A Novel Probabilistic Approach for Efficient Information Retrieval

Sonia Bansal and Reena Garg
YMCA University of Science and Technology, Faridabad, Haryana, INDIA

ABSTRACT
Information on the World Wide Web is increasing tremendously. To get the relevant information from very large data sets is essential. In traditional retrieval systems, the query is given to large corpus to retrieve the relevant documents. The traditional models for information retrieval are just one subclass of retrieval techniques that have been studied in many years. Although many techniques share common characteristics in the information retrieval hierarchy, they all share a core set of similarities that justify their own class and these algorithms are design for isolated datasets. But in most of cases, relationships among different datasets are always existed. A new probabilistic Hidden Markov model is proposed and based on this model new information retrieval (IR) technique is presented. Hidden Markov models (HMMs) are widely used in science, engineering and many other areas. In a HMM, there are two types of states like hidden states and observable states. HMM is powerful modeling of context as well as the current observations. Hidden Markov model is finite state machine which offer a good balance between simplicity and expressiveness of context. IR is performed by determining the sequence of states that was most likely to have generated the entire document, and retrieving the information that were associated with certain designated target states. Determining this sequence is efficiently performed by dynamic programming with the Viterbi algorithm.

Keywords: Information Retrieval, Statistical model, Hidden Markov Model, Viterbi algorithm

1. INTRODUCTION

Information Retrieval in most cases is searching relevant information. The process of retrieving information from the result pages yielded by a search engine is termed as web information extraction. Searching interesting information is one of the most important tasks in Information Retrieval (IR). An IR system accepts a query from a user and responds with a set of documents. The system returns both relevant and non-relevant material. Generally a search engine presents the retrieved document set as a ranked list of document titles. The documents in the list are ordered by the probability of being relevant to the user's request. The highest ranked document is considered to be the most likely relevant document; the next one is slightly less likely and so on. This organizational approach can be found in almost any existing search engine [5,7,9]. It is assumed that the user will start at the top of the list and follow it down examining the documents one at a time. There are many models developed for information retrieval. The majority of IR system is based on the Boolean model. The vector model is the most frequently used in experimental environment. Apart from this other models are connectionstic, fuzzy-logic, semantic, Rule based, cluster and probabilistic model. The obtained information can be less of completeness and accuracy because the user may unintended lose certain information hiding within the text. There are many real world information searching tasks that absolutely require syntactic information and yet there are restricted enough to be traceable. As a result, one normally has to put considerable effort to further review the web search outcomes so as to filter out the unnecessary data [2]. Such tasks can be very time-consuming. Also, the obtained information can be less of completeness and accuracy because the user may unintended lose certain information hiding within the text. More precise and automatic information retrieval technique is clearly required in order to achieve accurate information retrieval in a smarter way. In this paper we proposed statistics-based methods, the Hidden Markov Model (HMM) which has strong theoretical foundation with a well-established training algorithm and HMM can process data quite robustly. In this paper we present the Hidden Markov Chain to capture both static and dynamic data for efficient retrieval by using similarity coefficient. The remainder of this paper is organized as follows: A brief review of HMM is presented in Section 2. Section 3 presents Hidden Markov Model. Section 4 describes the framework of Information retrieval using HMM and Viterbi algorithm . Finally, Section 5 concludes the paper.

2. RELATED WORK

With respect to information retrieval, it may be viewed as a process of selecting documents from a collection according to the presence of keywords assigned by an indexer, while information extraction may be defined as a type of concept extraction that automatically recognizes significant vocabulary items [12]. Currently, the function of most web search engines are more close to fundamental information retrieval in which the user inputs keywords then obtains the outputs through word-matching process. If the user is more interested in information extraction, the results returned by web search engines will be mostly too rough to truly fulfil the user's need due to the limited mining capability to those search engines. A number of alternative document organization approaches have been developed over the recent years [1, 4, 10, 16]. These approaches are normally based on visualization and presentation of some relationships among the documents, terms, or the user's query.

Especially, when dealing with semi-structured or free-structured documents, wrapper normally performs more conservative [8]. As a result, automatic training models, which are more flexible and robust in handling free-structured information extraction, have attracted extensive interest. Among those, Hidden Markov models can be regarded as a symbolic representative [11]. Although originally HMM was mainly implemented for speech recognition [13], in recent years the robustness of HMM have also been extended to the field of information extraction and have contributed preliminary success. The Hidden Markov Model (HMM) is a popular statistical tool for modelling a wide range of time series data. In the context of natural language processing(NLP), HMMs have been applied with great success to problems such as part-of-speech tagging and noun-phrase chunking. Sigletos, Paliouras, and Karkaletsis used HMM for role identification in the field of financial articles [14]. More recently, Ching, et. al. adopted HMM for customer relationship management [3]. Song, Song, Hu, and Allen [15] extended HMM to the field of mining biomedical information. It can be seen that the studies shown above in using HMM for information extraction are mostly focusing on dealing with structured or semi-structured data, while free-structured text remains largely unexplored. This study therefore concentrates on applying HMM's to deal with free texts.

As cases are specific in nature, it is not efficient to cluster documents using keywords or links alone. To date, most strategies are based only on static information, whereas they should incorporate dynamic links and also utilize access patterns. For example, if two documents are frequently retrieved one after the other, it is advantageous to store the images in close proximity. In practice, the user is usually prompted (either by link or query) to next retrieval by the information currently on view. Thus, the decision to retrieve the next document is based primarily on the last retrieval. Barristers and lawyers repeatedly tend to retrieve those documents that are currently most relevant. For a period of time therefore, the document access patterns and relevant links display marked similarity. However, document selection evolves naturally as the case develops.

3. HIDDEN MARKOV MODEL

A Hidden Markov Model (HMM) is a statistical model in which the system is assumed to be a Markov process with unknown parameters and the hidden parameters are found from the observable parameters. In a HMM, the state is not directly visible, but variables influenced by the state are visible. Each state has a probability distribution. HMM is a popular statistical tool for modelling a wide range of time series data. HMM distinguished from a general Markov model in which the states in an HMM cannot be observed directly (i.e. hidden) and can only be estimated through a sequence of observations generated along a time series (or called steps). The probability of the system being a particular state at time t depends both on the probabilities of states at immediately preceding time t-1 (i.e. Markovianity) and the observation drawn at time t. Assume the total number of states being N, and let both qt and ot each denotes the system state and the observation at time t. An HMM, k, can be formally characterized by three types of parameters, namely, A, B, and Π, where A is a matrix of transition probability between states, B is a matrix of observation probability densities relating to states, and Π is a matrix of initial state probabilities, respectively. Specifically, matrices A, B, Π each is further represented as

A = {aij = P(qj at t+1 | qi at t)}, where P(a | b) is the conditional probability of a given b, $t \geq 1$ is time, and $qi \in Q$.

- Informally, A is the probability that the next state is q_j given that the current state is q_i.
- $B = \{b_{ik} = P(o_k \mid q_i)\}$, where $o_k \in O$.
- Informally, B is the probability that the output is o_k given that the current state is q_i.
- $\Pi = \{p_i = P(q_i \text{ at } t=1)\}$.

HMM model and relating parameters namely, A, B, and Π, are shown in Fig. 1.

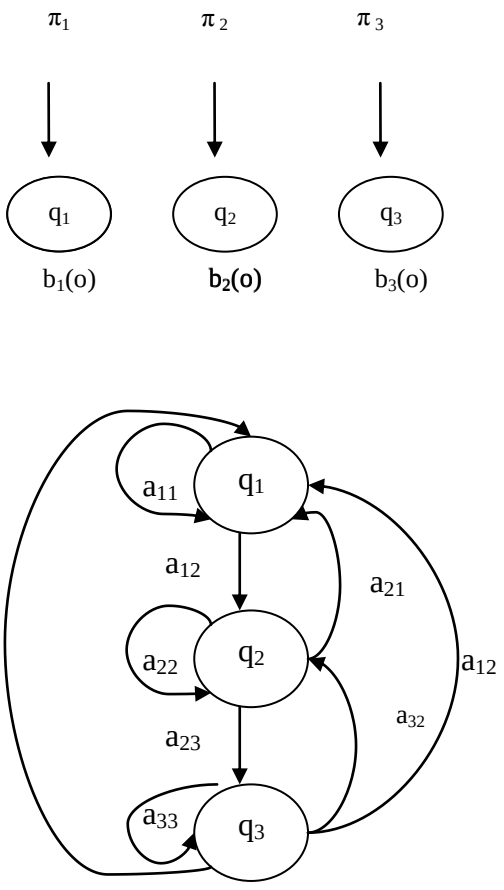

Fig. 1. HMM structure and parameter matrices A, B and Π

Given an HMM λ and observation sequence $O = \{o_1, o_2, \ldots, o_T\}$, one may obtain the hidden states in terms of a variety of optimality criteria. For example, one may choose the optimality criterion which maximizes the expected number of correct individual

states, or to maximize the expected number of correct pairs of states, and so on. However, the most widely used criterion is to trace the best state sequence Q* = $\{q_1,q_2, \ldots ,q_T\}$ along the whole HMM structure using Viterbi dynamic programming approach so as to maximize P(Q*|, O ,λ) [53]. In order to make Q* meaningful, one has to well set up the model parameters A, B, and Π. The purpose of such training process is to estimate the well-suited model parameters so as to make P(O| λ) maximized, i.e. maximize the probability of observations O under the model λ.

4. INFORMATION RETRIEVAL BASED ON HMM

4.1. Framework Design

In order to well perform the information retrieval tasks through HMM, the HMM hidden states have to be carefully determined. The success of building up a good HMM relies on the user's understanding to both the retrieval items of interest and document organizations and is shown in Fig. 2. In this framework on the basis of query the relevant and non relevant information is extracted by applying the similarity measures.

This retrieval and non retrieval information is pass to the Hidden Markov Model where the training set is present. The HMM model is performed in two parts. The first part focuses on unique term of interest. For each target term, a corresponding HMM is built exclusively serving for the extraction task of that term. While the second stage resolves a more complex, multiple terms, defense acquisition information retrieval issue. After finding the most relevant information, the ranked list of documents is produced to the user.

The ways of state transitions is quite important, since it determines whether the underlying state transitions can fit the extraction issue or not. The previous studies [6,18] only allows background state outgoing transition to the prefix state and incoming transition only from the suffix state. However, in this study, the state transitions are allowed between any states because, within the free-structured format, the states are distributed in several regions of the text and must be extracted in fragments. Once the state transition structure is determined, the remaining issue is to determine the suitable parameters. For the HMM structures with the observations determining a unique path through the states, the model parameters are estimated by maximum likelihood with the ratios of counts from training samples. Once the model parameters have been resolved, one can then use Viterbi algorithm to dig out the hidden states.

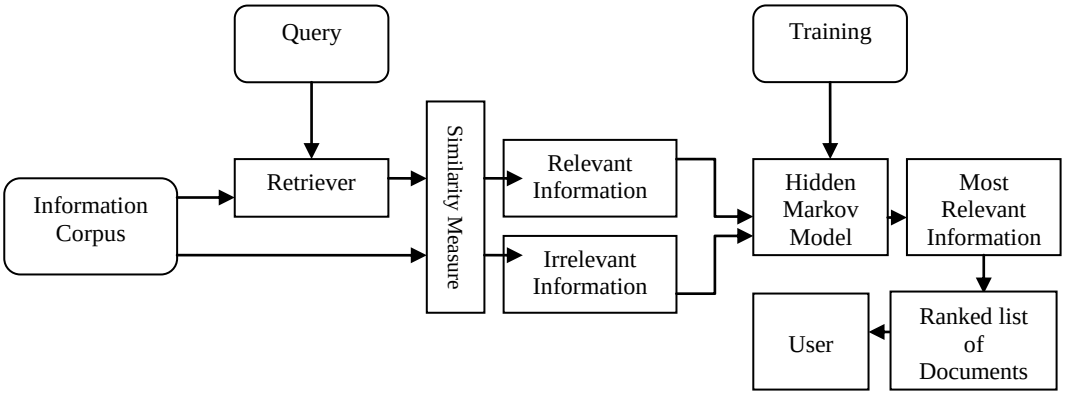

Fig. 2. Framework of HMM Information Retrieval

4.2 Viterbi Algorithm

The algorithm makes a number of assumptions. First, both the observed events and hidden events must be in a sequence. This sequence often corresponds to time. Second, these two sequences need to be aligned, and an instance of an observed event needs to correspond to exactly one instance of a hidden event. Third, computing the most likely hidden sequence up to a certain point *t* must depend only on the observed event at point *t*, and the most likely sequence at point $t - 1$. These assumptions are all satisfied in a first-order hidden Markov model. The first part of the assignment is to build an HMM from data. Recall that an HMM involves hidden state that changes over time, as well as observable evidence, henceforth called the output of the HMM. An HMM is defined by three sets of probabilities:

Step 1. Exhaustive search for a solution: For each state *s*, the probability of observing each output retrieved *o* at that state ($P(E[t]=o \mid X[t]=s)$)

Step 2. Reducing complexity using recursion: From each state *s*, the probability of traversing to every other state *s'* in one time step ($P(X[t+1]=s' \mid X[t]=s)$)

- Partial probabilities (δ's) and partial best paths: Thus δ(i,t) is the maximum probability of all sequences ending at state i at time t, and the partial best path is the sequence which achieves this maximal probability.
- Most probable sequence of hidden states: The

best interpretation given the entire context of the observations and decide the execution sequence

Step 3. Distribution over the start state ($P(X[0])$).

Regarding step 3, there is a single dummy start state, distinct from all other states, and to which the HMM can never return. Even so, there is need to estimate the probability of making a transition from this dummy start state to each of the other states.

For step1 and 2, compute estimates of these probabilities from data. Here training data is provided which consist of one or more sequences of state-output pairs, i.e., sequences of the form $x[1]$, $e[1]$, $x[2]$, $e[2]$, ..., $x[n]$, $e[n]$. During this training phase, The state variables are visible. Given these sequences, estimate the probabilities that define the HMM. For instance, to estimate the probability of output *o* being observed in state *s*, you might simply count up the number of times that output *o* appears with state *s* in the given data, and divide by a normalization constant and hence the probabilities of all outputs from that state add up to one. In this case, that normalization constant would simply be the number of times that state *s* appears at all in the data. Athough this approach corresponds to the meaning of a conditional probability, when making estimates of this sort; it is often preferable to smooth the estimates. These outputs are used for analysis purpose. Next section explains the accuracy parameter.

4.3 Measurement for Performance Evaluation

The performances are evaluated in terms of the precision and recall [4], which are widely used to evaluate information retrieval and extraction systems. Precision defines the correctness of the data records identified while recall is the percentage of the relevant data records identified from the web page. These measures for information extraction define a correspondence between the extracted items and facts within the documents. Precision answers the question that, for every item in the extracted outcomes, if there is a corresponding fact in the documents.
Recall rate naturally corresponds to the question that, for every fact in the documents, if there a corresponding item shown in extracted outcomes. Precision and Recall rate are expressed as

$$\text{Precision} = \frac{\alpha}{\alpha + \beta}$$

$$\text{Recall rate} = \frac{\alpha}{\alpha + \gamma}$$

where α denotes the number of relevant items matching the facts, β denotes the number of non relevant items, respectively and γ denotes the number of facts failing to be retrieved. Precision and Recall rate have natural correspondences to both the development cycle and the user's environment.
In the user's environment, low recall can be fixed by increasing the redundancy of the corpus, and low precision can be improved by adding more constraints in the system processing loop. However, if a user is interested in taking care of both precision and recall, the F-measure can be a good propose and is expressed as

$$\text{F -measure} = \frac{2PR}{P + R}$$

where P denotes precision and R the recall rate, respectively.
F-measure exhibits the desirable properties of being highest when both recall and precision are high. In this study, the above measures all have been included for evaluating the extraction performance in order to obtain a more objective view into the retrieval process.

5. CONCLUSION AND FUTURE WORK

In this paper we present the concept of Markov model based on both the static and dynamic characteristics of document retrieval. This model has important implications to improve document retrieval speeds. This model allows all observation symbols to be emitted from each state with a finite probability, which makes the model much more expressive and able to better represent the retrieved information. One state depends upon the other hence there is relation between one state to another state so it provide a good basis for actual physical partitioning of the network of documents. Simple estimation methods for the transition probabilities among the hidden states are discussed. The estimation methods are better than the traditional algorithm in both the quality of estimation and the computational complexity. The HMM model is well defined for information retrieval from the relational information. In future try to implement this method for the large datasets and hyperlinks. Moreover extended method will be explored for efficient retrieval of information from the large corpus.

REFERENCES

1. A. Leuski and J. Allan (2000). Evaluating a visual navigation system for a digital library. International Journal on Digital Libraries, 3(2),170-184.

2. B.S. Everitt (1993), Cluster Analysis, 3rd edn., Edward Arnold, University Press, Cambridge, UK.

3. Ching, W., Ng, M., & Wong, K. (2004). Hidden Markov model and its applications in customer relationship management. IMA Journal of Management Mathematics, 15, 13–24.

4. D. Dubin(1995). Document analysis for visualization. In Proceedings of ACM SIGIR, pages 199-204.

5. D. Harman and E. Voorhees (1997), editors. The Fifth Text REtrieval Conference TREC-5). NIST.

6. Freitag, D., & McCallum, A., (2000). Information extraction with HMM structures learned by stochastic optimization. In Proceedings of 17th national conference on artificial intelligence and 12th AAAI conference, Austin TX, USA, pp. 584–589.

7. Google. http://www.google.com/.

8. Hsu, C. N., & Dung, M. T. (1998). Wrapping semi-structured web pages with finite-state transducers. In Proceedings of conference on automatic learning and discovery (CONALD-98).

9. Infoseek. http://www.infoseek.com/.

10. J. Allan. Building hypertext using information retrieval (1997). Information Processing and Management,33(2):145-159.

11. Kushmerick, N., & Thomas, B. (2002). Adaptive information extraction: Core technologies for information agents. Intelligents information agents R&D. In M. Klusch, S. Bergamaschi, P. Edwards, & P. Petta (Eds.), Europe: An agent link perspective (pp. 79–103). NY: Springer- Verlag.

12. Linoff, G. S., & Berry, M. J. A. (2001). Mining the Web: Transforming customer data into customer value. NY: John Wiley & Sons.

13. Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77, 257–286.

14. Sigletos, G., Paliouras, G., & Karkaletsis, V. (2002). Role identification from free text using hidden Markov models. In Proceedings of 2nd hellenic conference on artificial intelligence SETN 2002. Lecture notes in artificial intelligence (vol. 2308, pp. 167–178). London, UK: Springer- Verlag.

15. Song, M., Song, I., Hu, X., & Allen, R. (2005). Integrating text chunking with mixture hidden Markov models for effective biomedical information extraction. In International conference on computational science (ICCS) 2005, Atlanta USA.

16. W. B. Croft and R. H. Thompson (1987). I3R: A new approach to the design of document retrieval systems. Journal of the American Society for Information Science, 38, 389-404, 1987.

17. Y. P Chang & Brandt (2007), Mining free structured information based on hidden Markov Models. Expert Systems with Applications 32, 97-102.

18. Zhang, N. R. (2001). Hidden Markov models for information extraction. Final Project, Stanford NLP Group, Stanford University, CA, USA.