Conference on Information Technology and Data Science

Deep Learning-Based Method for Detecting Cassini-Huygens Spacecraft Trajectory Modifications*

Ashraf ALDabbas[a], Zoltán Gál[b]

[a]University of Debrecen, Doctoral School of Informatics
Ashraf.Dabbas@inf.unideb.hu

[b]University of Debrecen, Faculty of Informatics
Gal.Zoltan@inf.unideb.hu

Proceedings of the 1st Conference on Information Technology and Data Science
Debrecen, Hungary, November 6–8, 2020
published at http://ceur-ws.org

Abstract

During the last 13.5 year motion cycle of the interplanetary research project, there were necessary flight path modifications of the Cassini spacecraft. In the order of signal travel time (approximatively 80 minutes) on the Earth-Cassini long sized channel, complex event detection of orbital modifications requires special investigation and analysis of the collected large trajectory dataset. This paper presents a sophisticated, in-depth learning approach for detecting Cassini spacecraft's trajectory modifications in post-processing mode. The model uses neural networks with Long Short-Term Memory (LSTM) to extract useful data and learn the time series' inner data pattern, together with the penetrability of the LSTM layers distinguish dependencies between the long- and short-term phases.

Keywords: Cassini-Huygens interplanetary project, complex event, sensory data, big data, artificial intelligence, pattern processing, knowledge representation

AMS Subject Classification: 65C60, 60G35, 68T05, 68T20

*This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund. The paper was supported by the QoS-HPC-IoT Laboratory, too.

1. Introduction

A complex event's significance is related to processing several events, followed by paying attention to discriminate distinct occurrences within a time series of events [2]. There are situations where the obtainable knowledge to represent some method or device is merely observation inspection; with the scale of big data, there is an significant simplification of the problem, which is to identify an extreme event [5]. The spacecraft launched in October 1997 arrived at its target on 1 Jul 2004 [8]. This case is called Cassini-Huygens' Saturn Orbit Insertion (SOI). It took the spacecraft 6.7 years from Earth's launch to reach Saturn's destination (SOI). For reaching Saturn manoeuvres for correcting the orbital momentum of the spaceraft relative to Sun were necessary. They were gravity assisted by doing flybys of planets. Then flybys of Titan, the biggest moon of Saturn were carried out for further gravity assisted orbit corrections needed for reaching other moons of Saturn [3]. Visualization of the large-scale trajectory shown in Figure 1 below represents the last 393,977 pieces of Cassini trajectory coordinates sampled between Earth and Saturn planets [9]. The last 13.5 years of the trajectory defined by roughly large-scale semi-ellipse and evaluated by us begins with a circular marker in the bottom, while the top square symbolizes the end of the trajectory on September 14, 2017. In our research framework we focus on detecting Cassini's orbiter trajectory changes.

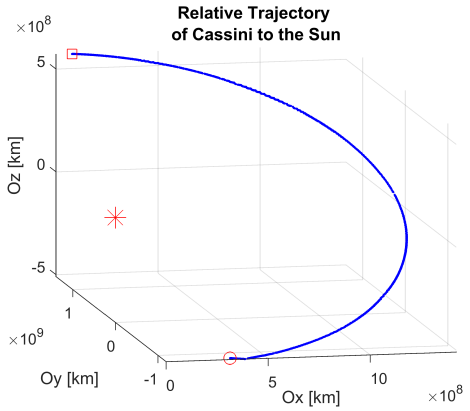


Figure 1. Cassini large scale trajectory around Sun.

The trajectory represented contains just the last 13.5 Earth years including approaching phase, SOI event and orbiting around the Saturn of the Cassini spacecraft. This duration is approximately half period time of the Saturn trajectory around the Sun. The star, circle, and square characters mark the Sun, the first sample, and the last sample, respectively. After the SOI, the Cassini trajectory is a dynamic curve compound by an ellipse-based helicoid around Saturn, orbiting around the Sun in its own ellipse. It should be mentioned that the first amount

of samples are before SOI, consequently the time interval between SOI and end of project is just 13.3 years. With the established method, we aim to detect events in post processing mode relevant to Cassini's trajectory modifications. To this end, we put forward our model that uses the capacity of Long-Short Term Memory neural networks to produce useful data and learn the internal data structure of the trajectory time series and leverage the memory dependence LSTM potential.

In section 2 we provide a brief literature review of several related studies within the field. Section 3 describes extraction of sampling and trajectory characteristics of the Cassini database (Imaging Science Subsystem (ISS), Saturn EDR Data Sets (Volume 1 – Volume 116) [9], which was indexed by National Aeronautics and Space Administration of the USA (NASA). Section 4 discusses spacecraft trajectory modifications detection by LSTM based artificial intelligence algorithm along with the adopted method for the detection of trajectroy changes. Section 5 contains the experimental results of the trajectory manoeuvres detection and section 6 summarizes our research conclusions.

2. Related Work and Previous Studies

Cassini's project's trajectory was separated into three classes of activities occurring in phases: i) launching and journey to Saturn, ii) approaching and arrival at Saturn, and iii) science phase. Controlling the trajectory required processing a variety of status information including certain step length, velocity, etc. The approaching and arrival process provided complete project trajectory information. Basic role of trajectory maps in the science phase is to position the spacecraft in a specific location related to Saturn, which has been meticulously planned and has sufficient entrance conditions consistent with spacecraft velocity and path angle. Paper [11] offers recent research results on the Cassini project orbiter remodeling and mostly shifting our view of the Cassini orbiter; like in the last half of 2016, one of Cassini's near flybys changed the trajectory of Cassini to form a sequence of 20 rings containing marvelous orbits.

Paper [7] surveyed artificial intelligence developments in the concept of spacecraft control and guidance dynamics and focused on evolutionary logic and deep learning as the cornerstone to potential systemic space science. The method includes artificial intelligence and automatic logic to monitor the navigation and the remote sensing of the external space mission trajectories. Gated Recurrent Unit (GRU) provides a periodic neural network algorithm for real-time trajectory prediction, where its parameters are acquired as an initial step by batch processing, then the qualified feedback for trajectory prediction [6]. In work [10] a model-based reinforcement learning is proposed to conduct almost quintessential reconfiguration in establishing flying spacecraft. Along with two other algorithms, the LSTM layer network and reverse reinforcement learning were used to remodel and forecast possible trajectories to acquire collision-free maneuvers. These merits encouraged us to use LSTM networks, where the LSTM approach suits our study field.

3. Extracting Sampling and Trajectory Characteristics of the Cassini-Huygens Database

There were two pieces of the spacecraft: Huygens probe and Cassini orbiter. Cassini-Huygens (C-H) arrived at Saturn in 2004, sending useful data back to Earth. Huygens moved through Titan's atmosphere, Saturn's biggest moon, plunged down by parachute to the furthest point so far, landed on his surface. Huygens took samples, testing them, and submitting the findings to Cassini, who then returned these signals to Earth. Remote sensing devices gathered data from vast distances remotely. The acquired image data was provided by NASA's Imaging Science Subsystem (ISS). The ISS comprises of two detached wide-angle video cameras and a narrow-angle camera. The dataset of ISS picture volumes comprises an immense number of images and their corresponding labels. The dataset is freely accessible by reference [9]. Table 1 shows the project's time stamp.

Table 1. Tasks time stamps of main phases (UTC).

Task	*Starting Date*	*Ending Date*
C_H Project	15 October 1997	15 September 2017
Analyzed DB by us	6 February 2004	15 September 2017
Prime Mission	1 July 2004	1 July 2008
Equinox Mission	1 July 2008	11 October 2010
Solstice Mission	11 October 2010	15 September 2017

We evaluated the 116 volumes of the data collection from the above NASA source. The staring study had time stamp 02:07:06 on February 6, 2004, and end stamps on September 14, 2017, all clarified in UTC (Coordinated Universal Time). The C-H project's five key objectives (i.e., Saturn study, Moon Titan, Saturn rings, ice satellites, and magnetosphere) involved trajectory maneuvers. Reference [4] gives the amount of expected and performed adjustments in the trajectory of each mission. Table 2 provides the number of Saturn orbits along with the planned and executed maneuvers. The percentage of performed and expected trajectory maneuvers for the Prime and Equinox project is 69.5% and 67.3%, respectively.

Table 2. Number of Saturn orbits and manoeuvres.

Mission	*No. of Orbits*	*Trajectory Maneuvers Planned*	*Trajectory Maneuvers Executed*
Prime	75	161	112
Equinox	64	104	70
Solstice	155	206	141

With no official details on the Solstice mission's performed trajectory modifications, our forecast is 68.4 percent (average of the previous two) of the scheduled

maneuvers, providing 141 trajectory modifications.

4. Conditions of the Complex Event Detection of the Cassini Trajectory

Our classifier's key purpose is to specify complex events from sensory produced data; we expanded the sensory data index to detect temporal semantics for complex event detection. Potential dynamic data set occurrences are where observation sequences shift. We consider severe trajectory adjustment as the Cassini orbiter velocity vector shifts more than a threshold metric. Since velocity is a measure vector, extreme trajectory occurrences mean satisfying either of the following two conditions: extreme alteration of the velocity direction or of the acceleration vector's magnitude.

4.1. Modification per Time of the Velocity Vector Direction

The speed of the angle modification $\Delta\varphi_i$ between consecutive velocity vectors v_i and v_{i+1} is given by the following formula:

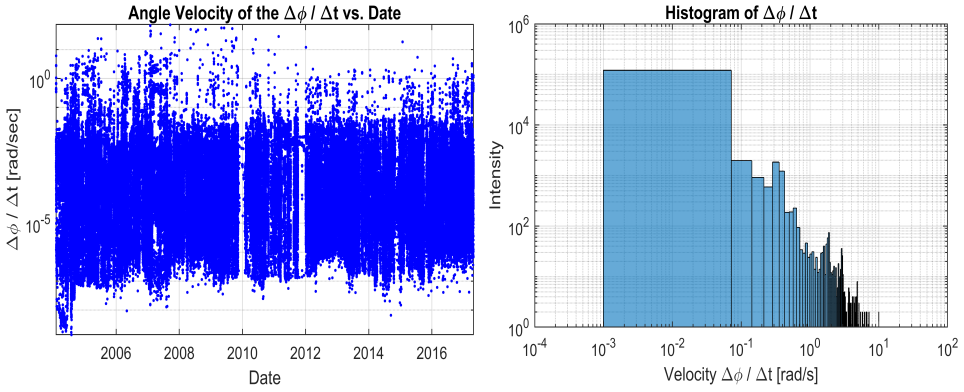
$$\frac{\Delta\varphi_i}{\Delta t_i} = \frac{\cos\left(\frac{\overline{v_{i+1}}\cdot\overline{v_i}}{\|\overline{v_i}\|\cdot\|\overline{v_{i+1}}\|}\right)}{t_{i+1}-t_i}, \quad i = 1, 2, \ldots, N-1 \tag{4.1}$$

where $\overline{v_i}$ and $\overline{v_{i+1}}$ are two consecutive velocity vectors of the orbiter, $\|\overline{v_i}\|$ is the magnitude of the vector, Δt_i is the time interval between two consecutive samplings and $i = 1, 2, \ldots, N-1$. The number of vectors is the total number of samples in the Prime, Equinox and Solstice missions: $N = 407,303 - 13,326 = 393,977$. Value around 0 of the $\Delta\varphi/\Delta t$ means a very small modification of the direction per unit of time. Such cases were at the beginning of the project (see values before 2005 on the Figure 2a).

This amount of samples analysed by us belongs to the last 13.3 years of the C-H project. Starting with the SOI event, the angle of the consecutive samplings of the velocity direction modified in a higher range per unit of time. The spacecraft trajectory has been updated many times, but no specific knowledge on these incidents is accessible to the public. NASA's open 116-volume archive includes samples of high time dispersion. Values of the $\Delta\varphi/\Delta t$ in the scale of over 1 rad/sec were sampled in case of relatively short delay time between consecutive samplings. The distribution of the $\Delta\varphi/\Delta t$ is exponential conform to the right hand side histogram of Figure 2.

4.2. Modification of the Acceleration Vector Magnitude

Trajectory adjustment happens as the magnitude of velocity vector v (v_x, v_y, v_z) varies during successive sampling by a higher value than the threshold Th_v. The



(a) Angle modification vs. date. **(b)** Histogram of angle modification.

Figure 2. Basic properties of angle modification velocity vs. date. Majority of angle velocity are less than 0.01 rad/sec.

estimation of the adjustment magnitude per unit of time of the velocity vector between two consecutive samples (being the moving average acceleration) is centered on the velocity components defined in the database as follows:

$$\overline{v} = \overline{v_x} + \overline{v_y} + \overline{v_z} \tag{4.2}$$

$$a_i = \frac{\|\overline{\Delta v_i}\|}{\Delta t_i} = \frac{\|\overline{v_{i+1}} - \overline{v_i}\|}{t_{i+1} - t_i}, \quad i = 1, 2, \ldots, N-1. \tag{4.3}$$

The magnitude of velocity modification can be derived using the following relation:

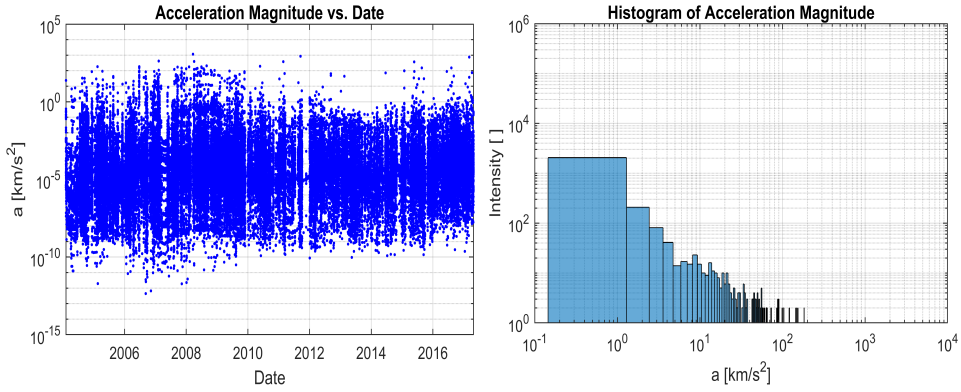
$$\|\overline{v_{i+1}} - \overline{v_i}\|^2 = (\Delta v_{x,i})^2 + (\Delta v_{y,i})^2 + (\Delta v_{z,i})^2 \tag{4.4}$$

where $\Delta v_{x,i}$, $\Delta v_{y,i}$, $\Delta v_{z,i}$ are the modification of the orthogonal velocity components in the sampling interval i, and $i+1$. The acceleration magnitude of the Cassini can be seen on the left-hand side of Figure 3. It can be observed that the distribution of the acceleration magnitude is power function conform to the right-hand side histogram of Figure 3.

The histogram denotes a structure in which the amplitude of acceleration events calculated in a specified magnitude duration is spread through possible magnitude values. Each level within the generated histogram expresses the acceleration rate among the acceleration span. A linear function, resulting in the histogram's power function dependency, will approximate the log-log scale histogram.

4.3. Complex Event Detection of the Cassini Orbiter Trajectory

Our classifier's key purpose is to specify complex events from sensory produced data; we expanded the sensory data index to detect temporal semantics for complex



(a) Acceleration magnitude vs. date. **(b)** Histogram of acceleration magnitude.

Figure 3. Acceleration magnitude of the Cassini vs. date. Majority of the magnitudes of a are less than 1 km/s^2.

event detection. Potential dynamic data set occurrences are where observation sequences shift. Let's have trajectory adjustment indexes where I set unique events:

$$I_\varphi = \{1 < i < N-1 \mid \frac{\Delta\varphi_i}{\Delta t_i} \geq Th_\varphi\} \tag{4.5}$$

$$I_a = \{1 < i < N-1 \mid a_i \geq Th_a\} \tag{4.6}$$

$$I = I_\varphi \cup I_a \tag{4.7}$$

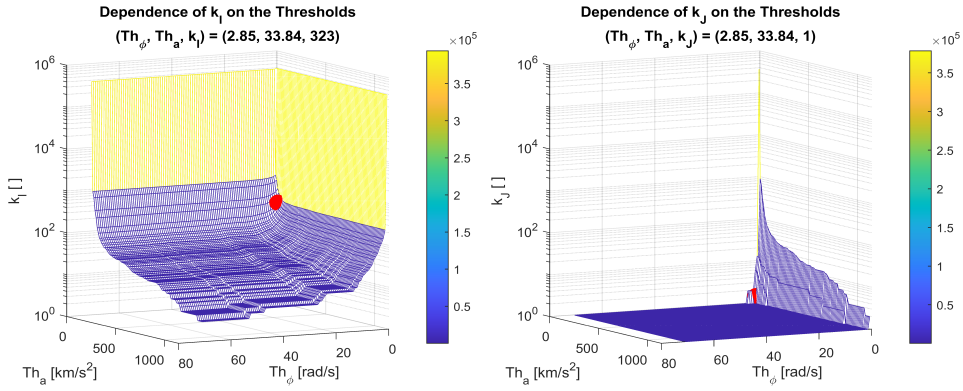
$$J = I_\varphi \cap I_a \tag{4.8}$$

where I_φ and I_a are sample indexes of the analyzed NASA database for which the velocity direction modifications or the acceleration magnitude are greater than the corresponding threshold values. Set J is used to sense the individual effect simultaneously of the two conditions mentioned in subsections 3.1 and 3.2. If the set J's cardinality is low, then conditions (4.5) and (4.6) are not strongly dependent and both of them help to detect complex events on the trajectory. The resulting set I contains all the sampling indexes detected by the proposed complex event detector.

$$I = i_1, i_2, \ldots, i_k \tag{4.9}$$

After the SOI phase in 2004, the spacecraft performed many trajectory adjustments in compliance with the Earth commands sent by the supervisor squad. Cardinality k_I and k_J of the sets I and J, respectively, give the number of extreme events called Cassini trajectory modifications dependent on circumstances (4.7) or (4.8) over the last 13.3 years of the project studied. Obviously, number of trajectory variations should be less than the amount of observations listed in Table 2.

Working points are represented with red bubble markers and are placed in the extreme modification coordinates of the gradient of the surfaces. Based on Table 1. the amount of conducted maneuvers is considered 323. By leveraging the dependence of severe values on the threshold values Th_φ and Th_a within our model, we may evaluate the working point in three-dimensional space. Figure 4 sets out this dependency as a surface plot. The working point on both surfaces is placed on cordinates with extreme modification of the surfaces. To fulfill the total number of trajectory manoeuvres the values of the thresholds are $(Th_\varphi, Th_a) = (2.85 rad/s, 33.84 km/s^2)$. For these threshold values, the cardinality of the set I and J were found to be $(k_I, k_J) = (323, 1)$. It was found that approximately two times more extreme events appear in set I_φ than in set I_a. These two sets are not disjunctive because in several samples, both of the conditions (4.5) and (4.6) fulfill. The union of these two sets gives precisely 323 extreme trajectory adjustment cases. In the continuation, we demonstrate the approach for detecting these trajectory maneuvers with recurrent neural network.



(a) Cardinality of set I vs. thresholds. **(b)** Cardinality of set J vs. thresholds.

Figure 4. Dependence of the cardinality of sets I and J.

4.4. Deep Learning Method of Detecting Trajectory Modifications

The work presented in [1] aids in analyzing the effect on the trajectory of spacecraft, depending on how two or more ideas or artifacts are related between the orbital elements. Any framework's productive research includes the time-domain scientific findings reference. The suggested solution is an amalgamated structure that may define trajectory adjustments in the C-H expedition project. Trajectory analysis enables knowledge to be gained, not only regarding spacecraft motion but also for improved machine learning-based motion analysis. Our system collects the trajectory data as inputs and analyses them momentarily and spatially based on the sample number and pacing alongside the spacecraft's velocity.

The input to the RNN system is a sequence of sample ID, $i \in \{1, \ldots, N-1 = 393{,}976\}$, sampling intervals $\Delta t_i = t_{i+1} - t_i$, modification of the position coordinates $(\Delta x_i, \Delta y_i, \Delta z_i)$ and modification of the velocity components $(\Delta v_{x,i}, \Delta v_{y,i}, \Delta v_{z,i})$ among the last 13.3 years of the studied time interval. The input X of the neural network is a $7xN$ type matrix conform to the formula below:

$$X = [X_1, X_2, \ldots, X_{N-1}], \tag{4.10}$$

where column vectors X_i have the following elements:

$$X_i = [\Delta t_i, \Delta x_i, \Delta y_i, \Delta z_i, \Delta v_{x,i}, \Delta v_{y,i}, \Delta v_{z,i}]^T. \tag{4.11}$$

The data set having $N-1$ samples is divided into the subsets of objects conform to Table 3. We used half of the samples for learning and the remaining half of samples is divided equally for validation and testing. Based on the formula (4.11) each matrix has 7 rows and N columns.

Table 3. Dimension of the data subsets.

Data Subset	*Start Column Index*	*End Column Index*	*No. of Matrix Rows*	*No. of Matrix Columns*
XTrain	1	(N-1)/2	7	(N-1)/2
YTrain	1	(N-1)/2	7	(N-1)/2
XValidate	(N-1)/2+1	3*(N-1)/4	7	(N-1)/4
YValidate	(N-1)/2+1	3*(N-1)/4	7	(N-1)/4
XTest	3*(N-1)/4+1	N-1	7	(N-1)/4
YTest	3*(N-1)/4+1	N-1	7	(N-1)/4

The human control team from Earth sent adjustment orders on the trajectory due to the mission's numerous science and celestial goals. We use LSTM neural network layers to better sense the memory property of the trajectory because of inertia of the Cassini. Figure 5 provide the adopted architecture of the recurrent neural network.

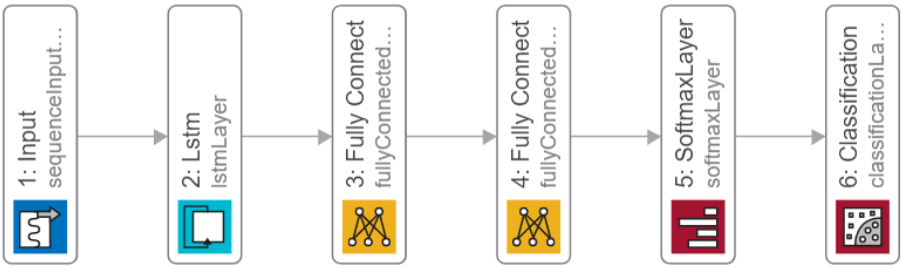


Figure 5. Architecture of the adopted recurrent neural network.

The RNN method conducts a binary trajectory sample classification. The sequences of the sampled multidimensional time series will be detected depending on the trajectory's extreme events. The orbiter's automatic adjustments have been produced to hold the orbiter on the complicated helicoid described in Section 3.

It is common to stack LSTM layers for better modeling capacity, particularly when a large amount of training data is available based on its ability to manipulate previous computation knowledge. In principle, RNNs should accommodate arbitrary long-term dependencies in the input series to prevent the gradient issue where long-term dependencies occur. Our each neural network machine performance is a binary feature that demonstrates how the Cassini orbiter trajectory is highly changed. Each neural network has formation algorithm style ADAM; the threshold gradient method is L2Norm. Table 4 offers additional criteria for the used set of neural networks.

Table 4. Training option values of the RNN.

Option	*Value*
GradientDecayFactor	0.9000
SquaredGradientDecayFactor	0.9900
InitialLearnRate	0.0200
GradientThreshold	1.0000
MaxEpochs	100
Number of Classes	2

The number of classes for the tested neural networks is two to detect the trajectory's complex events. As the CED conditions (see relations 4.5, 4.6) are fulfilled for any trajectory samples, that sample is classified True, otherwise is classified False.

5. Experimental Results of the Trajectory Manoeuvres Detection

Training and validation of trajectory research data sets are conducted in the last 13.3 years of the Cassini project using a set of twelve separate recurrent neural networks. Half of them are LSTMs, while the remainders are BiLSTMs. The changing input parameters (mini-batch scale, number of hidden units, number of classes on layer three and layer four) and measured performance metrics (subsequent learning time and the precision of these networks' identification) are shown in Table 5. It should note that for each RNN type we got detection accuracy over 99 %. Big difference is detected between the learning time of LSTM and BiLSTM network types. It is obvious that having both direction of the signal propagation the BiLSTM requires extra processing relative to the LSTM structure.

Table 5. Detection accuracy of the trajectory modification of different RNNs.

RNN Type	*Mini Batch Size* []	*Hidden Units# on L2*	*Classes# on L3*	*Classes# on L4*	*Learning Time* [s]	*Detection Accuracy [%]*
LSTM	2500	10	100	2	654.2	99.86
LSTM	2500	100	100	2	749.1	99.78
LSTM	5000	10	100	2	625.9	98.39
LSTM	5000	100	100	2	748.8	99.98
LSTM	10000	10	100	2	623.2	99.92
LSTM	10000	100	100	2	748.0	99.06
BiLSTM	2500	10	100	2	1089.8	98.76
BiLSTM	2500	100	100	2	1172.0	99.95
BiLSTM	5000	10	100	2	1093.4	99.11
BiLSTM	5000	100	100	2	1174.7	99.19
BiLSTM	10000	10	100	2	1092.4	99.36
BiLSTM	10000	100	100	2	1175.8	98.20

A popular challenge in machine learning is research and building algorithms that can learn from and render data predictions. These algorithms operate by rendering data-driven forecasts or decisions by creating a theoretical framework from input data. The accuracy levels provided in the deep learning-based model's spacecraft maneuvering behavior are acquired within the training dataset. The results support the view that deep models are important in the detection of trajectory modifications detection of sensory data.

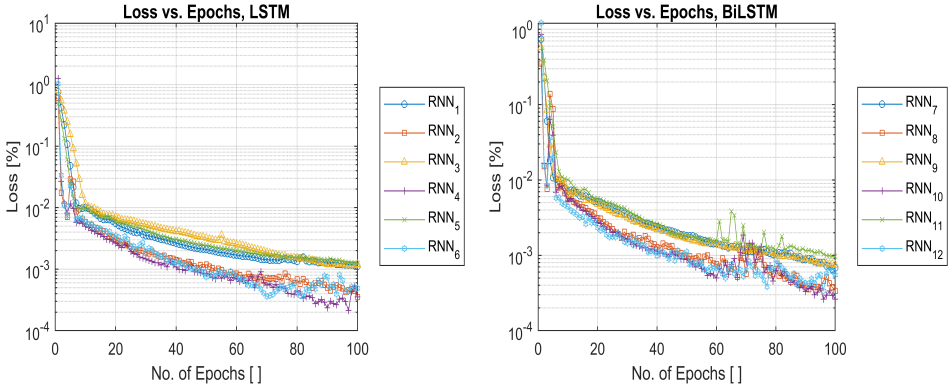
Figure 6 (left) and Figure 6 (right) visualize the deficit during the learning process. It can be found that L2's number of secret units affects learning loss. The fewer the number of hidden units are, the greater the loss are. For 10 hidden units, the learning loss doubles. The related activity has LSTM and BiLSTM networks, but the last one is learned on the L2 layer for 100 hidden units.

The ideal learning rate is bound to be exposed to the lack of the results' learned behavior, which depends on the dataset and architecture used. The loss value shows how effectively or adequately the model performs during each learning iteration. Figure 6a reveals that RNN4 LSTM offers the best loss relative to other versions. We can see that RNN8 BiLSTM has the strongest loss on the figure's right hand relative to other versions. The efficiency of identification is the ratio between the number of identified trajectory corrections and the number of real trajectory corrections. In this context we name it accuracy metric like in popular statistical software tool environments.

Generally defined model accuracy after measuring the parameters of the this model is explained in percentage. It is called the model accuracy metric that compares the model forecast with the correct data. Figure 7 represent dependency of the accuracy on the learning time for each of the analysed RNN. All LSTM

networks need lower learning time than for BiLSTM networks.

The model accuracy is identified after calculating the parameters of the used model with a percentage style; this is clear in the figure above; as an illustration, RNN4 is hitting an accuracy that exceeds the value of 99.9% with a minimum learning time with MiniBatch size = 5000, 100 hidden units and is able to learn the behaviour with 99.98% accuracy of the trajectory in less than 20.5 minutes on a desktop computer with 64 GB RAM and 12 CPU cores.



(a) Minibatch learning loss (LSTM). **(b)** Minibatch learning loss (BiLSTM).

Figure 6. Dependence of the minibatch learning loss on the number of epochs of the RNNs.

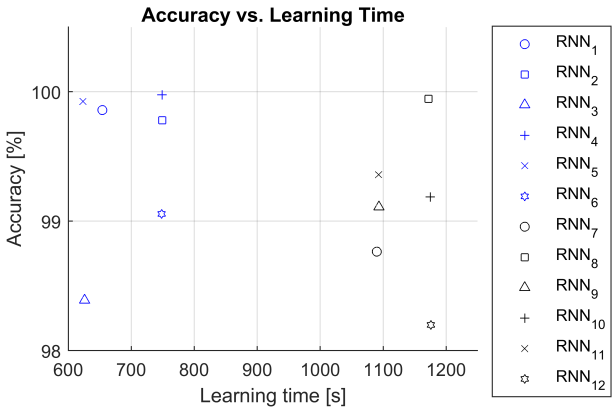


Figure 7. Scatter plot of the learning time and accuracy of the RNNs.

6. Conclusions

C-H spacecraft trajectory modifications detection method was conducted using LSTM/BiLSTM networks; as far as we know, this study is the first to detect the events of spacecraft trajectory modifications. By the provided results, we show that our test study clearly identifies that LSTM models with the chosen parameters have a reasonable option for specifying Cassini's trajectory modifications. Their usage and extra stacked layers produce a noticeable boost in increasing detection process efficiency. It is worth mentioning that the models used can be comfortably generalized to cover a large scientific field relevant to estimation. With more specific information, the provided models present a robust processing step in employing the inner features and time-series representation via LSTM time dependencies for precise detection. The proposed detection model was capable of identifying 99.98% of the trajectory modifications of the Cassini orbiter.

References

[1] A. ALDabbas, Z. Gal: *On the Complex Event Identification Based on Cognitive Classification Process*, in: 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2019, pp. 29–34.

[2] A. ALDabbas, Z. Gál: *Complex Event Processing Based Analysis of Cassini–Huygens Interplanetary Dataset*, in: Intelligent Computing Paradigm and Cutting-edge Technologies: Proceedings of First international conference on Innovative Computing and Cutting-edge Technologies (ICICCT 2019), Istanbul, Turkey: Springer, 2019, pp. 51–66,
DOI: https://doi.org/10.1007/978-3-030-38501-9.

[3] A. ALDabbas, Z. Gál: *Getting facts about interplanetary mission of Cassini-Huygens spacecraft*, in: 10th Hungarian GIS Conference and Exhibition, Debrecen, Hungary, 2019.

[4] B. Buffington: *Designing the Cassini Solstice Mission Trajectory, ASK Magazine, pages. 15-18*, (Accessed on 30/10/2020),
URL: https://appel.nasa.gov/wp-content/uploads/2013/04/513854main_ASK_41s_designing.pdf.

[5] G. Dematteis, T. Grafke, E. Vanden-Eijnden: *Rogue waves and large deviations in deep sea*, Proceedings of the National Academy of Sciences 115.5 (2018), pp. 855–860.

[6] P. Han, W. Wang, Q. Shi, J. Yang: *Real-time Short-Term Trajectory Prediction Based on GRU Neural Network*, in: 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC), IEEE, 2019, pp. 1–8.

[7] D. Izzo, M. Märtens, B. Pan: *A survey on artificial intelligence trends in spacecraft guidance dynamics and control*, Astrodynamics (2019), pp. 1–13.

[8] M. Meltzer: *Building an international partnership and preventing mission cancellation*, in: The Cassini-Huygens Visit to Saturn, Switzerland: Springer, 2015, pp. 27–46,
DOI: https://doi.org/10.1007/978-3-319-07608-9_2.

[9] NASA: *National Aeronautics and Space Administration of the USA, Cassini ISS Online Data Volumes, Imaging Science Subsystem (ISS), Saturn EDR Data Sets*, (Accessed on 30/10/2020),
URL: https://pds-imaging.jpl.nasa.gov/volumes/iss.html.

[10] S. Silvestrini, M. R. Lavagna: *Spacecraft Formation Relative Trajectories Identification for Collision-Free Maneuvers using Neural-Reconstructed Dynamics*, in: AIAA Scitech 2020 Forum, 2020, p. 1918.

[11] L. Spilker, S. Edgington: *Cassini-Huygens: Recent Science Highlights and Cassini Mission Archive*, EPSC 2019 (2019), EPSC–DPS2019.