusik, R., Vasioukhin, V. Hepsin promotes prostate cancer and metastasis. *Cancer Cell*, 2004

[14] Kuefer, R., Varambally, S., Zhou, M., Lucas, P.C., Loeffler, M., Wolter, H., Mattfeldt, T., Hautmann, R.E., Gschwend, J.E., Barrette, T.R., Dunn, R.L., Chinnaiyan, A.M., Rubin, M.A. alpha-Methylacyl-CoA racemase: expression levels of this novel cancer biomarker depend on tumor differentiation, 2002

[15] Pizer, E.S., Pflug, B.R., Bova, G.S., Han, W.F., Udan, M.S., Nelson, J.B. Increased fatty acid synthase as a therapeutic target in adrogen-independent prostate cancer progression. *Prostate* 47(2):102-110, 2001

[16] Dong, Y., Zhang, H., Gao, A.C., Marshall, J.R., Ip, C. Androgen receptor signaling intensity is a key factor in determining the sensitivity of prostate cancer cells to selenium inhibition of growth and cancer-specific biomarkers. *Mol. Cancer Ther* 4(7):1047-1055. 2005

[17] Ignatiuk, A., Quickfall, J.P., Hawrysh, A.D., Camberlain, M.D., Anderson, D.H. The smaller isoforms of ankyrin 3 bind to the p85 subunit of phosphatidylinositol 3'-kinase and enhance platelet-derived growth factor receptor down-regulation. *J Biol. Chem.* 281(9):5956-5964, 2006

[18] Ivanova, A., Liao, S.Y., Lerman, M.I., Ivanov, S., Stanbridge, F.J. STRA13 expression and subcellular localisation in normal and tumour tissues: implications for use as a diagnostic and differentiaion marker. *J Med. Genet.* 42(7):556-576. 2005