Centro Universitario de la Defensa
en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Evaluación de un sistema de inteligencia artificial para la detección de anomalías en rutas marítimas

Grado en Ingeniería Mecánica

ALUMNO: Jaime López Ulibarri

DIRECTORES: Miguel Rodelgo Lacruz

Andrés Suárez García

CURSO ACADÉMICO: 2019-2020

Universidade Vigo

Centro Universitario de la Defensa
en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Evaluación de un sistema de inteligencia artificial para la detección de anomalías en rutas marítimas

Grado en Ingeniería Mecánica

Intensificación en Tecnología Naval

Cuerpo General / Infantería de Marina

Universidade Vigo

RESUMEN

Inmersos en la revolución de las tecnologías 4.0 la Inteligencia Artificial obtiene un gran protagonismo gracias a sus importantes avances en numerosos campos. Entre ellos, el aprendizaje autónomo no supervisado destaca como herramienta de procesado de datos y detección de anomalías.

Por otro lado, la Armada dentro de sus misiones y líneas de actuación incorpora la defensa de espacios marítimos, así como el control y vigilancia de las actividades que en estos se desarrollan para evitar la proliferación de ilegalidades. El brazo ejecutor de la Armada en este ámbito es el Centro de Operaciones y Vigilancia Marítima (COVAM), desde él se lleva a cabo entre otras cosas la monitorización de todo el tráfico marítimo que atraviesa las aguas de interés nacional. Esta vigilancia requiere de procesar un continuo flujo de datos en tiempo real que un operador humano difícilmente puede llevar a cabo con éxito y con la rapidez suficiente como para generar una reacción eficaz.

De este problema surge la idea de implementar algoritmos de detección de anomalías y analizar las capacidades de estos en escenarios marítimos. Para ello se han estudiado las capacidades de detección de anomalías de NuPIC una aplicación en Python de IA que utiliza aprendizaje autónomo no supervisado basado en HTM (Una adaptación de las redes neuronales humanas a la tecnología). También se han generado escenarios sintéticos donde se han forzado diversas situaciones consideradas como sospechosas, según criterios de entidades del ámbito de la seguridad marítima como son la Armada y la Guardia Civil. Y finalmente se han analizado las capacidades del algoritmo procesando datos reales de rutas de tráfico marítimo.

PALABRAS CLAVE

Inteligencia Artificial, detección de anomalías, seguridad marítima, HTM, Numenta.

AGRADECIMIENTOS

A mis padres por su apoyo constante durante todo este tiempo y en especial por su ayuda en todo lo que estuvo a su alcance.

A mis tutores Don Miguel Rodelgo Lacruz y Don Andrés Suarez porque a pesar de sus distintos problemas desde la distancia han seguido apoyándome y guiándome estos meses.

A mis padrinos el Teniente de Navío Don Álvaro Fernandez Martín y el Alférez de Navío Don Borja Moriano Rivera, por servirme de guía en lo que a la Armada se refiere no solo en este trabajo sino durante todos estos años.

A Don Guillermo Supervielle Pérez Capitán de la Marina Mercante por su ayuda desinteresada con información y experiencia en temas de tráfico marítimo, así como a Don José Maraver Romero Subjefe de CCS/STM de Tarifa por toda la información aportada de Salvamento Marítimo.

A los Capitanes Don Gonzalo Lafita Becerril y Don Jesús Montes Gutiérrez de la Guardia Civil por su paciencia y ayuda para obtener información relativa a la labor de su institución en el ámbito marítimo.

Y por último a mi ayuda incansable y apoyo constante en todas las dificultades de estos meses.

CONTENIDO

Contenido 1
Índice de Figuras 4
Índice de Capturas 5
Índice de Tablas 6
1 Introducción y objetivos 1
1.1 Introducción y motivación del trabajo 1
1.2 Descripción del problema y objetivos 2
1.3 Organización de la memoria 3
2 Estado del arte 1
2.1 Introducción. El transporte marítimo y la seguridad en la mar 1
2.2 Seguridad marítima en España 3
2.2.1 El entorno marítimo en la Seguridad Nacional 3
2.2.2 Actividades ilícitas en el espacio marítimo 5
2.3 La acción del Estado en la mar 7
2.3.1 Salvamento Marítimo y el control del tráfico marítimo 7
2.3.2 La Guardia Civil y el SIVE 8
2.3.3 Otros actores de la seguridad marítima en España 9
2.4 M.S.O. Maritime Security Operations 10
2.4.1 El interés de la Armada en la Seguridad Marítima 10
2.4.2 MCVSM 10
2.4.3 COVAM 10
2.5 Inteligencia Artificial 11
2.5.1 Introducción. Tecnologías 4.0 11
2.5.2 Orígenes y desarrollo 12
2.5.3 Definición y desarrollo teórico 14
2.5.4 Aprendizaje autónomo 15
2.6 HTM (Hierachical Temporal Memory) 16
2.6.1 Los pilares de las HTMs 17
2.6.2 El aprendizaje en HTM 19
2.7 NuPIC (Numenta Platform for Intelligent Computing) 20
2.7.1 Numenta 20
2.7.2 NuPIC 20
2.7.3 NiPIC Geospatial 21

2.8 Detección de anomalías 23
2.9 Trabajos previos en la materia y su evolución 24
3 Desarrollo 1
3.1 Estudio de necesidades y capacidades en Seguridad Marítima 1
3.1.1 Integración de los actores de la seguridad marítima. 1
3.1.2 Recepción y procesado de información 2
3.1.3 Las capacidades de la IA frente a las necesidades de COVAM 3
3.2 Herramientas de trabajo e instalación 3
3.2.1 Sistema operativo, Ubuntu. 3
3.2.2 Python como lenguaje de programación 3
3.2.3 Nupic 4
3.2.4 Nupic geospatial 5
3.3 Generar rutas sintéticas 6
3.3.1 Microsoft Office Excel 7
3.3.2 Open CPN 7
3.3.3 Conversión a .csv 8
3.4 Pruebas iniciales con NuPIC 8
3.4.1 Ejecutar NuPIC Geospatial 9
3.4.2 Capacidad de aprendizaje 11
3.4.3 Rutas sintéticas sencillas 11
3.4.4 Rutas sintéticas marítimas 11
3.5 Evaluación de la relación posición-anomalía 11
3.5.1 Objetivo 12
3.5.2 Generar la ruta para el análisis 12
3.5.3 Desarrollo de las pruebas 12
3.6 Pruebas con datos reales 13
3.6.1 TV32. Obtención de datos reales 13
3.6.2 Pruebas con el conjunto de datos y problemas derivados 15
4 Evaluación y resultados 1
4.1 Análisis de rutas sencillas 1
4.1.1 Capacidad de aprendizaje 1
4.2 Evaluación de la relación desvío – anomalía 2
4.2.1 Escala 1 metro 2
4.2.2 Escala 2 metros 4
4.2.3 Escala 3 metros 5
4.2.4 Resultados y conclusiones 8

4.3 Prueba con datos reales 8
5 Conclusiones y líneas futuras 1
5.1 Capacidades destacables del algoritmo 1
5.2 La detección de anomalías en escenarios marítimos 1
5.3 El creciente desarrollo de las redes neuronales 1
5.4 Líneas futuras de la investigación 2
5.4.1 Procesado en tiempo real 2
5.4.2 Unión máquina-humano. Analizar las posibilidades de integración. 2
6 Bibliografía 1
Anexo I: Diccionario de siglas 6
Anexo II: geospatial_anomaly.py 7
Anexo III: run.py 10
Anexo IV: modelparams.py 13
Anexo V: aismsg.py 17
Anexo VI: decodificación de un mensaje ais 18

ÍNDICE DE FIGURAS

Figura 1-1 Sala de control COVAM [13] 2
Figura 2-1 Porcentaje del comercio mundial transportado por vía marítima. [1] 1
Figura 2-2 Ranking de países desarrollados con mejor conectividad marítima. [1] 2
Figura 2-3 Puertos europeos de mayor relevancia en el último cuarto de 2018. [6] 2
Figura 2-4 Objetivos generales de la Seguridad Nacional [2] 3
Figura 2-5 Características de los espacios comunes globales. [2] 4
Figura 2-6 Rescate de una patera frente a la costa andaluza [62] 6
Figura 2-7 Distribución de los centros de Salvamento Marítimo [68] 8
Figura 2-8 Sensores del SIVE [8] 9
Figura 2-9 Esquema de la evolución de la industrialización [10] 11
Figura 2-10 Ejemplos de tecnologías 4.0 [67] 12
Figura 2-11 Interfaz de Eugene Goostman. [53] 13
Figura 2-12 Atlas durante diversas pruebas de funcionamiento. [52] 13
Figura 2-13 Técnicas y campos de la IA. Fuente: Elaboración propia. 15
Figura 2-14 Representación de neuronas biológicas y artificiales. Fuente: Elaboración propia 17
Figura 2-15 Modelo de jerarquía simplificado de cuatro regiones HTM [17] 18
Figura 2-16 Representación de una región HTM [16] 18
Figura 2-17 Ejemplo de SDR. Fuente: Elaboración propia 19
Figura 2-18 Formula para calcular el grado de anomalía [17] 21
Figura 2-19 Comparativa de tipos de rutas. [19] 22
Figura 2-20 SDR a partir de posición GPS. [18] 23
Figura 3-1 Resumen de áreas de responsabilidad del COVAM 2
Figura 3-2 DST del Estrecho de Gibraltar en OpenCPN 8
Figura 3-3 Diagrama de ejecución de NuPIC. Modificado del original [12] 10

ÍNDICE DE CAPTURAS

Captura 3-1 Versión Python 4
Captura 3-2 Instalación con éxito NuPIC Geospatial 6
Captura 3-3 Ejemplo ejecución NuPIC 10
Captura 3-4 Captura de pantalla del TV32, tráfico en el estrecho 13
Captura 3-5 Archivo .log generado por el TV32 14
Captura 3-6 Petición de fichero AIS 14
Captura 4-1 Capacidad de aprendizaje NuPIC ruta recta sencilla 1
Captura 4-2 Ruta aleatoria sencilla 2
Captura 4-3 Media de anomalías - Escala 1M 3
Captura 4-4 Anomalías de los desvíos frente a la ruta original con 1 m de escala 3
Captura 4-5 Anomalías rutas escala 1m 4
Captura 4-6 Ruta - desvío de 5 a 8 metros 4
Captura 4-7 Media anomalías escala 2 metros 4
Captura 4-8 Gráficas de los distintos desvíos respecto a la ruta base con escala de 2 metros 5
Captura 4-9 Ruta - desvío de 1 a 4 m (Escala 3m) 6
Captura 4-10 Escala 3 metros media de anomalías 6
Captura 4-11 Escala 3m. Desvío de 12 a 22 m 7
Captura 4-12 Escala de 3m. Desvío de 8 a 12 m 7
Captura 4-13 Escala 3m. Desvio de 5 a 7m 7

ÍNDICE DE TABLAS

Tabla 1 Opciones de ejecución de NuPIC [25] 9
Tabla 2 Parámetros de ruta para analizar variaciones 12
Tabla 3 Umbral de anomalías.....8

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Introducción y motivación del trabajo

La Estrategia de Seguridad Nacional, documento que establece las líneas de acción del gobierno de la nación en materias de seguridad define como una de las amenazas y desafíos de la seguridad nacional la vulnerabilidad del espacio marítimo. Se considera uno de los espacios comunes globales que permiten el libre flujo de bienes, información y servicios. Son espacios que se caracterizan por carecer de fronteras físicas, así como soberanía y jurisdicción de los Estados. Por ello, las amenazas para la Seguridad Nacional los adoptan como escenarios de actuación idóneos.

Por otro lado, la Ley Orgánica 5/2005 de la Defensa Nacional encomienda la misión de vigilancia y seguridad de los espacios marítimos comunes a las Fuerzas Armadas con carácter permanente. De esta Ley surgen las actividades y operaciones de Vigilancia y Seguridad Marítima cuya responsabilidad recae en ALMART (Almirante de Acción Marítima) bajo la dependencia operativa del CMOPS (Comandante del Mando de Operaciones). Estas operaciones suponen toda actuación relacionada con asegurar la integridad, la defensa de los derechos fundamentales de los españoles y de la legalidad vigente en el espacio marítimo.

Para asesorar a ALMART y permitirle un control continuo de todas las actividades llevadas a cabo en la mar territorial, así como en los territorios de ultramar donde España tenga intereses, dispone del Centro de Operaciones y Vigilancia de Acción Marítima (COVAM), con disponibilidad 24/7. En él, se coordinan las operaciones de seguridad marítima (MSO), las actuaciones en el ámbito de la cooperación con otras agencias del entorno marítimo de carácter civil como militar ya sea de carácter nacional o internacional. Todo ello se traduce en un ingente flujo de información centralizado en COVAM (ver Figura 1-1), entre ellos la información de todos los buques que desarrollan sus actividades en aguas de soberanía nacional o de interés estatal.

Figura 1-1 Sala de control COVAM [25]

Para procesar toda esa cantidad de información requiere de personal cualificado que realiza una vigilancia continua las 24 horas del día monitorizando todo el tráfico marítimo nacional entre otros cometidos. Aquí es donde surge el problema planteado en este estudio y es la incapacidad de un operador humano de analizar las infinitas rutas de tráfico mercante, monitorizarlas y ser capaz de detectar una anomalía con la antelación suficiente como para reaccionar ante el suceso.

1.2 Descripción del problema y objetivos

Este problema encuentra una posible solución en las capacidades de los recientes avances tecnológicos. Pudiendo definir como una herramienta capaz de integrarse en el proceso de evaluación de la información recibida en COVAM y su procesado. Permitiendo una automatización del proceso de detección de anomalías en el inmenso flujo de información recibido por el centro de control.

Con este trabajo se pretende desarrollar los siguientes objetivos:

- Analizar las capacidades de los algoritmos de inteligencia artificial, concretamente en el ámbito del aprendizaje autónomo no supervisado y en la detección de anomalías.
- Confirmar las aportaciones que puedan surgir de una posible integración de estos sistemas en las operaciones de vigilancia y seguridad del tráfico marítimo.
- Estudiar el desarrollo de una aplicación que ayude a monitorizar y detectar actividades ilícitas o anomalías en el flujo de información monitorizada por los centros de vigilancia de tráfico marítimo, especialmente en el COVAM en tiempo real.
- Analizar NuPIC como ejemplo de IA de aprendizaje autónomo no supervisado y sus capacidades en la detección de anomalías en datos de tráfico marítimo.

1.3 Organización de la memoria

A continuación, se describe la organización de los distintos apartados de la memoria y la información desarrollada en ellos:

En el capítulo 1 se introduce la motivación que ha llevado a desarrollar el TFG así como el problema que se plantea y los posibles objetivos asumidos para resolver este problema.

En el capítulo 2 se lleva a cabo una descripción del ámbito en el que se va a trabajar, es decir las operaciones de vigilancia marítima, así como los medios, instituciones y otros sistemas que operancon el mismo objetivo y por otro lado se explica los conceptos básicos sobre la tecnología empleada, el estado actual, así como una breve introducción a sus características y capacidades.

En el capítulo 3 se presentan inicialmente las herramientas empleadas en la investigación y una explicación de las pruebas realizadas para conseguir los objetivos planteados originalmente. Estas pruebas se han dividido en tres grandes bloques coincidiendo con los tres objetivos fundamentales. Un primer estudio de las capacidades de la aplicación, una segunda parte donde la ponemos a prueba con escenarios propios de actividades ilícitas y estudiamos sus reacciones, así como la configuración más acertada. Y por último se introducen datos en tiempo real y se analizan con el algoritmo para ver su capacidad de procesado frente a estos.

En el capítulo 4 se definen los resultados en base a los objetivos definidos en un principio, así como su adecuación a los requerimientos de la investigación.

En el capítulo 5 se plasman las conclusiones obtenidas, posibles mejoras o líneas futuras de este análisis destacando las capacidades y posibilidades de esta herramienta.

2 ESTADO DEL ARTE

2.1 Introducción. El transporte marítimo y la seguridad en la mar

El transporte marítimo constituye una de las principales vías de comercio a nivel internacional. En torno a un 70% del comercio mundial se mueve por mar como podemos ver en las estadísticas de [1] en la gráfica Figura 2-1 que representa el porcentaje de cargas y descargas mundial por vía marítima desde 2006. Esto genera un escenario marítimo complejo protagonizado por una red de rutas que enlaza por mar los distintos estados involucrados. La mayor parte de las mercancías que se exportan e importan viajan entre continentes a bordo de buques a lo largo de caóticas líneas y conexiones que abarcan casi la totalidad de la superficie navegable del planeta.

Según datos estadísticos de la Conferencia de las Naciones Unidas de Desarrollo y Comercio (UNCTAD) del año 2018, el aumento en número de buques y conexiones de la flota mercante internacional es continuo correspondiente a un aumento del 3.3%. El volumen de comercio marítimo internacional ronda los 10.7 billones de toneladas en el año 2017.

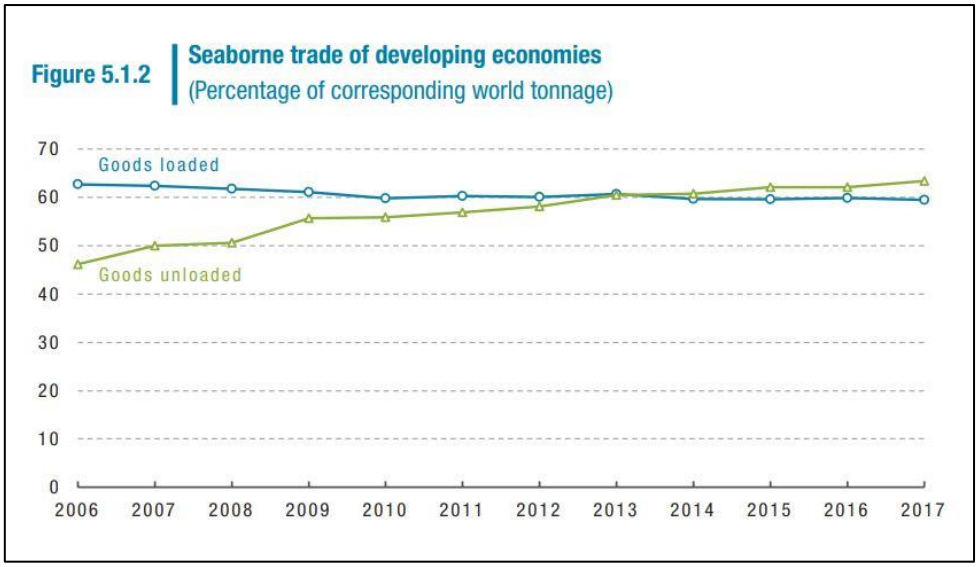


Figura 2-1 Porcentaje del comercio mundial transportado por vía marítima. [1]

Por otro lado, hay que destacar la mejora en la conectividad entre regiones y líneas de tráfico mercante. En este entorno destaca España como país que ostenta el puesto número seis (ver Figura 2-2) por delante de Japón o Francia en el ranking de países desarrollados con mejor conectividad mundial en cuanto a líneas de tráfico marítimo. Esto se traduce en mayor número de rutas que conectan nuestras costas y mayores flujos de buques mercantes lo que supone un reto para los objetivos que se pretenden alcanzar en este trabajo.

Economy (Ranked by 2017 value)	2007	2012	2016	2017
United States of America	84	92	92	94
Netherlands	85	89	90	91
Belgium	74	79	85	90
Germany	89	91	94	90
United Kingdom	77	84	92	89
Spain	71	74	80	88
France	65	70	70	76
Japan	63	63	79	73
Italy	59	66	66	66
Sweden	26	49	53	58

Figura 2-2 Ranking de países desarrollados con mejor conectividad marítima. [1]

A pesar de que, a nivel global los datos estratosféricos del comercio asiático hacen sombra a las capacidades marítimas de nuestro país. España a nivel Unión Europea se encuentra bien posicionado en lo referente a capacidades y flujo de mercancías. Un factor fundamental son sus 5.978 km de costas[1] y su posicionamiento estratégico entre Asia, Europa del Norte y América. Como prueba de la actividad

Top 5 European ports by gross weight of goods handled during the 4th quarter 2018

	2015	2016	2017	2018			2018			
	Q4	Q4	Q4	Q1	Q2	Q3	Q4			
	Gross weight of goods (in Mio tonnes)						Gross weight of goods (in Mio tonnes)	Growth rate on previous quarter (%)	Growth rate on same quarter of previous year (%)	Annual growth rate (%)
Rotterdam (NL)	108.7	109.5	106.2	111.6	106.7	110.2	**113.0**	**+2.5**	**+6.4**	**+1.9**
Antwerpen (BE)	48.3	48.8	50.5	52.8	54.5	52.3	**52.4**	**+0.2**	**+3.8**	**+5.4**
Hamburg (DE)	29.0	28.8	28.0	28.1	29.9	33.2	**33.9**	**+2.2**	**+20.9**	**+5.3**
Amsterdam (NL)	23.0	25.4	24.6	26.1	23.4	24.2	**25.7**	**+6.5**	**+4.6**	**+1.0**
Algeciras (ES)	20.3	20.9	22.0	22.0	22.5	22.9	**21.3**	**-7.1**	**-3.0**	**+6.2**

Note: Turkish ports are not included because data are not available in 2018.
Source: Eurostat (online data code: mar_qg_qm_pwh)

Figura 2-3 Puertos europeos de mayor relevancia en el último cuarto de 2018. [6]

[1] Dato obtenido de la web del Ministerio de Fomento [3]

marítima de nuestro país se presenta el ranking de puertos con mayor peso bruto de bienes transportados en el último cuarto del año 2018 como se puede ver en la Figura 2-3.

Se pone así en relevancia el papel de España como agente europeo en el ámbito del tráfico mercante y de alguna forma dar a conocer la realidad de este sector en nuestro país. Con ello se pretende que el lector adquiera conciencia de la dificultad que supone tener un control sobre este flujo continuo de buques, así como la importancia de mantener la seguridad marítima como prioridad estatal, factor que desarrollaremos a continuación.

2.2 Seguridad marítima en España

2.2.1 El entorno marítimo en la Seguridad Nacional

Dentro de los ámbitos descritos de la Seguridad Nacional en la Estrategia definida por el gobierno del país en el año 2017 [2] se incluye la seguridad en el entorno marítimo (ver Figura 2-4). Este documento define las líneas de actuación del estado en cuanto a seguridad como pilares básicos de una sociedad abierta y democrática como la española y como condición indispensable para el progreso de esta. Destacamos así el concepto de "Seguridad" como la necesidad de los Estados de proteger la vida, la libertad, los derechos y el estado de bienestar de la ciudadanía y cómo toma un carácter crítico en el desarrollo de un país.

Figura 2-4 Objetivos generales de la Seguridad Nacional [2]

Asegurar la "Seguridad Nacional" definida como todas aquellas medidas destinadas a garantizar la paz, la libertad y el estado de bienestar del país conforma una de las misiones fundamentales de las Fuerzas Armadas. Como se recoge en las Reales Ordenanzas para las Fuerzas Armadas [3] en su artículo 3 – Primer deber del militar: "*La disposición permanente para defender a España, incluso con la entrega de la vida cuando fuera necesario, constituye el primer y más fundamental deber del militar…"* Por lo que podemos tomar este documento como aquellas misiones fundamentales que las Fuerzas Armadas deben tener como objetivo.

De entre todos los ámbitos expuestos en el documento nos centraremos en uno que afecta de forma directa a la Armada y al desarrollo de este trabajo, la Seguridad Marítima. Dada su situación geoestratégica, la magnitud de los espacios marítimos de soberanía e interés y por la importancia del sector para el país resulta evidente la necesidad de abordar el problema y establecer una política adecuada al respecto.

Definido como espacio común global, el espacio marítimo destaca por ser escenario ideal para el desarrollo de algunas de las actividades que amenazan la seguridad de la nación. Comparte este puesto de relevancia con el ciberespacio y el espacio aéreo y ultraterrestre. Ambos tienen en común cuatro características principales (ver Figura 2-5):

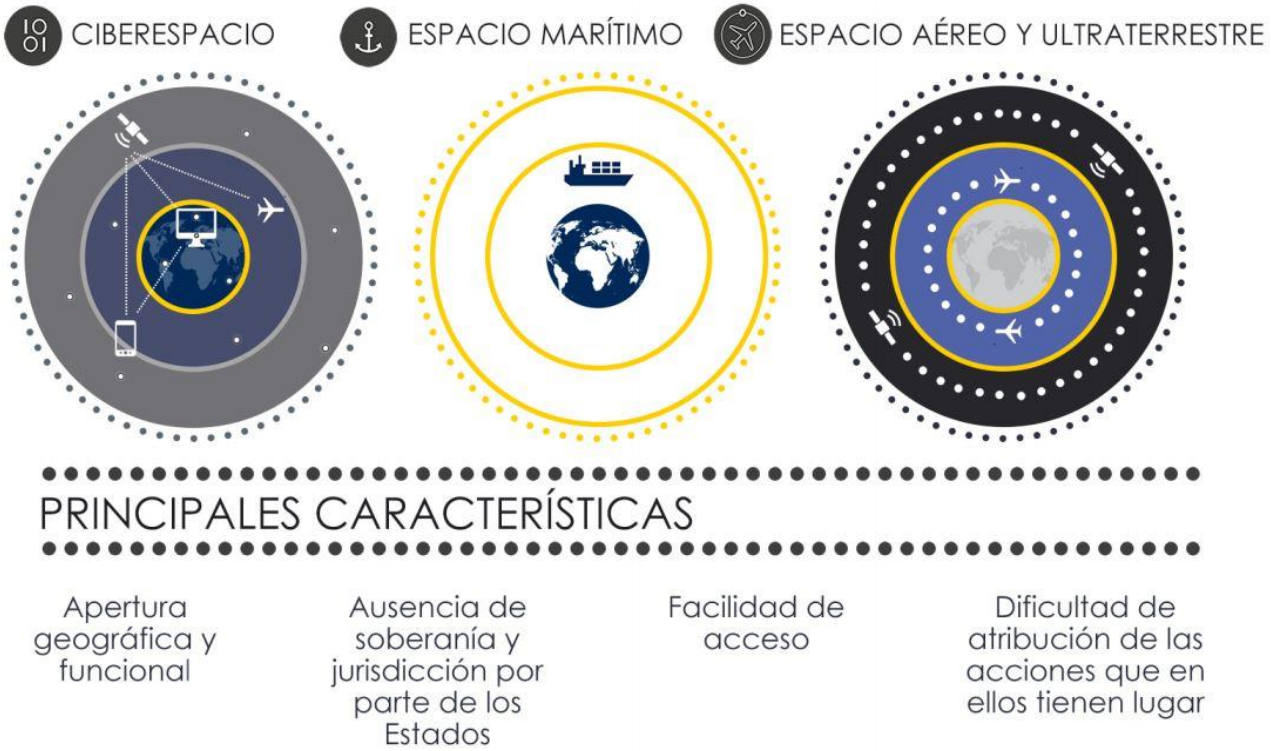


Figura 2-5 Características de los espacios comunes globales. [2]

Por estas razones podemos definir estos espacios como escenarios idóneos que conectan el mundo y carecen de un control positivo por parte de los Estados de manera que esta cierta relajación de las restricciones y control de las autoridades permite que sea aprovechada para el desarrollo de actividades ilícitas.

Centrando esfuerzos en el espacio marítimo podemos destacar su importancia para España como potencia eminentemente marítima y destacar su gran valor estratégico. Además, el tráfico marítimo constituye como ya vimos anteriormente un vector de gran valor para el comercio y el transporte además de para el aprovisionamiento energético. Ya que a diferencia de otras naciones España carece de recursos energéticos por lo que las líneas mercantes constituyen su principal fuente de abastecimiento. *"Las rutas marítimas son vitales para el transporte, el comercio y el suministro energético."* [2]

Dentro de las amenazas desarrolladas en el entorno marítimo y derivadas de actos intencionados y de naturaleza delictiva, destacamos las siguientes definidas en la Estrategia de Seguridad Nacional:

- Piratería
- Tráficos ilícitos
- Actos contra la conservación del patrimonio cultural subacuático
- Redes de inmigración ilegal por vía marítima
- Explotación incontrolada de recursos como la pesca.

Por lo tanto, convergen en un mismo ámbito dos conceptos completamente opuestos. Por un lado, un entorno que supone para el país el origen de parte de su desarrollo económico, de los recursos energéticos que consume y de las relaciones comerciales con otros países. Sin embargo, también es el escenario ideal para el desarrollo de actividades ilegales. Por esto, el Estado define la Seguridad Marítima como un objetivo principal de la Estrategia de defensa nacional.

Estas actividades constituyen una constante preocupación en el ámbito de la seguridad y protagonizan los objetivos de las fuerzas armadas a la hora de combatirlas para garantizar un entorno marítimo seguro. A continuación, se desarrollan un poco más estas amenazas para conocerlas mejor y poder definirlas y caracterizarlas posteriormente en el desarrollo del trabajo de tal forma que se tengan claras las características de las anomalías que se buscan.

2.2.2 Actividades ilícitas en el espacio marítimo

El abanico de actividades desarrolladas en los escenarios marítimos amparados por las características de estos como la cierta ausencia de soberanía y la facilidad de acceso de cualquiera a sus actividades es muy amplio. Sin embargo, a nivel global destacan una serie de ellas que suponen un riesgo para la integridad, y bienestar de la ciudadanía. Además, no solo amenazan a las actividades desarrolladas en el marco de los espacios marítimos, sino que tienen gran influencia en las actividades económicas, políticas y sociales del país. A continuación, se describen y definen las actividades que más interesan como objetivos de estado en el entorno marítimo.

2.2.2.1 Piratería

Entendida como las actividades destinadas a interceptar de forma ilícita los buques que transitan por rutas de tráfico, secuestrarlos mediante la fuerza y pedir un rescate o fianza por su puesta en libertad. No es una práctica común en aguas territoriales nacionales, pero sí tienen influencia en las actividades de interés desarrolladas en escenarios marítimos ya que corresponde a las Fuerzas Armadas y concretamente a la Armada establecer un marco seguro para que la flota de empresas mercantes y pesqueras nacionales ejerzan su actividad con seguridad en el exterior.

2.2.2.2 Tráfico de mercancías ilícitas

El control del tráfico de mercancías ilegales es otro factor que afecta a los intereses de España y de la comunidad internacional ya que a diferencia de otras vías de comunicación como pueden ser la aérea o la terrestre resulta muy difícil vigilar el acceso al país por vía marítima. Por lo que aparecen una serie de vectores que introducen mercancías ilegales en nuestro territorio.

2.2.2.3 Redes de inmigración ilegal por vía marítima

Debido a la proximidad con regiones de extrema pobreza, también se tiene como objetivo el control de la inmigración ilegal. A lo largo de la costa norte de África numerosas mafias se aprovechan de la desesperada situación de personas en países subdesarrollados que, inmersos en la extrema pobreza deciden abandonar sus vidas y buscar un futuro mejor en Europa. Esto lleva a un continuo flujo de inmigración procedente del norte de África y que debido a la proximidad de las costas españolas genera un flujo de inmigración ilegal que debe ser controlado por el estado.

No solo supone el acceso de inmigrantes de forma irregular al país, sino que las condiciones en las que arriban a nuestras costas en muchas ocasiones provocan un riesgo para la integridad de esas personas y su seguridad, ya que en muchos casos ponen en riesgo sus vidas en busca de un futuro mejor, como se puede apreciar en Figura 2-6. Supone un deber conforme a los Derechos Humanos luchar contra estas mafias y acabar con la inmigración ilegal para evitar que se repitan estas escenas en las que la vida y la integridad de estos seres humanos se encuentra en peligro.

Figura 2-6 Rescate de una patera frente a la costa andaluza [68]

2.2.2.4 Explotación incontrolada de recursos como la pesca

El ejercicio de la pesca en las aguas interiores y mar territorial español está reservado exclusivamente a los pesqueros españoles, con las limitaciones que contenga la legislación vigente en materia de pesca. Existen convenios con otros países que regulan esta normativa. Sin embargo las especies marinas en este mundo globalizado deben estar también protegidas para promover una explotación de los recursos que nos ofrece el mar de forma sostenible y castigar a aquellos que atenten con la integridad de estos espacios donde se debe llegar a un equilibrio entre la obtención de recursos y la protección de la biodiversidad. Surge la necesidad de perseguir, castigar e investigar todas aquellas actividades que vayan en contra de las leyes de protección de las aguas territoriales, de la plataforma continental, así como de la intrusión de otros países para la extracción de manera ilícita de recursos propios de las aguas de soberanía nacional. Para ello, ya existe un complejo sistema de inspección y control pesquero con el que la armada colabora como describiremos brevemente a continuación.

2.3 La acción del Estado en la mar

Una vez conocidos los problemas que amenazan la seguridad de nuestras aguas, se presentan los distintos agentes estatales involucrados en garantizar la seguridad en la mar, así como las actividades que realizan estos con dicho fin. Estos organismos mantienen una estrecha relación con la Armada y trabajan conjuntamente cada vez más para alcanzar objetivos comunes.

La seguridad marítima se plantea ya en sí como un concepto muy amplio y que abarca muchos factores y condicionantes. Por este motivo también involucra a numerosos actores que con diferentes medios proporcionan una ayuda eficaz para combatir las irregularidades que puedan desarrollarse en la mar. Las necesidades de coordinación de estos medios suponen un punto interesante a tener en cuenta ya que estos actores poseen objetivos comunes en los escenarios de seguridad y con sus distintas capacidades y medios pueden contribuir a un escenario marítimo nacional seguro.

La Armada como elemento naval del Ministerio de Defensa, tiene atribuidas responsabilidades en todos los ámbitos de seguridad marítima, sin embargo, sus objetivos se han ido desarrollando en escenarios más allá de nuestras costas y fuera de nuestros espacios marítimos dejando la responsabilidad de ciertas materias de seguridad marítima entidades que a continuación se relacionan.

2.3.1 Salvamento Marítimo y el control del tráfico marítimo

Hay que destacar previamente que la información a continuación desarrollada esta proporcionada por el Subjefe de la Estación de Vigilancia de Tarifa de Salvamento Marítimo, así como por un extrabajador de la estación. Ambos poseedores de una prolongada experiencia en el sector, así como en las actividades desarrolladas en el ámbito marítimo. Algunas citas de esta sección han sido obtenidas de la comunicación vía email con los mismos.

El control y la vigilancia del tráfico marítimo se encuentra delegada en la Sociedad de Salvamento y Seguridad Marítima según el artículo 268 de Real Decreto Legislativo 2/2011, de 5 de septiembre [4]. En él cito textualmente: “Constituye el objeto de la Sociedad de Salvamento y Seguridad Marítima la prestación de los servicios públicos de salvamento de la vida humana en la mar (…) la prestación de los servicios de seguimiento y ayuda al tráfico marítimo, de seguridad marítima y de la navegación (…).”

Los numerosos medios de los que disponen, así como los sistemas de monitorización lo dotan de un organismo fundamental en la prevención de accidentes en la mar. Si bien, sus cometidos se centran más en la prevención de abordajes y el salvamento de la vida humana, sin embargo, su colaboración puede llegar a resultar un elemento crucial en la detección de actividades ilícitas y asegurar la seguridad en la mar.

Salvamento marítimo cuenta con 20 estaciones repartidas por toda la costa nacional (ver Figura 2-7 Distribución de los centros de Salvamento Marítimo Figura 2-7), estos centros cuentan con todos los medios necesarios para la detección, motorización, seguimiento y control de los buques que transitan por nuestras costas lo que puede suponer una importante ayuda en la labor de COVAM a la hora de coordinar y vigilar el tráfico en nuestras costas. Estas estaciones están dedicadas a una doble misión. Por un lado, monitorizan los buques que transitan por los DST de sus áreas de responsabilidad, así como el tráfico de entrada y salida de puerto. Por otra parte, coordinan cualquier actuación en materia de salvamento según lo especificado en el convenio SOLAS (Safety Of Life at Sea).

Figura 2-7 Distribución de los centros de Salvamento Marítimo [5]

En estos centros se encuentra personal disponible las 24 horas del día, todos los días el año controlando el tráfico marítimo. Estas actividades se encuentran reguladas por la OMI (Organización Marítima Internacional) a través de uno de sus organismos, en este caso el encargado de certificación y regulación de ayudas a la navegación, la IALA (International Association of Lighthouse Authorities). Este organismo promulga una serie de estándares que deben de cumplir los centros de vigilancia también conocidos como VTS (Vessel Traffic Service) para controlar y mantener la seguridad de la navegación [5].

Esta capacidad de control y vigilancia se la dan los medios electrónicos con los que cuentan en estas estaciones, en concreto la VTS de Tarifa cuenta con algunos elementos como:

- Radares TERMA Scanter 2001, 6 banda X y 3 banda S
- 3 sensores radiogoniometría, RHODE & SCHWARTZ integrados en sistema de presentación
- 3 estaciones meteorológicas GEONICA, integrados en pantallas VTS
- AIS integrado en doble señal [6].
- WIS de interpretación de datos para geolocalización de datos AIS y radar sobre pantalla
- Todos estos sistemas están integrados en las pantallas VTS con 5 puestos para operación en red.

Los VTS no solo vigilan el tráfico en los DST, sino que gracias a las aportaciones del AIS, permite la vigilancia en todo el litoral peninsular e insular. Se atiende a la normativa general estatal (dentro de las 12 millas) y específica establecida por cada Capitán Marítimo para cada zona Marítima con respecto a lugares de fondeo e incluso a la estancia o deriva de buques a menos de 12 millas de la costa española.

Sin embargo, la detección de anomalías en estos centros se realiza mediante: “El trabajo constante en pantalla” lo que supone un factor a tener en cuenta en esta investigación ya que, en la búsqueda de la seguridad marítima centralizada en el COVAM, la información de los VTS puede ser de gran utilidad por la cantidad de medios e información que manejan.

2.3.2 La Guardia Civil y el SIVE

Otra de las instituciones que intervienen en este ámbito de seguridad marítima, es la Guardia Civil mediante el control y vigilancia de las fronteras marítimas del país. Cuando España, en un momento determinado, tuvo que asumir el control de una parte significativa de la frontera sur de la Unión Europea para hacer frente a los desafíos que planteaban, principalmente, el tráfico ilícito de sustancias

estupefacientes y la inmigración irregular, La Guardia Civil tuvo que enfrentarse al reto apoyándose en soluciones imaginativas y oportunas. Resultaba evidente que el esfuerzo, la profesionalidad y la dedicación de tantos y tantos Guardias Civiles precisaban, para el mejor cumplimiento de los cometidos indicados, el complemento y la ayuda de determinados medios específicos.

El SIVE fue la respuesta a esa necesidad; se trataba de un sistema de detección temprana que aportaba la posibilidad de transmitir, en tiempo real, la información precisa y necesaria para facilitar y asegurar la adopción de decisiones. Además, se encuentra en una continua fase de modernización, buscando también introducir nuevas tecnologías en él "el objetivo, a medio-largo plazo, de alcanzar la frontera inteligente." [7] Como se puede ver no solo la Armada se esfuerza en mejorar sus sistemas de la mano de la IA.

El SIVE da una posible solución a dos de los principales problemas de la seguridad marítima en España, la inmigración ilegal y el tráfico de drogas. Mediante un reconocimiento temprano de los vectores de riesgo, se permite la coordinación de los medios de la Guardia Civil para atacar la amenaza.

Al igual que posee Salvamento Marítimo, en la Guardia Civil cuentan con una serie de sensores radar, optrónicos y equipos de comunicaciones que le permiten llevar a cabo la detección, seguimiento y reconocimiento de toda embarcación independientemente de su porte que se aproxime a nuestro territorio.

Figura 2-8 Sensores del SIVE [8]

Toda la información de estos sensores y medios es recogida en unos centros de control donde personal cualificado la evalua y toma decisiones. En este caso vemos otro ejemplo de procesado de información en grandes flujos por factores humanos. Además, en el libro [7] proporcionado por personal del Cuerpo, no tiene en cuenta la contribución de otros elementos como la Armada a estas operaciones.

2.3.3 Otros actores de la seguridad marítima en España

Otras instituciones de gran importancia como la Secretaría General de Pesca (SEGEPESCA) o la Dirección Adjunta de Vigilancia Aduanera (DAVA), cuentan con embarcaciones, así como

profesionales encargados de velar por sus intereses en la mar. Por ello se hace necesarios mencionarlos para destacar la importancia de las actividades aquí expuestas, así como el alcance de este problema.

La Armada, y en concreto el Almirante de Acción Marítima es consciente de ello, y por eso existen documentos que regulan las colaboraciones con los actores de la seguridad marítima [8] [9], así como ejercicios (MARSEC) que permiten desarrollar en conjunto escenarios ficticios que involucran a varias de estas agencias y a sus medios. De ahí que resulte importante mencionarlos en este trabajo y mantener presente una visión en conjunto del problema.

2.4 M.S.O. Maritime Security Operations

2.4.1 El interés de la Armada en la Seguridad Marítima

Para el desempeño de la defensa nacional y el cumplimiento de sus misiones la Armada tiene como objetivo el desempeño de labores de seguridad marítima como elemento fundamental para el desarrollo del Estado. Sin embargo, ¿por qué si ya existen suficientes organismos para cubrir las necesidades en materia de seguridad en la mar, la Armada sigue cubriendo estas capacidades?

El principal teatro de operaciones de los buques de la Flota es la mar, y más concretamente como parte de sus Fuerzas Armadas tiene la misión permanente de defender la integridad de España por mar. Por lo tanto, tiene la obligación permanente de asegurar un entorno controlado para sus operaciones navales. Esto se lleva a cabo gracias a sus contribuciones a la Seguridad Marítima, si bien no son los actores principales, desarrolla permanentemente una vigilancia de las zonas de interés marítimo nacional. Además, supone una fuerza de proyección de esa seguridad marítima más allá de nuestras fronteras por lo que su contribución permanente a nivel internacional a un reconocimiento de todo el tráfico marítimo supone una ventaja táctica a la hora de enfrentarse a escenarios de intensidad.

2.4.2 MVSM

Dentro de las Fuerzas Armadas buscando la visión de un ámbito conjunto en todos los aspectos se crea el Mando de Vigilancia y Seguridad Marítima encargado de desarrollar la misión permanente del Estado Mayor de la Defensa en acciones de seguridad en los entornos marítimos nacionales. “El CMVSM (Comandante del Mando de Vigilancia y Seguridad Marítima), bajo la dependencia operativa del CMOPS, realiza sus operaciones de carácter permanente con la fuerza transferida a la cadena operativa y que constituye el Núcleo de la Fuerza Conjunta número 1 (NFC1), que actualmente cuenta con el CG de ALMART y su COVAM, patrulleros de altura (OPV), buques de medidas contraminas (MHC), aviones de vigilancia marítima, helicópteros, unidades de buceo y de desactivación de explosivos (EOD), Equipos de Seguridad (EOS) de la Fuerza de Protección y Unidades de Operaciones Especiales en el ámbito marítimo (SOMTU) de la Fuerza de Guerra Naval Especial.”

2.4.3 COVAM

“Con este fin se han ido integrando bajo la dirección del Mando de la FAM, que recae sobre el Almirante de Acción Marítima (ALMART), todas las funciones relativas a la Seguridad Marítima desarrolladas por la Armada, la dependencia de las unidades encargadas de desempeñarlas y la generación del "conocimiento del entorno marítimo", producto de la compilación, análisis, correlación y fusión de toda la información relacionada con el ámbito marítimo”. [11] El COVAM surge como elemento centralizado de los esfuerzos en seguridad marítima y se busca que sea capaz de procesar y coordinar información y acciones derivadas de todos los agentes que participan en este ámbito. Por ello, se desarrolla

2.5 Inteligencia Artificial

2.5.1 Introducción. Tecnologías 4.0

Una vez presentado en las líneas anteriores el ámbito en el que se pretende trabajar apreciando la cantidad de variables y datos que se deben procesar, se presenta la Inteligencia Artificial (IA) como herramienta que permita desarrollar un control sobre dicho ingente flujo de información para facilitar las labores de detección de anomalías y actividades ilícitas. Una tecnología que, a pesar de su continua evolución, conforma la base de grandes avances industriales y tecnológicos actuales.

Una actualidad inmersa en una revolución industrial y técnica en constante desarrollo, donde las telecomunicaciones y la informática juegan un papel crucial (ver Figura 2-9Figura 2-9 Esquema de la evolución de la industrialización). Hablamos de la revolución 4.0, término acuñado por primera vez en abril de 2013 en la presentación del documento: "Recommendations for implementing the strategic initiative Industrie 4.0" publicado por Acatech (la academia alemana de ciencias y tecnologías) [10] durante la celebración de la "Hannover Messe" la feria industrial más importante del mundo.

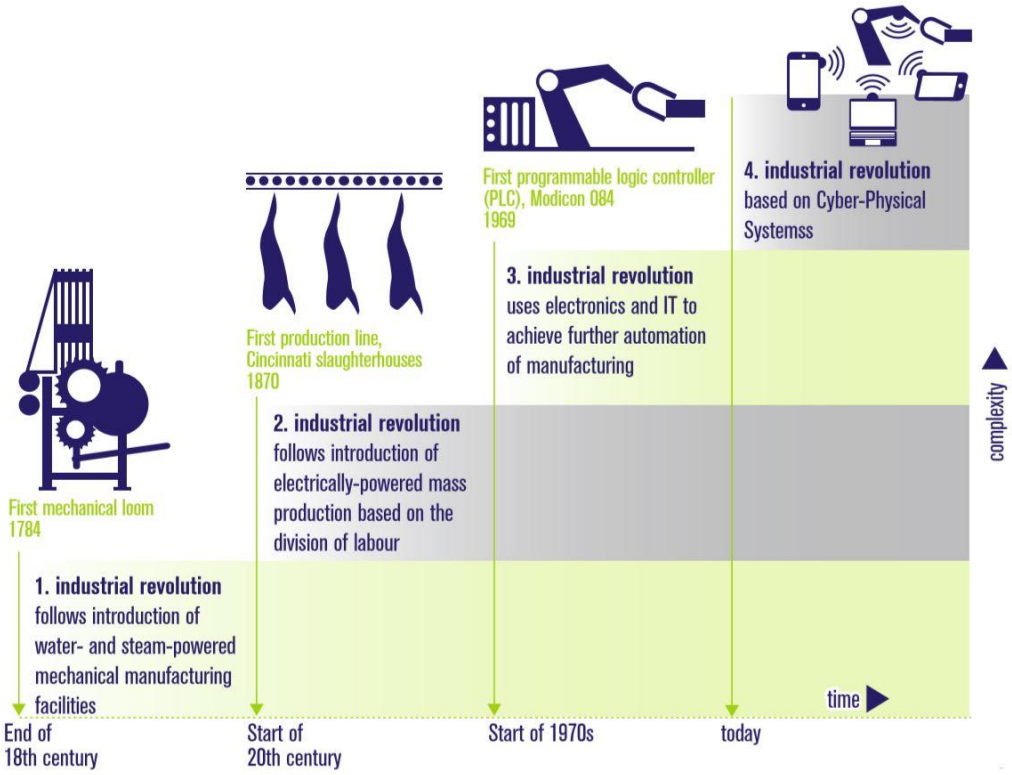


Figura 2-9 Esquema de la evolución de la industrialización [11]

Se presenta entonces un concepto que se ha ido desarrollando hasta la fecha y que va más allá de la industria y sus actividades adquiriendo un carácter global. Podemos decir que estas tecnologías 4.0 toman parte en el conjunto de actividades de la sociedad e incluso interviene en las actividades cotidianas. Sin saberlo, se encuentran rodeados de aplicaciones y tecnología que poco a poco integran estas tecnologías como herramientas de mejora de sus servicios: el reconocimiento facial, las sugerencias de búsquedas adaptadas a nuestros propios gustos, las bases de datos de grandes empresas de cuyos servicios nos beneficiamos a diario e incluso en nuestra propia casa con dispositivos de reconocimiento de voz capaces de conectar en una red local electrodomésticos y controlarlos, así como la realidad de la interconexión a través de redes de todo lo que nos rodea.

El internet de las cosas, Big Data, las simulaciones 3D, o la realidad aumentada son algunos de los pilares en los que esta revolución está apoyando su desarrollo (ver Figura 2-10) , sin embargo, muchos expertos reconocen a la Inteligencia Artificial como el motor que está impulsando este desarrollo y que supondrá en un futuro un antes y un después en el desarrollo humano, siendo comparado incluso con el descubrimiento de la electricidad o de la máquina de vapor.

Figura 2-10 Ejemplos de tecnologías 4.0 [67]

Una tecnología de tal envergadura como es la inteligencia artificial no surge de la noche a la mañana, sino que es fruto de aportaciones de grandes personalidades de las ciencias y de la técnica a lo largo de varias décadas por lo que se presenta a continuación un breve resumen de los grandes hitos que hasta el día de hoy han contribuido a su evolución.

2.5.2 Orígenes y desarrollo

A pesar de parecer un desarrollo de la actualidad, la idea de crear máquinas capaces de reproducir la lógica humana ha sido un problema planteado por los seres humanos a lo largo de la historia. Se conoce que ya en la Edad Antigua, filósofos y pensadores se afanaban por descubrir la forma de automatizar procesos lógicos del ser humano con máquinas. Incluso en el siglo XIV de la mano de Ramón Llull en su obra el Ars Magna [11], aparece una primera aproximación a la IA al expresar su idea de generar el razonamiento mediante su implementación en un sistema mecánico. A pesar de ello, los avances tecnológicos no acompañaron a estas ideas que no llegaron a verse desarrolladas hasta varios siglos después.

No será hasta el siglo XX cuando se desarrolle de la mano de la computación la idea de poder hacer que unas máquinas tan capaces pudieran aportar las suficientes capacidades como para tomar sus propias decisiones sin le necesidad de la existencia de un humano que programe su funcionamiento. Esto ocurrirá de la mano de Alan Turing, el padre de la computación y descubridor de las claves de la máquina enigma. Alan Turing no solo fue capaz de crear un complejo sistema electromecánico que sentaría los cimientos de la informática y la teoría de la computación, sino que fue más allá y predijo que estas máquinas en un futuro podrían alcanzar las capacidades de los seres humanos.

Adelantando estos conocimientos en dos publicaciones de gran relevancia [12] [13], donde A. Turing argumenta cómo las máquinas pueden llegar a pensar como humanos, así como desarrolla su "Imitation game" lo que más tarde se conocerá como Test de Turing en el que pone a prueba a las máquinas haciendo a un ser humano (interrogador) intentar distinguir si se encuentra conversando con una máquina a la cual no puede ver o con un humano. Llegando a afirmar: "Creo que dentro de unos cincuenta años será posible programar computadoras (…) para que jueguen tan bien al juego de imitación que un interrogador promedio no tendrá más del 70 por ciento de posibilidades de hacer la identificación correcta después de cinco minutos de interrogatorio" [12].

El principal germen de esta corriente de investigación aparece en la Conferencia de Dartmouth en 1956, donde J. McCarthy acuña por primera vez el término de Inteligencia Artificial y se establecen las principales líneas de investigación en dicha materia. Sin embargo, entre los años 60 y 90 la investigación en IA sufre una gran desaceleración debido a la falta de medios y capacidades para hacer realidad las aspiraciones de los científicos. Llevando a que la escasez de resultados palpables permitiese un atisbo de confianza en ella. A finales de los 80 los avances tecnológicos y los beneficios económicos de la IA avivan su investigación. Desde entonces, muchos han sido los hitos ocurridos en materia de inteligencia artificial, destacamos:

- Aparición de los primeros vehículos autónomos en 1982, con la furgoneta Dickmanns.
- El nacimiento de Internet en 1991 supone el acceso a uno de los grandes potenciales de la IA, datos en cantidades incontables.
- En 1997 "Deep Blue" gana al célebre jugador de ajedrez Kaspárov a una partida.
- En 2008 Google desarrolla la primera App que reconoce la voz.
- En 2011 Watson de IBM, gana a humanos al concurso de Jeopardy.
- Las técnicas de machine learning permiten a una IA reconocer gatos en 2012
- Atlas, de Boston Dynamics un sistema robótico bípedo capaz de mantener el equilibrio y de sortear distintos obstáculos. (ver Figura 2-12)
- En 2014 fue diseñado Eugene el primer sistema de IA capaz de pasar el test de Turing con éxito (ver Figura 2-11).

Esta pequeña lista resume los grandes hitos relacionados con IA, sin contar los numerosos proyectos que hoy en día se están llevando a cabo para incorporar estos algoritmos y desarrollar sus capacidades.

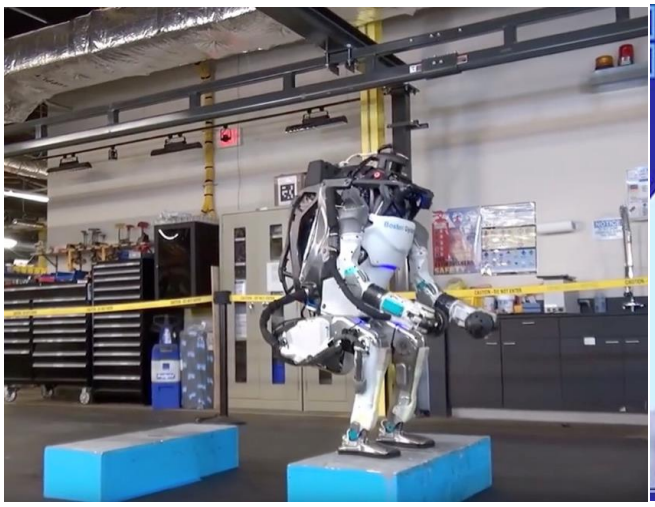


Figura 2-12 Atlas durante diversas pruebas de funcionamiento. [52]

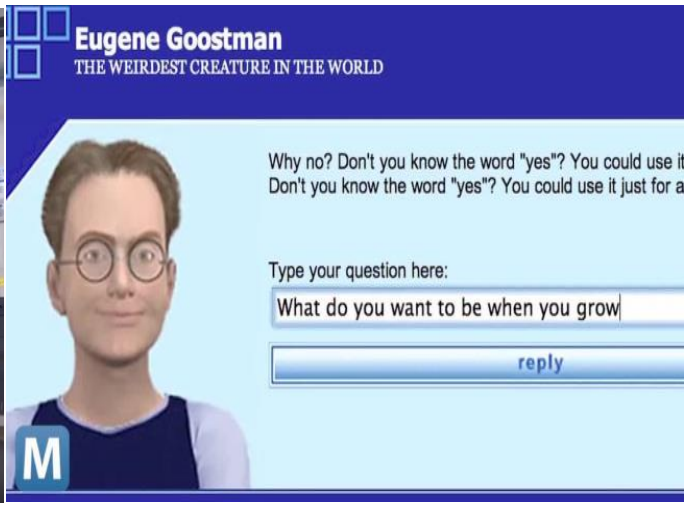


Figura 2-11 Interfaz de Eugene Goostman. [53]

2.5.3 Definición y desarrollo teórico

Expuestos los inicios y el breve desarrollo de la IA, debemos abordar los aspectos teóricos de esta tecnología, así como sus principales características y capacidades. La realidad sobre la IA es que no existe una definición consensuada entre los distintos expertos que trabajan con ella. Entre ellas, John McCarthy en [14] la define como: *"la ciencia y la ingeniería de crear máquinas inteligentes, especialmente programas de computación inteligentes. Está relacionada con la tarea similar de utilizar ordenadores para comprender la inteligencia humana, pero la IA no se limita a métodos que sean observables biológicamente".*

A pesar de las distintas versiones para definir el concepto, todas se encuentran en torno a la idea de imitar o de crear el funcionamiento neuronal humano a través de una máquina. Si buscamos en el diccionario de la RAE la definición de máquina: "*f. Conjunto de aparatos combinados para recibir cierta forma de energía y transformarla en otra más adecuada, o para producir un efecto determinado."* En este caso la cierta forma de energía se corresponde con flujo de datos o información procedentes de numerosos sensores y bases de datos que la informática e internet han hecho posibles. Por otra parte, el efecto deseado es el comportamiento acorde con las respuestas de un humano a los distintos estímulos que recibe a través de sus sentidos.

La Inteligencia Artificial, surge entonces como herramienta fundamental de apoyo a las actividades técnicas y científicas. La principal meta de esta corriente de investigación y desarrollo consiste en adaptar algoritmos matemáticos capaces de imitar las capacidades del cerebro humano.

Como hemos visto anteriormente, hay muchos ejemplos de algoritmos ya desarrollados que dan fe de sus capacidades inteligentes. Todos ellos tienen en común que desarrollan una actividad concreta llegando a superar las capacidades de los humanos, aunque estos logros no se asemejan a la idea de generar un programa global. Por ello se clasifica la IA según su alcance o su ámbito de aplicación en tres tipos:

- Inteligencia artificial débil o estrecha es aquella que se corresponde con todo lo que conocemos hoy en día como inteligencia artificial. Es decir, programas o algoritmos que se encuentran especializados en una capacidad concreta de nuestra inteligencia, ya sea aprendizaje, reconocimiento de voz, análisis de imágenes y un largo etc.

- Inteligencia artificial general nos referimos a ella como aquel programa o algoritmo de IA capaz de desarrollar el amplio espectro de capacidades que la inteligencia nos permite a los humanos llevar a cabo. Abandona la concepción de un sistema específico capaz de realizar una tarea concreta, sino un sistema capaz de abarcar capacidades distintas. Además, esa inteligencia artificial general no sería una gran unión de inteligencias artificiales débiles, cada una para resolver un problema, sino que sería capaz de realizar juicios y razonar ante una situación de incertidumbre.

- Inteligencia artificial fuerte se concibe como aquella que tiene conciencia de sí misma y posee los estados mentales propios de los humanos (sentimientos). Protagonista de numerosas obras de ciencia ficción se les supondría capaces de plantear conflictos éticos y debatir sobre su propia existencia.

A pesar del interés sobre la temática, podríamos estar describiendo de estos sistemas durante páginas y páginas sin embargo la realidad es que actualmente solo conocemos desarrollos en inteligencia artificial estrecha o débil y es donde se centra la investigación. Existen distintos campos en los que es posible imitar el comportamiento de la inteligencia humana, los que se relacionan en la Figura 2-13. Estas protagonizan las principales líneas de la investigación.

Para el problema planteado en el trabajo de tener sistemas capaces de aprender patrones de comportamiento es fundamental por lo que destaca el aprendizaje autónomo o machine learning como línea de estudio del conjunto de distintos sistemas desarrollados en el campo de la IA. A continuación, se desarrollará un poco más esta rama y profundizándose en las utilidades que ofrece.

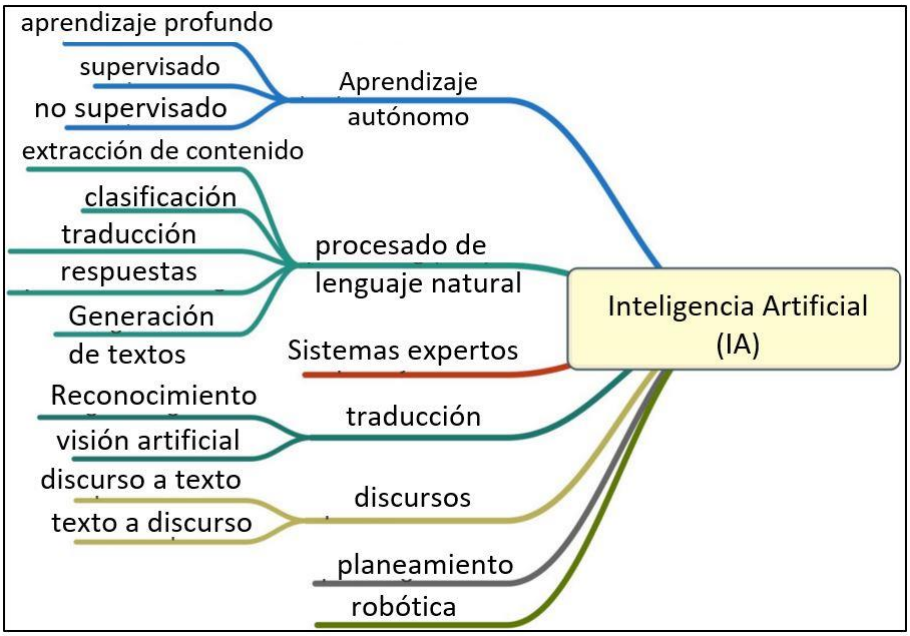


Figura 2-13 Técnicas y campos de la IA. Fuente: Elaboración propia.

2.5.4 Aprendizaje autónomo

Dentro de la inteligencia artificial débil o estrecha, uno de los campos que se abordan en las numerosas capacidades específicas de esta rama de la IA es el "machine learning" (aprendizaje automático). En él redes neuronales artificiales tratan de imitar el funcionamiento neurológico del ser humano a la hora de aprender patrones. A través de complejos algoritmos matemáticos se pretende buscar en un conjunto de datos y tratar de hacer predicciones en base a estos. Estas predicciones confirman las capacidades de aprendizaje permitiendo relacionar patrones de distintos tipos de datos y predecirlos algo así como funciona nuestro cerebro a la hora de relacionar conceptos que ya conocemos.

No existe mejor forma de conocer el funcionamiento de estos sistemas que entendiendo aquel que toma como origen, concretamente el proceso de aprendizaje humano. Ayudándonos de la psicología y de uno de los grandes estudiosos de las teorías del aprendizaje humano, Jean Piaget quién destaca la importancia de los estímulos en los niños y su capacidad para adaptarse a ellos. Es decir, podemos simplificar el aprendizaje humano en su sentido más básico en la recepción de un estímulo o desequilibrio y nuestra capacidad de adaptarnos a él. Citando textualmente una de las obras del reconocido psicólogo [15]: *"el desarrollo mental aparece entonces, en su progresiva organización como una adaptación siempre más precisa a la realidad".* Por lo tanto, se puede obtener del fragmento que los estímulos y nuestra adaptación a ellos son la base del proceso de aprendizaje.

Por un lado, entendemos por estímulo toda información capaz de ser entendida por el ser humano. En el caso de la máquina se traduce en un flujo de datos de distinta magnitud, tipo y procedencia. Si comparamos con la capacidad humana, nuestro cuerpo nos limita a información procedente de únicamente 5 fuentes: vista, oído, tacto, olfato y gusto. Para la inteligencia artificial, esta información puede ser prácticamente de cualquier fuente gracias a los numerosos sensores desarrollados con los que podemos medir prácticamente cualquier variable y a las ventajas de la globalización de Internet.

Atendiendo a el otro ámbito de aprendizaje, la adaptación es el principal logro que la computación ha añadido a la IA. Conseguir que complejos sistemas matemáticos reproduzcan las redes neuronales de manera que permitan esta adaptación y a los estímulos en forma de flujo de datos que reciben aporten una respuesta variable en lugar de programada. Hasta ahora la computación solo se basaba en respuestas planeadas hasta la llegada de la IA que ofrece esa adaptación a los datos procesados dando respuestas en función de la información procesada.

Según ofrecemos esa información al algoritmo podemos distinguir dos tipos fundamentales de aprendizaje autónomo:

- **Aprendizaje autónomo supervisado.**

En el aprendizaje autónomo supervisado el algoritmo requiere de datos etiquetados. Es decir, se deben de mostrar o cargar aquellos patrones que debe de aprender a diferenciar dentro de un conjunto de datos. Para ello, se etiquetan estos y se le ofrecen numerosos tipos de patrones hasta que el algoritmo es capaz de aprenderlos y diferenciarlos dentro de un conjunto de datos.

- **Aprendizaje autónomo no supervisado.**

En este caso el aprendizaje resulta algo más complejo ya que el humano no tiene el control de los parámetros o criterios utilizados por el algoritmo para asociar conjuntos de datos y patrones. En este caso es el propio algoritmo el que a base de procesar un flujo continuo de datos establece un criterio de clasificación y memoriza dichos patrones. Es un sistema más capaz en cuento a procesado de cantidad de información.

Entre ambas posibilidades como veremos posteriormente, el aprendizaje no supervisado constituye una herramienta más potente para los fines elegido ya que nos permite que todo el flujo de información sea procesado sin necesidad de entrenar al algoritmo previamente con los casos de anomalías que queremos detectar. Sin embargo, también constituye el principal reto: adaptar el criterio de aprendizaje del motor de IA para que detecte aquello que buscamos en ese gran flujo continuo de datos y evitar en la medida de lo posible la aparición de falsos negativos que hagan que el sistema no sea preciso en la detección de anomalías.

2.6 HTM (Hierachical Temporal Memory)

Para generar algoritmos de IA, ¿Por qué no utilizar como modelo el cerebro y reproducir su funcionamiento sustituyendo el sistema nervioso y sus estímulos electroquímicos por sistemas digitales? En eso se basa la teoría de Jeff Hawkins autor de "On Intelligence" donde podemos leer [16]: "*Mi deseo no se limita a comprender qué es la inteligencia y cómo actúa el cerebro, sino que pretendo aprender a construir máquinas que funcionen de la misma manera.*" Estudiar el funcionamiento del cerebro y del neocórtex[2] de tal forma que se lleve a cabo un estudio de ingeniería inversa para desarrollar un algoritmo computacional que comparta estos principios de funcionamiento.

El neocórtex conforma la base del pensamiento inteligente en nuestro cerebro. Pensar, tocar, el lenguaje, la planificación son actividades llevadas a cabo por esta región del cerebro. Un gran número de procesos que cabe pensar se relacionan con una gran diversidad de algoritmos neuronales. Lo cierto es que no, la actividad neuronal que subyace en estos procesos se corresponde a unos patrones uniformes de comportamientos neuronales que pueden llegar a ser más o menos fácilmente reproducibles por redes neuronales artificiales.

[2] Neocórtex: parte del cerebro responsable de la inteligencia y de la mayoría de nuestras capacidades lógicas.

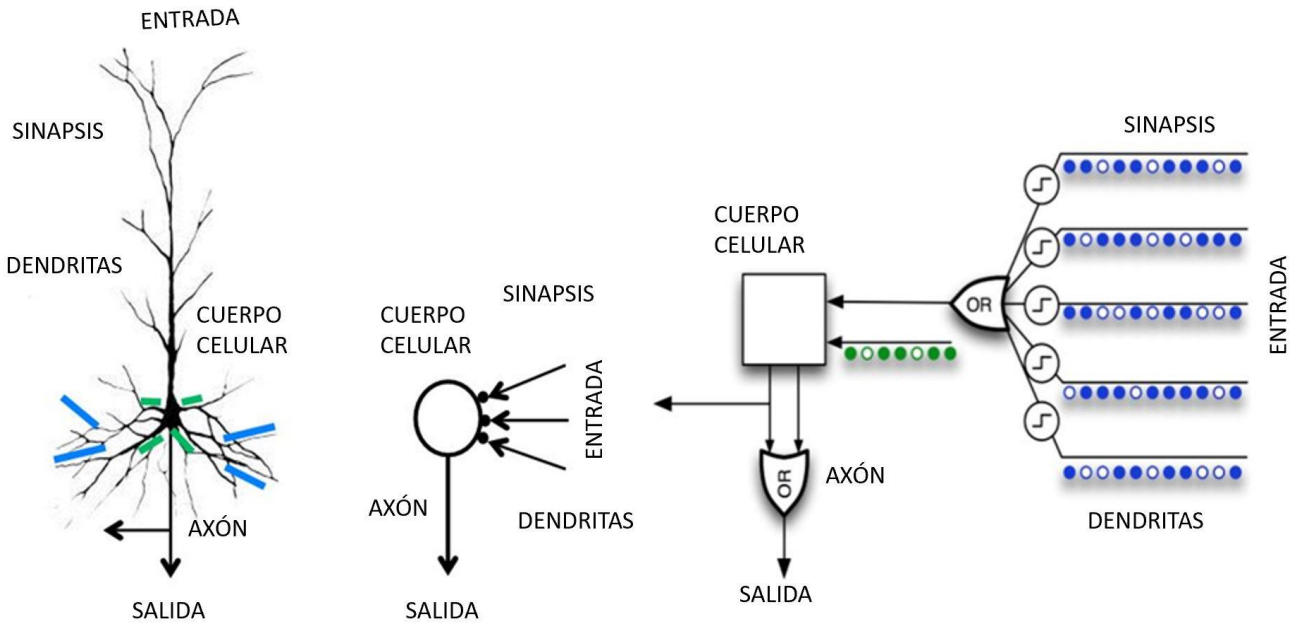


Figura 2-14 Representación de neuronas biológicas y artificiales. Fuente: Elaboración propia

En base a esto crea los algoritmos HTM en los que desarrolla una teoría aplicando los conocimientos sobre el cerebro y su funcionamiento. Jeff Hawkins define las HTMs como un tipo de red neuronal artificial las cuales precisan de ser entrenadas con la exposición a un flujo de datos. Siendo estos datos un factor determinante en su proceso de aprendizaje. Estos datos no precisan de una organización previa o un marcado, es decir, las HTMs son estructuras de aprendizaje no supervisado.

Fundamentalmente su funcionamiento se basa en la memoria. Los ordenadores convencionales generalmente poseen una memoria plana en la cual tenemos controlado dónde y cómo se almacena la información. Sin embargo, en HTM la memoria es más restrictiva, tiene una jerarquía establecida y se encuentra basada en el tiempo. Como se ha dicho anteriormente, HTM es una red neuronal artificial así que emulando a la red neuronal humana se encuentra dividida en regiones y a su vez en columnas y jerarquizada en varios niveles interrelacionados tanto entre niveles como entre columnas.

2.6.1 Los pilares de las HTMs

Se definen los principios básicos de la tecnología HTM en cuatro grandes conceptos. La importancia de la organización jerárquica, cómo se estructuran las regiones HTM, las capacidades de una representación distribuida dispersa a la hora de almacenar datos y la influencia de la información basada en el tiempo.

Una red HTM consiste en regiones organizadas en una jerarquía. Cada región representa un nivel en la jerarquía lo que permite que a medida que se asciende exista convergencia. El beneficio de trabajar con una red jerárquica es la eficiencia. A medida que se asciende en la jerarquía los patrones aprendidos son combinados y reutilizados. Es decir, de una forma más sencilla, la jerarquización permite aprender patrones de manera más sencilla añadiéndole cada vez más simplicidad. Se puede asemejar a un proceso de estudio teórico de un problema de ingeniería. Cuando intentamos hacer cálculos de estructuras complejas a través de simplificaciones y suposiciones reducimos la dificultad del problema y poco a poco le añadimos variables. De lo sencillo a lo complejo es nuestra forma de aprendizaje que se refleja

en las capacidades de esta red neuronal, de tal forma que cuando haya otro objeto con alguna característica básica similar no sea necesario reaprenderla.

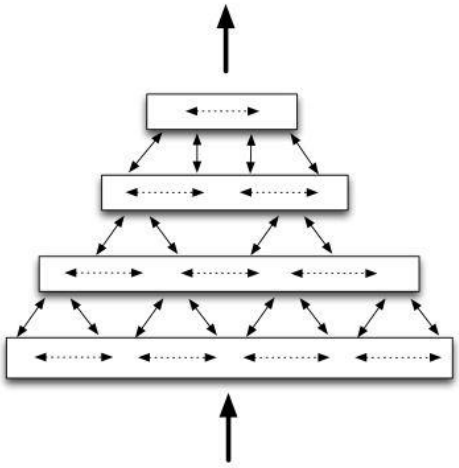


Figura 2-15 Modelo de jerarquía simplificado de cuatro regiones HTM [18]

Además, es posible combinar varias redes HTM cada una correspondiendo a un sensor, de manera que se pueda procesar información de más de una fuente. Y se preguntarán, ¿Qué beneficios nos aporta la organización jerárquica? Los beneficios obtenidos se entienden en materia de eficiencia. Reduciendo el tiempo de entrenamiento de la red y el uso de la memoria.

La región es una unidad principal de memoria y cada región representa un nivel de jerarquía. Las regiones están formadas por células interconectadas organizadas en columnas. La jerarquía está definida por la conectividad existente entre las regiones. Por así decirlo cada región se corresponde con una capa de células interconectadas entre ellas que almacenan un tipo de información en concreto.

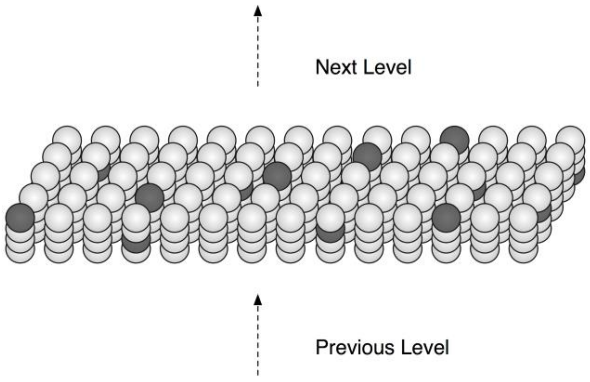


Figura 2-16 Representación de una región HTM [18]

El segundo pilar de las HTM son las representaciones distribuidas dispersas. Todas las neuronas se encuentran interconectadas entre ellas, sin embargo, lo que las caracteriza es que solo un pequeño porcentaje de ellas se encuentran activas en cada momento. Esto es lo que se conoce como representaciones distribuidas dispersas y es la forma que tiene el cerebro de representar la información que procesa. En el modelo de HTM también se trabaja con representaciones distribuidas dispersas en la forma de intentar que la mayoría de los bits tengan valor 0 y solo una dispersión de ellos estén activos.

Estas representaciones se denominan SDR (Sparse Distributed Representation). Es la forma en la que esta tecnologíarepresenta y procesa la información. Lo primero que hace una región HTM al recibir la información de entrada es transformarla en un SDR donde solo una pequeña parte de las células están activas. No son más que vectores de longitud variable con un número reducido de bits activos a 1 y el resto a 0 (ver Figura 2-17).

0100000000000000001000000000000000000000000000000000000000010000.....1011000000

Figura 2-17 Ejemplo de SDR. Fuente: Elaboración propia

El tiempo supone un importante factor para tener en cuenta. La información que recibimos de los sentidos necesita de una continuidad en el tiempo para mejorar su reconocimiento. Por ejemplo, nos sería muy difícil reconocer un sonido con solo un instante de este. El cerebro necesita de una mínima progresión de la información a lo largo del tiempo para poder llevar a cabo una asociación del estimulo con un concepto aprendido.

Para poder aprender los sistemas de HTM necesitan ser expuestos a entradas que varían en el tiempo durante el entrenamiento. Además de una sucesión de datos, estos deben de tener coherencia entre ellos si no, seríamos incapaces de establecer un patrón que los relacionase y facilitar su aprendizaje. Además, un factor importante: el aprendizaje tiene que ocurrir continuamente, y el aprendizaje tiene que ocurrir en la presencia de ruido. Aprender y reconocer secuencias es la base de la formación de las predicciones. Una vez que un HTM aprende qué patrones son los que siguen otros patrones, puede predecir los que se supone que serán los siguientes conociendo la entrada actual y las inmediatamente anteriores.

2.6.2 El aprendizaje en HTM

Lo cierto es que la aproximación de HTM al aprendizaje humano no es exacto ya que el algoritmo no entiende lo que está procesando, es decir no existe un proceso lógico en la predicción de patrones sino un sistema estadístico trabajando con los SDR generados a partir de esa información recibida.

Esta capacidad de ofrecer predicciones es lo que permite llevar a cabo una detección de anomalías. Permiten al sistema hacer una comparativa con lo que hasta ahora considera como un comportamiento normal y la información que recibiría en tiempo real. Así una región sería capaz de saber que ha ocurrido una anomalía cuando ambos valores no coinciden en tiempo o forma.

2.7 NuPIC (Numenta Platform for Intelligent Computing)

2.7.1 Numenta

En este apartado se presenta la herramienta de IA elegida para estudiar las posibilidades de estos algoritmos en cuanto a detección de anomalías en entornos marítimos. El algoritmo seleccionado es NuPIC una herramienta de software libre ofrecida para posibles desarrollos de aplicaciones de IA en detección de anomalías usando HTM. Jeff Hawkins, autor ya mendionado en apartados anteriores de esta memoria, además de desarrollar numerosos estudios y publicaciones sobre HTMs, es cofundador de Numenta, empresa estadounidense cuya principal línea de investigación es la neurociencia y sus implicaciones en el ámbito de la IA.

Numerosos estudios avalan la experiencia de Numenta y sus miembros en máquinas inteligentes y en su concepción de estudiar el cerebro humano para desarrollar distintos proyectos que permitan a las máquinas alcanzar las capacidades del sistema nervioso. Por otro lado, desde Numenta hacen grandes esfuerzos para acercar esta tecnología al resto de la comunidad científica y permitir a otros desarrolladores aportar ideas y mejoras en todos sus proyectos. Esta es la base de que sustenta NuPIC como algoritmo de software libre. Además, desde la propia empresa favorecen la participación y las aportaciones de todo interesado con una visión de comunidad.

2.7.2 NuPIC

Podemos definir NuPIC como una herramienta de aprendizaje autónomo no supervisado que implementa algoritmos de HTM para la detección de anomalías en grandes flujos de datos. Forma parte de una de las aportaciones de Numenta dentro de sus investigaciones en el campo de las HTMs. Se establece como algoritmo de uso libre cuyo código es compartido y además de este lleva asociado a toda una comunidad de desarrolladores e investigadores que a través de foros de debate, conjunto de datos, tutoriales y videos explicativos facilitan el trabajo a todos aquellos que se inician en el manejo de la aplicación. El objetivo de NuPIC es acercar de manera sencilla los algoritmos de IA y demostrar su utilidad en el campo de la detección de anomalías.

A continuación, se procede a exponer algunas de las características de NuPIC y una breve introducción a su modo de trabajo en lo referente a funcionamiento y capacidades. A pesar de ser una herramienta de software libre y de permanecer en desarrollo, se puede afirmar que es un algoritmo potente con una gran capacidad.

El código de NuPIC utiliza Python como lenguaje de programación, y definimos los siguientes componentes como parte fundamental de su funcionamiento, respaldándonos en el trabajo previo en [18]:

- **Datos de entrada.** NuPIC utiliza archivos .csv como fuente de datos. Las variables que se pretenden sean estudiadas por la aplicación deben de estar organizadas en este tipo de archivos (CSV del inglés: “Comma Separated Values”).
- **Codificadores.** Permiten transformar los datos de entrada que pretendemos procesar con el algoritmo en SDRs. Utilizaremos un codificador u otro en función de la importancia o del tipo de dato que queramos incorporar.
- **Regiones jerarquizadas.** En este módulo es donde NuPIC tiene su red neuronal arificial. Aquí se incorporan dos regiones, una temporal y otra espacial. Con ellas se identifican patrones espaciales, concretamente con el “spatial pooler”(SP) y los temporales con el “temporal pooler”(TP). Una vez definidos estos patrones genera un SDR como predicción del siguiente dato que espera recibir.
- **Clasificador CLA.** Genera una función de distribución de probabilidad para cada campo predicho caracterizándose mediante dos parámetros:

- *Alpha: valor con el que se calcula la media móvil.*
- *K: pasos hacia el futuro que se buscan para la predicción*

En líneas generales utiliza las SDRs generadas y obtiene como producto una predicción.

- **Grado de anomalía (anomaly score).** Es el valor que NuPIC genera para cada entrada. Analizan el número de coincidencias de 0 y 1 existente entre el SDR introducido y la predicción que él genera. Este valor se obtiene de la fórmula de la Figura 2-18.

$$AnomalyScore = \frac{|A_t - (P_{t-1} \cap A_t|}{|A_t|}$$

$A_t = Columnas\ predichas\ en\ el\ instante\ t$

$P_{t-1} = Columnas\ activas\ en\ el\ instante\ t$

Figura 2-18 Formula para calcular el grado de anomalía [18]

2.7.3 NiPIC Geospatial

Finalmente, la gran aportación de NuPIC a esta investigación, viene de la mano de la capacidad de generar SDRs mediante posiciones geográficas. Esto es lo que fundamentalmente lleva a cabo el "Geospatial Coordinate Encoder" del cual explicaremos su funcionamiento a continuación. Debemos tener en cuenta que buscar anomalías en rutas ya genera cierta complejidad a la codificación, más aún si se pretende abarcar cualquier tipo de ruta desde un hombre a pie a un avión despegando. En ambos casos una misma ruta será anómala según qué caso.

Posiblemente con el siguiente ejemplo se pueda clarificar la información. Si se realiza un desplazamiento a pie de un punto A hacia un punto B desplazándonos solo 3 metros tal y como se describe en Figura 2-19, la posición GPS será prácticamente la misma a diferencia de si nos desplazamos 20 metros que ya la diferencia en las posiciones es sustancial, así como el tiempo del trayecto. Sin embargo, si tomamos otro medio de transporte como es un coche, los desplazamientos son más rápidos y a más distancia por lo que en un viaje en coche 20 metros la variación de la posición respecto a la totalidad del desplazamiento sería casi despreciable. Si tomamos ahora una separación de 2000 metros, la diferencia volvería a ser sustancial. Con esto se viene a explicar una de las capacidades de los codificadores espaciales y es que adaptan su forma de análisis de las posiciones en función de la velocidad del desplazamiento. Se detalla más adelante las dificultades que añade esto a la caracterización y estudio del algoritmo.

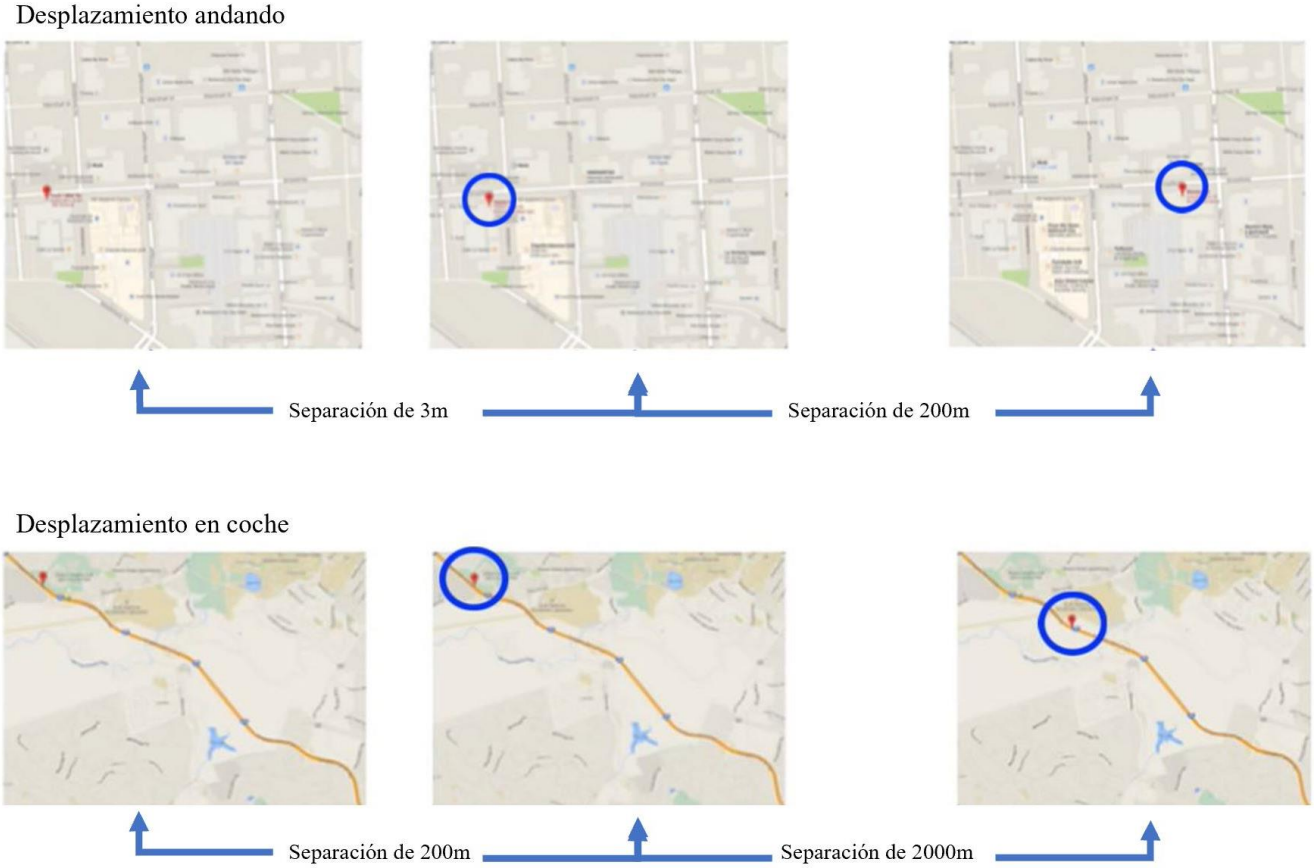


Figura 2-19 Comparativa de tipos de rutas. [20]

La codificación geoespacial tal y como explica en [19] uno de los desarrolladores de "NuPIC Geospatial" tiene dos factores a tener en cuenta: La velocidad de la ruta y la escala elegida como se explica en 3.4.1. Y a grandes rasgos lo que lleva a cabo este codificador es una división del espacio declarado en la escala en pequeños cuadrados a los cuales asignados valores aleatorios. El primer valor se corresponde con el "orden" de la celda y es un número del 0 al 1 en función de la latitud y la longitud que ocupe. El segundo valor, es la posición del bit que representa esa celda. Por último (ver Figura 2-20) se seleccionan los k valores más altos y se activan los bits que correspondan con esas celdas. El número de celdas va asociado al número de bits activos de la SDR. Obteniendo así un vector, o codificador de la posición.

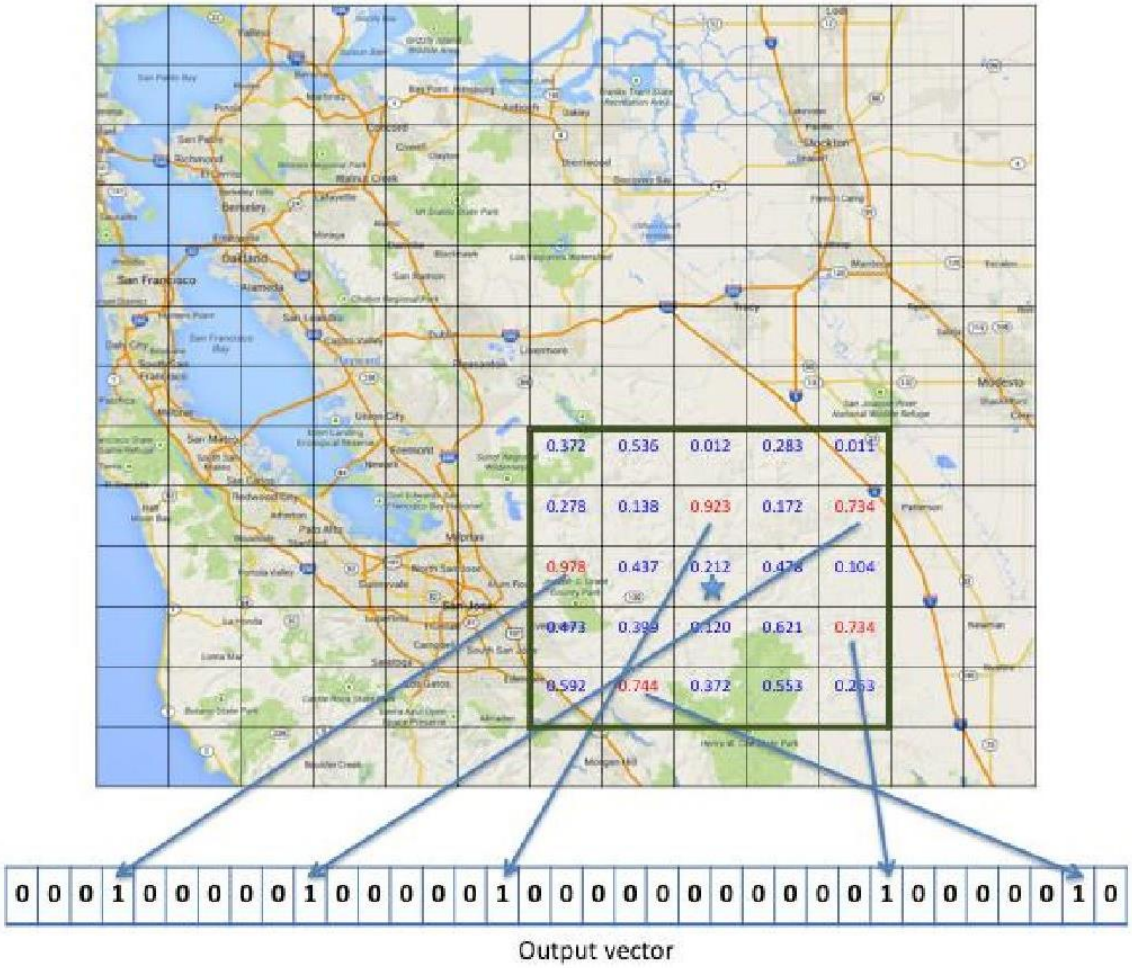


Figura 2-20 SDR a partir de posición GPS. [19]

2.8 Detección de anomalías

Para presentar mejor las capacidades de la tecnología HTM y dar a conocer sus posibilidades, desde Numenta se ha generado una herramienta que permite evaluar distintas aplicaciones de detección de anomalías en tiempo real generando una clasificación en función de determinados parámetros, esta aplicación recibe el nombre de NAB (Numenta Anomaly Benchmark) [20]. El propósito de NAB es proporcionar una herramienta con la que calificar los distintos desarrollos en cuestión de detección de anomalías.

2.9 Trabajos previos en la materia y su evolución

Esta investigación tiene su origen en una línea de proyectos basados en la herramienta de Numenta y los cuales han supuesto un importante apoyo para no partir de cero en lo que a la investigación con NuPIC se refiere. En todos ellos se trata la detección de anomalías como un hecho importante, sin embargo, no se ofrece una aplicación real a las capacidades de la herramienta. Seguidamente se redacta un breve resumen en líneas generales de las aportaciones de los TFG previos en la materia.

El primero de todos [21] se presenta como un acercamiento a las capacidades de NuPIC como herramienta de detección de anomalías en rutas de tráfico marítimo. Este, se centra en la detección de anomalías centrando los esfuerzos en el análisis del DST de Fisnisterre y en las rutas generadas por un contacto concreto, el Volcán de Tindaya.

Posteriormente en el año 2017 fueron dos los trabajos desarrollados en la materia. Uno de ellos [22] pretende abarcar la capacidad de detección en tiempo real de las anomalías y el segundo de ellos [18] pilar fundamental de apoyo a esta investigación trataba de abarcar tanto el tráfico marítimo, como el aéreo además de representarlo de forma visual.

Por último, siguiendo con la línea de [18] otro alumno profundiza en su trabajo [23] en el análisis del tráfico aéreo. Este último resulta de poca relevancia para este TFG, sin embargo, forma parte de la línea de investigación en materia de detección de anomalías de los últimos años.

Teniendo en cuenta todos ellos y gracias al interés desatado en la Armada por la innovación y la Inteligencia Artificial durante el presente curso 2019 – 2020, se decide retomar esta línea de trabajo. Surge entonces una colaboración entre el COVAM y el Centro Universitario de la Defensa para desarrollar algún tipo de herramienta de IA que facilite las tareas del centro de vigilancia. La adaptación y la incorporación de los requerimientos de COVAM y de la seguridad marítima protagonizan esta nueva aportación a la investigación. Es por ello por lo que se han presentado los dos temas fundamentales de este trabajo, y lo que se pretende estudiar es la viabilidad de la incorporación de los problemas derivados de la seguridad marítima que interesan a la Armada con las características y posibilidades de la detección de anomalías mediante algoritmos de IA.

3 DESARROLLO

En este capítulo, se van a desarrollar las líneas de investigación llevadas a cabo en este proyecto. En primer lugar, exponer brevemente de forma teórica las distintas capacidades y ventajas que pueden aportar los algoritmos de IA y la detección de anomalías respecto a las actividades llevadas a cabo en el ámbito de la seguridad marítima. A continuación, se exponen las herramientas necesarias para trabajar con el algoritmo y por último una descripción de las distintas pruebas a la que ha sido sometido.

3.1 Estudio de necesidades y capacidades en Seguridad Marítima

Para argumentar correctamente esta investigación, la investigación comenzó un proceso de investigación sobre las necesidades del Estado en materia de seguridad marítima, así como los actores que intervienen en ella y sus necesidades. De esta forma se busca estudiar las capacidades con las que ya se cuentan y analizar la posibilidad de mejora mediante la implementación de los algoritmos de HTM y de detección de anomalías mediante aprendizaje autónomo, objetivo de este trabajo.

3.1.1 Integración de los actores de la seguridad marítima.

Durante las primeras líneas del trabajo se indican el porqué de la importancia de la seguridad en la mar. También se definen los aspectos que más preocupan al Estado Español en este tema, así como una breve descripción de estos problemas. Por último, se describen los distintos organismos públicos que intervienen, y algunas de sus competencias. Con esto se puede afirmar que no existe un órgano central o entidad con plenas capacidades en todo lo que concierne a las actividades en la mar.

La creación del Mando de Vigilancia y Seguridad Marítima supone un importante hito, ya que centraliza los esfuerzos a nivel Fuerzas Armadas, sin embargo, fuera de ellas órganos como la Guardia Civil, la Policía Nacional o Salvamento Marítimo realizan esfuerzos en la misma materia y existe una notable carencia de integración entre organismos. En el artículo [24], se presenta el COVAM como ese elemento integrador entre entidades, por lo que un gran flujo de información debe de gestionarse desde este organismo, como queda reflejado en Figura 2-16.

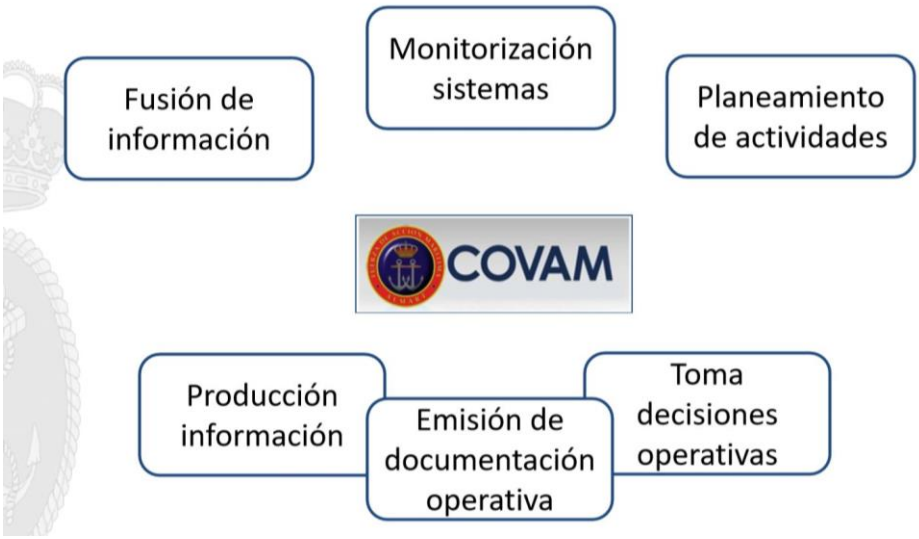


Figura 3-1 Resumen de áreas de responsabilidad del COVAM

Para todas estas áreas de trabajo de trabajo existe una aplicación de los algoritmos de IA, sin embargo, destacan dos:

- Toma de decisiones operativas
- Monitorización de sistemas

En este aspecto se destaca la cantidad de información recibida en el COVAM, que requiere de ser procesado por los operadores de este. Toda esta información caracteriza la situación marítima procedente de distintas fuentes, ya que además de la coordinación de las actividades y actores en materia de seguridad marítima, el COVAM es encargado de tener el control de todo buque que navegue en las aguas donde España tiene algún interés. Por lo tanto, como se ha desarrollado anteriormente las capacidades del ser humano a la hora de detectar problemas en grandes flujos de datos son muy limitadas. Así se presentan los algoritmos de IA como solución fundamental a este problema.

3.1.2 Recepción y procesado de información

No solo en la Armada, sino que hemos visto que tanto en Salvamento Marítimo, como en la Guardia Civil la vigilancia del espacio marítimo queda protagonizada por personal cualificado monitorizando la información recibida de numerosos sensores en sistemas diseñados para tal efecto, pero sin embargo la detección debe ser efectuada por el factor humano sin ningún sistema capaz de procesar toda esa información y ofrecer una detección de cualquier patrón similar a una anomalía o comportamiento sospechoso.

Por lo que el estudio de todos los actores que intervienen en la seguridad marítima ha permitido conocer y definir el problema no solo en el ámbito de la Armada, sino a nivel nacional y poder afirmar que actualmente no existe ninguna aplicación capaz de recibir toda la información que se posee sobre el tráfico y las actividades marítimas gracias a sistemas como el AIS y detectar anomalías.

3.1.3 Las capacidades de la IA frente a las necesidades de COVAM

Por lo tanto, un sistema de computación inteligente se plantea como una solución para mejorar la capacidad de detección de anomalías. Los algoritmos de IA se pretenden que incluyan las capacidades del cerebro humano a la hora de asimilar conceptos y aprender patrones de comportamiento. En la actualidad no poseen exactamente esas capacidades, sin embargo, muchos son las empresas que están volcando sus esfuerzos en evolucionar esta tecnología. Todo esto, sumado a la velocidad de trabajo de estas los hacen una solución muy adecuada al problema de procesado de información planteado.

Una vez asumidas las capacidades de los sistemas de IA frente a los problemas planteados, se efectuaron una serie de pruebas con el sistema inteligente de Numenta, NuPIC para corroborar las capacidades de estos sistemas en materia de seguridad marítima, para asistir al COVAM en sus actividades de vigilancia y contribución al conocimiento del entorno marítimo producto de la compilación, análisis, correlación y fusión de toda la información relacionada con el ámbito marítimo.

3.2 Herramientas de trabajo e instalación

Para conocer el desarrollo de este trabajo se necesita introducir las herramientas elegidas para ayudar en el proceso de evaluación. Así como las capacidades que se buscan de ellas y su aportación al proyecto.

3.2.1 Sistema operativo, Ubuntu.

Debido a las ventajas que nos ofrece a pesar de ser un sistema operativo poco común en nuestros ordenadores personales pero que ofrece muchas ventajas a nivel de trabajo con aplicaciones y desarrollo de código se decide trabajar sobre Ubuntu.

Como sistema operativo de trabajo se ha elegido Ubuntu. Debido a que las pruebas requeridas no suponen grandes esfuerzos de hardware se utiliza el ordenador personal del alumno donde se corre una máquina virtual.

El factor fundamental para trabajar en Ubuntu es que la aplicación ha sido desarrollada en él, lo que nos va a permitir el acceso a mayor cantidad de recursos y de información. Por lo que a primera escala utilizamos una máquina virtual para llevar a cabo las pruebas de menor entidad, utilizando la versión de Ubuntu 18.04.3 LTS.

3.2.2 Python como lenguaje de programación

Para entender un poco más el funcionamiento de NuPIC, surge como primer problema conocer el lenguaje en el que está programado y la forma en la que se implementa para poder comunicarse con el algoritmo. NuPIC se encuentra escrito en Python, un lenguaje de programación que permite una amplia gama de posibilidades y resulta bastante intuitivo y sencillo de utilizar.

La primera semana de trabajo se emplea para familiarizarse con este lenguaje, sus posibilidades y distintas librerías. Además de conocer en paralelo el código de NuPIC para poder adaptar lo que se va aprendiendo del lenguaje con el uso que se da de él en el algoritmo.

3.2.3 Nupic

Se presenta ahora la principal herramienta de este Trabajo de Fin de Grado, ya que vamos a analizar este algoritmo de Inteligencia Artificial y generar datos sobre las capacidades de esta herramienta. Como ya se ha presentado anteriormente, NuPIC es el motor de IA que se elige para ser analizado. El objetivo en gran manera que se pretende es definir las capacidades de este algoritmo y dar valores a determinados aspectos que se desconocen sobre él.

El primer paso necesario es la instalación de este en el sistema operativo. A continuación, se explica el proceso de instalación de la herramienta y los posibles inconvenientes que se encontraron durante el mismo.

3.2.3.1 Requisitos previos a la instalación.

Para instalar NuPIC se deben seguir los pasos que ofrece Numenta en su Wiki en GitHub. Para facilitar el trabajo Numenta ofrece además un vídeo en el que Matthew Taylor, *Open Source Community Flag-Bearer* de Numenta explica paso a paso el proceso de instalación de NuPIC desde cero en Ubuntu.

Antes de comenzar la instalación, es importante asegurar que se han instalado y actualizado todos los requisitos se detallan en la referencia [25]:

- Python 2.7 & development headers
- pip version 8.1.2 o superiores
- wheel versión 0.29.0 o superiores
- numpy
- Compilador de C++11

Se comienza con la instalación de los requerimientos que se detallan, algunos de los cuales ya se encuentran instalados por lo que solo se comprueba la versión instalada.

Se comprueba primero de todo, la versión de Python. Para ello, se introduce en el terminal los siguientes comandos para ver si se encuentra instalado, y cuál es la versión que se posee (ver Captura 3-1):

```
python --version
```

Captura 3-1 Versión Python

En caso negativo se ejecutan los siguientes comandos en el terminal de Ubuntu:

```
hg clone https://hg.python.org/cpython
hg update 2.7
sudo apt-get build-dep python2.7
sudo apt-get install python-dev
```

Para instalar pip, el segundo de los requerimientos de NuPIC se siguen las instrucciones de la web de pip para así instalar la última versión.

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.p
curl https://bootstrap.pypa.io/get-pip.py | sudo python
```

En resumen, se han instalado todos los requerimientos previos de NuPIC debido a que los restantes se encuentran incluidos o en la instalación de Python, como es el caso de Wheel. Y esta versión de Ubuntu ya incluye un compilador de C++11 por lo que se prosigue con los siguientes pasos de la instalación.

3.2.3.2 Instalación de NuPIC

Para instalar Nupic, simplemente se introduce en el terminal:

```
pip install nupic  --user
```

A diferencia de versiones anteriores utilizadas en otros TFG, los desarrolladores de NuPIC han conseguido optimizar su instalación, por ello solo se necesita este comando para instalar el algoritmo en el sistema. Indicándose tras unos minutos de espera que su instalación se finaliza con éxito.

A continuación, se clona el directorio de GitHub de NuPIC para poner todos los archivos del directorio en la máquina. Tras ello se puede decir que se ha instalado NuPIC en el sistema.

3.2.3.3 Test NuPIC

Para verificar una correcta instalación se ejecutan los test de NuPIC y se verifican los resultados obtenidos. Necesitamos instalar py.test para correr las pruebas contenidas en el directorio que se ha copiado previamente de GitHub, se ejecuta lo siguiente:

```
sudo apt-get install python-pytest
```

Por último, se ejecuta el test de NuPIC:

```
python ./scripts/run_nupic_test.py -u
```

Se obtienen algunos problemas menores derivados de las numerosas pruebas que ejecuta este test, sin embargo, nada que pueda afectar al correcto funcionamiento del algoritmo, por lo que después de este resultado se puede considerar que la instalación de NuPIC ha sido un éxito.

3.2.4 Nupic geospatial

El motor fundamental que se ejecuta en el análisis es el algoritmo de NuPIC desarrollado para la detección de anomalías en rutas. Su instalación no se integra en el proceso de instalación por lo que se debe incorporar aparte. Al igual que con NuPIC los desarrolladores añaden un videotutorial para ayudar a los usuarios a su instalación, además de la descripción del proceso de instalación por lo que se resume en seguir las pautas que se indican.

Primero de todo se copian los archivos volcados en el entorno de GitHub relacionados con NuPIC Geospatial, para ello se ejecuta:

```
git clone https://github.com/numenta/nupic.geospatial.git
```

Seguidamente, se instala NuPIC geospatial ejecutando el siguiente comando:

```
pip install -r requirements.txt --user
```

Al ejecutarse, se obtiene el siguiente error:

```
ImportError: No module named Cython.Distutils
```

Quedando el error solventado tras la instalación de cython.

```
pip install cython
```

Se vuelve a ejecutar el comando, esta vez llevándose a cabo la instalación con éxito (ver Captura 3-2).

```
Successfully built pycapnp
Installing collected packages: pycapnp, nupic.bindings, nupic
  Found existing installation: pycapnp 0.6.3
    Uninstalling pycapnp-0.6.3:
      Successfully uninstalled pycapnp-0.6.3
  Found existing installation: nupic.bindings 1.0.6
    Uninstalling nupic.bindings-1.0.6:
      Successfully uninstalled nupic.bindings-1.0.6
  Found existing installation: nupic 1.0.5
    Uninstalling nupic-1.0.5:
      Successfully uninstalled nupic-1.0.5
Successfully installed nupic-1.0.1 nupic.bindings-1.0.0 pycapnp-0.5.12
```

Captura 3-2 Instalación con éxito NuPIC Geospatial

Con todo ello se poseen las herramientas necesarias para procesar rutas y estudiar las posibilidades del programa en cuanto al tráfico marítimo. Ahora se necesita desarrollar un método para generar rutas que se adapten a los requerimientos para el estudio de las capacidades de NuPIC.

3.3 Generar rutas sintéticas

Para conocer NuPIC en detalle y desarrollar sus capacidades, se ve necesario crear una serie de rutas sintéticas adaptadas a las características que se quieren estudiar y donde se tengan sus parámetros muy controlados. Este factor no permite desarrollarlo la incorporación de datos AIS, ya que proviene de contactos reales. Cierto es que la fuente de datos de NuPIC se pretende que sea el sistema AIS, sin embargo, para poder controlar los parámetros que se introducen y ver así la respuesta de la máquina ante las distintas rutas generadas se opta por generarlas de forma sintética.

Como ya se comenta en los primeros apartados, NuPIC es una aplicación de aprendizaje autónomo no supervisado por lo que se debe preparar el entorno para ver que decisiones toma el algoritmo y establecer qué considerar como anomalía y en qué condiciones. En cierto modo, lo único que entiende

NuPIC Geospatial a la hora de procesar información son archivos .csv[3] por lo que el primer reto que se plantea es encontrar una herramienta que permita al alumno generar rutas con posiciones de latitud y longitud y velocidad.

3.3.1 Microsoft Office Excel

La herramienta elegida para generar archivos .csv es Microsoft Excel, a través de una hoja de cálculo. Debido fundamentalmente a dos factores: la familiarización del alumno con el programa obtenida a lo largo de su formación y las capacidades de cálculo y procesado de datos del programa que han permitido no solo pre-procesarlos para introducirlos en NuPIC, sino también analizar los resultados obtenidos.

3.3.2 Open CPN

Las capacidades de este software muy utilizado en la Armada para generar rutas y estudiarlas en cartografía náutica electrónica se han utilizado para dar realismo a las citadas rutas sintéticas. Este software permite la posibilidad de generar rutas en entornos marítimos teniendo en cuenta los diversos factores que afectan a la navegación y aprovecharlas para estudiarlas con NuPIC (ver Figura 3-2).

OpenCPN permite generar rutas sobre la cartografía digital y presentar parámetros de cambios de rumbo, velocidad, tiempo de tramo etc. También presenta los DST (Dispositivos de Separación de tráfico)[4], los distintos puertos, zonas pesqueras e incluso fondeaderos. Resulta entonces una herramienta muy útil a la hora de generar escenarios sintéticos para entrenar el motor de IA con datos y referencias del medio marítimo. Además, cuenta con la capacidad de exportar las distintas rutas a archivos .gpx para después poder procesarlo con NuPIC.

[3] Archivos .csv proviene del inglés (“coma separated values”) es decir, valores divididos en columnas separados por comas en un archivo.

[4] Los DST son zonas en las que por motivos de concentración del tráfico marítimo se establecen una serie de carriles de tránsito, zonas de separación y normas especiales que mejoran la seguridad de la navegación.

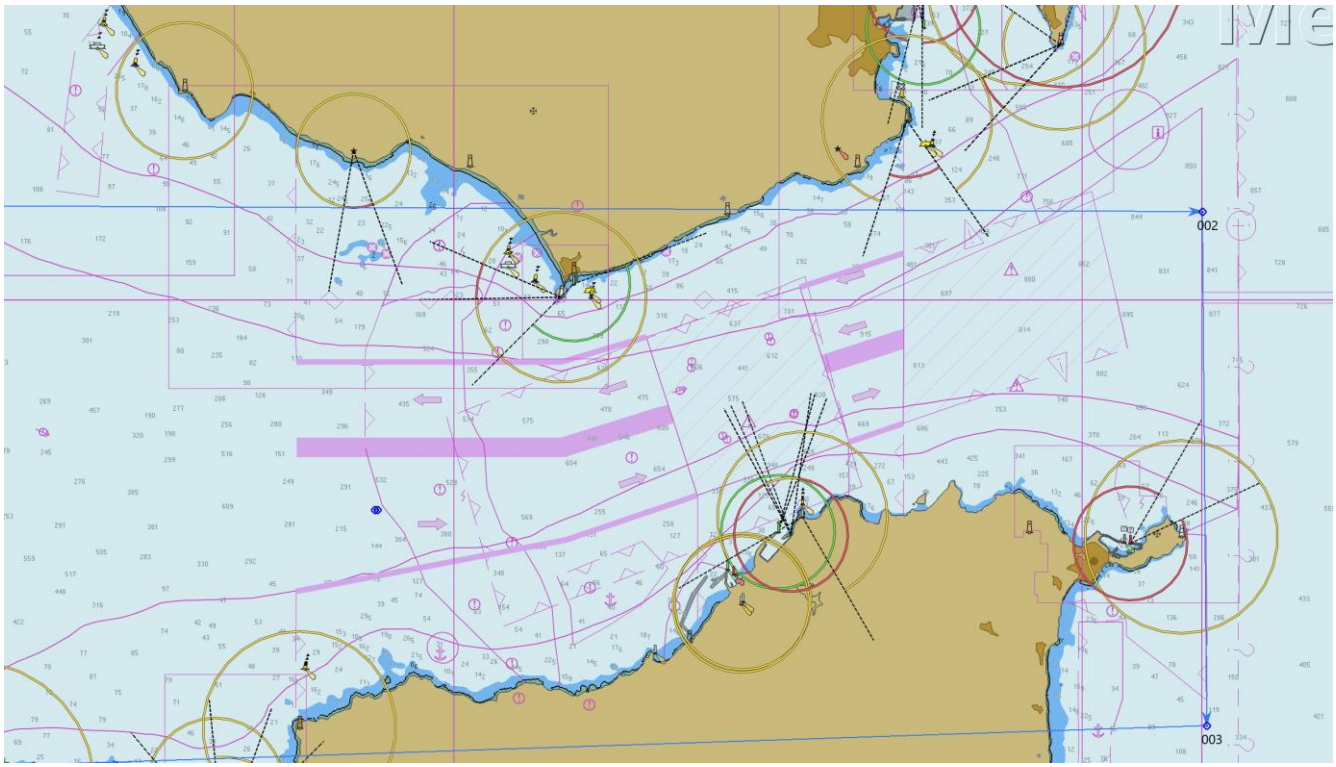


Figura 3-2 DST del Estrecho de Gibraltar en OpenCPN

3.3.3 Conversión a .csv

Una vez exportadas las rutas, se decidió recurrir a un pequeño programa de software libre que permite transformar archivos .gpx en archivos .csv de tal forma que estos últimos podamos leerlos con Excel y realizar los cambios pertinentes para ejecutar el archivo con NuPIC.

La realidad es que surge un problema de compatibilidad entre los datos generados por OpenCPN y los que lee el algoritmo. Por ello se genera una hoja de Excel que transforma los datos obtenidos del programa de cartografía y efectúa cambios como la transformación a tiempo Unix[5] del grupo fecha hora o el orden de las columnas. Proceso tras el cual se exporta un archivo Excel perfectamente compatible con NuPIC.

3.4 Pruebas iniciales con NuPIC

En este título se describen brevemente las primeras pruebas llevadas a cabo con el algoritmo para conseguir una familiarización con el modo de funcionamiento del código, los parámetros que se necesitan declarar al ejecutarlo y los formatos de salida y entrada. El objetivo de estas pruebas es meramente pedagógico para entender cómo ejecutar el algoritmo desde el terminal de Ubuntu y en líneas generales como obtener la información necesaria.

[5] Medida temporal correspondiente al número de segundos transcurridos desde la medianoche UTC del 1 de enero de 1970, sin contar segundos intercalares.

3.4.1 Ejecutar NuPIC Geospatial

Para entender el funcionamiento de NuPIC Geospatial, se debe conocer los programas que lo constituyen y que utilidad tienen cada uno de ellos en el proceso. Los programas fundamentales que intervienen en el algoritmo son:

run.py: Este programa es el principal y el que usaremos para arrancar la máquina virtual. Enlaza todos los demás segmentos de código y sirve de alguna forma como herramienta de control entre el hombre y la máquina ya que es aquí donde se declaran los parámetros que afectaran al desarrollo del aprendizaje del algoritmo, estos son:

Opciones	Comando
Escala	-s “escala”, --scale= “escala elegida”
Verbose	-v, --verbose
Cambiar directorio salida	-o “directorio salida”
Secuencia manual	-m, --manual-sequence
Codificador temporal	-t, --time-encoder

Tabla 1 Opciones de ejecución de NuPIC [26]

Estos parámetros que se relacionan en Tabla 1Tabla 1 Opciones de ejecución de NuPIC, modifican el funcionamiento de NuPIC o le indican al programa como debe de ejecutarse. Todos ellos deben de declararse a la hora de ejecutar el programa run.py a continuación se explican la utilidad de cada opción.

Escala: Adapta la codificación geospatial en función de las necesidades de los datos que se van a procesar, sin embargo, se desconoce las capacidades exactas de este parámetro que será objeto de estudio en el siguiente apartado.

Verbose: Activa la opción de que se imprima por pantalla todos los procesos que el algoritmo va ejecutando muy útil para en caso de fallo conocer en que parte está el problema.

Cambiar directorio salida: Tal como su nombre indica, permite seleccionar el directorio de salida para los archivos generados por NuPIC donde quedan registrados los resultados.

Secuencia manual: Permite dividir la secuencia de posiciones de forma manual, en lugar de la asignación automática mediante secuencias temporales.

Codificador temporal: Añade dentro de los parámetros del modelo el codificador temporal que incluye el momento del día.

Después run.py llama a *preprocess_data* lo guarda en el archivo *preprocessed_data.csv*, se lo pasa a la función r*unGeospatialAnomaly* y vuelca los resultados en *data.js* con el objetivo de permitir visualizarlo posteriormente.

preprocess_data.py: Es la función que lee los datos de entrada del archivo .csv y los procesa para evitar que haya errores en el conjunto eliminando posiciones repetidas o datos incoherentes como valores de velocidad extraños, posiciones erróneas, etc.

runGeospatialAnomaly.py: Es la función que configura el modelo, lo ejecuta y obtiene los índices de anomalía.

model_params.py: Es el archivo que contiene los parámetros de configuración del modelo.

A la hora de ejecutar el programa debemos de llamar al programa *run.py*, añadir el directorio del archivo .csv de entrada y seguidamente incluir las opciones de ejecución que se precisen incluidas en Tabla 1. Como podemos ver en Captura 3-3 una imagen de la ejecución de un archivo de ejemplo cualquiera, en este caso: *valor4.*

```
alumno@tfg-mar:~/nupic.geospatial$ python run.py /home/alumno/nupic.ge
ospatial/pruebas/valor4.csv -m -v -s10
```

Captura 3-3 Ejemplo ejecución NuPIC

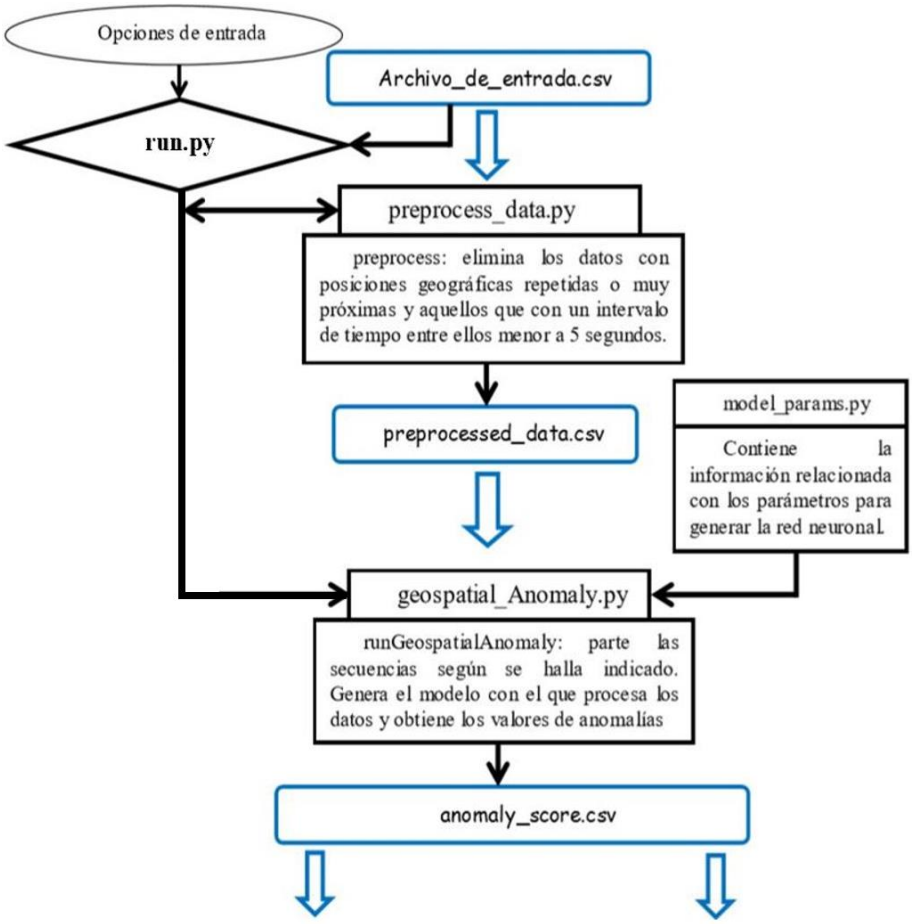


Figura 3-3 Diagrama de ejecución de NuPIC. Modificado del original [12]

Una vez finalizado el proceso completo de aprendizaje, NuPIC genera dos archivos, en el primero de ellos (*anomaly_score)* presenta los datos en un archivo de salida .csv donde se relacionan las distintas posiciones introducidas con el valor de anomalía que le corresponde. Este archivo será el de más utilidad para el trabajo ya que el otro (*preprocessed_data)* solo es útil para emplearlo con el servidor de NuPIC y visualizar la ruta en un mapa con las anomalías según colores. Todos los programas descritos se encuentran en los distintos directorios generados tras la instalación del algoritmo. A modo de resumen se presenta en Figura 3-3 un diagrama de flujo con el proceso de ejecución de NuPIC con los distintos programas que intervienen en su funcionamiento obtenido de [18] y modificándolo ya que en este trabajo se van a procesar los datos tal cual se obtienen del algoritmo sin realizar ningún post-procesado que pueda alterar los datos erróneamente.

Descrito el proceso de ejecución y funcionamiento de NuPIC Geospatial, se describirán las distintas pruebas y descubrimientos llevados a cabo con el algoritmo, para intentar conocer un poco más en profundidad el interior de esa "caja negra" a la que se refieren los estudiosos de la temática que conforman los algoritmos de IA. Ya que en muchas ocasiones se obtienen resultados, pero se desconoce su origen.

3.4.2 Capacidad de aprendizaje

Lo primero que se pretende definir del algoritmo es la capacidad que tiene para aprender una ruta determinada. Para ello, se entrena con una serie de rutas de más sencilla a más compleja para determinar cuántas repeticiones de la misma ruta necesita el algoritmo para dejar de considerarla anómala.

3.4.3 Rutas sintéticas sencillas

Para comenzar a analizar las capacidades de NuPIC se han generado una serie de rutas sintéticas sencillas con las que estudiar las capacidades del algoritmo, así como familiarizarse con su uso. En estas rutas no se tienen en cuenta las posibles características del tráfico marítimo, simplemente se introducen comportamientos básicos y se analiza la reacción de la red neuronal.

El principal problema de este trabajo es que resulta muy complejo conocer las operaciones en las que el algoritmo se basa para aprender y dar una predicción del siguiente dato. El programa tiene unos criterios con los que considera un dato anómalo con otro que no lo es y el operador humano desconoce sus razones. Para conocer cómo reacciona a distintos entornos se llevan a cabo estas pruebas.

3.4.3.1 Ruta recta

En el primer experimento se probó una ruta recta sin variar la escala. Esta ruta mantenía la latitud constante y la longitud se modificó en un incremento constante para crear una línea recta.

3.4.3.2 Cambio de rumbo

Otra prueba interesante es comprobar cómo se comporta el algoritmo ante un cambio de rumbo en la ruta bastante acentuado. En este caso a la ruta recta anterior se le incorpora un cambio de rumbo para estudiar la reacción del algoritmo ante este.

3.4.4 Rutas sintéticas marítimas

Después de realizar las pruebas sencillas, con rutas rectas o cambios sencillos de posición o velocidad se genera una ruta algo más aleatoria y próxima a lo que podría ser la realidad de

3.5 Evaluación de la relación posición-anomalía

En este apartado se presenta un análisis pormenorizado de la correspondencia entre un desvío de una ruta una determinada distancia con una determinada escala y el valor de la anomalía obtenido. Para ello se genera una ruta recta sencilla con la separación entre posiciones adecuada para que NuPIC no las considere como realiza una conversión de variaciones de latitud y longitud a distancia en metros y se comprueba el valor de anomalía obtenido.

3.5.1 Objetivo

El objetivo de esta prueba es, conocer partiendo de la escala mínima, cuanto se tiene que desviar en una dirección una misma ruta para que el algoritmo la considere anómala.

Una vez realizadas las sucesivas pruebas básicas, se ha detectado que se desconoce qué considera el algoritmo como una posición anómala en función de la separación con respecto a la posición original. Por ello se ve necesario darle un valor concreto a la distancia que debe de separarse de su ruta habitual un determinado contacto para que (teniendo en cuenta la escala elegida) NuPIC lo considere como anómalo.

3.5.2 Generar la ruta para el análisis

La ruta generada para estas pruebas consiste en una línea recta formada por 100 posiciones, esto conforma la ruta en sí, la cual se repite una serie finita de veces. Todas las repeticiones tienen como objetivo que el algoritmo aprenda la ruta hasta que deje de considerarla anómala. En la última de las 10 repeticiones se introducirá un desplazamiento al este de todos los puntos que generan la ruta. A continuación, en Tabla 2 se relacionan los parámetros que definen la ruta. De estos parámetros utilizaremos dos de ellos para variarlos la escala y la longitud de las rutas.

Parámetros	**Datos**
Latitud	Separación 4 m entre posiciones
Longitud	Variable
Velocidad	10 nudos
Tiempo	6 segundos
Nº Rutas	10
Escala	Variable

Tabla 2 Parámetros de ruta para analizar variaciones

3.5.3 Desarrollo de las pruebas

Según la teoría si se asigna una escala muy pequeña y las posiciones se salen de ellas, estas posiciones siempre serán anómalas. O por lo menos, NuPIC no será capaz de generar un patrón que coincida con la ruta que está recibiendo. Por lo tanto, comenzaremos gradualmente de menos a más para ver cómo afectan pequeñas variaciones de posición a una misma ruta y manteniendo la escala de 1m.

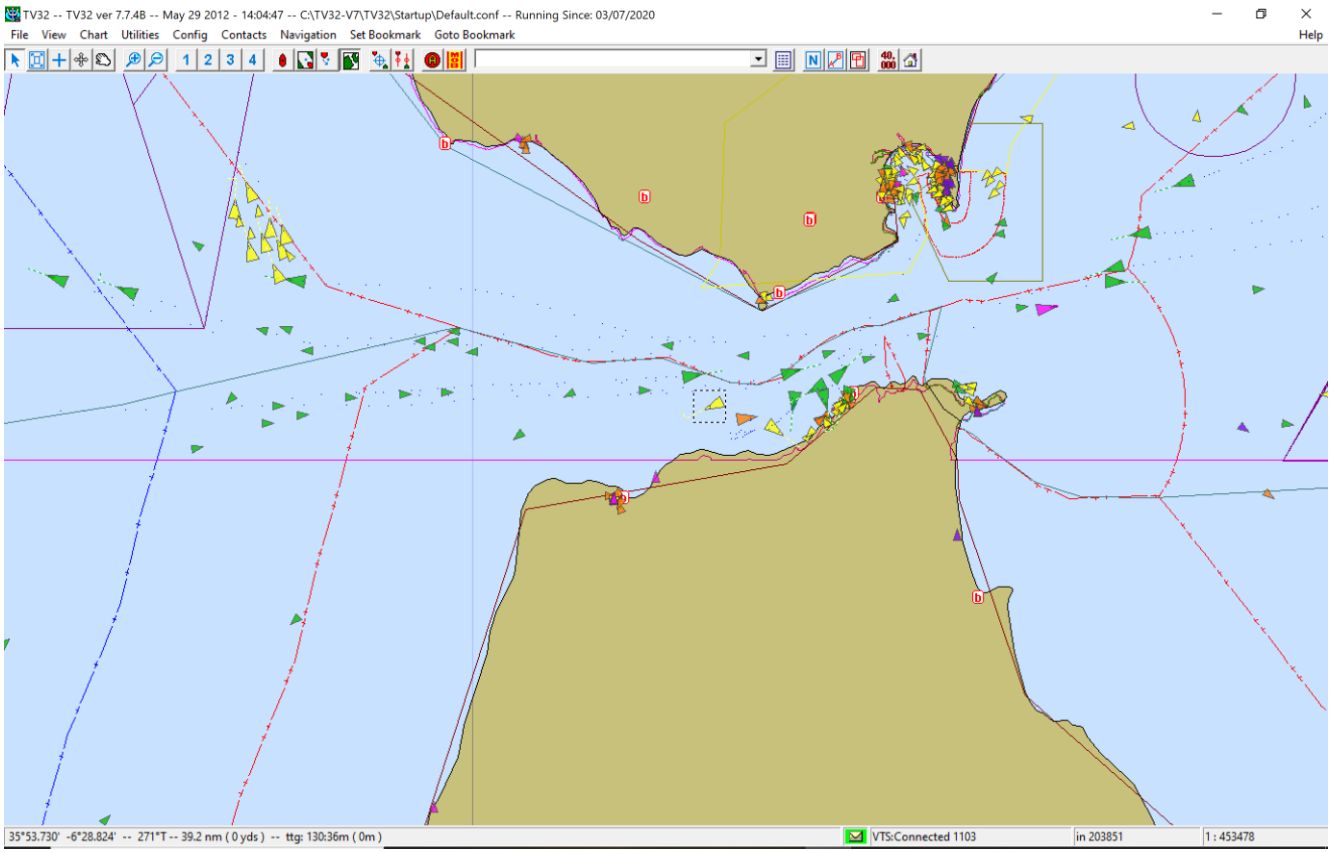
Se llevan a cabo tres pruebas cada una con una escala, 1m, 2m y 3m. Dentro de cada prueba se varia la última ruta o repetición (ruta nº10) y se analiza los cálculos de anomalías cada vez que desplazamos todas las posiciones de la ruta 1 metro al este en cada prueba. Se desplazan tantos metros como sean necesarios para evaluar y determinar los márgenes de cada escala y dependiendo del desplazamiento, el valor de anomalía dado.

3.6 Pruebas con datos reales

A demás de las pruebas realizadas con escenarios sintéticos se pretende estudiar la capacidad de la herramienta de reaccionar a un flujo de información procedente de mensajes AIS. Destacar que no se llevará a cabo un análisis en tiempo real de los datos, lo que se podrá desarrollar en líneas futuras, sino que se ha usado una pequeña base de datos para analizar la información y procesarla con el algoritmo.

3.6.1 TV32. Obtención de datos reales

Se utiliza el programa TV32 facilitado por COVAM para monitorizar el tráfico marítimo y obtener de él datos de los contactos que presenta. El programa posee una conexión a través de internet que permite recibir los mensajes AIS en tiempo real.



Captura 3-4 Captura de pantalla del TV32, tráfico en el estrecho

Por su parte el TV32 posee la capacidad de almacenar la información recibida de los mensajes AIS en una pequeña base de datos diaria en archivos .log Captura 3-5 y almacenarlos en una carpeta. Estos archivos se presentan en un formato que tras una serie de búsquedas se descubre que pertenecen a mensajes en formato AIS.

```
20200217.log
1  !AIVDM,1,1,,B,B33mDsP00830DC5b8NWQ3weUoP06,0*3B,1581927574,1581927576
2  !BSVDM,1,1,,A,1aNSO6h00TOGiT8HH6ob57vp050L,0*7C,1581927574,1581927576
3  !ABVDM,2,1,2,A,53FJR<02;Gp4D4UF221<E9@u<61HDTq@Tt<Pv20l1p?665ba0<PH0klUCPH8,0*3E,1581927574
4  !ABVDM,2,2,2,A,88888888880,2*2D,1581927574,1581927576
5  !ABVDM,2,1,5,B,53Eh2RT00000D7O?S@0PE8lTpTv1V0@thu8E<00N0`=33t0Ht011CR@@0000,0*15,1581927574
6  !ABVDM,2,2,5,B,00000000000,2*29,1581927574,1581927576
7  !ABVDM,1,1,0,B,13EdlwWOhLwDI=pHWl=K<arn20Rg,0*5E,1581927574,1581927576
8  !ABVDM,1,1,,A,H3`je9lUFC@@?M;@5jmqq01p?150,0*7D,1581927575,1581927576
9  !AIVDM,1,1,9,B,H3P;wuQ@P4hTl40000000000000,2*2D,1581927575,1581927576
10 !ABVDM,1,1,,A,13Edn:gP1IwdKt:Dwmg33gvl0L0Q,0*5A,1581927575,1581927576
11 !ABVDM,2,1,3,B,53e`4J82DFlgTP7C7?@<l60<Ln18Tv0M84p@D017?RhHH4aQ0S3@D53@0000,0*1B,1581927575
12 !ABVDM,2,2,3,B,00000000000,2*2F,1581927575,1581927576
13 !ABVDM,2,1,1,B,53Ebcd0252K0D5<n221H:18U<84r22222222220l10F37757N:4hp11H51CQ,0*16,1581927575
14 !ABVDM,2,2,1,B,DRAQH888880,2*5B,1581927575,1581927576
15 !ABVDM,2,1,1,B,54SD6b42=A>qK8MO:20ILr105Dh6222222222216@P:8:4b70<QDp8340H88,0*1B,1581927575
16 !ABVDM,2,2,1,B,88888888880,2*2D,1581927575,1581927576
17 !ABVDM,1,1,,B,13Ei9eWP1MwwbK`Fj<v@=gvn2<0m,0*59,1581927575,1581927576
18 !ABVDM,1,1,5,A,13cjLc002rPF64lFC9Ikbjvp0l07,0*4F,1581927575,1581927576
19 !AIVDM,1,1,,B,4028j:Qv@``7KOvPkhFl:OQ000S:,0*5D,1581927575,1581927576
20 !ABVDM,2,1,3,B,54`UtW@2?iw<?H8?:20L58T84h@V1=0U8UB2221@I0cEE4c<0OCBC1SlQ820,0*65,1581927575
21 !ABVDM,2,2,3,B,EQC`8888880,2*18,1581927575,1581927576
22 !ABVDM,1,1,,A,13F94s0P@wOPMW@E@Vut39Bp0<1a,0*33,1581927575,1581927576
23 !AIVDM,1,1,3,A,13`nQf0P@JO2P2t@S8JsAJ<p0H@C,0*01,1581927575,1581927576
24 !ABVDM,1,1,1,A,3DSaF20011OC<eHGsllf0s6n0P00,0*19,1581927575,1581927576
25 !ABVDM,2,1,5,A,58HuaT82Dsk;UKG7CF0H4pH58F2222222222221@BhN8D4a=0GhC1i@jDPDp,0*28,1581927575
```

Captura 3-5 Archivo .log generado por el TV32

Para poder trabajar con estos datos, que como vemos son incompresibles, se hace necesario la decodificación de estos mensajes. Para ello, se genera un sencillo programa en Python (puede verse en el Anexo III de este documento) que utiliza una librería para leer los datos AIS de este mensaje y decodificarlos en un archivo .csv y así poder interpretarlos. Para ello se debe renombrar los archivos generados por TV32 como archivos .ais y guardarlos en el directorio donde se almacena el programa. Tras ello lo ejecutamos introduciendo los siguientes comandos en el terminal:

```
cd conversor\ archivos\ TV32
python aismsg.py
```

Ejecutamos así el programa y nos pide que introduzcamos el nombre del archivo que queremos decodificar en un .csv tal como se muestra en Captura 3-6.

```
Introduzca nombre del fichero AIS:
"20200204.ais"
alumno@tfg-mar:~/conversor archivos TV32$
```

Captura 3-6 Petición de fichero AIS

Una vez ejecutado, se genera en el mismo directorio un archivo .csv con los datos de los buques registrados en el mensaje AIS habiendo decodificado un mensaje del estilo a lo reflejado en Captura 3-5 y generando un .csv como el que se observa en Anexo VI: decodificación de un mensaje ais. Todo esto según el procedimiento anteriormente explicado se ejecuta con NuPIC y se analizan los resultados obtenidos.

3.6.2 Pruebas con el conjunto de datos y problemas derivados

Como se verá en el apartado de resultados, al ejecutar el algoritmo con los datos obtenidos por NuPIC surge un gran problema y es que estos datos no siguen un patrón lógico. Como ya se conoce, la herramienta trabaja procesando datos en csv los cuales se encuentran agrupados en rutas, derivando entonces en el problema y es que los datos almacenados en memoria por el TV32 no se encuentran agrupados por rutas ya que cada contacto emite sus datos AIS en una franja temporal por lo que se van registrando salteados y sin coherencia.

Por ello, se desarrolla un nuevo programa que agrupa los datos recibidos por MMSI, uno de los parámetros elegidos del mensaje AIS y que identifican a los distintos buques. De esta forma, se lleva a cabo un pre-procesado de la información antes de entregársela al algoritmo.

Finalizamos el desarrollo de este trabajo con el procesado de datos procedente de mensajes reales AIS procedentes de una aplicación utilizada por COVAM y por las distintas unidades de la Armada lo que puede resultar interesante a la hora de implementar estos algoritmos en el citado organismo. Por último, se da paso a los resultados obtenidos en las pruebas anteriormente descritas, así como una pequeña evaluación de estas.

4 EVALUACIÓN Y RESULTADOS

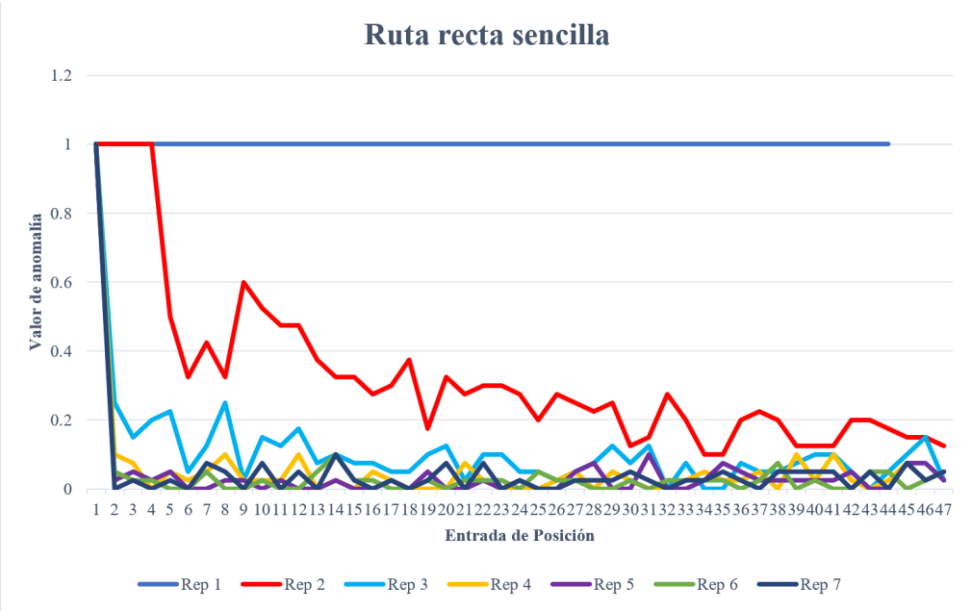
4.1 Análisis de rutas sencillas

Con estas pruebas se han podido comprobar algunas de las capacidades de NuPIC, entre ellas la velocidad de aprendizaje en función de la ruta procesada, la respuesta a una ruta recta o distintos cambios de dirección en ella y por último a unas posiciones más próximas a lo que podría generar un buque y sus trasmisiones AIS.

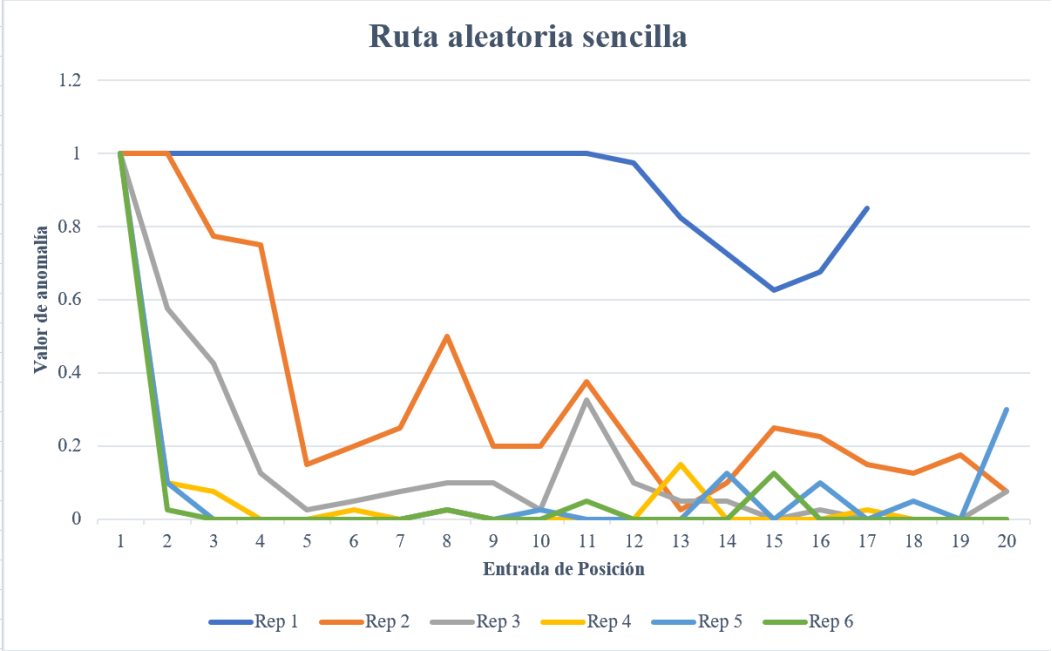
4.1.1 Capacidad de aprendizaje

Se ha visto la alta capacidad de aprendizaje que posee el algoritmo ya que en una ruta sencilla solo son necesarias 2 repeticiones para que el algoritmo nos de un valor de anomalía muy próximo a 0. Se puede ver en la Captura 4-1 que las anomalías se estabilizan con la tercera repetición entre 0.2 y 0. Todo esto para el caso de un ruta recta sin ninguna variación.

A continuación, evaluamos el caso de una ruta recta con posiciones pseudo aleatorias, es decir no siguen una línea recta, sino que son posiciones que van de un lado para el otro.



Captura 4-1 Capacidad de aprendizaje NuPIC ruta recta sencilla



Captura 4-2 Ruta aleatoria sencilla

En este caso hay un pico descendente hacia abajo en la primera ruta, cuando debería de ser todo unos y es que las posiciones aleatorias en esa zona se aproximan entre ellas. Sucede entonces que los SDRs de estas estén tan próximos que le resulte sencillo realizar una predicción, sin embargo, no es lo usual de ahí que se destaque la importancia de la escala en estas pruebas (ver Captura 4-2).

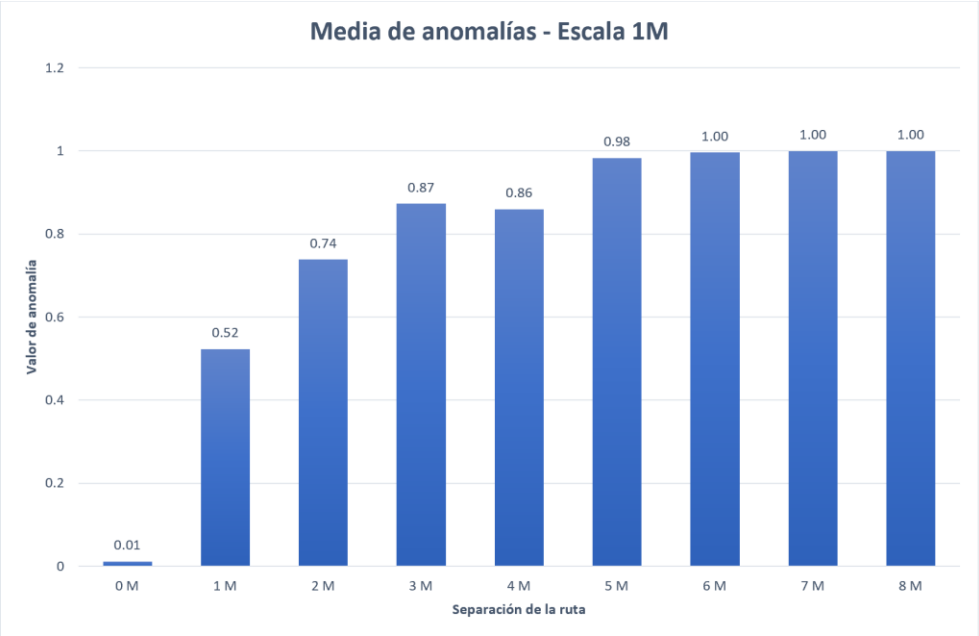
En este caso también podemos destacar la capacidad de aprendizaje del algoritmo que solo ha necesitado leer dos veces la ruta para generar una respuesta correcta y aprender el comportamiento patrón.

4.2 Evaluación de la relación desvío – anomalía

Al evaluar las capacidades de detección en función de la escala y la desviación en metros de la rutas que procesamos con NuPIC se han obtenido resultados interesantes hasta el punto de poder definir en una determinada escala a cuanto equivale en separación en metros un determinado valor de anomalía.

4.2.1 Escala 1 metro

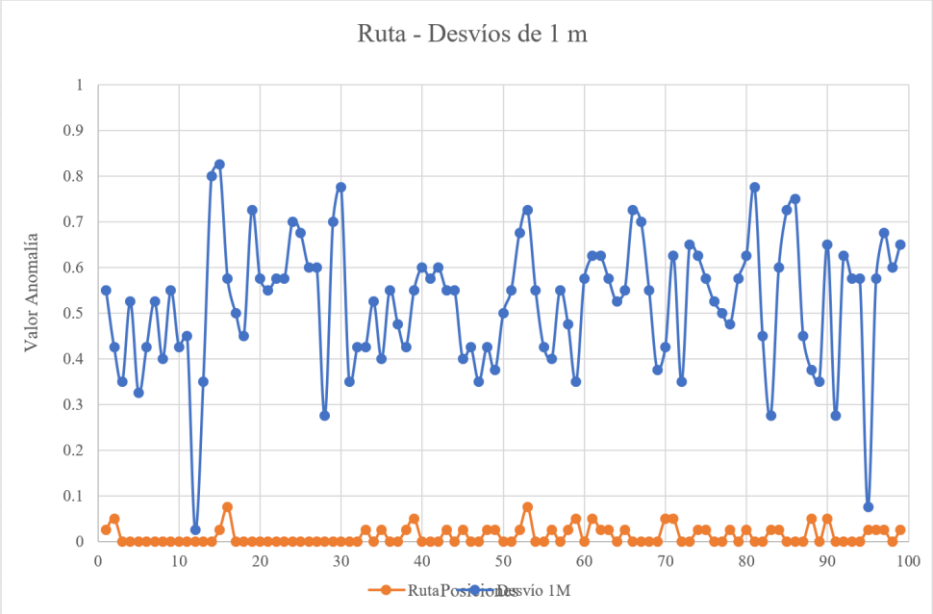
Eligiendo una escala de 1 metro y desviando la ruta nº 10, se han analizado los valores de anomalía obtenidos por NuPIC y se han obtenido los siguientes datos. En la primera gráfica se compara el valor de anomalía obtenido en función de la distancia a la que hemos separado la ruta de la original. El valor de 0M se corresponde con la ruta original y cómo podemos ver si separamos 1 metro la ruta respecto a la original obtenemos un valor medio de anomalía que va aumentando exponencialmente hasta estabilizarse entre el 0.9 y el 1 a partir de los 5 metros de separación.



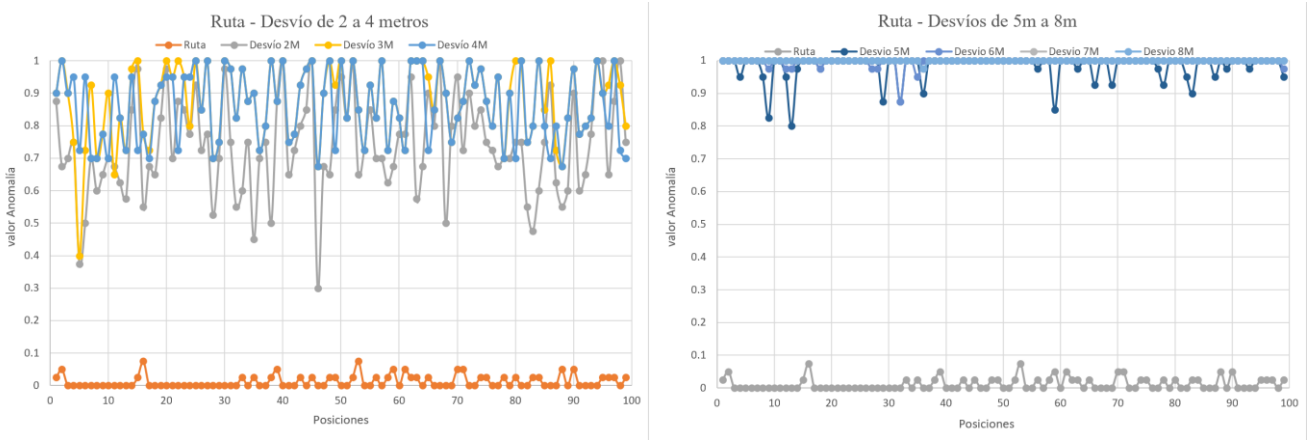
Captura 4-3 Media de anomalías - Escala 1M

Si por otro lado vemos los datos reales sin agruparlos en una media, se observa mejor como a partir de los 5 metros el programa entiende todas las posiciones como anómalas casi en su totalidad son un 1 lo que significa que la predicción no ha tenido a penas bits de coincidencia en el SDR.

Por otra parte, entre 1 metro y los 5 metros existe una zona de incertidumbre, aunque se aprecia claramente como entre el 0 y el 0.4 no existen a penas valores por lo que sería un buen margen para establecer un resultado como no anómalo. Estudiaremos el caso del resto de escalas para poder dar una afirmación.



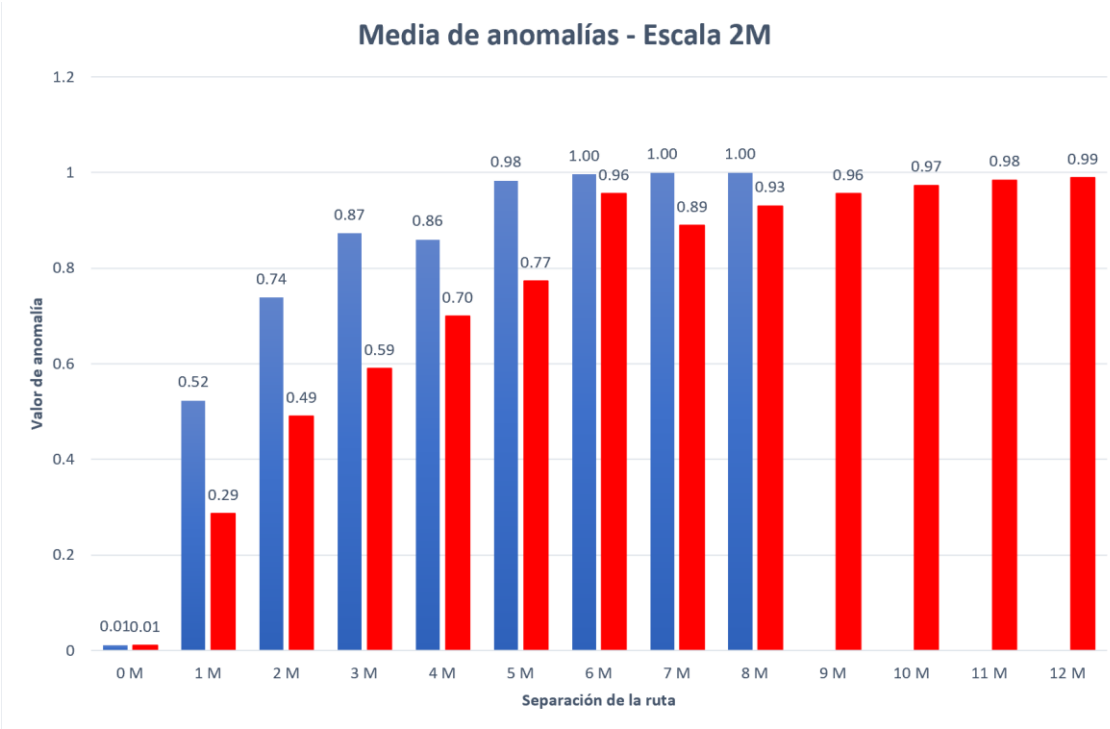
Captura 4-4 Anomalías de los desvíos frente a la ruta original con 1 m de escala



Captura 4-5 Anomalías rutas escala 1m

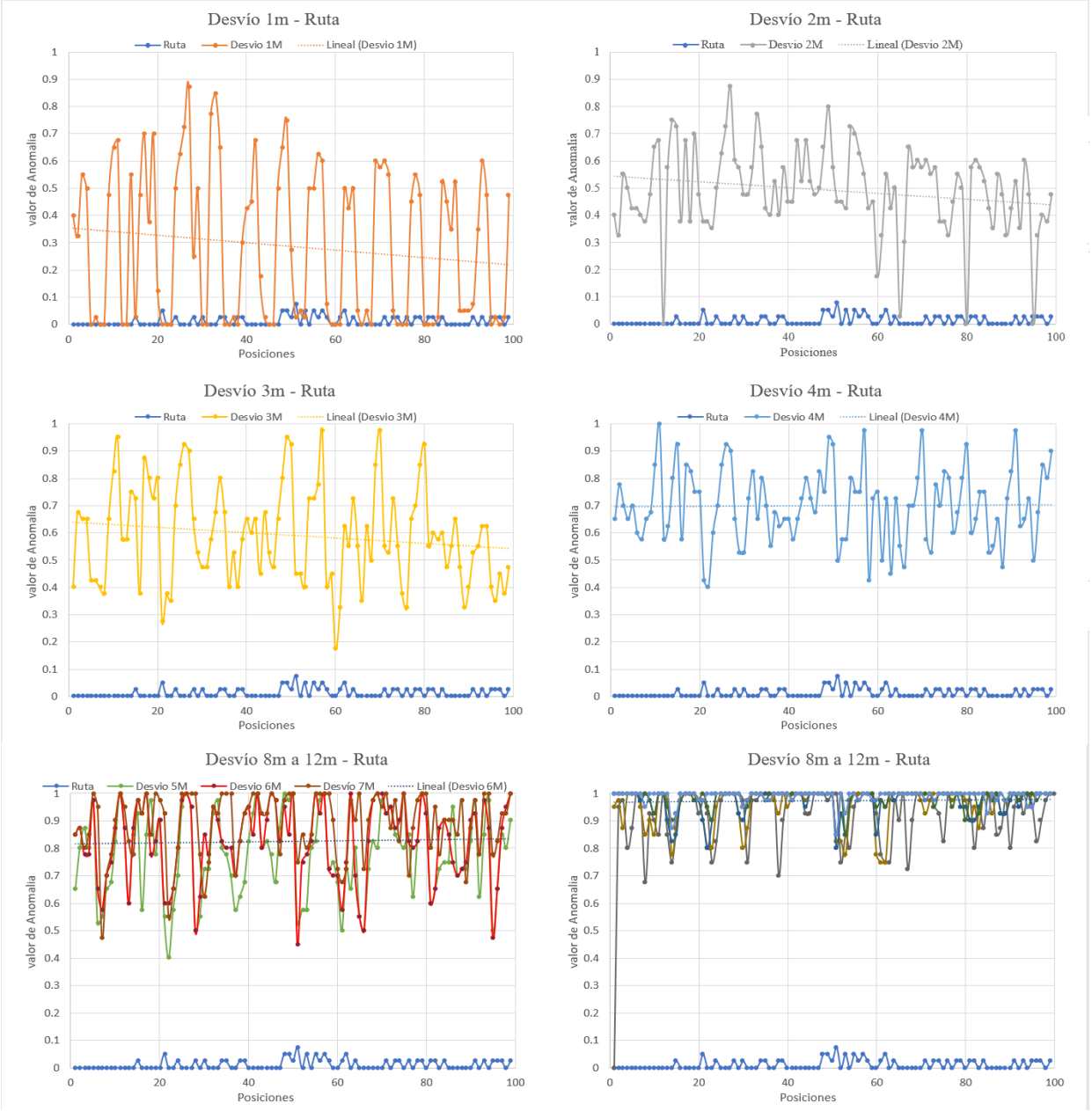
4.2.2 Escala 2 metros

En el siguiente intento, se procesa el mismo patrón de rutas, pero esta vez con una escala mayor. Sin embargo, se aprecia la necesidad de aumentar la separación con respecto a la ruta original en más distancia ya que no aparece un valor de anomalía significativo hasta los 6 metros de separación, entendiéndose por significativo entre 0.8 y 1.



Captura 4-7 Media anomalías escala 2 metros

Como se aprecia en la gráfica en una escala de 2 metros, se considera anómalo una separación respecto a la ruta original de 6 metros. Por lo tanto, su genera un umbral diferente al que teníamos con la escala anterior (ver Captura 4-7). En este caso la separación de 1 metro respecto a la ruta original se mezcla en la zona de incertidumbre como se ve en Captura 4-8 donde la gráfica de desvío de 1 metro tiene su línea de tendencia por debajo del margen establecido en la prueba anterior. Así se puede afirmar que a medida que aumentamos la escala la precisión del algoritmo se reduce y comienza a considerar normal, datos separados al menos 1 metro de la ruta original.

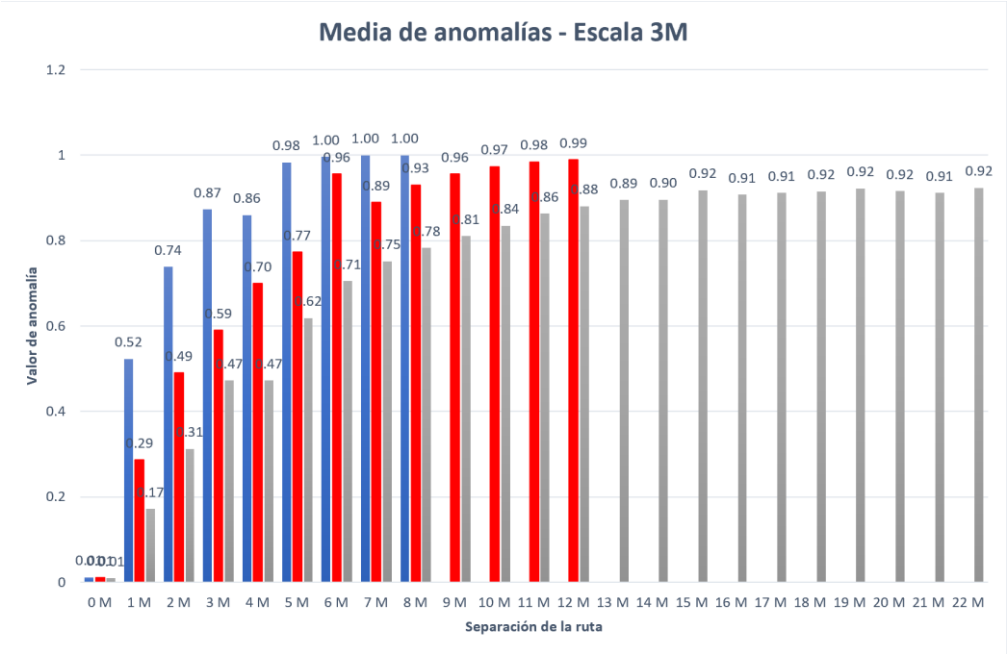


Captura 4-8 Gráficas de los distintos desvíos respecto a la ruta base con escala de 2 metros

4.2.3 Escala 3 metros

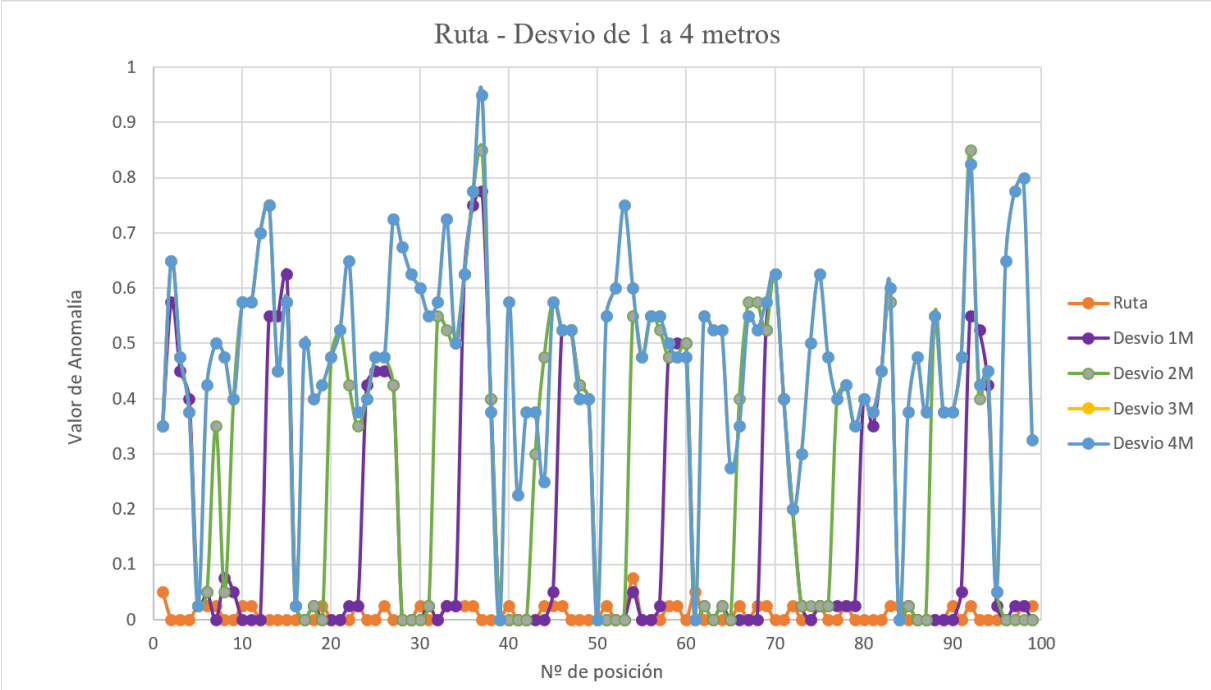
Por último, se utiliza una escala de 3 metros para analizar los datos, y como en la prueba anterior hay que aumentar la distancia de separación respecto a la ruta original. Los datos obtenidos son muy similares a los anteriores estableciéndose ahora la estabilización en los 12 metros, ver Captura 4-10.

Con estas tres pruebas se puede decir que cada rango de escala establecido en NuPIC, genera una zona de anomalía que no es exactamente el valor de esta escala, es decir, con la escala de 1 metro considerará perfectamente anómalo todo lo que se encuentre con una separación de 5 metros, 9 metros en el caso de la escala nº2 y 12 metros en la nº3. Pudiendo establecer la hipótesis de que con cada aumento de escala esta zona se separa 4 metros de la ruta original.

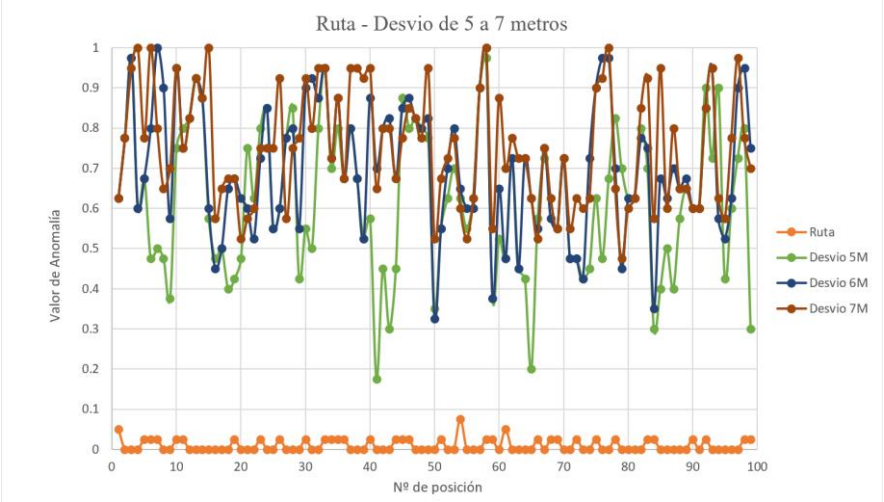


Captura 4-10 Escala 3 metros media de anomalías

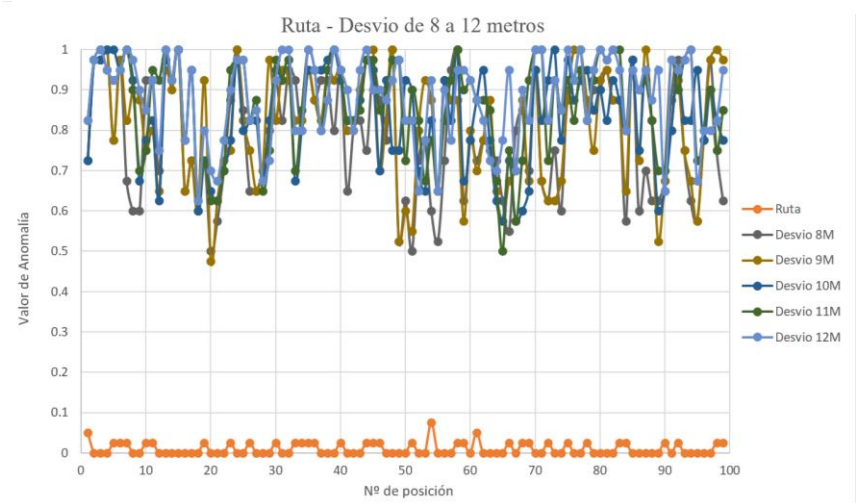
Además, se puede advertir que esa zona de imprecisión de la anomalía también se agranda en este caso, ahora existe una zona de 4 metros de separación respecto al original donde podemos encontrar datos con un valor de anomalía de 0. Sin embargo, se puede ver que se mantiene una zona clara de separación. Por lo tanto, se puede decir que el algoritmo un índice de anomalía entre 0 y 0,3 es bueno. Entre 0.3 y 0.7 existe una zona de incertidumbre donde se produce el tránsito hasta la separación donde cualquier posición es considerada anómala.



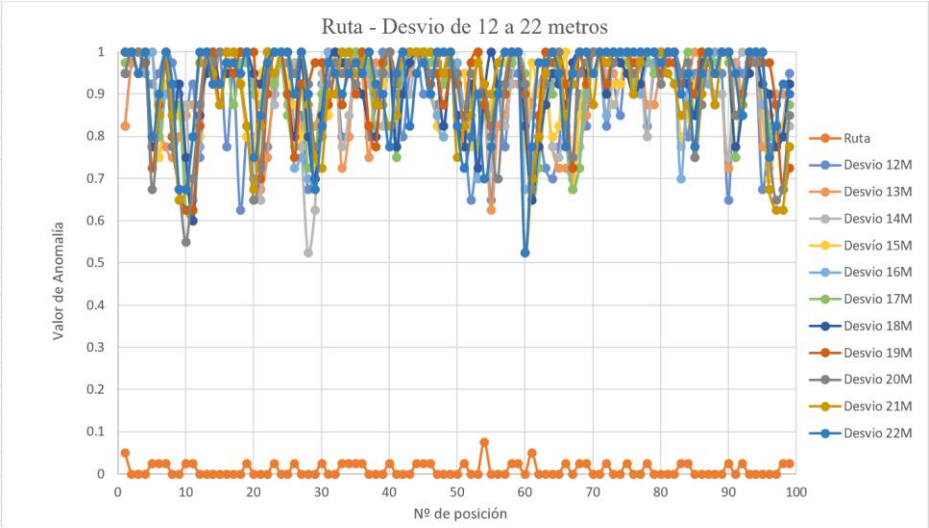
Captura 4-9 Ruta - desvío de 1 a 4 m (Escala 3m)



Captura 4-13 Escala 3m. Desvio de 5 a 7m



Captura 4-12 Escala de 3m. Desvío de 8 a 12 m



Captura 4-11 Escala 3m. Desvío de 12 a 22 m

4.2.4 Resultados y conclusiones

Como se puede apreciar en las gráficas donde se hayan representados los valores de anomalía en cada posición. Tal y como se ha dicho se genera una concentración de anomalías a partir de los 12 metros de separación que se mantiene constante hasta los 22 metros. Ver Captura 4-9, Captura 4-13, Captura 4-12 y Captura 4-11.

En resumen, con esta prueba se han podido establecer dos hipótesis sobre las capacidades de NuPIC:

- La escala: A medida que la vamos aumentando, esta no solo permite una zona de tolerancia mayor, es decir una zona donde las posibles desviaciones puedan considerarse como algo normal, sino que también existe una separación respecto a la ruta aprendida que se amplía con la escala de 4 en 4 metros y a partir de la cual se estabiliza el valor de anomalía y todo lo que se encuentra a partir de ahí es anómalo.
- La parametrización de los valores de anomalías. Se ha podido establecer mediante una ruta sencilla una serie de rangos en los que trabaja NuPIC y dentro de los cuales podemos considerar como algo anómalo o no.

Anómalo	**Zona de transición**	**No Anómalo**
1 – 0.7	0.7 – 0.3	0.3 - 0

Tabla 3 Umbral de anomalías

Por lo tanto, basado en la experimentación se sugiere esta hipótesis para dar valores de anomalía. Los valores de la zona de transición podrán ser objeto de otra investigación su tratamiento, pero lo cierto es que cuando NuPIC genera valores en esa zona es porque est a llevando a cabo una transición a alguna de las otras zonas.

4.3 Prueba con datos reales

El objetivo de estas pruebas es comprobar la capacidad de NuPIC de procesar archivos con mucha información y ofrecer un valor de anomalía correcto. Y los resultados son muy satisfactorios, la realidad es que el algoritmo tras 40 minutos de trabajo continuo ha sido capaz de procesar 68660 posiciones procedentes de un archivo .log grabado con el TV32 el día 28 de febrero de 2020.

El principal problema que surge es que al ser un archivo grabado por el programa tal y como se reciben las posiciones reportadas por AIS no ha permitido obtener datos de anomalías muy concluyentes, sin embargo, destacar que a pesar de que la mayoría de vaores de anomalía son 1, cuando procesa una cadena de datos de un mismo contacto (mismo MMSI), es capaz de aprender y dar valores de anomalía bastante bajos. Esta prueba, requiere de un estudio en mayor profundidad, así como barajar la posibilidad de previa ejecución con NuPIC generar un programa en Python que genere el formato adecuado al tiempo que lee el archivo almacenado para permitir procesar las rutas correctamente.

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Capacidades destacables del algoritmo

Lo cierto es que como se ha podido comprobar NuPIC presenta unas características muy buenas a la hora de procesar grandes cantidades de datos y generar un valor de anomalía para ellos. Las distintas pruebas y la precisión de metros con la que es capaz de detectar anomalías lo convierten en una herramienta muy capaz.

Desde una ruta recta sencilla a un conjunto de datos reales procedentes de AIS, de los cuales, si bien no se ha conseguido el resultado esperado, se ha podido comprobar que es capaz de aprender una ruta pequeña dentro de un conjunto de miles de datos casi aleatorios. Por lo tanto, el algoritmo como ejemplo de las capacidades de la IA en la detección de anomalías ha demostrado poseer unas características adecuadas

5.2 La detección de anomalías en escenarios marítimos

Si bien no se ofrecen los resultados esperados, mediante el desarrollo de un programa previo que recoja los datos y los organice de forma que el algoritmo pueda tratarlos, puede resultar ser una herramienta muy potente. Se ha podido comprobar la gran cantidad de datos que es capaz de procesar en muy poco tiempo y con medios de hardware limitados. Si se analizan estos datos, podemos evaluar que con los medios y datos correctos un algoritmo de aprendizaje autónomo puede procesar escenarios marítimos y ofrecer datos muy útiles a los actores de la seguridad marítima.

5.3 El creciente desarrollo de las redes neuronales

La realidad de los numerosos descubrimientos en el campo de la IA puede llegar a provocar que Numenta abandone la iniciativa como una de las empresas principales de desarrollo de redes neuronales para la detección de anomalías, en favor de otras que se encuentran desarrollando sistemas similares. Por ello se hace interesante un análisis de las innovaciones en este campo y las posibles mejoras que puedan desarrollarse. Posiblemente en un futuro, la competencia en el sector permita el desarrollo de algoritmos más complejos y adecuados a las necesidades de la seguridad marítima. Como se ha visto en [27] actualmente ha aumentado el número de empresas interesadas en este campo de investigación, por lo que resulta interesante conocerlas y ver que capacidades podían aportar a este proyecto. [20]

5.4 Líneas futuras de la investigación

Como principal vía futura de investigación se presenta la posibilidad de desarrollar una aplicación de inteligencia artificial que utilice algoritmos de aprendizaje autónomo no supervisado para detectar anomalías en tiempo real en el tráfico de información recibido por el Centro de Operaciones y Vigilancia de Acción Marítima (COVAM). Como se ha demostrado, las capacidades de esta herramienta son indescriptibles teniendo en cuenta que las pruebas realizadas con el algoritmo se han desarrollado con un ordenador personal y a través de una máquina virtual. Los potentes servidores existentes en el mercado pueden hacer de esta herramienta un asistente muy potente en la detección de comportamientos anómalos en los teatros de interés nacional.

5.4.1 Procesado en tiempo real

La posibilidad de integrar el algoritmo en tiempo real se presenta como un reto futuro de vital importancia. En este trabajo, se han analizado rutas completas que eran procesadas según una base de datos de información AIS una detrás de la otra, sin embargo, la posibilidad de que el algoritmo procese la información en tiempo real se plantea como un gran reto. El objetivo de este proyecto sería la detección de anomalías de tal forma que permitan activar una serie de respuestas coordinadas desde el COVAM en los organismos encargados de la seguridad marítima. Por ello se define la rapidez en la detección como rasgo fundamental, y conseguir que se efectúe al mismo tiempo que se reciben los datos podía ser un futuro desarrollo interesante de investigación.

5.4.2 Unión máquina-humano. Analizar las posibilidades de integración.

Por otro lado, se hace necesario tender hacia una concepción actual en desarrollo: “Human in the loop”. En esta teoría se concibe una unión máquina-humano aprovechando las capacidades de ambos y generando un sistema perfecto.

"El ordenador es increíblemente rápido, preciso y estúpido. El hombre es increíblemente lento, imprecise y brillante. Juntos, tienen un poder que va más allá de lo imaginado.” – Albert Einstein.

Con esto se busca una tendencia a integrar esta capacidad de la IA en una aplicación supervisada por operadores experimentados que permitan aumentar exponencialmente las capacidades de procesado de información del COVAM. Además, permitiría a estos centrar su atención en la toma de decisiones, dejando el análisis de datos a las máquinas las cuales generan resultados mucho más precisos, efectivos y rápidos. Con todo esto, sería interesante estudiar la forma de integrar las capacidades de detección de anomalías en los centros de vigilancia junto con las actividades de los operadores que monitorizan y procesan todo el flujo de información. Todo ello buscando mejorar la capacidad de reacción de los distintos actores en la búsqueda de problemas que amenacen la seguridad marítima y generar una respuesta lo más rápida y eficaz posible.

6 BIBLIOGRAFÍA

[1] United Nations Conference On Trade And Develpment - UNCTAD, "2018 HANDBOOK OF STATISTICS," Genova, 2018.

[2] Departamento de Seguridad Nacional - Presidencia del Gobierno, *Estrategia de Seguridad Nacional 2017*, 2017.

[3] Ministerio de Defensa, *Real Decreto 96/2009 Reales Ordenanzas para las Fuerzas Armadas*, Madrid: Boletin Oficial del Estado, 2009.

[4] Ministerio de Fomento, *Real Decreto Legislativo 2/2011, de 5 de septiembre, por el que se aprueba el Texto Refundido de la Ley de Puertos del Estado y de la Marina Mercante*, BOE, 2011.

[5] IALA(International Association of Marine Aids to Navigation and Lighthouse Authorities), *IALA Guideline No. 1068 Assistenace Service by Vessel Traffic Service*, 2009.

[6] IALA, *Guidelines on AIS as a VTS tool*, 2001.

[7] Guardia Civil , «Sistema Integrado de Vigilancia Exterior - SIVE,» 2019.

[8] ALMART, *Instrucción Permanente Num. 2.05 (CONFIDENCIAL)*, 2004.

[9] ALMART, *Instrucción Permanente NUM. 2.03 (CONFIDENCIAL)*, Cartagena: Publicación oficial Armada, 2008.

[10] ACATECH, Recommendations for implementing the strategic initiative INDUSTRIE 4.0, 2013.

[11] J. Velasco, «Inteligencia Artificial: Ramon Llull y el Ars Magna,» Marzo 2012. [En línea].

[12] A. M. Turing, *COMPUTING MACHINERY AND INTELLIGENCE*, 1950.

[13] A. Turing, *Intelligent Machinery*, 1948.

[14] J. McCarthy, *The Little Thoughts of Thinking Machines*, Stanford, CA 94305: Stanford University, 1983.

[15] J. Piaget, Seis Estudios de Psicología, vol. 2, Labor.

[16] J. Hawkins y S. Blakeslee, On Intelligence, Times Books, 2004.

[17] Numenta Inc, *Memoria Temporal Jerárquica - Algoritmos de Aprendizaje Cortical HTM*, vol. versión 0.2.1, 2011.

[18] S. Martín Marco y M. Rodelgo Lacruz, *Estudio y optimización de un sistema de inteligencia artificial para el análisis de tráfico marítimo y aéreo empleando AIS y ADS-B*, Marín: Universidad de Vigo, 2017.

[19] Numenta, «Geospatial Coordinate Encoder - YouTube,» [En línea]. Available: https://www.youtube.com/watch?v=KxxHo-FtKRo.

[20] Numenta, «Github - Numenta Anomaly Benchmark,» 2019. [En línea]. Available: https://github.com/numenta/NAB. [Último acceso: febrero 2020].

[21] C. A. Pérez-Seoane, *Desarrollo de un sistema de inteligencia artificial para la supervisión y detección de anomalías en rutas marítimas,* Centro Universitario de la Defensa, 2016.

[22] M. P. Hernández, *Desarrollo de un sistema de representación de rutas marítimas con detección de anomalías en tiempo real,* Centro Universitario de la Defensa, 2017.

[23] A. R. Carmona, *Desarrollo de un sistema de Inteligencia Artificial para la identificación del tráfico aéreo,* Centro Universitario de la Defensa, 2018.

[24] M. d. l. G. García, «El COVAM de la Armada al servicio de la Seguridad Marítima,» *Instituto Español de Estudios Estratégicos,* vol. 55/2011, 27 Julio 2011.

[25] R. V. Bueno, «Automatic detection of signals by using artificial intelligence techniques,» , 2012. [En línea]. Available: https://dialnet.unirioja.es/servlet/tesis?codigo=102112. [Último acceso: 10 2 2020].

[26] Numenta, "GitHub NuPIC Install," Numenta, [Online]. Available: https://github.com/numenta/nupic. [Accessed 01 2020].

[27] A. Lavin y S. Ahmad, «Evaluating Real-time Anomaly Detection Algorithms – the Numenta Anomaly Benchmark,» *ICMLA 2015,* Oct 2015.

[28] Ministerio de Transportes, Movilidad y Agenda Urbana, «Web oficial del Ministerio de Transportes, Movilidad y Agenda Urbana,» [En línea]. Available: https://www.mitma.gob.es/. [Último acceso: 25 Febrero 2020].

[29] Union Europea, «Eurostats,» European Commission, [En línea]. Available: https://ec.europa.eu/eurostat/web/main/home. [Último acceso: 25 Febrero 2020].

[30] Naciones Unidas, *Convención de las Naciones Unidas sobre el Derecho del Mar,* Montego Bay, 1982.

[31] I. M. Organisation, *Guidelines for the onboard openational use of shipborne automatic identification sistem (AIS),* vol. Resolution A.917(22), 2001.

[32] «Github - Numenta,» [En línea]. Available: https://github.com/numenta/NAB. [Último acceso: 12 02 2020].

[33] O. Berges Bergàda y A. Martín Mallofré, *Estudio de las actividades ilegales en el trensporte marítimo,* Universidad Politecnica de Cataluña, 2011.

[34] U. S. D. o. State, *Volume I. Drug and Chemical Control,* 2019.

[35] A. (. J. d. E. M. d. l. Armada), *Adición1. Lineas Generales de la Armada 2017,* Ministerio de Defensa. Armada Española, 2017.

[36] A. (. G. J. d. E. M. d. l. A. T. L. Calderón, *Lineas Generales de la Armada,* Madrid: Armada Española, 2017.

[37] IMO (International Maritime Organisation), *Resolution A.851(20),* 1997.

[38] O. (. M. Internacional), *Anexo 2 - Aspectos Tecnicos. Resolución a.572(14).*

[39] NMEA, *NMEA 0183 - Standar for Interfacing Marine Electronic devices,* Version 4.11, 2018.

[40] NMEA, *Automatic Identification System,* 2008.

[41] J. McCarthy, M. Minsky, N. Rochester y C. Shannon, «A PROPOSAL FOR THE DARMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE,» 31 August 1995.

[42] F. I. d. C. C. y. P. D. I. P. AN (CIA-MO-EO) Lamas López, «Hacia un mantenimiento predictivo basado en la condición (CBM) de equipos embarcados utilizando redes neuronales,» *Boletín Técnico de Ingenieria de la Armada,* nº 12, Mayo 2017.

[43] D. M. P. ALGARRA y A. (. D. F. L. LÓPEZ, «DESCRIPCIÓN DE METODOLOGÍA DE CLASIFICACIÓN DE MODOS DE FALLO PARA INTEGRACIÓN EN SISTEMA PREDICTIVO,» *Boletin técnico de Ingenieria de la Armada,* nº 17, Diciembre 2019.

[44] A. H. R. Prieto, J. L. Pérez y A. Padrón., «EL MODELO NEURONAL DE McCULLOCH Y PITTS Interpretación Comparativa del Modelo,» *Congreso Nacional de Instrumentación,* 2016.

[45] S. Purdy, *Encoding Data for HTM Systems,* Redwood City, California, USA : Numenta, Inc.

[46] Ministerio de defensa, *Estrategia de Tecnología e Innovación para la Defensa ETID,* 2010.

[47] Minsiterio de Defensa. DIRECCIÓN GENERAL DE ARMAMENTO Y MATERIAL, *Estrategia de Tecnología e Innovación para la Defensa,* 2015.

[48] NATO SCIENCE AND TECHNOLOGY COMMITTEE , *ARTIFICIAL INTELLIGENCE: IMPLICATIONS FOR NATO'S ARMED FORCES,* 2019.

[49] V. Torra, «La Inteligencia Artificial,» *Lychnos. Fundación general CSIC,* 2011.

[50] R. L. Mántaras, «El futuro de la IA: hacia inteligencias artificiales realmente inteligentes,» *¿Hacia una nueva Ilustración? Una década trascendente,* 2019.

[51] J. Pastor, *Qué es la inteligencia Artificial,* 2018.

[52] The Guardian, «Eugene Goostman is a real boy – the Turing Test says so,» Jun 2014.

[53] A. T. L. Calderón, VISION ARMADA 4.0.

[54] ALMART, *OPTASK MSA (MARITIME SITUATIONAL AWARENESS) (CONFIDENCIAL),* Cartagena: Mensaje oficial Armada, 2011.

[55] International Maritime Organisation, *ResolutionA.917(22) - GUIDELINES FOR THE ONBOARD OPERATIONAL USE OF SHIPBORNE AUTOMATIC IDENTIFICATION SYSTEMS (AIS),* 2001.

[56] Jefatura del Estado, *Ley Orgánica 2/1986 de Fuerzas y Cuerpos de Seguridad,* Boletin Oficial del Estado, 1986.

[57] J. d. Estado, *Ley 14/2014 de Navegación Marítima,* Boletin Oficial del Estado, 2014.

[58] Jefatura del Estado, *Ley Orgánica 12/1995, de 12 de diciembre, de Represión del Contrabando,* BOE, 1995.

[59] NATO, *ATP-02.1 NAVAL COOPERATION AND GUIDANCE FOR SHIPPING (NCAGS) – GUIDE TO OWNERS, OPERATORS, MASTERS AND OFFICERS,* 2014.

[60] ALMART, *OPTASK RMP (CONFIDENCIAL),* 2011.

[61] Fuerza de Acción Marítima , *Acción derivada núm.2: "Definición de necesidades de investigación para el CUD". Anomalías de patrones de comportamiento de bueques. (USO OFICIAL),* 2020.

[62] M. d. l. G. García, «EL COVAM DE LA ARMADA AL SERVICIO DE LA COMUNIDAD MARÍTIM,» *IEEE Instituto Español de Estudios Estratégicos,* vol. 5, 2011.

[63] F. J. V. Álvarez, *Seguridad Marítima y Piratería,* 2013.

[64] A. (.-M. D. F. Llamas, «Uso de redes neuronales como técnica de clasificación de eventos sobre datos de SICP,» *Boletín Técnico de Ingenieria de la Armada,* nº 15, Diciembre 2018.

[65] D. P. L. Cánovas y F. AN (CIA) Lamas López, «Introducción a las técnicas de clasificación y regresión asociadas al aprendizaje automático,» *Boletín Técnico de Ingenieria de la Armada,* nº 16, Mayo 2019.

[66] «Glokor Research and Development Center,» [En línea]. Available: http://glokor.eu/industry-4-0-conference-industrial-automation-artificial-intelligence-october-26th-2017-gdansk/. [Último acceso: febrero 2020].

[67] E. Codina, «Interceptadas seis pateras en las costas andaluzas en los tres primeros días del año,» *El País,* Enero 2017.

[68] Ministerio de Transportes, Movilidad y Agenda Urbana, «Web Oficial Salvamento Marítimo,» [En línea]. Available: http://www.salvamentomaritimo.es/conocenos/nuestros-medios/20-centros-de-coordinacion-de-salvamento. [Último acceso: marzo 2020].

La información de las referencias bibliográficas clasificadas como CONFIDENCIAL no ha sido incluida en esta memoria, solo como referencia para estudio y conocimiento de la materia a desarrollar.

ANEXO I: DICCIONARIO DE SIGLAS

En este anexo se desarrolla una relación de las siglas y acrónimos empleados a lo largo de la memoria acompañados por su significado original y entre paréntesis la traducción al castellano cuando corresponda.

AJEMA: Almirante Jefe de Estado Mayor de la Armada

AIS: Automatic Identification Sistem

ALMART: Almirante Jefe de la Fuerza de Acción Marítima

COVAM: Centro de Operaciones y Vigilancia de Acción Marítima.

CSV: Coma Separated Value (valores separados por comas)

DST: Dispositivo de separación de tráfico

FAM: Fuerza de Acción Marítima

HTM: Hierarchical Temporal Memory (Memoria Temporal Jerárquica)

MMSI: Maritime Mobile Service Identity (Identificación del Servicio Móvil Marítimo)

IA: Inteligencia Artificial

NAB: Numenta Anomaly Benchmark (Comparador de anomalías de Numenta)

SIVE: Sistema Integrado de Vigilancia Exterior

SOLAS: Safety Of Life At Sea (Seguridad de la Vida en la Mar)

ANEXO II: GEOSPATIAL_ANOMALY.PY

```
1.      #!/usr/bin/env python
2.      # ----------------------------------------------------------------------
3.      # Numenta Platform for Intelligent Computing (NuPIC)
4.      # Copyright (C) 2014, Numenta, Inc.  Unless you have an agreement
5.      # with Numenta, Inc., for a separate license for this software code, the
6.      # following terms and conditions apply:
7.      #
8.      # This program is free software: you can redistribute it and/or modify
9.      # it under the terms of the GNU Affero Public License version 3 as
10.     # published by the Free Software Foundation.
11.     #
12.     # This program is distributed in the hope that it will be useful,
13.     # but WITHOUT ANY WARRANTY; without even the implied warranty of
14.     # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
15.     # See the GNU Affero Public License for more details.
16.     #
17.     # You should have received a copy of the GNU Affero Public License
18.     # along with this program.  If not, see http://www.gnu.org/licenses.
19.     #
20.     # http://numenta.org/licenses/
21.     # ----------------------------------------------------------------------
22.
23.     """
24.     A simple client to create a CLA anomaly detection model for geospatial.
25.     """
26.
27.     import csv
28.     import datetime
29.     import sys
30.
31.     from nupic.frameworks.opf.model_factory import ModelFactory
32.
33.     import model_params
34.
35.
36.
37.     DEFAULT_DATA_PATH = "data/commute.csv"
38.     DEFAULT_OUTPUT_PATH = "anomaly_scores.csv"
39.
40.     ACCURACY_THRESHOLD = 80  # meters
41.     INTERVAL_THRESHOLD = 30  # seconds
42.
43.
44.
45.     def addTimeEncoders(params):
46.       params["modelParams"]["sensorParams"]["encoders"]["timestamp_timeOfDay"] = {
47.         "fieldname": u"timestamp",
48.         "name": u"timestamp_timeOfDay",
49.         "timeOfDay": (51, 9.5),
50.         "type": "DateEncoder"
51.       }
52.       return params
53.
54.
55.
```

```
56.        def setEncoderScale(params, scale):
57.          params["modelParams"]["sensorParams"]["encoders"]["vector"]["scale"] = \
58.            int(scale)
59.          return params
60.
61.
62.
63.        def createModel(useTimeEncoders, scale, verbose):
64.          params = model_params.MODEL_PARAMS
65.          if useTimeEncoders:
66.            params = addTimeEncoders(params)
67.          if scale:
68.            params = setEncoderScale(params, scale)
69.          if verbose:
70.            print "Model parameters:"
71.            print params
72.          model = ModelFactory.create(params)
73.          model.enableInference({"predictedField": "vector"})
74.          return model
75.
76.
77.
78.        def runGeospatialAnomaly(dataPath, outputPath,
79.                                 scale=False,
80.                                 autoSequence=True,
81.                                 useTimeEncoders=False,
82.                                 verbose=False):
83.
84.          model = createModel(useTimeEncoders, scale, verbose)
85.
86.          with open (dataPath) as fin:
87.            reader = csv.reader(fin)
88.            csvWriter = csv.writer(open(outputPath,"wb"))
89.            csvWriter.writerow(["timestamp",
90.                                "longitude",
91.                                "latitude",
92.                                "speed",
93.                                "anomaly_score",
94.                                "new_sequence"])
95.
96.            reader.next()
97.            reader.next()
98.            reader.next()
99.
100.           lastTimestamp = None
101.           lastTrackName = None
102.           outputFormat = "%Y-%m-%dT%H:%M:%S"
103.
104.           for _, record in enumerate(reader, start=1):
105.             trackName = record[0]
106.             timestamp = datetime.datetime.fromtimestamp(int(record[1]) / 1e3)
107.             longitude = float(record[3])
108.             latitude = float(record[2])
109.             speed = float(record[5])
110.             accuracy = float(record[7])
111.
112.             altitude = float(record[4]) if record[4] != "" else None
113.
114.             if accuracy > ACCURACY_THRESHOLD:
115.               continue
116.
```

```
117.            newSequence = False
118.            # Handle the automatic sequence creation
119.            if autoSequence:
120.              if lastTimestamp and (
121.                (timestamp - lastTimestamp).total_seconds() > INTERVAL_THRESHOLD):
122.                newSequence = True
123.            # Manual sequence resets depend on the track name
124.            else:
125.              if trackName != lastTrackName:
126.                newSequence = True
127.
128.            lastTimestamp = timestamp
129.            lastTrackName = trackName
130.
131.            if newSequence:
132.              if verbose:
133.                print "Starting new sequence..."
134.              model.resetSequenceStates()
135.
136.            modelInput = {
137.              "vector": (speed, longitude, latitude, altitude)
138.            }
139.
140.            if useTimeEncoders:
141.              modelInput["timestamp"] = timestamp
142.
143.            result = model.run(modelInput)
144.            anomalyScore = result.inferences["anomalyScore"]
145.
146.            csvWriter.writerow([timestamp.strftime(outputFormat),
147.                                longitude,
148.                                latitude,
149.                                speed,
150.                                anomalyScore,
151.                                1 if newSequence else 0])
152.            if verbose:
153.              print "[{0}] - Anomaly score: {1}.".format(timestamp, anomalyScore)
154.
155.        print "Anomaly scores have been written to {0}".format(outputPath)
156.
157.
158.
159.      if __name__ == "__main__":
160.        dataPath = DEFAULT_DATA_PATH
161.        outputPath = DEFAULT_OUTPUT_PATH
162.
163.        if len(sys.argv) > 1:
164.          dataPath = sys.argv[1]
165.
166.        if len(sys.argv) > 2:
167.          outputPath = sys.argv[2]
168.
169.        runGeospatialAnomaly(dataPath, outputPath)
170.
```

ANEXO III: RUN.PY

```
1.      #!/usr/bin/env python
2.      # ----------------------------------------------------------------------
3.      # Numenta Platform for Intelligent Computing (NuPIC)
4.      # Copyright (C) 2013, Numenta, Inc.  Unless you have an agreement
5.      # with Numenta, Inc., for a separate license for this software code, the
6.      # following terms and conditions apply:
7.      #
8.      # This program is free software: you can redistribute it and/or modify
9.      # it under the terms of the GNU Affero Public License version 3 as
10.     # published by the Free Software Foundation.
11.     #
12.     # This program is distributed in the hope that it will be useful,
13.     # but WITHOUT ANY WARRANTY; without even the implied warranty of
14.     # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
15.     # See the GNU Affero Public License for more details.
16.     #
17.     # You should have received a copy of the GNU Affero Public License
18.     # along with this program.  If not, see http://www.gnu.org/licenses.
19.     #
20.     # http://numenta.org/licenses/
21.     # ----------------------------------------------------------------------
22.     import os
23.     import sys
24.     from optparse import OptionParser
25.
26.     from tools.preprocess_data import preprocess
27.     from tools.anomaly_to_js_data import postprocess
28.     from model.geospatial_anomaly import runGeospatialAnomaly
29.
30.     DEFAULT_OUTPUT_DIR = "output"
31.     verbose = False
32.     scriptDir = os.path.dirname(os.path.realpath(__file__))
33.
34.
35.     parser = OptionParser(
36.       usage="%prog <path/to/input/file> [options]\n\nRun NuPIC on specified "
37.             "location file, which should already be in the proper format "
38.             "(downloaded from the simulator)."
39.     )
40.
41.     parser.add_option(
42.       "-m",
43.       "--manual-sequence",
44.       action="store_true",
45.       default=False,
46.       dest="manualSequence",
47.       help="Automatically breaks into sequences based upon time gaps."
48.     )
49.     parser.add_option(
50.       "-t",
51.       "--time-encoders",
52.       action="store_true",
53.       default=False,
54.       dest="useTimeEncoders",
55.       help="Adds time of day encoder to model params."
56.     )
```

```
57.     parser.add_option(
58.       "-v",
59.       "--verbose",
60.       action="store_true",
61.       default=False,
62.       dest="verbose",
63.       help="Print debugging statements."
64.     )
65.     parser.add_option(
66.       "-o",
67.       "--output-dir",
68.       default=DEFAULT_OUTPUT_DIR,
69.       dest="outputDir",
70.       help="Where to write the output file."
71.     )
72.     parser.add_option(
73.       "-s",
74.       "--scale",
75.       default=False,
76.       dest="scale",
77.       help="Meter resolution for Geospatial Coordinate Encoder (default 5m)."
78.     )
79.
80.
81.     def run(inputPath, outputDir, useTimeEncoders, scale, autoSequence):
82.
83.       outputPath = os.path.abspath(outputDir)
84.       if not os.path.exists(outputPath):
85.         os.makedirs(outputPath)
86.
87.       preProcessedOutputPath = os.path.join(outputPath, "preprocessed_data.csv")
88.       if verbose: print "Pre-processing %s..." % inputPath
89.       preprocess(inputPath, preProcessedOutputPath, verbose=verbose)
90.
91.       anomalyOutputPath = os.path.join(outputPath, "anomaly_scores.csv")
92.       if verbose: print "Running NuPIC on %s..." % preProcessedOutputPath
93.       runGeospatialAnomaly(preProcessedOutputPath,
94.                            anomalyOutputPath,
95.                            scale=scale,
96.                            autoSequence=autoSequence,
97.                            useTimeEncoders=useTimeEncoders,
98.                            verbose=verbose)
99.
100.      visualizationOutputPath = os.path.join(scriptDir, "static/js/data.js")
101.      if verbose: print "Creating visualization at %s..." % visualizationOutputPath
102.      postprocess(anomalyOutputPath, visualizationOutputPath)
103.
104.
105.    if __name__ == "__main__":
106.      (options, args) = parser.parse_args(sys.argv[1:])
107.      try:
108.        input_path = args.pop(0)
109.      except IndexError:
110.        parser.print_help(sys.stderr)
111.        sys.exit()
112.
113.      verbose = options.verbose
114.
115.      run(
116.        input_path,
```

```
117.        options.outputDir,
118.        options.useTimeEncoders,
119.        options.scale,
120.        not options.manualSequence)
121.
```

ANEXO IV: MODELPARAMS.PY

```
1.      # ----------------------------------------------------------------------
2.      # Numenta Platform for Intelligent Computing (NuPIC)
3.      # Copyright (C) 2014, Numenta, Inc.  Unless you have an agreement
4.      # with Numenta, Inc., for a separate license for this software code, the
5.      # following terms and conditions apply:
6.      #
7.      # This program is free software: you can redistribute it and/or modify
8.      # it under the terms of the GNU Affero Public License version 3 as
9.      # published by the Free Software Foundation.
10.     #
11.     # This program is distributed in the hope that it will be useful,
12.     # but WITHOUT ANY WARRANTY; without even the implied warranty of
13.     # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
14.     # See the GNU Affero Public License for more details.
15.     #
16.     # You should have received a copy of the GNU Affero Public License
17.     # along with this program.  If not, see http://www.gnu.org/licenses.
18.     #
19.     # http://numenta.org/licenses/
20.     # ----------------------------------------------------------------------
21.
22.     MODEL_PARAMS = {
23.         # Type of model that the rest of these parameters apply to.
24.         'model': "HTMPrediction",
25.
26.         # Version that specifies the format of the config.
27.         'version': 1,
28.
29.         # Intermediate variables used to compute fields in modelParams and also
30.         # referenced from the control section.
31.         'predictAheadTime': None,
32.
33.         # Model parameter dictionary.
34.         'modelParams': {
35.             # The type of inference that this model will perform
36.             'inferenceType': 'TemporalAnomaly',
37.
38.             'sensorParams': {
39.                 # Sensor diagnostic output verbosity control;
40.                 # if > 0: sensor region will print out on screen what it's sensing
41.                 # at each step 0: silent; >=1: some info; >=2: more info;
42.                 # >=3: even more info (see compute() in py/regions/RecordSensor.py)
43.                 'verbosity' : 0,
44.
45.                 # Example:
46.                 #     dsEncoderSchema = [
47.                 #       DeferredDictLookup('__field_name_encoder'),
48.                 #     ],
49.                 #
50.                 # (value generated from DS_ENCODER_SCHEMA)
51.                 'encoders': {
52.                     u'vector':    {
53.                         'fieldname': u'vector',
54.                         'n': 2048,
55.                         'w': 51,
56.                         'scale': 5,
57.                         'timestep': 10,
```

```
58.                         'name': u'vector',
59.                         'type': 'GeospatialCoordinateEncoder'
60.                     },
61.                 },
62.
63.                 # A dictionary specifying the period for automatically-generated
64.                 # resets from a RecordSensor;
65.                 #
66.                 # None = disable automatically-generated resets (also disabled if
67.                 # all of the specified values evaluate to 0).
68.                 # Valid keys is the desired combination of the following:
69.                 #  days, hours, minutes, seconds, milliseconds, microseconds, weeks
70.                 #
71.                 # Example for 1.5 days: sensorAutoReset = dict(days=1,hours=12),
72.                 'sensorAutoReset' : None,
73.             },
74.
75.             'spEnable': True,
76.
77.             'spParams': {
78.                 # SP diagnostic output verbosity control;
79.                 # 0: silent; >=1: some info; >=2: more info;
80.                 'spVerbosity' : 0,
81.
82.                 # Spatial Pooler implementation selector.
83.                 # Options: 'py', 'cpp' (speed optimized, new)
84.                 'spatialImp' : 'cpp',
85.
86.                 'globalInhibition': 1,
87.
88.                 # Number of columns in the SP (must be same as in TP)
89.                 'columnCount': 2048,
90.
91.                 'inputWidth': 0,
92.
93.                 # SP inhibition control (absolute value);
94.                 # Maximum number of active columns in the SP region's output (when
95.                 # there are more, the weaker ones are suppressed)
96.                 'numActiveColumnsPerInhArea': 40,
97.
98.                 'seed': 1956,
99.
100.                # potentialPct
101.                # What percent of the columns's receptive field is available
102.                # for potential synapses.
103.                'potentialPct': 0.8,
104.
105.                # The default connected threshold. Any synapse whose
106.                # permanence value is above the connected threshold is
107.                # a "connected synapse", meaning it can contribute to the
108.                # cell's firing. Typical value is 0.10.
109.                'synPermConnected': 0.1,
110.
111.                'synPermActiveInc': 0.0001,
112.
113.                'synPermInactiveDec': 0.0005,
114.
115.                'boostStrength': 1.0,
116.            },
117.
118.            # Controls whether TP is enabled or disabled;
```

```
119.            # TP is necessary for making temporal predictions, such as predicting
120.            # the next inputs.  Without TP, the model is only capable of
121.            # reconstructing missing sensor inputs (via SP).
122.            'tmEnable' : True,
123.
124.            'tmParams': {
125.                # TP diagnostic output verbosity control;
126.                # 0: silent; [1..6]: increasing levels of verbosity
127.                # (see verbosity in nta/trunk/py/nupic/research/TP.py and TP10X*.py)
128.                'verbosity': 0,
129.
130.                # Number of cell columns in the cortical region (same number for
131.                # SP and TP)
132.                # (see also tpNCellsPerCol)
133.                'columnCount': 2048,
134.
135.                # The number of cells (i.e., states), allocated per column.
136.                'cellsPerColumn': 32,
137.
138.                'inputWidth': 2048,
139.
140.                'seed': 1960,
141.
142.                # Temporal Pooler implementation selector (see _getTPClass in
143.                # CLARegion.py).
144.                'temporalImp': 'cpp',
145.
146.                # New Synapse formation count
147.                # NOTE: If None, use spNumActivePerInhArea
148.                #
149.                # TODO: need better explanation
150.                'newSynapseCount': 20,
151.
152.                # Maximum number of synapses per segment
153.                #  > 0 for fixed-size CLA
154.                # -1 for non-fixed-size CLA
155.                #
156.                # TODO: for Ron: once the appropriate value is placed in TP
157.                # constructor, see if we should eliminate this parameter from
158.                # description.py.
159.                'maxSynapsesPerSegment': 32,
160.
161.                # Maximum number of segments per cell
162.                #  > 0 for fixed-size CLA
163.                # -1 for non-fixed-size CLA
164.                #
165.                # TODO: for Ron: once the appropriate value is placed in TP
166.                # constructor, see if we should eliminate this parameter from
167.                # description.py.
168.                'maxSegmentsPerCell': 128,
169.
170.                # Initial Permanence
171.                # TODO: need better explanation
172.                'initialPerm': 0.21,
173.
174.                # Connected Permanence
175.                'connectedPerm': 0.5,
176.
177.                # Permanence Increment
178.                'permanenceInc': 0.1,
```

```
179.
180.                # Permanence Decrement
181.                # If set to None, will automatically default to tpPermanenceInc
182.                # value.
183.                'permanenceDec' : 0.1,
184.
185.                'globalDecay': 0.0,
186.
187.                'maxAge': 0,
188.
189.                # Minimum number of active synapses for a segment to be considered
190.                # during search for the best-matching segments.
191.                # None=use default
192.                # Replaces: tpMinThreshold
193.                'minThreshold': 3,
194.
195.                # Segment activation threshold.
196.                # A segment is active if it has >= tpSegmentActivationThreshold
197.                # connected synapses that are active due to infActiveState
198.                # None=use default
199.                # Replaces: tpActivationThreshold
200.                'activationThreshold': 6,
201.
202.                'outputType': 'normal',
203.
204.                # "Pay Attention Mode" length. This tells the TP how many new
205.                # elements to append to the end of a learned sequence at a time.
206.                # Smaller values are better for datasets with short sequences,
207.                # higher values are better for datasets with long sequences.
208.                'pamLength': 3,
209.            },
210.
211.            # Don't create the classifier since we don't need predictions.
212.            'clEnable': False,
213.            'clParams': None,
214.
215.            'anomalyParams': {  u'anomalyCacheRecords': None,
216.        u'autoDetectThreshold': None,
217.        u'autoDetectWaitRecords': 2184},
218.
219.            'trainSPNetOnlyIfRequested': False,
220.        },
221.    }
222.
```

ANEXO V: AISMSG.PY

```
import ais.stream

print("Introduzca nombre del archivo AIS: ")
file_in =input()
file_out = open('salida.csv', 'a')

# fields of interest

keys = ['mmsi','timestamp', 'y', 'x', 'cog', 'true_heading', 'sog']

# write csv header

header = ','.join(keys) + '\n'
file_out.write(header)

# transform ais msg to csv line
with open(file_in) as f:
    for msg in ais.stream.decode(f):
      try:
        vals = [str(msg[k]) for k in keys]
        line = ','.join(vals) + '\n'
        file_out.write(line)
      except:
        pass
```

ANEXO VI: DECODIFICACIÓN DE UN MENSAJE AIS

mmsi,timestamp,y,x,cog,true_heading,sog

257801500,22,28.1431666667,-15.4252333333,288.299987793,7,0.0

225439000,22,36.127815,-5.43483166667,87.3000030518,511,0.0

225434000,23,35.8955833333,-5.31237,302.200012207,511,0.0

224476170,23,40.6162483333,0.59637,240.600006104,71,0.0

224320160,22,38.87312,0.942915,167.699996948,511,2.79999995232

225990606,39,28.8571866667,-13.816015,360.0,511,0.0

311009600,23,28.1231566667,-15.3708483333,78.1999969482,109,1.10000002384

244860120,22,28.1250966667,-15.42486,322.200012207,511,0.10000000149

227425000,23,44.49152,-1.41470833333,346.100006104,511,3.59999990463

235013826,22,39.5623533333,2.62863666667,142.600006104,511,0.0

477620100,20,36.3188316667,-2.68127333333,81.9000015259,80,11.1000003815

244870861,23,41.5580583333,2.50752166667,195.199996948,511,0.0

255806171,23,38.3275216667,-0.496206666667,253.300003052,5,0.0

224306530,23,38.17209,-0.0190016666667,232.699996948,511,2.90000009537

224901000,23,39.560675,2.63324333333,67.3000030518,326,0.0

224209620,23,41.0603716667,1.30852666667,22.5,511,3.40000009537

224306340,42,37.8157233333,-0.78369,12.8999996185,511,0.0

319153200,40,42.279705,-8.70270166667,107.400001526,130,0.0

224265320,44,36.1354066667,-5.42555833333,66.4000015259,511,15.8000001907

306885000,43,36.14612,-5.42900166667,221.199996948,203,0.0

225989762,42,39.4397433333,-0.30872,112.900001526,511,0.5

225987279,16,43.5513333333,-5.68800166667,360.0,511,0.0

671657000,43,36.80083,2.4515,256.899993896,257,12.1000003815

224081290,43,28.9204966667,-13.6761316667,360.0,511,0.0

357751000,53,24.5652316667,-16.901915,9.0,10,14.3000001907

249107000,44,43.136855,-9.780335,16.0,18,15.3999996185

244130061,43,37.9724083333,-0.687441666667,337.200012207,511,0.20000000298

224386370,42,36.4605583333,-4.46746333333,68.1999969482,511,2.90000009537

210748000,44,34.6743833333,-7.25475166667,42.2999992371,38,10.8999996185

224066450,43,36.48778,-4.95178166667,352.399993896,511,0.10000000149