Boğaziçi University

Computer Engineering Department

CmpE 492: Graduation Project

Final Report

Version 1.0

Bayesian Medical Expert Systems

Student:
Mert Cem Taşdemir

Advisor:
Assoc. Prof. Ali Taylan Cemgil

June 5, 2017

Abstract

The project consists of two essential parts. One of them is to find the best set of diagnosis given a set of symptoms. The other is to find the best question to be asked to the patient to find the diagnosis. For the first part an exact inference algorithm called Quickscore[1] is used. For the second part, 4 question generation strategies tested using inference engine built for the first part.

Contents

1 Introduction

1.1 Motivation

The aim of this project is to design Bayesian Medical Expert Systems(MES) that can be used in differential diagnosis of a particular set of diseases given a set of symptoms. My main motivation is to delve into the pround notion of Bayesian sense and use the knowledge I gather in ML applications that I find interesting and challenging. In this project, I have taken a model built for a similar -if not identical- purpose, as the starting point and tried to understand what it does. I may say that I have accomplished the initial aim, by re-implementing the model. From now on, I will modify, or probably change, the model so that we can get better results in both question asking and inference. In this report I will describe what I have done so far and what I'm planning to do. Besides, I will present the test results of the initial model.

1.2 State of the Art

The field was initially tried to be plowed using knowledge-base systems, that is, it was driven by knowledge, rather than data at the beginning. Yet, due to the problems in knowledge representation, etc. the cutting edge was pushed to data-oriented systems. But, this brings another problem out, which is inadequacy of data. And this is the reason why the field is not mature enough. Even though the records are digital, the data collection is the foremost problem, as either the records are taped negligently or they are not shared even in an anonymised format.

Usually Bayesian Networks are used in this field. Since it is highly possible not to observe some symptoms even though the disease is present, using probabilistic models helps the program output to be more accurate.

Promedas[2] might also be equivalent to what we desire to accomplish and it is built on the same network model as we used.

1.3 Description

The project can be described in a nutshell as follows: In the approach we have inherited, MAP estimation is used to determine the best disease configurations that fit to the model given a set of symmtoms. And the graphical model is built on noisy-OR gates, that is, with the given states of a set of diseases, the probability of observing a symptom is determined using their non-occurence probabilities when those diseases are present. Besides the network model and the MAP estimation used for inference, another significant purpose of the project is to minimize the number of questions asked to find the posterior probability. In other words, we try to maximize the knowlenge we gather by knowing minimum number of states of symptoms.

1.4 Outline

In the model, we use Bayesian Inference to find maximum posterior value of disease occuring when given few observations of symptoms are present. The reason we use a Bayesian -or probabilistic, approach is basically that it's almost sure that there exist multiple missing data. *A small note here can be that the model should be used for a particular set of diseases because of the complexity of the system. Yet, the model should also be generic, that is, it should be applied to different sets.*

1.5 Data Model

We have synthetic data which is obtained using the following Generative Model:

$$ds_{ij} \sim \mathcal{BE}(\big[0,1\big]; \pi_0, 1-\pi_0)$$

where $\pi_0 \gg 1-\pi_0$ and ds_{ij} indicates if the symptom i is present for d_j, disease j. ds_{ij} i 1 when the symptom s_i is observed for d_j, and 0 when not.

2 Methodology

2.1 Ethical

One should initially discuss the ethical issues first when the object is to develop a MES. First of all, there might be catastrophic consequences if the MES fails because different diseases usually require different treatments and treatment for one might be fatal for a patient who has the other disease in her/his body, no matter how close the diseases are in reflection.

Second, the data used to train the MES have to be protected carefully. The reason is that, despite the fact that there won't be no name or any other identified on the data, the data itself might be unique to a person, that is, the patient's identity might be revealed by the data. This has to be forestalled though the probability of such an event approaches to zero.

2.2 Structural

While UzmanDoktar [3], the role model of this project, has the number of diseases on the order of 100 and the number of symptoms on the order of 500, the number of diseases are on the order of 30000 and the number of symptoms are on the order of several thousands for a human being. Hence, it's definitely not efficient to consider all the diseases at once. Instead, the system will be useful for sets of diseases that are similar to each other and need to be differentiated from each other. By doing this, the MES will not tell that: "either there is nothing wrong with the patient, or the patient has flu, or s/he has terminal stage lung cancer. That is, our aim is to differentiate diseases like pneumonia from rhinovirus related upper respiratory system infection. They have a common reflection, but can be distinguished from each other.

2.2.1 Network Model & Approach

In our Bayesian Network(BN), we tried to estimate maximum the probability of observing a disease combination d^* , given set of symptoms $s_{1:k}$ where $k < j$, and d^* consists of $d_{1:m<t}$'s where t is a constant that reflects the maximum number of concurrent diseases *and we took it* 3 *for the sake of computational complexity*. In other words, we tried to make a *Maximum a-Posteriori(MAP)* estimation.

$$d^* = \arg\max_d p(s|d)p(d)$$

Yet instead, we use negative-logposterior($\mathcal{L}$) for calculations, since we are also trying to minimize the entropy, that is, we want the disease to be close to the given symtoms.

As mentioned in the introduction, the Noisy-OR Network Model is used to calculate the likelihood: $p(s|d)$. The model is chosen to take advantage of independence of the causality of the symptoms -or namely diseases.

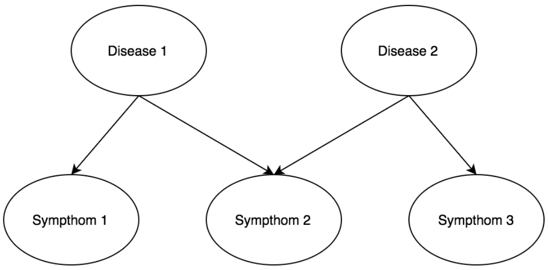

Figure 2.1: Example Bayesian Network(BN) model taken from UzmanDoktar[3, pg. 6]

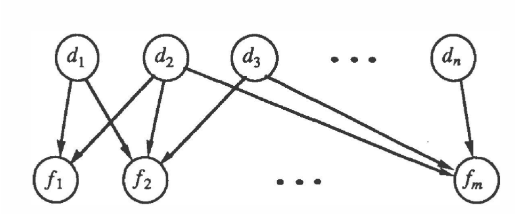

Figure 2.2: Two-layer multiply connected Bayesian Network-Generic model[1]

Probability of not observing s_i given a diease set d is:

$$p(s_i = 0|d) = \theta_0 \prod_j \theta^{ds_{ij}d_j}$$

Probability of observing s_j given a diease set d is:

$$p(s_i = 1|d) = 1 - \theta_0 \prod_j \theta^{ds_{ij}d_j}$$

where d is the set of diseases that is given and for which the probability of observing s_i is calculated and θ_0 is the probability of not observing a symptom when no disease related to it is present.

Now, suppose we know the values -observed, not observed- of some symptoms. Then multiply the probabilities of observing the observed ones and not observing the not observed ones and plug this into **??**. The model is a-priori independent, that is, the diseases are equiprobable.

2.2.2 Question Asking Strategies

2.2.2.1 Relative-Entropy Based

In this approach we want to choose a question *-or actually a symptom,* that reduces entropy more than any other question. In other words, we tried to find a question that can provide the most divergence from the current state. What we are looking for is a question that provides more information than the other if we know whether it's observed or not observed. The measure here is the Kullback-Leibler(KL) Divergence, so we want the posterior to be as far from the prior as possible.

This method is an informative one, since it learns from the data that the patient provide. But this also brings the computational complexity issues, as the complexity grows exponantially with the size of the network.
The algorithm is:

1. Calculate the probability of observing a diagnosis before the examined symptom is known
2. Calculate the probability of observing the diagnosis when the examined symptom is known
3. Compare the relative-entropy of the first two result
4. Choose the unknown symptom that maximizes the 3^{rd} part

2.2.2.2 Symptom Based

In this approach, we sort all the symptoms according to the number of their occurance in diseases and ask questions to the patient in this order. This approach is a non-informative one, that is, this does not learn anything from patient.
The algorithm is:

1. Rank the symptoms according to the number of diseases with which they are related
2. Chose the unknown symptom that has the highest rank

2.2.2.3 Disease Based

In this approach, we sort the diseases according to the number of symptoms that they cause. Afterwards, we ask to the patient whether the symptoms of the first diseases are present. When all the questions of this disease is asked, we continue asking the symptoms of the next disease, and so on so forth. This is also a non-informative approach, since it's only concentrated on the initial setting.
The algorithm is:

1. Rank the diseases according to the number of symptoms with which they are related

2. Choose the highest ranked disease
3. Ask the symtoms related to the chosen disease
4. When there is no symptom related to the chosen disease, choose the disease that has the highest rank amongst the diseases that are not examined yet

2.2.2.4 Hybrid of first two

Since the relative-entropy based strategy is computationally infeasible in large networks, a heuristic is devised based on the symptom based question generation approach.
The algorithm is:

1. Rank the symptoms according to the number of disease with which they are related
2. Choose choose highest X percentage of the symptoms
3. Calculate relative-entropy for chosen diseases
4. Choose the question that has the maximum relative-entropy i.e. causes maximum Shannon Entropy reduction
5. Note: X needs to be optimized, 25% seems like a good initialization

3 Evaluation

3.1 Test Results

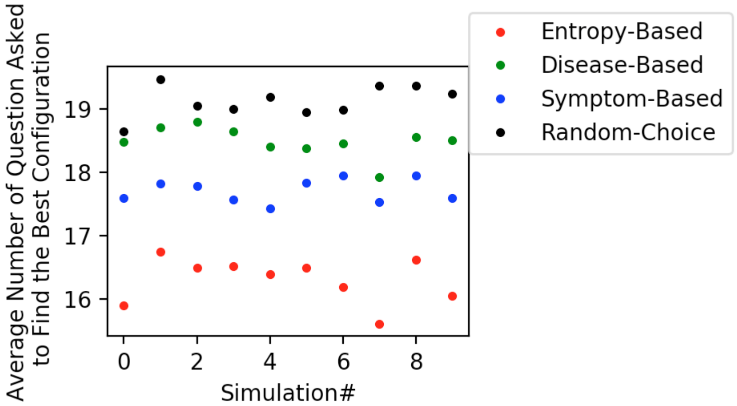

Figure 3.1: Small Network Test Result

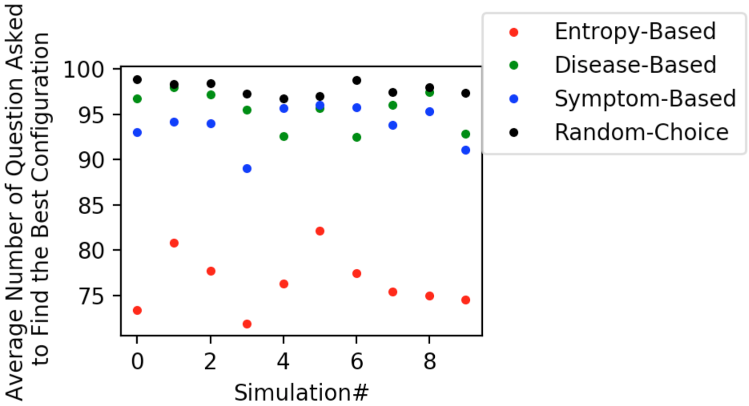

Figure 3.2: Large Network Test Result

The test configuration that give the output shown in Figure3.1 is a network with 10 diseases, and 20 symptoms. The test consists of 10 simulations. Each simulation indicates the average number of questions asked to receive the best disease configuration that one can get from the model given a set of symptoms for 100 runs. That is, we create 100 different symptom set and create the best disease configuration the model outputs. Then for each question asking strategy, we try to reach the best and count the number of questions to make it there for those symptom sets and get the average of those 100 for each question asking strategy.

The testing methodology is similar for the output shown in Figure3.2, but for the latter one we create 20 simulations and each simulation consists of 10 runs.

It's obvious that the Entropy-Based strategy beats all the others. From the figure, we may conclude that using that strategy, knowing 75% of the symptoms would be enough to infer the disease configuration that our network model outputs.

The other two strategies are doing slightly better than the random question generation. Those two are not as informative as the one mentioned above. It's because the orders of questions generated by those strategies don't change as the model learns from the symptoms. They only care about the initial setting of the model, and not information received from the data provided by the patient.

The hybrid one slightly outperforms the non-informative strategies and fails to beat relative-entropy based strategy. Yet it's computationally quite more efficient than the relative-entropy-based strategy.

When testing, the probability of missing a symtom when a disease is present is taken as 0.02 and probability of not observing a symptom when a disease is not present as 0.95. Lastly, the model is prior independent, that is prior probability of each disease is $1/\#of diseases$.

4 Conclusion

4.1 Future Work

Firstly, new question generation strategies will be applied to the same model. Even though the results that the relative entropy based strategy brought seems quite good, it's computationally infeasible as the number of computations grow exponantially with the number of parameters.

As mentioned, the number of concurrent diseases is assumed to be limited in our model. This restriction should be removed. So, we will implement the Heckermann's quickscore for small networks and a Gibbs sampler for larger ones.

Lastly, a parallel version of the relative entropy based strategy and the used exact inference method -Heckerman's Quickscore[1] will be implemented as it seems highly parallelizable.

Bibliography

[1] David Heckerman. A tractable inference algorithm for diagnosing multiple diseases. In *UAI-89*, pages 163–172, 1989.

[2] Promedas - probabilistic medical diagnostic advisery system.

[3] Hıdır Yüzügüzel and A. Taylan Cemgil. Uzmandoktar: An expert system for diagnosis of agricultural plant diseases. Boğaziçi University, 2013.

[4] Hıdır Yüzügüzel, A. Taylan Cemgil, and Emin Anarım. Query ranking strategies in probabilistic expert systems. In *22nd Signal Processing and Communications Applications Conference*, 2014.

[5] David Barber. Bayesian reasoning and machine learning, feb 2017.

[6] David Barber. Machine learning - a probabilistic approach, 2006.

[7] Tommi S. Jaakola and Michael I. Jordan. Variational probabilistic inference and the qmr-dt network. *Journal of Artificial Intelligence Research*, 10:291–322, may 1999.

[8] Feili Yu, Fang Tu, Haiying Tu, and Krishna Pattipati. Multiple disease (fault) diagnosis with applications to the qmr-dt problem. In *Proc. IEEE Int. Conf. Syst. Man Cybern.*, pages 1187–1192, Washington, DC, nov 2003. IEEE.

[9] William J. Long. Medical informatics: reasoning methods. *Artificial Intelligence in Medicine*, 23(1):71–87, aug 2001.

[10] Ethem Alpaydin. *Introduction to Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press, 2014.

[11] Bastian Wemmenhove, Joris M. Mooij, Wim Wiegerinck, Martijn Leisink, Hilbert J. Kappen, and Jan P. Neijt. Inference in the promedas medical expert system.

[12] Adam Zagorecki and Marek J. Druzdzel. Knowledge engineering for bayesian networks: How common are noisy-max distributions in practice? *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS*, 43(1), jan 2013.

[13] Jonathan D. Nelson. Finding useful questions: On bayesian diagnosticity, probability, impact, and information gain. *Psychological Review*, 112(4):979–999, 2005.

[14] Wim Wiegerinck, Bert Kappen, and Willem Burgers. Bayesian networks for expert systems, theory and practical applications.

[15] Igor Kononenko. Machine learning for medical diagnosis: History, state of the art and perspective. *Artificial Intelligence in Medicine*, 23(1):89–109, aug 2001.

[16] S R Soni, A Khaunteta, and M Gupta. A review on intelligent methods used in medicine and life science. In *International Conference and Workshop on Emerging Trends in Technology, Mumbai, India, February 25–26, 2011*, ICWET '11, pages 703–706, New York, NY, USA, 2011. ACM.