



Sistema automático para comparación y detección de anomalías en buses de comunicación de sistemas satelitales

Autor:

Ing. Raúl Marcelo Blanco Eliçabe

Director:

Dr. Marcelo Leandro Moreyra (FAIN UNCo)

Codirector:

Lic. Vicente Ramos García (FAI UNCo)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 21 de Octubre de 2021 y el 9 de diciembre de 2021.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar	5
2. Identificación y análisis de los interesados	8
3. Propósito del proyecto	8
4. Alcance del proyecto	8
5. Supuestos del proyecto.	9
6. Requerimientos	9
7. Historias de usuarios (<i>Product backlog</i>).	10
8. Entregables principales del proyecto	12
9. Desglose del trabajo en tareas	13
10. Diagrama de Activity On Node.	13
11. Diagrama de Gantt	16
12. Presupuesto detallado del proyecto	18
13. Gestión de riesgos	18
14. Gestión de la calidad	20
15. Procesos de cierre	23

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	21 de Octubre de 2021
1	Actualización hasta sección 9 inclusive	7 de Noviembre de 2021
2	Actualización hasta sección 12 inclusive	18 de Noviembre de 2021
3	Actualización hasta sección 15 inclusive	22 de Noviembre de 2021
4	Corrección final	29 de Noviembre de 2021
5		

Acta de constitución del proyecto

Buenos Aires, 21 de Octubre de 2021

Por medio de la presente se acuerda con el Ing. Raúl Marcelo Blanco Eliçabe que su Trabajo Final de la Carrera de Especialización en Inteligencia Artificial se titulará “Sistema automático para comparación y detección de anomalías en buses de comunicación de sistemas satelitales”, que consiste esencialmente en el desarrollo de un prototipo de sistema de evaluación y detección de datos anómalos en un simulador digital de un satélite real, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo, con fecha de inicio 21 de Octubre de 2021 y fecha de presentación pública 10 de octubre de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Lic. Nicolás Eduardo Horro
INVAP S.E.

Dr. Marcelo Leandro Moreyra
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

Los satélites artificiales dominan nuestros cielos, brindando información con objetivos de comunicación, científicos y/o militares. Al 16 de septiembre de 2021 orbitan nuestro planeta 7941 satélites artificiales, según Supriya Chakrabarti*.

La empresa de colocar un satélite funcional en órbita requiere inversiones económicas y tecnológicas y conlleva al desarrollo de hardware y software para garantizar su operatividad y confiabilidad durante la vida útil del mismo. Como en toda medición de parámetros, existen los errores, los satélites no se encuentran exentos a ellos. Estos pueden ser propios del método de medición, pero también pueden ser producidos entre otros, por desfases, por fallas en la sincronización entre los diferentes subsistemas que componen un satélite.

Para poder simular la operación de un satélite, las empresas diseñadoras y que llevan adelante la construcción de los mismos desarrollan modelos virtuales que emulan a su patrón real. Estos softwares reproducen teóricamente la totalidad del funcionamiento del satélite, permitiendo analizar diferentes escenarios durante la etapa de montaje y testeo del satélite. Al concepto de modelo virtual y su par funcional, se los conoce como gemelos digitales.

En la fase de ensablado, integración y testeo, los satélites son probados en tierra y para ello, se utiliza además del hardware propio del satélite, a su gemelo digital.

El gemelo digital, debe reproducir exactamente la misma información producida por el sistema real y llegado el caso de no poder hacerlo, se deberá adaptar para predecir el comportamiento.

A su vez, desde el desarrollo de nuevas técnicas, como la de LSTM ** se ha implementado la inteligencia artificial en la detección de anomalías en la telemetría de los sistemas satelitales en operación.

El presente proyecto consiste en el desarrollo de un algoritmo de detección y alerta de datos anómalos en los buses de comunicación entre sistemas. Este objetivo es extensible al hardware existente no solo a satélites, sino también a equipos con gran flujo de información de sensores y coordinación entre sistemas como vehículos de transporte, sistemas de control de procesos, entre otros.

En el diagrama en bloques de la figura 1, se observa de manera muy simplificada el sistema propuesto. Los datos físicos provenientes de los subsistemas del satélite, son colectados por el modelo virtual. Seguidamente, estos datos ingresan al autoencoder en donde, utilizando técnicas de Deep Learning, se determinan los datos anómalos arrojados por el simulador. Estos datos anómalos obtenidos, son comparados con los datos anómalos determinados por el sistema LSTM del satélite, permitiendo realizar ajustes en el comportamiento del modelo virtual.

Se busca desarrollar un sistema que permita evaluar los canales de datos y fijar un umbral de detección de anomalías que aprenda a medida que los datos son analizados, modificando este umbral dinámicamente, mediante Deep Learning. Esto permitirá un simulador que aprenda a simular de manera autónoma a su hermano gemelo físico.

Para INVAP, el desarrollo del presente proyecto implica adquirir nuevas herramientas en la confiabilidad de los sistemas satelitales, así como también de la información por ellos recolectada.

* Director del Centro de Ciencia y Tecnología Espaciales de la Universidad de Massachusetts, Lowell

** Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding

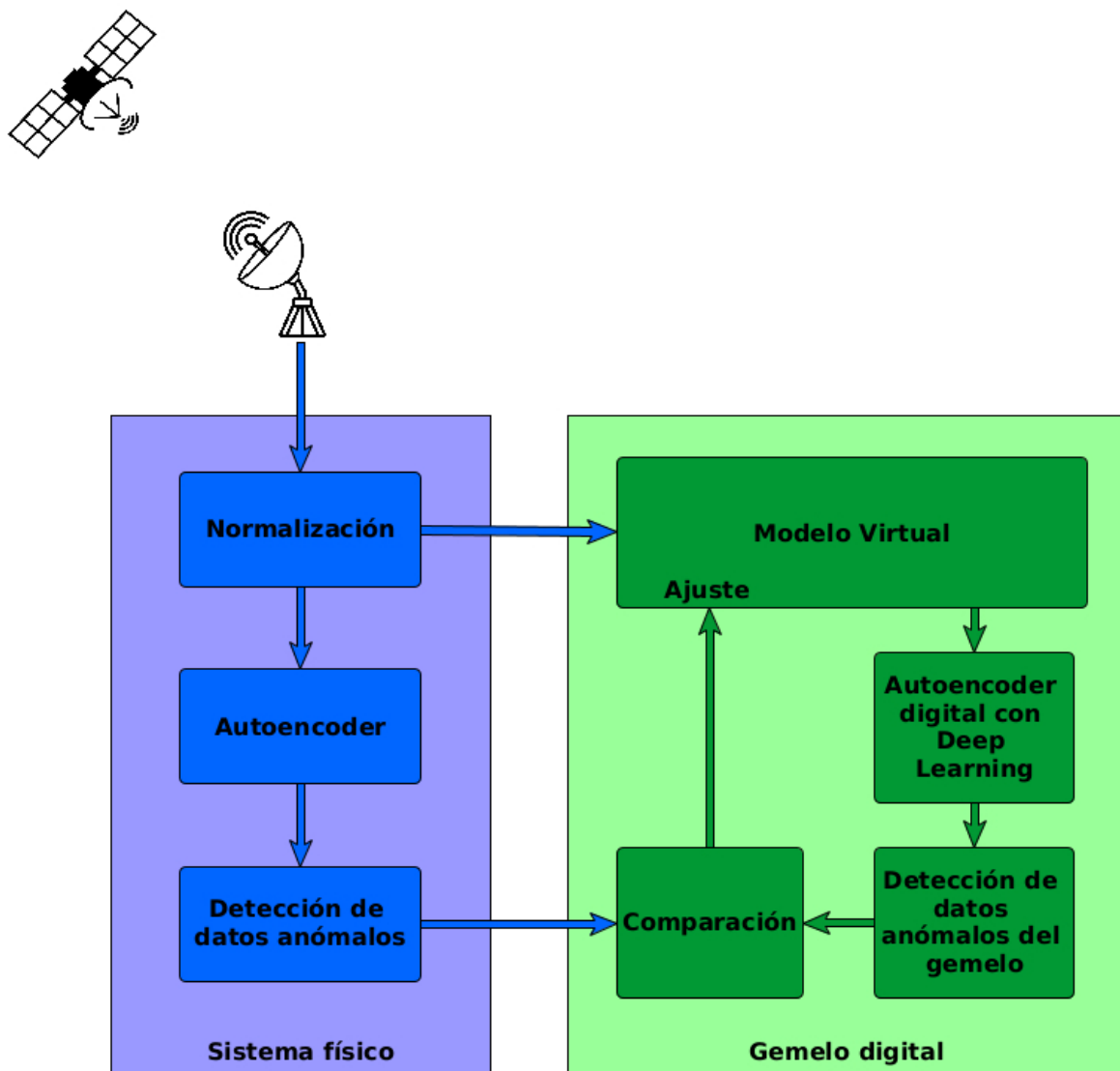


Figura 1. Diagrama en bloques del sistema

El presente trabajo es extrapolable a cualquier sistema complejo que tenga un bus de comunicación donde puedan generarse anomalías y sea necesario analizarlas para su corrección en el campo virtual.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Lic. Nicolás Eduardo Horro	INVAP S.E.	Arq. de SW
Responsable	Ing. Raúl Marcelo Blanco Eliçabe	FIUBA	Alumno
Orientadores	Dr. Marcelo Leandro Moreyra	FAIN UNCo	Director Trabajo final
	Lic. Vicente Ramos García	FAI UNCo	Codirector Trabajo final

- Cliente: Lic. Nicolás Eduardo Horro ayudará a definir el dolor de la empresa.
- Director: Dr. Marcelo Leandro Moreyra va a poder colaborar mucho con la metodología dada su trayectoria profesional.
- Codirector: Lic. Vicente Ramos García aportará sus conocimientos en software y en Python.

3. Propósito del proyecto

El propósito de este proyecto es investigar y desarrollar un sistema que permita el ajuste del gemelo digital, mediante la detección de errores anómalos en la fase de simulación de un satélite.

Se aprovechará la oportunidad, para colaborar con la ciencia y tecnología de la industria nacional.

4. Alcance del proyecto

El presente proyecto contempla el desarrollo del software descrito en la sección 1, en particular para detectar datos anómalos en la simulación de la información generada por un satélite. Se desarrollará el sistema y el software vinculado, para estudiar el comportamiento conjunto de un satélite y su gemelo digital.

Será necesario contar con datos pertenecientes a un sistema físico verdadero, así como también los datos de su simulador digital o gemelo digital. Estos datos podrán ser provistos por INVAP S.E. u obtenidos por otra fuente y serán utilizados exclusivamente para el presente trabajo, no pudiendo ser divulgados bajo ningún concepto.

Llegado el caso, en que INVAP S.E. no pueda compartir los datos de vuelos reales, se optará por utilizar datos de provenientes de equipos similares, buscando únicamente demostrar el concepto planteado para ser extrapolado al caso satelital.

El presente trabajo no contempla el desarrollo del software del simulador, así como tampoco la detección de datos anómalos en el sistema real.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Se podrá acceder a datos reales pertenecientes a un simulador real.
- Se podrá acceder a datos reales pertenecientes a un satélite real.
- Ambos set de datos corresponden a un par gemelo digital y sistema real.
- Los datos corresponden a una misma ventana de tiempo y eventos. Esto permitirá evaluar el comportamiento del gemelo digital y la detección de anomalías en el sistema digital.
- De no cumplirse los supuestos anteriores, se utilizarán datos similares, provenientes de otras fuentes.
- El hardware requerido para el entrenamiento del sistema se puede adquirir en el mercado.

6. Requerimientos

1. Requerimientos generales

- 1.1. El sistema deberá implementarse como un servicio de clasificación de anomalías que deberá procesar series temporales multivariable e indicar, los intervalos en los que se detecta la presencia de una anomalía. Internamente puede contener múltiples modelos. El usuario deberá poder seleccionar el modelo y los parámetros del mismo (por ejemplo el umbral) por argumentos pasados por línea de comando o desde un archivo de configuración.

2. Requerimientos funcionales

- 2.1. El sistema deberá poder consumir datos de un CSV o de alguna de las siguientes bases de datos: SQL, InfluxDB o Elasticsearch.
- 2.2. La entrada estará compuesta por series temporales multivariadas. Estas series, pueden ser de alguno de los siguientes tipos:
 - 1) Numérico: entero o decimal.
 - 2) Booleano.
 - 3) Categórico: representado como una cadena de texto.
- 2.3. Los modelos incluidos en la solución deben estar conformados por distintos tipos de preprocesamiento y clasificación, pudiendo incluir, a modo de ejemplo:
 - 1) Preprocesamiento:
 - Reducción de dimensiones.
 - Codificación de variables categóricas.
 - Transformación de distribuciones de entrada.
 - 2) Clasificación:
 - Isolation Forest.
 - Support Vector Machines.
 - Autoencoders con LSTM.
 - Máquinas de Boltzmann restringidas.

- Redes convolucionales de tiempo dilatadas.
- 2.4. El entregable final deberá incluir aquellos que hayan demostrado un mejor desempeño para los escenarios ensayados.
- 3. Requerimientos de Interfaz
 - 3.1. Entradas:
 - No se requiere interfaz de usuario gráfica, pero sí un mecanismo por archivos de configuración o argumentos por línea de comando para seleccionar la entrada, algoritmo clasificador y parámetros operativos, así como disponer de algún tipo de log del sistema para que pueda ser monitoreado.
 - 3.2. Salidas:
 - Un reporte formato tipo archivo de texto o PDF con un detalle indicando ventanas en las que se hallaron anomalías y la causa (umbral de error, clusters, correlaciones, etc.)
- 4. Requerimientos de ambiente
 - 4.1. El sistema deberá poder ejecutarse en un server con Linux.
- 5. Requerimientos de desempeño
 - 5.1. Se deberá destinar una partición de los datos suministrados para evaluación de los modelos.
 - 5.2. Los modelos deberán ser evaluados con la métrica AUC.
 - 5.3. Para cada escenario considerado, alguno de los modelos deberá permitir obtener un AUC mayor al 60 %.
- 6. Requerimientos de documentación
 - 6.1. La documentación deberá indicar recomendaciones del modelo utilizar en distintos tipos de escenarios.
 - 6.2. Se llevará un control de versiones en Gitlab.

7. Historias de usuarios (*Product backlog*)

Se identifican los siguientes roles/usuarios:

- **Usuario final:** En general se tratará de un ingeniero de sistema SuT/DuT. Este usuario, bien quiere entender la causa de una falla, o detectar de forma automática cuando los valores se alejan de los nominales. No necesariamente debe conocer de IA, pero si claramente sobre la metrología y el funcionamiento de los buses y los protocolos de los mismos.
- **Desarrollador con perfil de IA:** Este usuario comprende de IA y se dedica al desarrollo de modelos para ser agregados a los sistemas o bien modifica existentes. Sus tareas suelen ser entrenar a sistemas existentes para mejorar su desempeño. Conoce lenguajes de programación, como por ejemplo Phyton.
- **Sysadmin/Devops:** Este usuario tiene que ser capaz de poner en producción el algoritmo, incorporandolo a un gemelo digital y mantener la funcionalidad de la solución en ambiente de operación. Domina el diseño de SW.

La ponderación de cada historia de usuario, se obtendrá de una evaluación sobre tres ejes, mediante una escala entre 0 y 5 sobre cada eje. Seguidamente se realizará la suma algebraica de estos valores, y se aproximará al valor superior más cercano de la serie de Fibonacci: [0, 1, 2, 3, 5, 8, 13, 21...]

1. Dificultad/Tiempo
2. Complejidad
3. Incertidumbre/Riesgo

Historia 1 de usuario

Como ingeniero de sistema, deseo poder definir las entradas del sistema: una base de datos o un archivo CSV, una ventana de tiempo, el tipo de datos y el modelo a utilizar para proveer de la información necesaria al procesos de análisis.

- Dificultad: 3
- Complejidad: 2
- Incertidumbre: 1

Total: 6

Puntaje: 8

Historia 2 de usuario

Como Ingeniero de sistema, deseo poder alimentar al sistema con datos de entrada para obtener como salida un informe con las anormalidades detectadas, el origen de las mismas y su clasificación.

- Dificultad: 5
- Complejidad: 5
- Incertidumbre: 5

Total: 15

Puntaje: 21

Historia 3 de usuario

Como desarrollador de IA, deseo poder acceder al código fuente, para poder mejorarlo o completar la integración a las sistemas del INVAP.

- Dificultad: 1
- Complejidad: 2
- Incertidumbre: 1

Total: 4

Puntaje: 5

Historia 4 de usuario

Como desarrollador de IA, deseo poder entrenar a este sistema de IA y poder ajustarlo.

- Dificultad: 1
- Complejidad: 3
- Incertidumbre: 1

Total: 5

Puntaje: 5

Historia 5 de usuario

Como Devops engineer deseo poder incorporar lo desarrollado en este proyecto a los simuladores del INVAP para mejorar nuestros sistemas de Testing.

- Dificultad: 1
- Complejidad: 2
- Incertidumbre: 1

Total: 4

Puntaje: 5

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Manual de usuario.
- Código fuente.
- Informe final

9. Desglose del trabajo en tareas

1. Relevamiento y capacitación (60 hs)
 - 1.1. (ID1.1) Lectura de Papers específicos (40 hs)
 - 1.2. (ID1.2) Lectura de protocolos de comunicación (20 hs)
2. Bloque Autoencoder digital (200 hs)
 - 2.1. (ID2.1) Ingeniería de características (40 hs)
 - 2.2. (ID2.2) Modelación del autoencoder (40 hs)
 - 2.3. (ID2.3) Reducción de dimensiones (20 hs)
 - 2.4. (ID2.4) Calibraciones y correcciones (20 hs)
 - 2.5. (ID2.5) Entrenamiento del sistema (80 hs)
3. Modelo Predictivo (130 hs)
 - 3.1. (ID3.1) LSTM con autoencoder (60 hs)
 - 3.2. (ID3.2) Parámetros de simulación (30 hs)
 - 3.3. (ID3.3) Desarrollo de herramientas que permitan el diagnóstico (40 hs)
4. Bloque Comparación (90hs)
 - 4.1. (ID4.1) Procesamiento de datos anómalos del sistema físico (30hs)
 - 4.2. (ID4.2) Comparación de datos anómalos entre sistemas (30 hs)
 - 4.3. (ID4.3) Calculo de parámetros de ajuste (30hs)
5. Integración (80hs)
 - 5.1. (ID5.1) Despliegue de sistemas en Emulador Digital (40hs)
 - 5.2. (ID5.2) Ajuste de los bloques (40 hs)
6. Documentación (80 hs)
 - 6.1. (ID6.1) Confección del manual de usuario (20hs)
 - 6.2. (ID6.2) Documentación de algoritmos, memorias y resultados (30hs)
 - 6.3. (ID6.3) Confección del informe final (30hs)

Cantidad total de horas: 640 hs

10. Diagrama de Activity On Node

En la figura 2 se observa el diagrama Activity On Node. En el mismo, los tiempos se encuentran indicados en horas. Con trazo remarcado se puede observar el camino crítico del proyecto.

El diagrama comienza con la lectura de documentación específica sobre el problema a resolver (ID1.1).

A continuación, se realizará la lectura de los protocolos de comunicación típicos utilizados en las comunicaciones satelitales: CANBus, Spacewire, MIL1553-STD, RS-422, etc (ID1.2).

Seguidamente, se realizará la ingeniería de características, para poder determinar de mejor manera el modelo de predicción (ID2.1) Con la ingeniería de características, se procederá a modelar el autoencoder (ID2.2) y luego se realizará la reducción de dimensiones (ID2.3). La calibración del autoencoder se realizará en la siguiente etapa (ID2.4) para luego entrenar al sistema y lograr que sea representativo del sistema físico (ID2.5).

Mientras el modelo es entrenado, se comenzará con el desarrollo del modelo predictivo, iniciando con el estudio del LSTM (ID3.1), continuando con la definición de los parámetros de simulación (ID3.2) y el desarrollo de herramientas que permitan al usuario el diagnóstico del sistema predictivo (ID3.3).

En este punto, se contará con el modelo del autoencoder y del LSTM, lo que permitirá comparar su comportamiento y resultados. Para ello, se analizarán los resultados anómalos en el sistema físico y en el autoencoder (ID4.1 e ID4.2). Con estos resultados, se podrá calcular y ajustar los parámetros del autoencoder (ID4.3).

A continuación se procederá con la integración de los sistemas, comenzando con el despliegue del emulador digital (ID5.1). Los ajustes a los bloques se deberá hacer como final del proceso de integración (ID5.2).

Finalmente, se procederá a la creación de la documentación del sistema, alimentado por registros y resultados de ensayos.

Se creará el manual de usuario para ser entregado al INVAP (ID6.1), la documentación de algoritmos, y registros (ID6.2) y por último se redactará el informe final del proyecto (ID6.3).

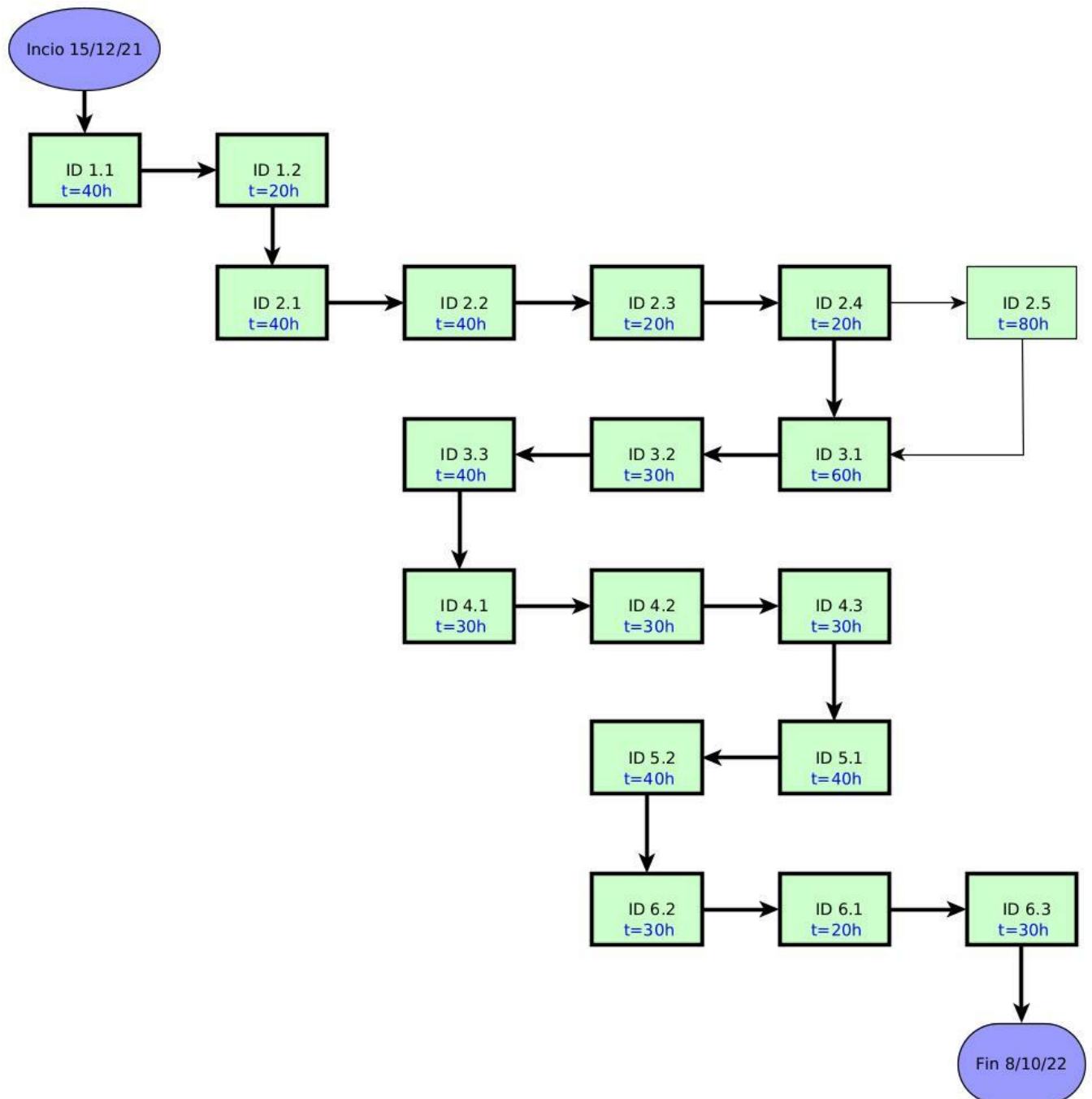


Figura 2. Diagrama en *Activity On Node*

11. Diagrama de Gantt

En la figura 3, se observa el diagrama de Gantt del presente proyecto utilizando como mano de obra, un único recurso, con una carga horaria de trabajo semanal de aproximadamente 11 horas. El proyecto completo podrá ser terminado, de esta manera, a principios de Octubre de 2022.

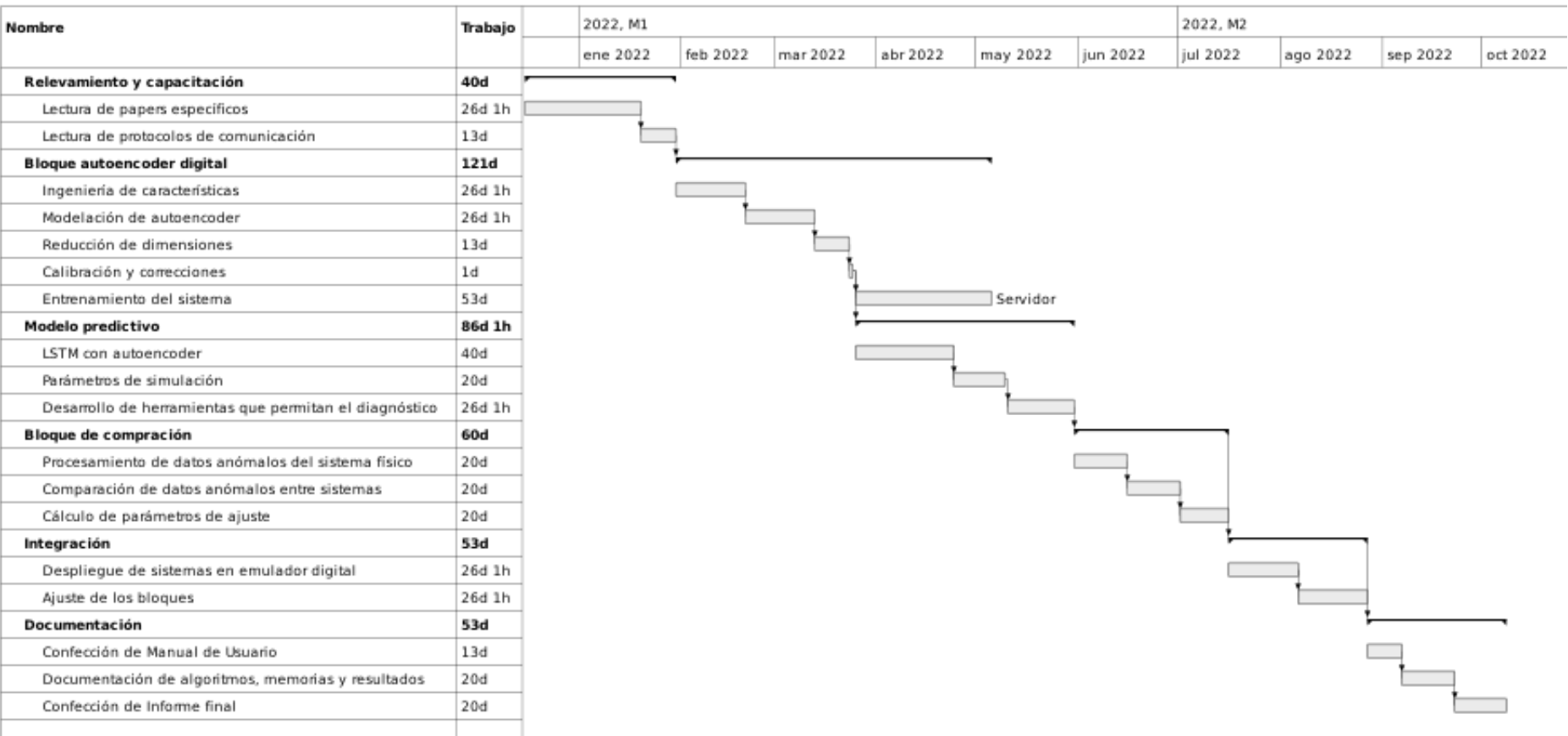


Figura 3. Diagrama de Gantt rotado

12. Presupuesto detallado del proyecto

El presupuesto estimado para el desarrollo del presente proyecto, se estima en función de la hora de ingeniería, disponible en el mercado argentino.

Se observa la necesidad de adquirir un servidor para realizar, en mayor medida el entrenamiento del sistema. El mismo deberá contar con las siguientes características:

- CPU Intel i7 3,4GHz
- 16 GB de RAM, DDR4
- GPU GTX 1050 Ti o similar con 2GB de RAM
- SSD 1TB

Asociado a los costos indirectos se dispondrá de una oficina de co-working que resolverá el acceso a internet, así como también la energía eléctrica y las instalaciones de desarrollo. La misma se localizará en la cercanía a mi domicilio particular, por lo que no se agregarán costos de traslado.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Horas de ingeniería	640 h	\$ 3000	\$ 1.920.000
Servidor Linux	1	\$ 150.000	\$ 150.000
SUBTOTAL			\$ 2.070.000
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Oficina co-working	10 meses	\$ 12.000	\$ 120.000
SUBTOTAL			\$ 120.000
TOTAL			\$2.190.000

13. Gestión de riesgos

A continuación de describirán los riesgos detectados para el presente proyecto. Para ponderarlo, se utiliza un análisis con dos ejes. Por un lado, se analizará:

- Probabilidad de ocurrencia: la posibilidad de que dicho riesgo ocurra, asignándole un valor entre 1 y 10, siendo 1 un riesgo con baja probabilidad de ocurrencia y 10 con la más alta probabilidad de ocurrencia.
- Severidad: impacto del riesgo con el proyecto. Se le asignará 1 al riesgo cuya severidad sobre el proyecto sea muy baja y 10 a aquel riesgo que impacte fuertemente en el desarrollo del proyecto.

Riesgo 1: No conseguir datos reales de vuelos. Invap no autoriza en tiempo, forma y cantidad la entrega de los datos impidiendo de esta manera contar con los datos para entrenar al sistema.

- Severidad (S): 10. Contar con datos reales es fundamental para el desarrollo del proyecto, ya que sin estos, el sistema no podrá aprender de la realidad. De contar con un set de datos pequeños, el sistema dispondrá poca experiencia real, obteniendo un resultado regular.
- Probabilidad de ocurrencia (O): 7. Existe burocracia dentro de la compañía y los datos reales de vuelos, son datos confidenciales. Esto genera respuestas sigilosas para con los datos reales.

Riesgo 2: No obtener los recursos de hardware. El servidor realizará el entrenamiento del sistema. Este requiere ciertas características de desempeño para realizar el entrenamiento en el tiempo estimado.

- Severidad (S): 2. El entrenamiento se puede realizar en cualquier servidor, sacrificando velocidad de computo. A su vez, el entrenamiento se realiza en simultáneo con otras tareas, teniendo de esta manera bajo impacto en el desarrollo final de proyecto.
- Ocurrencia (O): 4. Los recursos económicos son propios, por lo que la probabilidad de no contar con los mismos, depende de las condiciones económicas futuras del desarrollador.

Riesgo 3: El desarrollo del proyecto presenta mayor dificultad de la espera. Los resultados iniciales conllevaron más tiempo del esperado. Como consecuencia de esto, se debe reasignar más horas de trabajo para el desarrollo completo del proyecto.

- Severidad (S): 5. Se comprometen los plazos de la culminación del proyecto, pero no significa que no se obtenga el resultado esperado.
- Ocurrencia (O): 8. Inteligencia Artificial es una técnica que requiere el uso de herramientas de cálculo avanzado, requiriendo redoblar esfuerzos en el desarrollo de algoritmos.

Riesgo 4: No obtener los resultados esperados. El modelo desarrollado, presenta resultados y conclusiones ambiguas.

- Severidad (S): 8. El objetivo del proyecto se cumple a medias, pero los resultados no son concluyentes imposibilitando el uso futuro de la herramienta.
- Ocurrencia (O): 2. Existen desarrollos funcionales de herramientas realizadas por agencias espaciales, las cuales se encuentran en uso por sistemas satelitales.

Riesgo 5: No disponer de tiempo o conocimientos. Por falta de cualquiera de estos dos recursos, el proyecto queda postergado o abandonado. El vance de la cursada de la carrera de postgrado es fundamental para el normal desarrollo del proyecto.

- Severidad (S): 10. No se logra el proyecto, quedando el mismo en estado de abandono.
- Ocurrencia (O): 6. La carga horaria de cursado, sumado a la necesaria para el proyecto y al cumplimiento laboral del responsable, impiden el desarrollo programado del proyecto.

b) Tabla de gestión de riesgos: (El RPN se calcula como $RPN=S \times O$)

Riesgo	S	O	RPN	S*	O*	RPN*
1 - No conseguir datos reales de vuelos	10	7	70	2	3	6
2 - No obtener los recursos de hardware	2	4	8	-	-	-
3 - El desarrollo presenta mayor dificultad de la espera	5	8	40	2	8	16
4 - No obtener los resultados esperados	8	2	16	-	-	-
5 - No disponer de tiempo o conocimientos	10	6	60	10	2	20

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 20.

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Mitigación riesgo 1: Se pueden obtener datos publicos de métricas de vuelos para entrenar el sistema. Así mismo, se puede lograr entrenar el sistema con datos sintéticos. Esto requerirá tiempo para producirlos.

- Severidad (S): 2. El sistema obtenido obtiene su entrenamiento de datos similares o sintéticos, demostrando que el sistema, de contar con datos reales, aprenderá de los mismos.
- Probabilidad de ocurrencia (O): 3. Los datos son obtenidos desde fuentes alternativas.

Mitigación riesgo 3: Se debe reevaluar el proyecto, su alcance y definir nuevamente los requerimientos.

- Severidad (S): 2. Al reevaluar el proyecto, se acortará el alcance y los requerimientos, disminuyendo de esta manera la complejidad del mismo y por ende sus plazos.
- Probabilidad de ocurrencia (O): 8. No se puede reducir la probabilidad de ocurrencia de este riesgo.

Mitigación riesgo 5: Se optimizará el tiempo de trabajo, invirtiendo tiempo en jornadas extras. Así mismo, no se asumirán compromisos nuevos que demanden dedicación de tiempo en la jornada laboral. Se requiere ser metódico y programado.

- Severidad (S): 10. El éxito del proyecto, depende directamente del tiempo invertido en estudio y trabajo en el mismo. Esto es intransferible y por lo tanto no se puede mitigar la severidad. Tercerizar esto no es parte del proyecto.
- Probabilidad de ocurrencia (O): 2. Optimizar el tiempo de estudio y trabajo disminuirá la ocurrencia de este riesgo.

14. Gestión de la calidad

- Req #1: El sistema deberá implementarse como un servicio de clasificación de anomalías que deberá procesar series temporales multivariable e indicar los intervalos en los que se

detecta la presencia de una anomalía. Internamente puede contener múltiples modelos. El usuario deberá poder seleccionar el modelo y los parámetros del mismo (por ejemplo el umbral) por argumentos pasados por línea de comando o desde un archivo de configuración.

- Verificación: El sistema se alimentará con una base de datos con un formato predeterminado, pero seleccionable entre dos opciones. Asimismo, se ingresará al sistema la configuración mediante línea de comandos o con un archivo editable de configuración. Luego de ejecutar el sistema, se obtendrá un informe del análisis realizado, donde figure la configuración, las anomalías detectadas y la ventanas de tiempo donde surgieron las mismas.
 - Validación: Inspección visual.
- Req #2: El sistema deberá poder consumir datos de un CSV o de alguna de las siguientes bases de datos: SQL, InfluxDB o ElasticSearch.
 - Verificación: Se proveerá de un set de datos en formato CSV, MDF/NDF.
 - Validación: El cliente inspeccionará que el sistema permita archivos CSV, MDF/NDF de InfluxDB o ElasticSearch.
 - Req #3: La entrada estará compuesta por series temporales multivariadas. Estas series, pueden ser de alguno de los siguientes tipos:
 1. Numérico: entero o decimal.
 2. Booleano.
 3. Categórico: representado como una cadena de texto.
 - Verificación: Se ingresará una base de datos con series numéricas (entero o decimal), booleana y categórica pudiendo el sistema procesarlas independientemente.
 - Validación: El cliente realizará inspección visual.
 - Req #4: Los modelos incluidos en la solución deben estar conformados por distintos tipos de preprocesamiento y clasificación, pudiendo incluir, a modo de ejemplo:
Preprocesamiento:
 1. Reducción de dimensiones.
 2. Codificación de variables categóricas.
 3. Transformación de distribuciones de entrada.
 - Verificación: Se ejecutará el sistema para un mismo set de datos, eligiendo diferentes parámetros de configuración. Es de esperar que se obtengan por lo tanto diferente resultados.
 - Validación: El cliente realizará inspección visual.
 - Req #5: Los modelos incluidos en la solución deben estar conformados por distintos tipos de preprocesamiento y clasificación, pudiendo incluir, a modo de ejemplo:
Clasificación:
 1. Isolation Forest.
 2. Support Vector Machines.
 3. Autoencoders con LSTM.

4. Máquinas de Boltzmann restringidas.
5. Redes convolucionales de tiempo dilatadas.
 - Verificación: Se ejecutará el sistema para un mismo set de datos, eligiendo diferentes modelos y comparando los resultados. Es de esperar que se obtengan por lo tanto diferentes resultados los cuales tendrán una misma naturaleza.
 - Validación: El cliente realizará inspección visual.
- Req #6: El entregable final deberá incluir aquellos que hayan demostrado un mejor desempeño para los escenarios ensayados.
 - Verificación: Se realizarán ensayos de los modelos y los resultados de desempeño serán comparados, dejando operativos en el sistema, aquellos que presenten resultados con $AUC \geq 60\%$.
 - Validación: Ensayo cuantitativo.
- Req #7: Entradas: No se requiere interfaz de usuario gráfica, pero sí un mecanismo por archivos de configuración o argumentos por línea de comando para seleccionar la entrada, algoritmo clasificador y parámetros operativos, así como disponer de algún tipo de log del sistema para que pueda ser monitoreado.
 - Verificación: Se realizarán las instrucciones por comandos o por un archivo que alimenten al sistema con la configuración. Seguidamente se modificarán los parámetros observando cambios en el comportamiento del sistema.
 - Validación: Inspección visual.
- Req #8: Salidas: Un reporte tipo en formato de archivo de texto o PDF con un detalle indicando ventanas en las que se hallaron anomalías y la causa (umbral de error, clusters, correlaciones, etc.)
 - Verificación: Se generarán casos de prueba con anomalías sembradas y se observarán los resultados en el informe.
 - Validación: Ensayo cuantitativo.
- Req #9: El sistema deberá poder ejecutarse en un server con Linux.
 - Verificación: Realización de ensayo en server Linux
 - Validación: Inspección visual.
- Req #10: Se deberá destinar una partición de los datos suministrados para evaluación de los modelos.
 - Verificación: La base de datos suministrada, será dividida dejando una de sus partes reservada para la fase de ensayos.
 - Validación: Inspección visual.
- Req #11: Los modelos deberán ser evaluados con la métrica AUC.
 - Verificación: Para cada modelo, se levantará una curva de positivos reales versus falsos positivos y se medirá el área debajo de la misma.
 - Validación: Inspección visual.
- Req #12: Para cada escenario considerado, alguno de los modelos deberá permitir obtener un AUC mayor al 60%.

- Verificación: Para cada modelo, se levantará una curva de positivos reales versus falsos positivos y se medirá el área debajo de la misma. Los modelos con curvas con área mayor a 60 % serán los implementados en el sistema.
- Validación: Inspección cuantitativa.
- Req #13: La documentación deberá indicar recomendaciones del modelo a utilizar en distintos tipos de escenarios.
 - Verificación: Confección de manuales de usuario con recomendaciones sobre modelos, seteos en función de los datos a analizar.
 - Validación: Inspección visual.
- Req #14: Se llevará un control de versiones en Gitlab.
 - Verificación: Documentación en Gitlab.
 - Validación: Inspección visual.

15. Procesos de cierre

Cómo etapa final del proyecto, se realizará una revisión del desarrollo del mismo, así como el desempeño del sistema. Se realizarán los siguientes procesos de cierre:

- Se comparará el presente proyecto con la ejecución del mismo, analizando el cumplimiento del alcance y los requerimientos. También se evaluarán los tiempos consumidos en el desarrollo de las tareas descritas en la sección 9 y el cumplimiento de los supuestos. Esta tarea será desarrollada por el Responsable y el Director.
- Se revisarán los aciertos y desaciertos en las técnicas y procedimientos empleados en el desarrollo del proyecto. Se describirá como se abordaron y resolvieron los problemas que se presentaron. Esta tarea será desarrollada por el Responsable, el Director, pudiendo también participar el cliente para hacerse de la experiencia ganada.
- Finalmente, se realizará un acto de presentación del proyecto, permitiendo dar los agradecimientos oportunos a los interesados y colaboradores. Esta tarea será realizada por FIUBA en conjunto con el Responsable.