http://ansinet.com/itj

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

ANSInet

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (17): 4256-4261, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.4256.4261

# Hidden Markov Model and its Application in Natural Language Processing

Xuexia Gao and Nan Zhu
Computer and Information Engineering College, Xinxiang University,
Henan, China

**Abstract:** This study describes Hidden Markov Model and its application in natural language process, first introduces the basic concept of Hidden Markov Model, then introduces the three basic issues and the basic algorithm to solve the problems, finally gives the demonstration of application of Chinese part-of-speech tagging and speech recognition via Hidden Markov Model.

**Key words:** Natural language processing, chinese part-of-speech tagging, hidden markov model, speech recognition

## INTRODUCTION

Hidden Markov Model is the mathematical statistical model used to describe the statistical characteristics of random process; it is developed by the Markov chain (Lu, 2007; Wen *et al*., 2005). Because the actual problem is more complex than the description of a Markov chain model, the observed events are not corresponding to the status one-to-one but related through a set of probability distribution, that model is a Hidden Markov Model. It is a double random process, one of which is the Markov Chain; it describes the status transition probability (Wang and Guan, 2005). What other random process described is the statistical relationship between status and observation value. Through observation value observers perceive the existence and characteristics of hidden status, to form "hidden" Markov Model. Hidden Markov Model in the 1970s in the speech recognition field got great success (Daniel and James, 2005), after that it is widely applied to each domain of natural language processing, which becomes the important method of natural language processing based on statistics; it is one of important achievements in the last century in the field of statistical natural language processing (Li and Guo, 2009).

Hidden Markov Model is a quintuple group:

$$(Q_x, Q_o, A, B, \pi)$$

- Limited set of status: $Q_x$, $\{q_1, q_2, q_3, q_4, \ldots, q_n\}$, which represents the model status (i.e., Output), the number of states is N. For some practical applications, even the status is hidden but each status of the model is related to some physical meaning, meanwhile the status is connected with each other and which can transfer from one status to another status. $X_t$ is used to repress the status of t moment
- Limited set of observed values:

$$Q_o, \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \ldots, \sigma_M\}$$

The number of observed values corresponding to each status is M, observed value is corresponding to the actual output of the system model. $O_t$ is used to express the observation of time t

- Transition probability:

$$A = \{a_{ij}\},\ a_{ij} = P\ (X_{t+1} = q_j|X_t = q_i),\ 1\ \leq i, j \leq N$$

$a_{ij}$ meets $a_{ij} \geq 0$, $\forall i, j$ and:

$$\sum_i a_{ij} = 1, \forall i_o$$

- Output probability:

$$B = \{b_{ik}\} b_{ik}$$

Represents in the status $q_i$, the probability of $v_k$ appears at t moment, that is $b_{ik} = P\ (O_t = \sigma_k|X_t = q_i)$, $1\ \leq i \leq N$, $1 \leq k \leq M$. $b_{ik}$ meets $b_{ik} \geq 0$ and:

$$\sum_k b_{ik} = 1, \forall i_0$$

- Initial status distribution:

**Corresponding Author:** Xuexia Gao, Computer and Information College, Xinxiang University, Henan, China

$$\pi = (\pi_i),\ \pi_i = P\ (X_i = q_i),\ 1\ \leq I \leq N$$

that is the probability of $q_i$ at moment t = 1, $\pi_i$ meets:

$$\sum_i \pi_i = 1$$

Starts from initial status to transfer to the end status so far.

All the status experienced in such a transferring process according the order arrange into the vectors, this is called status chain and recorded as: $Q_t$, represents the source status of the No. t time; In such a transfer process, the vectors according to the output descending order in such a transferring process is called output chain, which is recorded as: O, $O_t$ represents the output of the No. t transfer. The output chain of Hidden Markov Model can be observed but the status chain is not visible.

**Three fundamental questions of HMM:** In the quintuple group ($Q_x$, $Q_o$, A, B, π) of HMM, if λ = {A, B, π} is given HMM parameter, O, {$\sigma_{k1}$, $\sigma_{k2}$, $\sigma_{k3}$, ..., $\sigma_{kt}$} is observation sequence, if its status set is S = {$q_{i1}$, $q_{i2}$, $q_{i3}$, ..., $q_{iT}$}, then HMM can solve the following three fundamental questions.

**Problem 1:** To a given observation sequence, the probability algorithm of forward probability can be solved: use observation sequence appearing before the T moment to extrapolate the probability of some observation value at current moment T.

The algorithm can be used to calculate the forward probability: Forward variable is defined as:

$$\begin{aligned} a_t\ (i) &= P\ (O\ \leq t,\ X_t = q_i) \\ &= P\ (O_1 = \sigma_{k1},\ \ldots,\ O_t = \sigma_{k1};\ X_i = q_i) \end{aligned}$$

which is the observed probability of O ≤t at t moment arrive status $q_i$:

Initialize:

$$\begin{aligned} \alpha_1(i) &= P(O_1 = \sigma_{k1}; X_1 = q_i) \\ &= b_{ik1} \bullet \pi_i, 1 \leq i \leq N \end{aligned}$$

Recursive:

$$\alpha_{t+1}(j) = \left\{\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right\} b_{ikt+1},$$
$$1 \leq t \leq T-1, 1 \leq j \leq N$$

Termination:

$$\begin{aligned} P(O) &= \sum_{i=1}^{N} P(O_1 = \sigma_{k1}, \cdots, O_t = \sigma_{k1}; X_t = q_i) \\ &= \sum_{i=1}^{N} \alpha_T(i) \end{aligned}$$

**Backward probability algorithm:** Backward and forward probability algorithms are very similar, backward variables are defined as:

$$\begin{aligned} \beta_t(i) &= P(O > t | X_t = q_i) \\ &= P(O_{t+1} = \sigma_{kt+1}, \cdots, O_T = \sigma_{k_T}, X_t = q_i) \end{aligned}$$

The calculation algorithm of backward probability is: Initialize:

$$\beta_T\ (i) = 1,\ 1\ \leq i \leq N$$

Recursive:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_{jkt+1} \beta_{t+1}(j),$$
$$t = T-1, T-2, \cdots, 1, 1 \leq i \leq N$$

Termination:

$$P(O) = \sum_{i=1}^{N} P(O_1 = \sigma_{k1}, X_1 = q_i),$$
$$P(O_2 = \sigma_{k2}, \cdots, O_T = \sigma_{kT} | X_1 = q_i) = \sum_{i=1}^{N} a_{ik1} \bullet \pi_i \bullet \beta_1(i)$$

After defining the forward probability, backward probability and their algorithms, the output probability P (O) can be calculated as:

$$P(O) = \sum_{i=1}^{N} \alpha_t(i) \beta_t(i)$$

**Problem 2:** An observation sequence is given; according to the existing Hidden Markov Model, a status sequence with the largest probability is found. For HMM, some observation sequence O seen from outside is not the only one inside the system corresponding to the status sequence Q but different sequences of status Q will produce different the probability of O. The task of biggest status sequence is to look for the most likely status sequence Q according to the system's observation sequence, which makes the chance of producing O become the biggest. To solve the problem the most common method is Veterbi algorithm. Veterbi is a kind of deformation of a dynamic programming algorithm, the summary is as follows:

Initialize:

$$\delta_1\ (i) = \pi_i b_{ik1},\ 1\ \leq i \leq N,\ \varphi_i\ (_i) = 0,\ 1\ \leq i \leq N$$

Recursive:

$$\delta_t(j) = \max_{1\leq i\leq N}\left[\delta_{t-1}(i)a_{ij}\right] b_{ikt},\ 2 \leq t \leq T, 1 \leq j \leq N$$

Termination:

$$\varphi_t(j) = \arg\max_{1\leq i\leq N}\left[\delta_{t-1}(i)a_{ij}\right] b_{ikt},\ 2 \leq t \leq T, 1 \leq j \leq N$$

Status series solution is:

$$P^* = \max_{1\leq i\leq N}\left[\delta_T(i)\right], P^{t^*} = \arg\max_{1\leq i\leq N}\left[\delta_T(i)\right]$$

$$P^*t = \phi_{t+1}\ (q^*_{t+1}),\ t = T\text{-}1,\ T\text{-}2,\ \ldots,\ 1$$

From this the best status sequence of P (O|λ) can be obtained: $q^*_1, q^*_2, \ldots, q^*_t$.

**Problem 3:** The estimation of model parameters is the training problem of HMM, that is, how to determine the model λ = {A, b, π} according to system output, makes the probability P (O|λ) being the largest observation value. Generally Baum-Welch algorithm is used to evaluate all parameters in the model.

## APPLICATION OF HIDDEN MARKOV MODEL IN PART-OF-SPEECH TAGGING

**Principle of part-of-speech tagging:** Hidden Markov Model in natural language processing in the beginning is used in Chinese word segmentation and part-of-speech tagging (Wu and Song, 2009). In part-of-speech tagging, part-of-speech sequence is equivalent to the status sequence hidden in the HMM, because part-of-speech sequence is hidden before mark, it is the goal of need to be solved, the given word string is the sequence of observation symbols and the known condition before the mark. If the part-of-speech tagging problem model is regarded as an HMM, the set of part-of-speech tagging is certain (the status number of HMM is certain), each word corresponding to part-of-speech is also certain, in the dictionary, every word has a definite one or several part-of-speech tags (Freitag and Mecallum, 2009; Kristie *et al*., 2009).

In HMM, part-of-speech tagging problem can be expressed as: In a given word sequence (observation value) W = ($w_1$, $w_2$, $w_3$, $w_4$, ..., $w_M$), solving the most probable part-of-speech (status) sequence T = ($t_1$, $t_2$, $t_3$, $t_4$, ..., $t_m$) makes the conditional probability P (T|W) is the largest. P (T|W) is hard to estimate currently, generally it is using the Bayesian principle for conversion, that is:

$$P(T\,|\,W) = \frac{P(T)P(T\,|\,W)}{P(W)}$$

In part-of-speech tagging, W is given, P (W) does not rely on T, therefore, in the calculation of P (T|W), P (W) needs not be considered, meanwhile applying the joint probability equation P (A, B) = P (A) P (B|A), there is P (T|W) = P (T) P (W|T) = P (W, T). The probability multiplication formula is making further application, there is:

$$\begin{aligned} P(T|W) &= P(w_1,\cdots,mt_1,\cdots,m) \\ &= \prod_{i=1\cdots m} P(W_i,t_i|W_1,\cdots,i-1,t_1,\cdots,i-1) \\ &= \prod_{i=1\cdots m} P(W_i|W_1,\cdots,i-1,t_1,\cdots,i-1) \\ &\quad \times P(t_i|W_1,\cdots,i-1,t_1,\cdots,i-1) \end{aligned}$$

In the expression:

$$\begin{gathered} W_{1,..}, i = W_1,\ W_2,\ \ldots,\ W_i,\ t_1,\ \ldots, \\ i = = t_2,\ t_2,\ \ldots,\ t_i,\ 1\ \leq i \leq m \end{gathered}$$

What needs to point out is what the above model reflects in ideal condition is the probability distribution of related part-of-speech tagging but because the estimated parameter space is too big, the model cannot actually be calculated. Therefore, in the actual part-of-speech tagging, in general the model needs to be simplified to reduce the parameter space. Thus some independence hypotheses are introduced (that is Markov assumption) as follows:

$$\begin{aligned} &P(t_i|w_1,\cdots,i-1,t_1,\cdots,i-1) \\ &\approx P(t_i|t_1,\cdots,i-1) \\ &\approx P(t_i|t_{i-N+1},t_{i-N+2}\cdots t_{i-1}) \end{aligned}$$

The emergence of $t_i$ part-of-speech tags only depends on limited prior N-1 part-of-speech tagging, that is N-POS model.

A word appearance does not depend on any words, only rely on prior part-of-speech tags and further suppose words $w_i$ only depends on part-of-speech tagging $t_i$, that is:

$$\begin{aligned} &P\left(w_i \middle| w_1, \cdots, i, t_1, \cdots, i\right) \\ &\approx P\left(w_i \middle| t_1, \cdots, i\right) \\ &\approx P\left(w_i \middle| t_i\right) \end{aligned}$$

After the above assumptions, a HMM with N-1 order of part-of-speech tagging can be obtained:

$$P(T\,|\,W) \approx \prod_{i=1,..,m} P(t_i\,|\,t_{i-N}, t_{i-N+2}, ..., t_{i-1})$$

In the expression, P ($W_i|t_i$) is called emission probability; Parameter P ($t_i|$ $t_{i-N+1}$, $t_{i-N+2}$… $t_{i-1}$) becomes the status transition probability.

If the emergence of part-of-speech tagging $t_i$ depends on just a limited prior part-of-speech $t_{I-1}$, the emergence of word $w_i$ only depends on part-of-speech tagging $t$, $_i$thus:

$$P(T\,|\,W) \approx \prod_{i=1,..,m} P(t_i\,|\,t_{i-N})\; P(W_i\,|\,t_i)$$

where, P ($t_i|t_{i-1}$) and P ($W_i|t_i$) are two key parameters in the expression, hereinto P ($W_i|t_i$) refers to the probability of part-of speech word $w_I$ in $t_i$; P ($t_i|t_{i-1}$) represents the times from part-of speech $t_{i-1}$ to the next $t_i$.

In the large-scale corpus, according to the law of large number, the expression can be obtained, P ($W_i|t_i$).C ($W_i$, $t_i$)/C ($t_i$), C ($W_i$, $t_i$) represents the times while the part-of-speech $W_i$ is $t_i$; C ($t_i$) represents the times of part-of-speech $t_i$:

$$P\ (t_i|t_{i-1}).\ C\ (t_{i-1}, t_i)\ /C\ (t_{i-1})$$

In the expression, C ($t_{i-1}$, $t_i$) represents the times from part-of-speech $t_{i-1}$ to next part-of-speech $t_i$, C ($W_i$, $t_i$), C ($t_{i-1}$), C ($t_i$) are all obtained from the corpus library with good marks and segmentation.

**Experimental data in part-of-speech tagging:** The corpus library in "People's Daily" in 1998 with part labels are used to test and train corpus to obtain transfer matrix table and frequency table of part-of-speech and are used to do HMM tagging, the samples are as follows:

- Bring (v.) the parents (noun) hope (verbs or noun) and (conjunction) entrust (noun) to arrive (v.) Beijing (noun)
- **Tagging process:** P (noun | auxiliary) = C (auxiliary, the noun) / C (auxiliary) = 45214/74792 = 0.6
- P (verb | auxiliary) = C (auxiliary verbs) / C (auxiliary) = 5235/74792 = 0.07
- P (hope | noun) = C (hope, noun) / C (noun) = 364/385325 = 0.0009
- P (hope | verbs) = C (hope, verbs) / C (v.) = 402/201424 = 0.002
- P (noun | auxiliary) P (hope | noun) = 0.6 * 0.0009 = 0.00054
- P (verb | auxiliary) P (hope | verbs) = 0.07 * 0.002 = 0.00014
- **Tagging results:** Bring (v.) the parents (noun) hope (noun) and (conjunction) entrust (noun) to arrive (v.) Beijing (noun)

## APPLICATION OF HIDDEN MARKOV MODEL IN SPEECH RECOGNITION

In the 1980s, J.K. Baker of CMU university applied the HMM to the speech recognition field, in the speech recognition there is a great success and which becomes the main method of speech recognition (Freitag and MeCallum, 2010).

According to the introduction of the hidden Markov model principle, we knew it was a double random process with speech recognition, these two random processes commonly describe the statistical properties of the speech signal, one is an implicit stochastic process that uses Markov chain with a finite state number to analog voice signal change and the other is an observation vector random process associated with each state of the Markov chain. So a certain period of characteristics of speech and time-varying signals are described by the corresponding state observation symbol of stochastic process and the change of signal at any time is described by the transfer probability of hidden Markov chain (Zheng and Zhang, 2002).

Based on hidden Markov model, a speech recognition algorithm count based on a large number of speech data and establishes the recognition statistical model, then extracts features from voice for identification, measures the similarity through comparison with the model and outputs the recognition result which is the category of highest similarity models. In general, hidden Markov model constitutes a speech recognition system or speaker recognition system, which needs to solve three basic problems.

The calculation of observing output probability P (o|$\lambda$): For a given the observed sequence O ($O_1$, $O_2$, $O_3$, …, $O_T$) and model $\lambda$ = {$\pi$+A+B}, the probability of model $\lambda$ to produce zero O can adopt forward probability and backward probability, in order to make the calculation reduced to $N^2T$ operations.

**Definition 1:** Forward probability uses observation sequence before the T moment to calculate the probability of an observed value at current time T, that is, using the probability of $Q_1$, $Q_2$, …, $Q_{t-1}$ to calculate the probability of appearing $Q_1$, $Q_2$, …, $Q_{t-1}$, $O_t$. It can be presented by $\alpha_t$ (i). Forward probability calculation algorithm is.

Initialization:

$$\alpha_1\ (i) = \pi_i b_i\ (o_1),\ 1\ \leq i \leq N$$

Recursion:

$$\alpha_{t+1}(j) = \{\sum_{i=1}^{N} \alpha_t(i)a_{ij}\}b_j(O_{t+1}), 1 \leq t \leq T-1, 1 \leq j \leq N$$

End:

$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

**Definition 2:** Backward probability uses $O_{t+2}$, $O_{t+3}$, …, $O_N$ to calculate the probability of $O_{t+2}$, $O_{t+3}$, …, $O_N$, which is presented by $\beta_t$ (i).

The backward ratio algorithm is listed as follows:

Initialization:

$$\beta_t\ (i) = 1,\ 1\ \leq i \leq N$$

Recursion:

$$\beta_t(i) = \sum_{j=1}^{N} \alpha_{ij} b_j(o_{t+1})\beta_{t+1}(j),$$
$$t = T-1, T-1, \ldots 1, 1 \leq i \leq N$$

End:

$$P(o \mid \lambda) = \sum_{i=1}^{N} \beta_T(i)$$

After defining forward probability backward probability and their algorithms, observation output probability P (o|λ) is easy to be obtained:

$$P(o \mid \lambda) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i)$$

**Search of best state sequence:** In HMM system, some sequence O observed from external is not the only in the internal system corresponding to state sequence Q but the probability of different Q producing O is not the same. The searching task of best status sequence is looking for the most likely state sequence Q according to the system output O, making the possibility of a state sequence to produce O become the maximum. The commonly used algorithm is Viterbi algorithm. The Viterbi algorithm is a kind of deformation of a dynamic programming algorithm, it can be got via recursion algorithm as follows:

Initialization:

$$\delta_1(i) = \pi_i b_i(o_1), 1 \leq i \leq N,\ \varphi_1(i) = 0, 1 \leq i \leq N$$

Recurse:

$$\delta_t(j) = \max_{1 \leq i \leq N}[\delta_{t-1}(i)a_{ij}]b_j(O_t), 1 \leq i \leq N$$
$$2 \leq t \leq T, 1 \leq j \leq N$$

$$\varphi_t(i) = \arg\max_{1 \leq t \leq N}[\delta_{t-1}(i)a_{ij}]$$
$$2 \leq t \leq T, 1 \leq i \leq N$$

End:

$$P^* = \max_{1 \leq i \leq N}[\delta_T(i)]$$
$$q_T^* = \arg\max_{1 \leq i \leq N}[\delta_T(i)]$$

**State sequence solving:** Therefore the best state sequence of P (o|$\lambda$) can be obtained: $q^*_1$, $q^*_2$, …, $q^*_t$.

**Model parameter estimation:** Model parameter estimation is a training problem of HMM model, that is, how to according to the given system output to determine the model λ = {π, A, B} and make P (O|λ) maximum. The researcher generally adopts Baum-Welch re-estimation method to do HMM model training.

Baum-Welch algorithm can be described as follows: Thus:

$$\xi_t(i, j) = \frac{\alpha_t(i)a_{ij}b_j(O_{i+1})\beta_{t+1}(j)}{P(O \mid \lambda)}$$

$$\gamma_t(i) = \sum_{j=1} \xi_t\ (i, j)$$

Thus the famous re-estimate formula in Baum-Welch algorithm:

$$\overline{\pi}_i = \gamma 1(t), \overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi t(i, j)}{\sum_{t-1}^{T-1} \gamma_t(i)}, \overline{b_{ij}} = \frac{\sum_{t=1, O_T=0}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

where, $\overline{\lambda} = \{\overline{\pi}, \overline{A}, \overline{B}\}$ is the model parameter after estimation and $P(O \mid \overline{\lambda}) \geq P(O \mid \lambda)$. The complex speech recognition problem is simply expressed and solved through hidden Markov model, hidden Markov model recognition system is better than any other system, in hidden Markov model recognition system there exist more training data statistical information and solving the difficulties in training and classification, therefore hidden Markov model used in speech recognition is a great success.

**Last word:** The application of Hidden Markov model in speech recognition and speech tagging is still in further perfect and the application of the model is not only in these two aspects. HMM as a good math model, its research is thorough, the algorithm is mature, efficiency is high, the effect is good and it is easy to train, in the model, if the observed value is taken as Chinese word, the status is regarded as English, which can solve the machine translation problems. HMM is not only widely used in natural language processing and has already been used in many other areas, such as signal processing, image processing, machine vision, even genetic engineering and other disciplines.

## CONCLUSION

Lots of experiment results show that based on the HMM speech tagging, the correct rate of mark results can reach 92%.

## REFERENCES

Daniel, J. and H.M. James, 2005. The Overview of Natural Language Processing. Electronic Industry Press, Beijing.

Freitag, D. and A. MeCallum, 2009. Information extraction with HMM structures learned by stochastic optimization. Proceedings of the 18th Conference on Artificial Intelligence, (AI'09), AAAI Press, Edmonton, pp: 584-589.

Freitag, D. and A. McCallum, 2010. Information extraction with HMM and shrinkage. Proceedings of the AAAI Workshop on Machine Learning for Information Extraction, (MLIE'10), AAAI Press, Orlando, pp: 31-36.

Kristie, S., M. Andrew and R. Rosenfeld, 2009. Learning hidden Markov model structure for information extraction. Proceedings of the AAAI Workshop on Machine Learning for Information Extraction, (MLIE'09), AAAI Press, Orlando, pp: 37-42.

Li, K.Y. and B.Y. Guo, 2009. Natural Language Processing. Science Press, Beijing.

Lu, W., 2007. The application of Hidden Markov in natural language understanding. Comput. Inform. Technol., 15: 33-34.

Wang, X.L. and Y. Guan, 2005. Computer Natural Language Processing. Tsinghua University Press, Beijing.

Wen, R., Q.M. Zhu and P. Li, 2005. The application of HMM and negative feedback model in part-of-speech tagging. J. Suzhou Univ., 7: 40-42.

Wu, X.M. and C.X. Song, 2009. Hidden Markov model used for protein sequence analysis. Biomed. Eng., 19: 455-458.

Zheng, J.H. and H. Zhang, 2002. Chinese organization name automatic identification based on the HMM. Comput. Appli., 22: 1-2.