Abnormality Detection in a Landing Operation Using Hidden Markov Model

Hasan Keyghobadi *, Alireza Seyedin

Data fusion Laboratory, Electrical Engineering Department, Ferdowsi University, Mashhad, Iran

Received 28 December 2015; revised 12 September 2016; accepted 9 December 2016; available online 16 March 2017

Abstract

The air transport industry is seeking to manage risks in air travels. Its main objective is to detect abnormal behaviors in various flight conditions. The current methods have some limitations and are based on studying the risks and measuring the effective parameters. These parameters do not remove the dependency of a flight process on the time and human decisions. In this paper, we used an HMM-based method which is among the main methods of situation assessment in data fusion. This method includes two clustering levels based on data and model. The experiments were conducted with B_777 flight data and the variables considered in the next generation of ADS_B. According to the results of this study, our method has high speed and sensitivity in detection of abnormal changes which are effective in the flight parameters when landing. With the dynamic modelling, there is no dependency on time and conditions. The adaptation of this method to other air traffic control systems makes its extension possible to cover all flight conditions.

Keywords: *Automatic Dependent Surveillance – Broadcast (ADS–B), Baum-Welch Algorithm, Data Fusion,. Expectation Maximization (EM) algorithm, Forward algorithm, Hidden Markov Model (HMM).*

1. Introduction

Throughout history, the improvement of air travels safety by airlines has been one of the hardest scientific and technological issues. Despite every progress in risk reduction strategies, air crashes still occur [1]. Today, the aircraft industry has turned to flight safety techniques, which predict the risks for taking necessary measures. The digital data obtained from Flight Data Recorder (FDR) is generally used by many airlines for risk detection process. All methods are capable of using part of many flight parameters in FDR.

The Flight Operations Quality Assurance (FOQA) known as Flight Data Monitoring (FDM) in Europe is one of the most famous effort in this matter. ED is used in the current business programs which alarms for exceeding the allowed area for a certain number of flight parameters. This alarm only produces a correct result for the previously studied events, not all possible events [2]. Other measures include the following:

The Morning Report Software Package is one of the last efforts for anomaly detection using FDR [3]. Subsequent measures in the methods of data analysis were taken for anomaly detection aerospace systems. While some use training-based methods, such as Inductive Monitoring System (IMS) which has limitations in application of time-dependent patterns [4], others have a non-observational approach and focus on application of sequence analysis algorithm

* Corresponding author. Email: h.keyghobadi7@gmail.com

for FDR discrete data, such as cockpit switch flips [5]. Das et al. in NASA developed MKAD which uses single-class SVM algorithm [6]. Srivastava et al. have introduced a method based on data that detects anomaly using clustering [7]. Recently, the information on the above algorithms have been summarized in a paper [8].

A system with the same function in all flight conditions for anomaly detection is absolutely essential, particularly because of ongoing development of public and private air transport, since experts in airport traffic control towers work with numerous parameters and there is possibility of error occurrence and the automatic systems for timely alarm are considered important challenges. This need doubles in airports and other high-traffic airways. One challenge is the fact that flight conditions are time-variable, therefore, the system should be able to detect dynamic behavior. Also, its adaptation to ADS_B which has been predicted in the next generation of air transport system is essential.

In this paper, the predicted variables in ADS_B were used for anomaly detection and the normal landing process is modeled using HMM and with its help, any abnormal change in the flight variables during the landing is detected very fast and can be used in alarm before the accident for better emergency assistance and help. This method is highly efficient in the modeling of random and dynamic processes. In fact, it is an attempt to create a framework for a system in order to detect flight anomaly based on HMM, because HMM has various structures and suitable algorithms to adapt to other processes.

In section 2, a description of level of HMM application in data fusion is given, then, in section 3 and 4, the theoretical examination of HMM and its application in anomaly detection of landing are described. Section 5 explains modeling and the results and in section 6, the potential future applications are discussed.

2. HMM in Data Fusion

Joint Director of Laboratories (JDL) is one of the most well-known models used in data fusion [9]. The following definition was presented by JDL for data fusion: “it is the process of data mixture for improvement of states estimation and prediction”. Data fusion is used in different levels. The application of HMM is related to the levels 2 and 3 for assessment of state and threat [9, 10]. The presented task in this paper is to assess the situation in landing.

The JDL model differentiates the types of data fusion functions into different fusion levels. The JDL Model has been used to develop an architecture paradigm for data fusion [11] and was originally

created by the Data Fusion Group at the JDL Joint Directors of Laboratories in USA.

The structure of the JDL Model is presented in Fig.1.

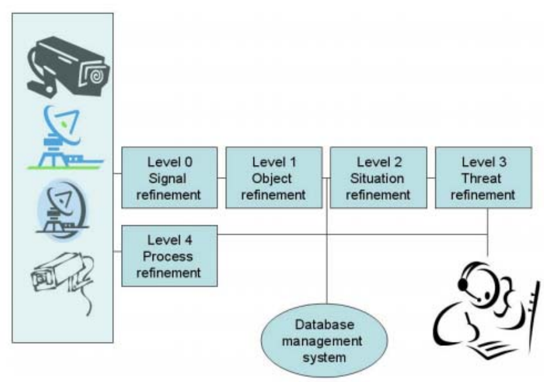

Fig. 1. Structure of the JDL model

3. Hidden Markov Model (HMM) [12]

HMM is a kind of double random Markov process stochastic finite-state automata, which consists of two Markov processes (Markov chain). One process is defined with state transition matrix and the other is expressed in any state with a distribution of observations in the form of probabilities matrix. In fact, the system moves from one state to another with the distribution of state transition probabilities and in each state, the system conditions are seen in the form of observations symbols using the distribution of observations probabilities in the output. This means that the system states are hidden, but the system's status and conditions are seen with observations (Fig. 2).

Three natural problems typically arise in applications that use HMM:

1) Computing the likelihood of an observation sequence (the forward algorithm): Find the probability of a given sequence of observations to be generated by the model.

2) Computing the most probable state sequence (the Viterbi algorithm): Find the most probable sequence of states that is likely to generate a given observation sequence.

3) Estimating the parameters of a model (the Baum-Welch algorithm): Find the most probable parameter set of a model given its structure and one or more observation sequences.

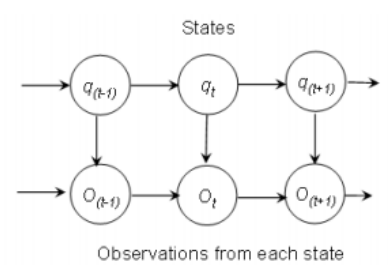

Fig. 2. HMM standard structure. qt is the hidden state and Ot is the observation

3.1.HMM Components [9]

A discrete HMM, λ, is expressed with N number of states and observations symbol M. The states space is S= {s_1, s_2, ..., s_N} and observations space is V= {v_1, v_2, …, v_M}. The λ model is described with the following parameters:

$$M, N, B, A, (\pi = \lambda) \quad (1)$$

$$\pi = \{\pi_i\} = \{ p(q_i = s_i)\} \quad (2)$$

$$A = \{ a_{ij} \} = \{ p(q_{t+1} = s_j \mid q_t = s_i)\} \quad (3)$$

$$B = \{ b_j(k)\} = \{ p(O_t = v_k \mid q_t = s_j)\} \quad 1 \le i,j \le N,\ 1 \le k \le M \quad (4)$$

Where A represents the state transition probabilities, B represents the probability distributions of the discrete observation symbols for all states, and π indicates the initial probability distribution of the states. a_{ij} is the probability of the system to change from state i to state j and $b_j(k)$ denotes the probability distribution of the symbols for a certain state j. The current state is denoted qt and the current observation is denoted O_t.

The parameters π, A, B are obtained via model training. In this research, training was conducted using Expectation Maximization (EM) algorithm, known as Baum Welch. When λ and its parameters are defined, we can use a sequence of observations for Likelihood (L) whose value is obtained from the equation (5).

$$L = p(O \mid \lambda) \quad (5)$$

4. Baum-Welch Algorithm for Learning HMM

Given an HMM M = (S, O, A, B, Π) and an observation sequence Y_1 ... Y_T , we want to re-estimate M as $\bar{M} = (S. O. \bar{A}. \bar{B}. \bar{\pi})$ so that $\bar{M}$ is more likely than M in the sense that $p(Y_1 \ldots Y_T|M) > p(Y_1 \ldots Y_T|\bar{M})$ In other words, the new model is more likely to produce the given observation sequence. This is done via an iterative Baum-Welch or forward-backward algorithm. The forward algorithm considered a forward variable $\alpha_t(s_i)$ as defined below:

$$\alpha_t(s_i) = p(Y_1 \ldots Y_t, X_t = s_i) \quad (6)$$

The forward variable represents the joint probability of the partial observation sequence Y_1 …Y_T and state s_i at time t. The forward algorithm computes the forward variable iteratively. In a similar manner, we define a backward variable β_t (s_i) as follows:

$$\beta_t(i) = p(O_{t+1}, O_{t+2}, ..., O_T | q_t = S_i, M) \quad (7)$$

The backward variable represents the probability of the partial observation sequence from 1 t + to the end given the state at time point t is si . The variable is

inductively computed as follows:

Step 1: Initialization

$$\beta_T(i) = 1 \quad 1 \le i \le L \quad (8)$$

Step 2: Induction

$$\beta_t(i) = \sum_{j=1}^{L} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \quad \begin{matrix} t = T-1, T-2, ..., 1 \\ 1 \le i \le L \end{matrix} \quad (9)$$

Given the sequence $Y_1 ... Y_T$ of observations and any two states s_i and s_j, the re-estimation a_{ij} of the parameter is simply the ratio of the expected number of transitions from state s_i to state s_j and the expected number of transitions from state s_i as a whole. Therefore, we define the following variable representing the posterior probability of transition from state s_i to state s_j given the observation sequence:

$$\xi_t(i, j) = p(q_t = S_i, q_{t+1} = S_j | \mathbf{O}, M) \quad (10)$$

Application of Bayes' rule converts ξ_t (s_i, s_j) as follows:

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{p(\mathbf{O}|M)} = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_t(n) a_{nm} b_m(O_{t+1}) \beta_{t+1}(m)} \quad (11)$$

Define the following variable γ_t (s_i) as the expected number of transitions from the state s_i at time t:

$$\gamma_t(i) = p(q_t = S_i | \mathbf{O}, M) \quad (12)$$

$$\gamma_t(i) = \sum_{j=1}^{L} \xi_t(i, j) \quad (13)$$

The re-estimated model parameters are the following:

$$\bar{\pi}_i = \gamma_1(i) \quad (14)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (15)$$

$$\bar{b}_j(k) = \frac{\sum_{t=1,\ s.t. O=v_k}^{T-1} \gamma_t(i)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (16)$$

5. Forward Algorithm [9]

O is the sequence of observations at T points of time, i.e. O= (o_1, o_2,...,o_T). In a recognition problem, the objective is to compare O to a specific model and find its similarity to that model (L). Forward algorithm is used to calculate L.

$$\alpha_t(i) = p(O_1 O_2 \ ... \ O_t, q_t = s_i | \lambda) \quad (17)$$

In forward algorithm, α_t(i) shows the value of a chain's observation probability (O_1, O_2,..., O_t) at state i and time t

$$\alpha_1(i) = \pi_i b_i(O_1) \quad 1 \le i \le L \quad (18)$$

For other observations, α which is called forward variable, is calculated in the following form:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{L} \alpha_t(i) a_{ij} \right] b_j(O_{t+1}) \quad \begin{matrix} 1 \le t \le T-1 \\ 1 \le j \le L \end{matrix} \quad (19)$$

The equation (19) indicates how we reach the state s_j at time t+1 from N possible states. The calculations for all states are done at time t and repeated for all times t=1, 2, ..., T. The final result is obtained by summation of final forward variable for all states (equation 20).

$$p(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \quad (20)$$

During the calculation procedure there are several multiplications with probabilities well below 1 which brings the final result closer to zero. The problem gets more complex when the final result is less than the smallest number that the computer can demonstrate. The problem can however be solved by scale factor c_t in equation (21).

$$c_t = {}^{1}\!/_{\sum_{i=1}^{N} \alpha_T(i)} \quad (21)$$

A useful equation for calculations is:

$$\log[\, p(O|\lambda)\,] = -\sum_{t=1}^{T} \log c_t \quad (22)$$

6. Experiment and Results

We used the database of 365 flights of B_777 airplane which were also utilized in MKAD project by NASA. The data is applied in the flight control by control towers and navigation systems. The data includes situation, altitude, horizontal distance, horizontal speed and vertical speed. In using HMM for dynamic phenomenon modelling, model training is essential. Therefore, in discrete HMM, it is important how to create symbols from continuous data. Also, the number of these symbols is effective in the model sensitivity. We used the value of variables which were divided into 100 clusters using K_means algorithm. The symbols are as follows: 1, 2, 3, ..., 99, 100.

In Table 1 illustrated some symbols and their related coordinate values of cluster centers.

Table. 1. Some of symbols and their related coordinate values of cluster centers

symbol	altitude (m)	horizontal distance (m)	Deviation from runway (m)	horizontal speed (m/s)	Vertical speed (m)
1	183.56	15273.49	0.39	48.31	-2.41
2	40.74	19869.68	-2.97	31.79	-0.75
6	192.73	15084.44	0.30	48.46	-2.44
7	481.86	9769.49	0.22	49.47	-3.31
10	656.26	7501.15	0.71	50.90	-4.02
14	1129.78	1127.65	3.11	25.28	-2.44
19	1023.22	1340.93	3.40	26.83	-2.25
24	902.42	4474.63	1.54	50.90	-4.02
28	61.71	18348.83	0.51	33.96	-1.05
33	733.84	6391.11	0.89	50.96	-4.04
37	280.67	12961.63	-0.70	49.07	-2.37
42	603.72	12467.48	-0.60	49.01	-2.40
46	1055.99	2813.58	2.04	50.60	-4.11
51	769.53	11272.80	8.81	42.19	-2.23
55	345.26	11693.82	-0.48	48.98	-2.50
60	642.57	7789.30	0.78	50.81	-3.97
64	685.47	3800.36	2.45	44.32	-4.56
69	56.85	18773.86	1.41	32.41	-0.97
73	988.30	2213.08	2.79	38.28	-3.22
78	714.25	6663.24	0.82	50.88	-4.04
82	1033.97	1607.86	3.15	26.77	-2.25
87	951.70	3944.33	1.74	50.93	-4.02
91	1005.36	1895.08	3.16	28.72	-2.40
96	139.04	16202.09	1.55	45.80	-2.09
97	373.82	11157.57	-0.41	49.21	-2.69

The landing process data were used in this paper which includes 20 successful landings and 10 unsuccessful landings. The model includes two states. S1: No change or small change in flight variables. S2: Significant change in flight variables. Since a logical and hidden relationship with the symbols is necessary, these states were selected. When the observations' symbols were constructed, the successful landing data were used by Baum Welch algorithm for model training. The model's matrixes were presented in the equations 22, 23, and 24.

$$\pi = \begin{bmatrix} 0.55 \\ 0.45 \end{bmatrix} \quad (23)$$

$$A = \begin{bmatrix} 0.45 & 0.55 \\ 0.61 & 0.39 \end{bmatrix} \quad (24)$$

$$B = \begin{bmatrix} 0.01 & 0.011 & \cdots & 0.125 & \sim 0 \\ 0.01 & 0.011 & \cdots & 0.124 & \sim 0 \end{bmatrix} \quad (25)$$

After the construction of a normal landing model, now we should detect abnormal behavior using successful and unsuccessful landings data. To do so, the forward algorithm was used and the symbols of each landing was given to the model as input and the, L, algorithm output was the rate of this landing's similarity to the successful landing model. We found out that a suitable method for calculation and representation of L is by using a five-symbol window on the data which moves on the symbols one by one. Therefore, the forward algorithm was conducted for 5 symbols that accelerated the processes.

The results for 4 different successful landings have been shown in Fig. 2. After examining all successful landing data, we specified a threshold for log (L), i.e. L<-23.02 indicates an abnormal behavior, and L>-23.02 is indicative of a normal behavior when landing.

The results for 2 different unsuccessful landings were obtained and shown in fig. 3.

In the F5 landing, first, the vertical speed was abnormal in proportion to the horizontal speed, then it was adjusted, but again went out of control. This was correctly detected by the model. Although the F6 landing had lower altitude, it had correspondence with the vertical and horizontal speeds. Then, its vertical speed exceeded the safe value and this was detected by the model. The calculations time for each landing in offline mode is as follows: F1: 0.2581 seconds, F5:0.2265 seconds, and F6: 0.2283 seconds. The calculation time of the forward algorithm for 5 symbols is 0.000286 seconds in average. Considering the high speed in detection, online processing is practically possible.

The main distinction between this method and MKAD, ED, and Srivastava et al.'s clustering method is using dynamic modeling along with data clustering. In addition to increasing speed and accuracy, online training and improvement of the model becomes possible. Also, the model's accuracy is independent of the data being continuous or discrete.

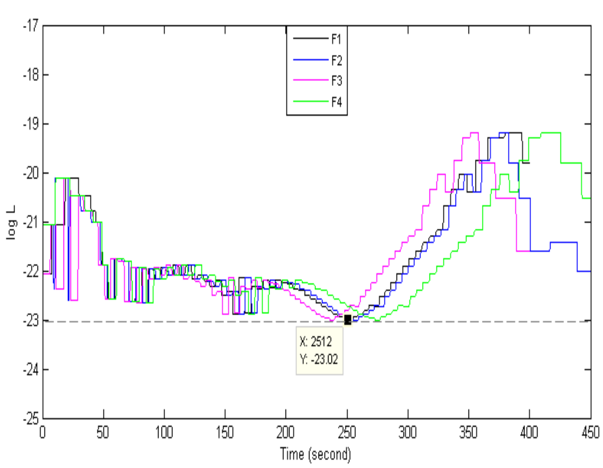

Fig. 3. Logarithm L with time for 4 successful landings and the allowable threshold for L.

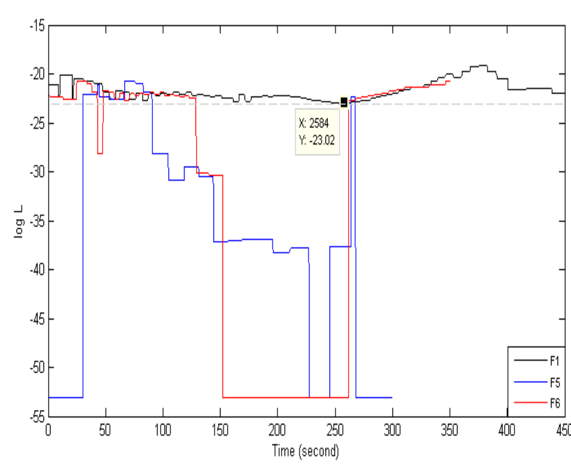

Fig. 4. Logarithm L with time for one successful landing (F1) and two different unsuccessful landings (F5, F6).

7. Discussion and Conclusion

The main subject of this paper is to use HMM for modeling of flight anomaly. As it was expected, this method has a high accuracy and sensitivity in this application such as speech and video processing and modeling of dynamic phenomena. Considering the main application of HMM in the modeling and detection of dynamic and random phenomena, the results indicate that this model is suitable for the flight process, because this process is dynamic and time-dependent. Because of the nature of its application, clustering, it uses two clustering levels. At data level, in addition to converting abundant data to a chain of symbols, it specifies the flight parameters' limit. At model level, HMM creates a dynamic probability model from a successful landing and has a high capability in applying time-dependent patterns. The algorithms' high speed makes possible the implementation of the project. This model is extendable for different parts of a flight according to various weather conditions and type of the airplane.

References

[1] Logan, T. J., "Error prevention as developed in airlines," International Journal of Radiation Oncology, Biology, Physics, vol. 71, pp: S178-S181, (2008).

[2] Li, L.; Das, S.; John Hansman, R.; Palacios, R.; Srivastava, A. N., "Analysis of Flight Data Using Clustering Techniques for Detecting Abnormal Operations," Journal of Aerospace Information Systems.

[3] Amidan, B. G.; Ferryman, T. A., "Atypical event and typical pattern detection within complex systems," IEEE Aerospace Conference, (2005).

[4] Iverson, D. L., "System health monitoring for space mission operations" IEEE Aerospace Conference, pp: 1-8 (2008).

[5] Budalakoti, S.; Srivastava, A. N.; Otey, M. E., "Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 39, pp: 101-113, (2009).

[6] Das, S.; Matthews, B. L.; Srivastava, A. N.; Oza, N. C., "Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study," in Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp: 47-56 (2010).

[7] Sriastava, A., "Discovering system health anomalies using data mining techniques," M. F. NASA Ames Research Center, CA, United States , (2005).

[8] Matthews, B. L.; Das, S.; Bhaduri, K.; Das, K.; Martin, R.; Oza, N. C.; et al., "Discovering Anomalous Aviation Safety Events Using Scalable Data Mining Algorithms," Journal of Aerospace Information Systems, vol. 11, pp: 482, (2014).

[9] Das, S., High-Level Data Fusion: AR TECH HOUSE, (2008).

[10] Holsopple, J.; Yang, S. J.; Sudit, M., "TANDI: Threat assessment of network data and information," in Defense and Security Symposium, pp: 62420O-62420O-12 (2006).

[11] Llinas, J.; Bowman, C.; Rogova, G.; Steinberg, A., Waltz, E.; F- White, "Revisiting thr JDL model II", The 7th Int. Conf. on Information Fusion, Stockholm, Sweden, June 28 – July 1, (2004).

[12] Rabiner, L. R., "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, vol. 77, pp: 257-286, (1989).