

Universidad Abierta Interamericana

ANALITICOS VISUALES CON INTELIGENCIA ARTIFICIAL APLICADOS EN CIENCIA MARINA

Tesista: Lic. Luis Dario Ceballos

Director: Dr. Marcos Daniel Zárate

Trabajo presentado para obtener el título de

//// DOCTOR EN INFORMÁTICA ////

RESUMEN //

La presente tesis doctoral desarrolla un estudio integral sobre el diseño e implementación de sistemas de analítica automatizada aplicada en ciencia marina, integrando inteligencia artificial, tecnologías semánticas y visualización de datos. El trabajo presenta los fundamentos teóricos y metodológicos de la visualización de datos científicos y los modelos de lenguaje extensos (LLM), junto con una revisión del estado del arte en plataformas de visualización y la aplicación específica en datos de ciencia marina. Este análisis permite identificar una brecha significativa en los sistemas que combinan razonamiento semántico e interacción en interpretación de datos para generar representaciones visuales adecuadas de forma accesible. Con el objetivo de superar dicha brecha, se propone VisBot, un sistema experimental que integra la recomendación automatizada de analíticos visuales impulsada por inteligencia artificial aplicado en un caso de estudio que utiliza datos de biodiversidad marina. Esta tesis pretende contribuir al campo del desarrollo de herramientas de análisis visual automatizado con el fin de facilitar la exploración de datos, favorecer el descubrimiento de conocimiento y promover la convergencia entre datos complejos, inteligencia artificial y generación de visualizaciones accesibles para usuarios con diversos niveles de habilidad técnica.

Palabras clave: *Visualización de Datos, Large Language Models, Análisis Automatizado, Ciencia Marina.*

DEDICATORIA //

Este trabajo está dedicado a mis hijos, Uriel y Milo, pilares fundamentales frente a los desafíos que propone el ámbito tanto personal como profesional y quienes me acompañan y sostienen a lo largo del camino.

Agradezco a su vez a Dios, a mis padres que no llegaron a ver este resultado pero sin su generosidad y entendimiento no hubiese sido posible, a la familia, a mis hermanos y amigos que son familia también y sobre todo al hacer las cosas con buenas intenciones.

RECONOCIMIENTOS //

A la ciencia argentina, especialmente a CENPAT-CONICET por sostener día a día a tantos quienes como yo apostamos por la carrera científica y académica.

A CESIMAR por brindar un espacio de investigación adecuado y profesional donde descubrir conocimiento sea parte cotidiana de la actividad.

A los colegas, y amigos con quienes compartí en mayor o menor medida este proceso.

A la UAI (Universidad Abierta Interamericana) por ofrecer un espacio de formación de excelencia, no solo desde lo académico sino también, desde lo humano. Por ello no puedo dejar de agradecer especial y enormemente al Dr. Carlos Neil, quien con su apoyo e impulso fue fundamental para seguir adelante frente a los obstáculos.

A mi Director de doctorado en CONICET Dr. Claudio Delrieux, por su enorme generosidad.

A mi Director en el doctorado UAI, Dr. Marcos Zárate, por su acompañamiento y guía.

ÍNDICE

1 INTRODUCCIÓN..... 1

1.1. Contexto y Justificación..... 1

1.2. Planteamiento del Problema..... 5

1.3. Objetivos de la Investigación..... 8

1.3.1. Objetivo General..... 8

1.3.2. Objetivos Específicos..... 9

1.4. Metodología..... 13

1.5. Estructura de la Tesis..... 17

2 MARCO TEÓRICO..... 19

2.1. Fundamentos de Visualización Científica..... 19

2.1.1. Percepción Visual..... 20

2.1.2. Modelos de Transformación de Datos..... 23

2.1.3. Taxonomías de Tipos de Datos..... 27

2.1.4. Taxonomías de Tareas Analíticas..... 32

2.2. Aplicaciones en Datos Científicos Marinos..... 33

2.2.1. Iniciativas de Datos Abiertos en Ciencia Marina..... 34

2.2.2. Sistemas Especializados de Visualización Científica..... 38

2.3. Tecnologías Emergentes..... 40

2.3.1. Large Language Models..... 41

2.3.2. Datos Enlazados y Web Semántica..... 44

2.3.3. Sistemas de Recomendación de Visualización..... 46

3 ESTADO DEL ARTE..... 48

3.1. Análisis Crítico de la Literatura..... 48

3.1.1. Evolución de Visualización de Datos Científicos..... 48

3.1.2. Sistemas de Recomendación Basados en Aprendizaje..... 50

3.1.3. Integración de Modelos de Lenguaje en Analítica Visual..... 52
3.2. Plataformas Interactivas de Visualización de Datos..... 54
3.2.1. Plataformas Comerciales..... 54
3.2.2. Librerías de Código Abierto..... 57
3.2.3. Entornos de Desarrollo Integrado para Análisis Estadístico.....59
3.2.4. Herramientas de Arquitectura Multidimensional.....61
3.3. Inteligencia Artificial Aplicada en Visualización..... 62
3.3.1. Arquitecturas de Integración LLM-Visualización.....63
3.3.2. Ingeniería de Prompts..... 65
3.4. Visualización de Datos Enlazados.....66
3.4.1. Sistemas de Detección Automática..... 67
3.4.2. Sistemas de Especificación Declarativa.....68
3.4.3. Sistemas con Modelo Formal de Visualización..... 69
4 CASO DE ESTUDIO.....71
4.1. Introducción..... 71
4.2. Contexto y Motivación..... 73
4.3. Arquitectura y Componentes Funcionales..... 76
4.3.1 Selección del Dataset para el caso de estudio..... 76
4.3.2. Arquitectura de Sistema y Flujo de Datos.....77
4.3.3. Generación de Visualizaciones..... 86
4.4. Resultados.....90
4.4.1. Métricas y Criterios de Evaluación.....90
4.4.1.1 Validación Específica con Datos Oceanográficos..... 95
Detección Automática de Variables Geoespaciales Marinas..... 95
Visualizaciones Geoespaciales Especializadas.....96
Análisis Temporal de Series Oceanográficas..... 96
Integración con Estándares de Biodiversidad..... 97
Limitaciones Identificadas en Contexto Marino.....97
4.4.2. Resultados Cuantitativos Comparativos..... 98
4.4.3. Resultados Cualitativos.....102

4.4.3.1 Percepción de Utilidad y Relevancia..... 103
4.4.3.2 Observaciones de Patrones de Interacción..... 104
4.4.3.3 Aspectos Positivos Identificados..... 105
4.4.4. Análisis Comparativo Frente a Herramientas de Referencia.....105
Aplicabilidad en Dominios Científicos Especializados..... 107
5 DISCUSIÓN.....109
5.1. Capacidades de VisBot..... 109
5.2. Limitaciones y Áreas de Mejora Identificadas.....110
5.2.1. Limitaciones Metodológicas Críticas..... 111
Dependencia de Infraestructura Comercial Propietaria.....111
5.2.2. Limitaciones Técnicas Moderadas..... 113
Capacidades de Personalización Visual Limitadas.....113
Soporte Limitado para Tipos de Datos Especializados..... 115
5.2.3. Limitaciones Menores de Diseño e Integración.....116
Refinamiento Estético de la Interfaz Gráfica..... 116
Ausencia de Integración con Entornos de Análisis Programático..... 117
Capacidades Colaborativas y Multiusuario Ausentes.....118
5.3. Estándares y consideraciones éticas..... 119
6 CONCLUSIONES.....121
REFERENCIAS.....124

CAPÍTULO

1 INTRODUCCIÓN

1.1. Contexto y Justificación

El crecimiento exponencial de datos en las últimas décadas ha transformado radicalmente la manera en que se realiza la presentación de información en múltiples disciplinas. Este fenómeno, denominado comúnmente 'Big Data' y ampliamente analizado por Mumuni et al. (2025), presenta desafíos significativos en el ámbito de las Ciencias de Datos, particularmente en tres áreas críticas que definen el ciclo de vida de la información digital, la integración y gestión de datos heterogéneos provenientes de fuentes diversas, el análisis sistemático y la extracción de conocimiento significativo mediante técnicas computacionales avanzadas, y finalmente, la comunicación efectiva de insights complejos a usuarios con diversos niveles de expertise técnico, desde investigadores especializados hasta tomadores de decisiones sin formación técnica formal. En el contexto específico de la inteligencia artificial aplicada a la visualización de datos como proponen Yang et al. (2024) y He et al. (2025), los LLM han emergido como una tecnología verdaderamente disruptiva que promete revolucionar la forma en que los usuarios interactúan con información compleja. Según demuestra la investigación reciente de Wu et al. (2024), los modelos como GPT-4 (OpenAI, 2023) han alcanzado capacidades notables para interpretar descripciones expresadas en lenguaje natural y generar código de visualización apropiado de manera automática, abriendo posibilidades completamente nuevas para la democratización del análisis de datos. Esta capacidad representa un cambio paradigmático fundamental, ya que tradicionalmente la creación de visualizaciones efectivas ha requerido no solamente conocimientos técnicos sobre herramientas específicas, sino también una comprensión profunda de principios de diseño visual y cognición humana. Paralelamente a estos desarrollos en inteligencia artificial, el paradigma de Linked Open Data (LOD) y la Web

Semántica han madurado significativamente hasta convertirse en estándares de facto para la publicación y conexión de datos científicos en Internet. Como señalan Bizer et al. (2011) en su trabajo seminal sobre la evolución de LOD, este enfoque permite que los recursos de datos se identifiquen de manera única mediante Uniform Resource Identifiers y que las relaciones entre ellos se definan explícitamente mediante vocabularios estandarizados del Resource Description Framework, facilitando así la interoperabilidad y reutilización de datos a escala web. Esta arquitectura técnica ha demostrado ser particularmente valiosa en dominios científicos donde la integración de múltiples fuentes de información heterogéneas resulta esencial para avanzar en el conocimiento. La intersección de estas dos tecnologías emergentes —los LLM por un lado y los datos enlazados por el otro— representa un área de investigación particularmente prometedora desde la perspectiva de las Ciencias de la Computación. Sin embargo, a pesar del potencial evidente, persisten desafíos fundamentales que requieren investigación rigurosa. La automatización inteligente de la visualización constituye uno de estos desafíos cruciales, ya que la selección apropiada de técnicas de visualización requiere comprender no solamente la estructura sintáctica de los datos, sino también la semántica profunda del dominio y las tareas analíticas específicas que el usuario intenta realizar, como han documentado extensamente Peña et al. (2016) y Po et al. (2020) en sus revisiones del estado del arte. El procesamiento eficiente de datos enlazados complejos representa otro desafío computacional significativo. Los grafos de conocimiento, que constituyen la estructura fundamental de LOD, representan datos multidimensionales con relaciones complejas que no se ajustan fácilmente a los modelos tabulares tradicionales utilizados por la mayoría de las herramientas de visualización convencionales. Como argumentan Brunetti et al. (2013) en su modelo formal de visualización de datos enlazados, esta incompatibilidad estructural requiere nuevos enfoques arquitectónicos y algorítmicos. Además, la usabilidad para usuarios no técnicos permanece como un obstáculo persistente, dado que las tecnologías semánticas tradicionalmente han requerido conocimientos especializados en lenguajes de consulta como SPARQL, modelado ontológico y el ecosistema RDF en general, limitando severamente su adopción fuera de la comunidad especializada de la Web Semántica, tal como documenta el estudio de Dadzie y Rowe

(2011). Esta tesis doctoral aborda estos desafíos desde una perspectiva fundamentalmente informática, proponiendo soluciones de arquitectura modular y desarrollo de métodos de ciencia de datos que integran técnicas avanzadas de inteligencia artificial con principios establecidos de visualización efectiva y tecnologías maduras de datos enlazados. Para validar empíricamente estas contribuciones teóricas y demostrar su aplicabilidad práctica, se utilizan como dominio de aplicación las ciencias del mar, un campo científico que presenta características particularmente desafiantes desde el punto de vista computacional. Este dominio se caracteriza por una heterogeneidad de datos que abarca variables de biodiversidad, información biológica sobre especies marinas, coordenadas geoespaciales y series temporales complejas, como describen Hardisty y Roberts (2013) en su análisis de la informática de biodiversidad. El volumen de datos generados en ciencias marinas crece exponencialmente debido a la proliferación de múltiples plataformas de observación remota satelital y sensores in-situ distribuidos, fenómeno documentado por Xie et al. (2019) en su estudio sobre tecnologías de observación oceánica. La elección de las ciencias del mar como dominio de validación no es en absoluto arbitraria ni conveniente, sino estratégicamente fundamentada. Este campo representa un caso de estudio ideal para evaluar sistemas de visualización inteligente debido a varios factores convergentes. Primero, la complejidad inherente y diversidad de tipos de datos requiere exactamente el tipo de razonamiento sofisticado que los sistemas basados en inteligencia artificial prometen proporcionar. Segundo, la disponibilidad de iniciativas de datos abiertos, como por ejemplo Pampa Azul en Argentina, programa que aunque en la actualidad se encuentra discontinuado por razones de financiamiento, sin embargo tuvo como objetivo coordinar esfuerzos nacionales de investigación marina, y por otro lado, la iniciativa GeoLink en Estados Unidos que integra repositorios de geociencias, proporciona acceso a datasets reales y complejos para validación empírica rigurosa, como documentan Cheatham et al. (2018) y Zárate et al. (2020). Tercero, la diversidad de usuarios potenciales en este dominio permite evaluar la usabilidad del sistema en condiciones realistas con stakeholders auténticos. El desarrollo de sistemas de recomendación de visualización basados en inteligencia artificial se alinea perfectamente con prioridades estratégicas identificadas a nivel global en las Ciencias

de la Computación. El plan estratégico de CONICET para temas de transformación digital, publicado en 2021, identifica explícitamente la inteligencia artificial como una disciplina fundamental para abordar problemas complejos que requieren aproximaciones genuinamente interdisciplinarias. Específicamente, esta investigación doctoral realiza contribuciones sustanciales en cuatro áreas prioritarias del desarrollo informático contemporáneo. En primer lugar, representa un avance significativo en inteligencia artificial aplicada mediante el desarrollo de sistemas híbridos que combinan razonamiento simbólico tradicional, implementado a través de ontologías formales y grafos de conocimiento estructurados, con técnicas modernas de aprendizaje automático materializadas en LLM para tareas de recomendación contextual. En segundo lugar, contribuye al campo de la ingeniería de software para datos científicos mediante el diseño e implementación de arquitecturas modulares y escalables que permiten el procesamiento, análisis y visualización de datos en tiempo real. En tercer lugar, avanza el estado del arte en interfaces humano-computadora inteligentes al proponer sistemas adaptativos que reducen significativamente la carga cognitiva del usuario mediante automatización inteligente, pero sin sacrificar el control del usuario ni la transparencia del proceso de razonamiento subyacente, un balance delicado que representa uno de los desafíos centrales en el diseño de sistemas basados en inteligencia artificial. Finalmente, proporciona una implementación práctica de los principios FAIR para datos científicos. Por lo que brindan un marco conceptual que enfatiza que los datos deben ser localizables (Findable) mediante identificadores persistentes, accesibles (Accesible) a través de protocolos estandarizados, interoperables (Interoperable) mediante el uso de vocabularios compartidos y reutilizables (Reusable) gracias a licencias claras y metadatos ricos, como bien articulan Wilkinson et al. (2016) en su propuesta que ha sido adoptada por agencias de financiamiento científico a nivel mundial. Devaraju et al. (2019) analiza la relación entre repositorios científicos y artículos mediante metadatos FAIR, conectando infraestructura de datos abiertos. Finalmente, Malik y Foster (2012) exponen los desafíos en la gestión y compartición de grandes volúmenes de datos científicos distribuidos.

1.2. Planteamiento del Problema

A pesar de los avances significativos y continuos tanto en el campo de la visualización de datos como en el desarrollo de tecnologías semánticas durante las últimas dos décadas, persiste una brecha considerable y problemática entre las capacidades técnicas teóricamente disponibles en laboratorios de investigación y su adopción efectiva por parte de científicos practicantes y analistas de datos trabajando en problemas reales. Esta brecha entre potencial y realización no es meramente un problema de difusión tecnológica, sino que refleja limitaciones fundamentales en cómo se diseñan, implementan y evalúan las herramientas computacionales para análisis de datos. La brecha de automatización representa quizás la manifestación más visible de este problema sistémico. Las herramientas tradicionales de visualización, incluyendo plataformas comerciales, así como bibliotecas de fuente abierta que permiten máxima flexibilidad, imponen requisitos cognitivos y técnicos sobre sus usuarios. Estos sistemas requieren que los usuarios posean conocimiento previo sustancial sobre qué tipos de gráficos resultan apropiados para diferentes combinaciones de variables y tareas analíticas, una forma de metaconocimiento que típicamente se adquiere sólo a través de años de experiencia práctica. Además, los usuarios deben comprender profundamente la estructura interna de sus propios datos, incluyendo tipos de variables, rangos de valores, presencia de valores faltantes y relaciones estructurales entre diferentes atributos. Finalmente, deben realizar manualmente el proceso complejo de transformación de datos crudos y mapeo de variables específicas a encodings visuales concretos como posiciones en ejes, colores, tamaños o formas, un proceso que involucra múltiples decisiones técnicas donde una decisión equivocada puede producir visualizaciones erróneas. La investigación de Chen et al. (2022) demuestra que incluso usuarios experimentados cometen errores sistemáticos y predecibles en la selección de visualizaciones apropiadas, particularmente cuando los datos presentan características inusuales. Estos hallazgos motivan fuertemente el desarrollo de sistemas de recomendación inteligente que puedan asistir incluso a usuarios expertos, no para reemplazar su juicio sino para complementar su experiencia con análisis computacional sistemático de patrones en los datos y mejores prácticas

documentadas en la literatura de visualización. La complejidad inherente de los datos introduce dimensiones adicionales de dificultad que las herramientas tradicionales simplemente no están equipadas para manejar. La naturaleza fundamental del grafo que caracteriza a LOD difiere profundamente de las estructuras tabulares que constituyen el modelo de datos asumido por la vasta mayoría de software de análisis. Como señalan persuasivamente Brunetti et al. (2013) en su análisis formal, la visualización efectiva de LOD debe abordar simultáneamente varios desafíos técnicos sin precedentes en visualización tradicional. Los datasets enlazados típicamente emplean múltiples vocabularios ontológicos diferentes incluso dentro de un solo dataset, lo que significa que el mismo concepto puede ser representado usando términos de diferentes esquemas de metadatos, requiriendo reconciliación semántica sofisticada. Los recursos individuales frecuentemente incluyen enlaces externos que referencian entidades definidas en grafos de conocimiento completamente separados y mantenidos por otras organizaciones, creando una red distribuida de información que debe ser navegada y potencialmente integrada para análisis completo. Las propiedades en datos enlazados pueden ser genuinamente multivaluadas, donde un solo atributo contiene múltiples valores de diferentes tipos de datos simultáneamente, una característica que viola las asunciones fundamentales de bases de datos relacionales. Finalmente, las jerarquías taxonómicas y las relaciones de subsunción entre clases ontológicas crean estructuras de generalización-especialización complejas que deben ser respetadas en cualquier visualización que pretenda preservar la semántica original de los datos. Estas características estructurales hacen que las técnicas de visualización clásicas desarrolladas para datos tabulares resulten no solamente subóptimas sino frecuentemente inadecuadas o directamente engañosas cuando se aplican ingenuamente a datos enlazados. Se requieren enfoques fundamentalmente nuevos que respeten la semántica rica codificada en ontologías mientras mantienen la interpretabilidad visual para usuarios humanos. La barrera de entrada para usuarios no expertos constituye otro obstáculo crítico que limita severamente la adopción de tecnologías de datos enlazados fuera de círculos especializados. La Web Semántica y sus tecnologías asociadas han desarrollado un ecosistema técnico sofisticado que, aunque poderoso, presenta una curva de aprendizaje pronunciada. El lenguaje de

consulta SPARQL, que permite extraer información de grafos RDF, requiere que los usuarios comprendan conceptos de lógica de grafos, sintaxis de triple patterns y operadores de consulta especializados que difieren significativamente de SQL y otros lenguajes más familiares. El modelado de conocimiento mediante ontologías formales demanda comprensión de lógicas descriptivas, razonamiento automático y principios de diseño ontológico que típicamente requieren formación especializada. Incluso conceptos aparentemente básicos como la distinción entre recursos, literales y blank nodes en RDF pueden resultar confusos para usuarios sin background en representación de conocimiento. Como documenta extensamente el estudio de Dadzie y Rowe (2011), estas barreras técnicas han confinado efectivamente el uso de tecnologías semánticas a una comunidad relativamente pequeña de expertos, impidiendo la realización del potencial completo de la Web de Datos. Los desafíos específicos que presentan las ciencias del mar como dominio de aplicación ilustran vívidamente por qué este campo constituye un caso de prueba particularmente exigente para sistemas de visualización inteligente. Los datos geoespaciales marinos requieren visualización precisa de información georeferenciada con coordenadas de latitud y longitud, pero frecuentemente también incluyen componentes verticales de profundidad que añaden una tercera dimensión espacial crítica para procesos oceanográficos. Las series temporales multivariadas capturan la evolución simultánea de docenas de parámetros oceánicos físicos, químicos y biológicos a través del tiempo, creando datasets de alta dimensionalidad que desafían técnicas de visualización convencionales. Los perfiles verticales de la columna de agua pueden contener mediciones de treinta o más variables diferentes a múltiples profundidades, desde la superficie hasta miles de metros de profundidad, generando estructuras de datos complejas que combinan aspectos espaciales, temporales y multivariados. La heterogeneidad de sensores utilizados en oceanografía observacional, cada uno con diferentes precisiones, frecuencias de muestreo, protocolos de calibración y posibles fuentes de error, complica adicionalmente tanto el análisis como la visualización de datos integrados. Estos desafíos multifacéticos motivan la pregunta de investigación principal que guía esta tesis doctoral:

RQ1 *¿Cómo puede la integración de modelos de lenguaje extensos (LLM), principios de*

visualización efectiva y tecnologías de datos enlazados generar sistemas de recomendación automática de visualizaciones que sean precisos, interpretables y accesibles para usuarios con diversos niveles de expertise?.

Esta pregunta principal se descompone en otras preguntas secundarias más específicas que estructuran la investigación.

Primero, desde una perspectiva de prompt engineering y configuración de modelos de lenguaje, **RQ2** *¿Qué estrategias específicas de diseño de prompts, estructuración de contexto y configuración de parámetros de LLMs producen recomendaciones adecuadas de visualización efectiva?*

Segundo, desde una perspectiva de evaluación para tareas analíticas específicas documentadas en taxonomías como la de Amar et al. (2005), surge la **RQ3** *¿Qué metodologías son adecuadas para evaluar la calidad de visualizaciones recomendadas automáticamente?*

1.3. Objetivos de la Investigación

1.3.1. Objetivo General

El objetivo general que guía esta investigación doctoral es desarrollar, implementar y validar empíricamente un sistema completo de recomendación inteligente de visualizaciones que integre de manera innovadora tres componentes tecnológicos principales: LLM de última generación para procesamiento de lenguaje natural y razonamiento contextual, principios establecidos y validados de analítica visual efectiva derivados de investigación en percepción humana y cognición, y capacidades avanzadas de procesamiento de LOD que permitan manejar la complejidad semántica de grafos de conocimiento, como bien explican Vandenbussche et al. (2017). La validación de este sistema debe ser rigurosa y multifacética, combinando evaluaciones cuantitativas automatizadas que midan precisión y relevancia objetivas con estudios de usabilidad controlados que involucren usuarios reales con diversos perfiles de experiencia, desde estudiantes sin formación técnica hasta profesionales con

experiencia en análisis de datos, todos trabajando con datos científicos reales del dominio marino que presenta los desafíos de complejidad y heterogeneidad característicos de aplicaciones científicas modernas.

1.3.2. Objetivos Específicos

El primer objetivo específico de esta investigación se enfoca en la investigación exhaustiva del estado del arte mediante una revisión sistemática y crítica de la literatura existente. Esta revisión debe abarcar tres cuerpos de literatura distintos pero interrelacionados: la investigación sobre visualización de datos enlazados que ha evolucionado en la última década, identificando no solamente las técnicas actuales y herramientas disponibles sino crucialmente las limitaciones documentadas y oportunidades de mejora que permanecen sin abordar satisfactoriamente. Paralelamente, debe analizarse rigurosamente el estado del arte en sistemas de recomendación de visualización, con énfasis particular en enfoques relativamente recientes basados en aprendizaje automático y especialmente en técnicas de deep learning aplicadas a tareas de visualización. Finalmente, este objetivo incluye evaluar herramientas emergentes que integran LLM para generación automática de visualizaciones, incluyendo sistemas como LIDA desarrollado por Dibia (2023), LLM4Vis propuesto por Wang et al. (2023), ChartGPT propuesto por Tian et al. (2024) que enfatiza accesibilidad, Chat2VIS que permite consultas en lenguaje natural como describen Maddigan y Susnjak (2023), y V-RECS (Podo et al., 2024), que combina recomendaciones y justificación de las mismas. Esta evaluación no debe limitarse a describir capacidades, sino identificar específicamente gaps en funcionalidad, limitaciones de usabilidad y oportunidades para contribuciones originales. El segundo objetivo específico consiste en diseñar una arquitectura modular que: (a) integre procesamiento de datos, (b) incorpore análisis mediante GPT-4 (OpenAI, 2023) con tiempos de respuesta inferiores a 5 segundos, y (c) genere visualizaciones interactivas en formato estándar con descripción de las recomendaciones. Esta arquitectura se propone fundamentalmente modular, permitiendo que componentes individuales sean desarrollados, probados y potencialmente reemplazados independientemente mientras mantienen interfaces bien definidas. La arquitectura debe integrar tres

subsistemas principales: un módulo de procesamiento de LOD capaz de consumir datos en múltiples formatos RDF, un motor de análisis mediante LLMs que razone sobre estructura de datos y tareas del usuario, y un generador de visualizaciones interactivas que produzca gráficos finales. El diseño debe especificar pipelines eficientes para cuatro procesos secuenciales críticos: análisis exploratorio automatizado de datasets que identifique características estadísticas y estructurales relevantes, detección automática de tipos de datos y patrones que clasifique variables y descubra relaciones, recomendación contextual de visualizaciones que considere tanto los datos como las intenciones del usuario, y generación y customización interactiva de gráficos que permita refinamiento iterativo. Crucialmente, la arquitectura debe implementar mecanismos sofisticados de caching y optimización que garanticen tiempos de respuesta aceptables por debajo de cinco segundos desde la carga de datos hasta la presentación de recomendaciones, un requisito esencial para mantener la fluidez de la experiencia de usuario en aplicaciones interactivas. El tercer objetivo específico se enfoca en el desarrollo de un prototipo funcional completo, que integre la arquitectura propuesta denominado VisBot (Ceballos, 2025), que materialice concretamente la arquitectura propuesta en un sistema ejecutable y evaluable. Este desarrollo debe implementar un sistema web completamente interactivo accesible mediante navegadores estándares sin requerir instalación de software especializado, maximizando así la accesibilidad. El sistema debe desarrollar capacidades robustas de procesamiento automático tanto de datos tabulares convencionales en formatos como CSV y Excel como de grafos RDF expresados en notaciones como RDF/XML, o JSON-LD, manejando apropiadamente variaciones y errores comunes en datos reales. La integración con GPT-4 (OpenAI, 2023) debe implementarse mediante estrategias cuidadosamente diseñadas de prompt engineering, incluyendo la estructuración óptima de mensajes del sistema que establezcan el rol del modelo como experto en análisis de datos especializado en visualización, la inclusión de contexto apropiado sobre el dataset siendo analizado incluyendo esquema y estadísticas descriptivas, y el formateo de consultas que expliciten respuestas estructuradas parseables automáticamente. El módulo de generación de visualizaciones debe utilizar la biblioteca estándar implementando soporte para múltiples tipos de gráficos incluyendo

scatter plots, gráficos de barras, líneas, histogramas, box plots y bubble charts, con opciones de customización interactiva que permite a usuarios ajustar ejes, colores, tamaños de puntos y otros parámetros visuales en tiempo real. Particular atención debe prestarse al soporte especializado para datos geoespaciales, implementando mapas mediante scatter_mapbox y scatter_geo que permitan visualizar distribuciones geográficas con precisión, una capacidad crítica para aplicaciones en ciencias marinas donde la dimensión espacial resulta fundamental. El cuarto objetivo específico aborda la evaluación cuantitativa sistemática del sistema implementado mediante el diseño y aplicación de un conjunto comprehensivo de métricas de evaluación. Estas métricas deben capturar múltiples dimensiones de calidad incluyendo relevancia medida en escala Likert que evalúa la alineación entre visualizaciones recomendadas y datos o tareas analíticas, diversidad que cuantifica la variedad de tipos de gráficos sugeridos para evitar monotonía, Normalized Discounted Cumulative Gain calculado sobre las cinco primeras recomendaciones para evaluar la calidad del ranking, tasa de éxito que mide el porcentaje de tareas completadas exitosamente por usuarios siguiendo recomendaciones del sistema, precisión semántica que evalúa la correctitud en la representación de relaciones semánticas presentes en los datos originales, correctitud de mapeos que verifica la apropiación de encodings visuales seleccionados según principios establecidos, y tasa de error que cuantifica la frecuencia de gráficos inválidos técnicamente o potencialmente misleading perceptualmente. La evaluación debe realizarse sobre datasets estandarizados que incluyan entidades geográficas, personas y organizaciones con sus relaciones y que permitan comparación con otros sistemas, incluyendo datos oceanográficos reales del Golfo San Jorge, y datasets de uso genérico pero estandarizados en el testeo de aplicaciones de manipulación de datos. Los resultados cuantitativos deben compararse sistemáticamente con herramientas existentes en dimensiones clave como calidad de interfaz gráfica de usuario, nivel de customización ofrecido y requisitos de habilidades técnicas del usuario. El quinto objetivo específico se centra en la validación externa mediante estudios de usabilidad controlados que involucren usuarios reales representativos de las poblaciones objetivo del sistema. Estos estudios deben conducirse con rigor metodológico, reclutando aproximadamente cincuenta participantes estratificados en tres grupos que

representen diferentes perfiles de experiencia: un primer grupo de veinte estudiantes universitarios con experiencia mínima en análisis de datos o visualización, un segundo grupo de veinte profesionales con conocimientos básicos de análisis de datos adquiridos en sus actividades laborales, y un tercer grupo de diez expertos intermedios que han utilizado herramientas profesionales como Tableau o Power BI pero carecen de experiencia específica en sistemas basados en inteligencia artificial. El protocolo experimental debe medir múltiples dimensiones de usabilidad, además de estas medidas subjetivas, deben analizarse métricas objetivas como tiempos de completación de tareas específicas y tasas de éxito en escenarios analíticos realistas diseñados para reflejar casos de uso auténticos. Finalmente, debe recopilarse feedback cualitativo rico mediante encuestas estructuradas con preguntas abiertas y entrevistas semi-estructuradas que permitan a los participantes articular sus experiencias, identificar fortalezas y debilidades del sistema, y sugerir mejoras desde sus perspectivas únicas. El sexto y último objetivo específico aborda la aplicación práctica y transferencia tecnológica al dominio de las ciencias del mar, demostrando el valor del sistema desarrollado en un contexto científico real con impacto social tangible. Este objetivo requiere adaptar el sistema para manejar eficientemente datos oceanográficos específicos del Golfo San Jorge, una región marina de importancia ecológica y económica crítica para Argentina, así como datos de otras áreas prioritarias identificadas por la iniciativa, actualmente discontinuada, Pampa Azul que coordinaba investigación marina estratégica. La adaptación debe demostrar capacidades concretas de integración con grafos de conocimiento existentes como OceanGraph que estructura información sobre expediciones oceanográficas, variables ambientales y publicaciones científicas. Deben documentarse exhaustivamente casos de uso representativos que evidencien utilidad práctica tangible para investigadores marinos trabajando en problemas reales, desde la exploración inicial de datos de campañas hasta el análisis de relaciones entre variables físicas y distribución de especies. Finalmente, debe evaluarse cuidadosamente el potencial de transferencia tecnológica del sistema más allá del proyecto de investigación, considerando aspectos de sostenibilidad post-investigación, accesibilidad para instituciones con recursos limitados, y posibilidades de adopción por parte de la comunidad científica marina más

amplia tanto en Argentina como internacionalmente.

1.4. Metodología

Esta investigación doctoral adopta un enfoque metodológico fundamentalmente basado en diseño, conocido en la literatura anglosajona como Design-Based Research, un paradigma que resulta particularmente apropiado para desarrollos en Ciencias de la Computación que buscan simultáneamente dos tipos de contribuciones complementarias: contribuciones teóricas que avanzan el entendimiento conceptual de fenómenos computacionales y contribuciones artefactuales que producen sistemas funcionantes que resuelven problemas prácticos reales. Como argumenta persuasivamente Hevner et al. (2004) en su influyente artículo sobre investigación en sistemas de información, este enfoque dual permite que el desarrollo del artefacto informe y refine la teoría mientras que la teoría guía y justifica decisiones de diseño, creando un ciclo virtuoso entre teoría y práctica. La metodología concreta se estructura en cinco fases principales que se ejecutan de manera iterativa más que estrictamente secuencial, permitiendo que los hallazgos de fases posteriores informen retroactivamente el refinamiento de productos de fases anteriores. La primera fase del proyecto doctoral, se dedica a la investigación documental exhaustiva y el análisis sistemático de requisitos que fundamentan todo el desarrollo posterior. Las actividades centrales de esta fase incluyen una revisión sistemática de la literatura especializada en visualización de datos enlazados, siguiendo metodologías establecidas como la propuesta por Kitchenham para revisiones sistemáticas en ingeniería de software. Esta revisión debe sintetizar hallazgos de trabajos seminales como el survey comprehensivo de Peña et al. (2016) que examina enfoques para la visualización de LOD y la monografía técnica de Po et al. (2020) que analiza herramientas de visualización para Linked Data y Big Data. Paralelamente, debe realizarse un análisis comparativo riguroso de herramientas contemporáneas basadas en LLMs para visualización, examinando no solamente su funcionalidad superficial sino sus arquitecturas subyacentes, decisiones de diseño, fortalezas demostrado desde el mes treinta y uno hasta el treinta y seis, se dedica al análisis sistemático, síntesis de resultados y documentación comprehensiva de toda la investigación. Esta fase representa la convergencia de todo el trabajo previo

en contribuciones articuladas claramente que avancen en el campo de las Ciencias de la Computación. Las actividades centrales comienzan con análisis estadísticos rigurosos de los datos cuantitativos recopilados durante la fase de evaluación. Se aplican técnicas inferenciales apropiadas incluyendo análisis de varianza multifactorial para comparar rendimiento entre diferentes grupos de usuarios, pruebas post-hoc como la diferencia honestamente significativa de Tukey (1977) para identificar diferencias específicas significativas entre pares de grupos cuando el análisis de varianza revela efectos globales, y modelado de regresión múltiple para explorar relaciones entre variables continuas como experiencia previa del usuario y latencias de completación de tareas. Se calculan intervalos de confianza para todas las estimaciones de efectos para cuantificar incertidumbre estadística, y se reportan magnitudes de efecto además de significancia estadística para facilitar interpretación práctica de resultados según estándares de la American Psychological Association. Paralelamente, se conduce análisis cualitativo sistemático de las transcripciones de entrevistas y respuestas a interrogantes abiertas mediante metodologías establecidas de investigación cualitativa interpretativa. Se aplica codificación temática iterativa donde dos codificadores independientes revisan el material textual identificando temas recurrentes, constructos emergentes y patrones de experiencia compartidos entre participantes. Los códigos iniciales se refinan mediante discusión dialéctica entre codificadores hasta alcanzar consenso sobre un libro de códigos final que captura las dimensiones principales de la experiencia del usuario. Se calcula concordancia inter-codificador usando el coeficiente kappa de Cohen para verificar la fiabilidad del esquema de codificación. Los temas identificados se organizan jerárquicamente en categorías de nivel superior que estructuran los hallazgos cualitativos de manera coherente mediante redes temáticas. La triangulación metodológica constituye un procedimiento analítico crítico que integra hallazgos cuantitativos y cualitativos para desarrollar una comprensión holística de la efectividad del sistema. Se identifican convergencias donde evidencia cuantitativa y cualitativa convergen hacia las mismas inferencias, fortaleciendo la validez de esos hallazgos mediante corroboración múltiple. También se examinan divergencias donde diferentes fuentes de evidencia sugieren interpretaciones conflictivas, explorando posibles explicaciones para estas discrepancias que frecuentemente revelan matices

importantes sobre el fenómeno estudiado. Esta síntesis integrada informa el diseño de futuras iteraciones del sistema mediante desarrollo iterativo y contribuye a teoría generalizable sobre diseño de interfaces de inteligencia artificial para análisis de datos. La identificación explícita de limitaciones metodológicas y amenazas a la validez representa un componente esencial de investigación rigurosa que demuestra pensamiento crítico y autocrítica epistémica. Se examinan sistemáticamente amenazas a la validez interna considerando explicaciones alternativas plausibles para resultados observados y factores de confusión potenciales, validez externa analizando en qué medida hallazgos pueden generalizarse más allá del contexto específico del estudio hacia otras poblaciones de usuarios o dominios de aplicación, validez de constructo evaluando si los instrumentos utilizados realmente operacionalizar los constructos teóricos que pretenden medir mediante correspondencia epistemológica, y validez estadística verificando que supuestos paramétricos de pruebas estadísticas se cumplan y que el tamaño muestral proporciona potencia estadística adecuada para detectar efectos de magnitud prácticamente significativa. Esta fase también incluye la redacción de manuscritos científicos para publicación en foros académicos de alto impacto, un objetivo fundamental para diseminar contribuciones de la investigación hacia la comunidad científica internacional. Se preparan artículos enfocados en diferentes aspectos del trabajo, como un artículo sobre la arquitectura de integración entre LLM y datos enlazados dirigido a conferencias especializadas en Web Semántica, un artículo sobre evaluación empírica de usabilidad para foros de Interacción Humano-Computadora, y potencialmente un artículo de aplicación en el dominio marino para revistas especializadas en oceanografía computacional. Cada manuscrito se adapta cuidadosamente a las convenciones epistémicas y discursivas de su audiencia objetivo mientras mantiene rigor metodológico y precisión expositiva. La escritura de la disertación doctoral propiamente dicha representa la culminación del trabajo académico, sintetizando todos los componentes en un documento coherente que construye un argumento convincente desde la problematización inicial hasta las contribuciones finales y direcciones prospectivas. La escritura sigue convenciones establecidas de tesis en Ciencias de la Computación incluyendo exposición precisa de la problemática y contribuciones en la introducción, revisión crítica de literatura

relacionada que identifica vacíos epistémicos específicos, descripción detallada de métodos que permita reproducibilidad mediante especificación completa, presentación estructurada de resultados distinguiendo claramente observaciones empíricas de interpretaciones teóricas, y discusión reflexiva que contextualiza hallazgos dentro del campo más amplio reconociendo honestamente limitaciones junto con fortalezas metodológicas. Los productos concretos de esta fase final incluyen la disertación doctoral completa conforme a normativas de la institución otorgante del grado, artículos científicos sometidos o idealmente publicados en revistas o conferencias internacionales con arbitraje por pares, presentaciones preparadas para conferencias académicas donde los resultados serán diseminados hacia la comunidad científica, y crucialmente, el conjunto completo de datos de evaluación y código fuente publicados bajo licencias de código abierto apropiadas en repositorios persistentes como Zenodo o Figshare con identificadores de objeto digital que permitan citación y reproducibilidad por otros investigadores, contribuyendo así a la ciencia abierta y acumulativa. Las consideraciones deontológicas atraviesan todas las fases metodológicas asegurando que la investigación se conduzca con integridad científica y respeto por participantes humanos conforme a declaraciones internacionales de bioética. El consentimiento informado se obtiene de todos los participantes en estudios de usabilidad mediante formularios escritos que explican en lenguaje comprensible los objetivos de la investigación, los procedimientos específicos que experimentarán incluyendo captura de pantalla y audio, los riesgos mínimos anticipados que en este caso son principalmente incomodidad leve por incapacidad de completar tareas, y garantías de confidencialidad sobre manejo de sus datos. Los participantes se informan explícitamente que su participación es completamente voluntaria y pueden retirarse en cualquier momento sin consecuencia alguna. La privacidad de los participantes se protege mediante anonimización rigurosa donde todos los datos personales identificables se eliminan o reemplazan por identificadores alfanuméricos únicos que no pueden ser vinculados a individuos específicos sin una clave criptográfica mantenida separadamente de los datos. La transparencia sobre capacidades y limitaciones del sistema se mantiene durante interacciones con usuarios, evitando crear expectativas infladas sobre precisión o capacidades que el sistema no

posee. Finalmente, el compromiso con datos abiertos se cumple utilizando exclusivamente conjuntos de datos que son públicamente disponibles o para los cuales se han obtenido permisos explícitos de uso en investigación, y publicando resultados y código de manera que permita escrutinio y reproducción por la comunidad científica internacional.

1.5. Estructura de la Tesis

La presente tesis se organiza en seis capítulos, los cuales se estructuran de manera lógica y progresiva con el propósito de conducir al lector desde los fundamentos teóricos y metodológicos hasta la validación empírica del sistema propuesto y la exposición de sus proyecciones futuras. Cada capítulo se encuentra articulado con los objetivos generales y específicos planteados, conformando un desarrollo coherente y sistemático del proceso de investigación. Capítulo 1, presenta el marco general de la investigación, exponiendo la motivación, la justificación y los antecedentes que dieron origen al estudio. Se explicitan el problema de investigación, los objetivos generales y específicos, las preguntas de investigación y la metodología adoptada. Asimismo, se describen las fases de desarrollo y se anticipa la estructura del documento en su conjunto. Capítulo 2, expone los fundamentos conceptuales y epistemológicos que sustentan la investigación, abordando las nociones esenciales vinculadas a la visualización científica, los datos enlazados y la Web Semántica, y los LLM. Se realiza una revisión crítica de la literatura especializada y se establecen las bases teóricas que orientan el diseño del sistema propuesto, justificando además la elección del dominio marino como contexto de aplicación. Capítulo 3, desarrolla un análisis comparativo y crítico de los principales enfoques y herramientas existentes en el campo de la analítica visual y la inteligencia artificial aplicada. Se examinan tanto soluciones tradicionales como sistemas recientes basados en modelos de lenguaje, identificando limitaciones, oportunidades y vacíos de conocimiento. Este relevamiento permite delinear las áreas de innovación sobre las cuales se sustenta el aporte original de la presente investigación. Capítulo 4, constituye el núcleo empírico y experimental del trabajo. Se describe en detalle la arquitectura, el diseño funcional y la implementación del sistema denominado *VisBot (Ceballos, 2025)*, el cual integra modelos de lenguaje con datos

enlazados para la recomendación automatizada de visualizaciones científicas. Se documenta el proceso de desarrollo, la metodología de integración tecnológica, los mecanismos de evaluación y los casos de uso validados con datos reales del ámbito de la investigación oceanográfica. Se enfatiza la aplicabilidad práctica y la viabilidad del sistema en entornos científicos con recursos tecnológicos limitados. Capítulo 5, presenta una reflexión crítica sobre las principales limitaciones y desafíos observados durante el desarrollo de la investigación. Se analizan aspectos técnicos, metodológicos y operativos, tales como la escalabilidad, la dependencia de modelos externos, la interpretación semántica y los costos de procesamiento. Asimismo, se discuten los desafíos humanos e institucionales vinculados a la adopción de herramientas inteligentes en comunidades científicas, y se proponen estrategias para su abordaje y superación. Capítulo 6, cierra el recorrido investigativo mediante la síntesis de los resultados obtenidos y la valoración de los aportes originales de la tesis en los planos teórico, metodológico y tecnológico. Se reflexiona sobre el impacto potencial del sistema *VisBot (Ceballos, 2025)* en la democratización del acceso a la analítica visual inteligente, destacando su contribución a la integración de datos enlazados, lenguaje natural e inteligencia artificial en contextos científicos. Finalmente, se delinean líneas futuras de trabajo, orientadas a la incorporación de modelos multimodales, la ampliación del soporte multilingüe, la integración con entornos colaborativos de análisis y la optimización del rendimiento en grafos de gran escala. Este último capítulo consolida el cierre del proceso de investigación y proyecta su continuidad hacia desarrollos posteriores. La tesis concluye con Referencias Bibliográficas exhaustivas citando toda la literatura consultada y citada a lo largo del documento según normas de citación bibliográfica establecidas, facilitando que lectores accedan a fuentes primarias y verifiquen afirmaciones realizadas.

CAPÍTULO

2 MARCO TEÓRICO

Este capítulo desarrolla los fundamentos conceptuales que sustentan la investigación doctoral mediante revisión exhaustiva y síntesis crítica de tres cuerpos de literatura interdependientes. La estructura del capítulo responde a la naturaleza multidisciplinar del problema abordado, requiriendo integración de conocimientos desde visualización científica, ciencias marinas computacionales y tecnologías emergentes de inteligencia artificial. La primera sección examina fundamentos teóricos y empíricos de visualización de información desde perspectivas de percepción visual, cognición gráfica y diseño de sistemas interactivos. La segunda sección caracteriza el dominio de aplicación de ciencias del mar, documentando propiedades específicas de datos oceanográficos que presentan desafíos computacionales particulares y justifican selección de este campo como caso de validación empírica. La tercera sección revisa tecnologías emergentes que constituyen pilares técnicos del sistema propuesto, incluyendo LLM, paradigmas de datos enlazados y sistemas de recomendación basados en aprendizaje automático. La integración de estos tres dominios proporciona marco conceptual unificado que fundamenta decisiones de diseño arquitectónico, algorítmico e implementación en capítulos subsecuentes.

2.1. Fundamentos de Visualización Científica

La visualización científica constituye una subdisciplina fundamental de las Ciencias de la Computación dedicada a la representación gráfica de datos abstractos para facilitar la comprensión humana de fenómenos complejos mediante la explotación de capacidades perceptuales y cognitivas del sistema visual. Esta área de investigación se fundamenta en convergencia de múltiples campos disciplinares incluyendo ciencias de la computación gráfica que proporcionan algoritmos y técnicas de renderizado,

psicología de la percepción que caracteriza cómo humanos procesan información visual, ciencias cognitivas que modelan razonamiento espacial y construcción de modelos mentales, y diseño de información que articula principios normativos para comunicación efectiva. El análisis exploratorio de datos, concepto fundamental articulado por Tukey (1977), establece la filosofía de examinar datos gráficamente antes de aplicar técnicas estadísticas formales. Esta aproximación enfatiza el uso de visualizaciones para descubrir patrones inesperados, detectar anomalías, verificar supuestos y generar hipótesis. Los sistemas contemporáneos de recomendación de visualización automatizan precisamente este proceso exploratorio inicial. La investigación en visualización ha evolucionado desde representaciones estáticas unidireccionales hacia sistemas interactivos bidireccionales que facilitan exploración dinámica y descubrimiento iterativo de patrones en datos complejos, transformando la visualización de medio pasivo de presentación en instrumento activo de análisis.

2.1.1. Percepción Visual

El sistema visual humano representa el canal sensorial de mayor ancho de banda para transmisión de información hacia el cerebro, procesando aproximadamente diez millones de bits por segundo según estimaciones derivadas de teoría de la información aplicada a fisiología sensorial (Ware, 2012). La corteza visual ocupa aproximadamente treinta por ciento de corteza cerebral humana comparado con ocho por ciento para corteza auditiva y tres por ciento para corteza somatosensorial, indicando asignación evolutiva de recursos computacionales cerebrales priorizando procesamiento visual. Esta arquitectura neuronal paralela decodifica la eficacia de representaciones visuales para interpretar información compleja más rápidamente que representaciones puramente textuales o numéricas que requieren procesamiento serial. Sin embargo, la mera existencia de alta capacidad de transmisión no garantiza automáticamente la comprensión efectiva de la información visualizada. El diseño de analíticos visuales debe considerar cuidadosamente características específicas del procesamiento visual humano identificadas en psicología experimental y neurociencia cognitiva, por ejemplo, como propone Wong et.al (2000). La teoría de características preatentivas (Treisman y Gelade, 1980) identifica propiedades gráficas detectables en paralelo, sin

atención focal serial. Estas características procesadas preatentivamente incluyen color definido por matiz, saturación y luminosidad, orientación de segmentos lineales detectada por células simples en corteza visual primaria sensibles a ángulos específicos, movimiento de objetos procesado por vía visual dorsal especializada en información temporal y espacial, tamaño de elementos medido como área o longitud, y forma básica distinguiendo círculos, cuadrados y triángulos mediante detectores de características tempranas. La importancia de características preatentivas para diseño de visualizaciones radica en que permiten detección instantánea de patrones sin búsqueda serial exhaustiva cuya latencia escala linealmente con número de elementos. Esta distinción fundamental entre procesamiento paralelo preatentivo versus serial atentivo informa decisiones cruciales de diseño visual sobre qué variables de datos codificar mediante qué canales visuales. Experimentos psicofísicos sistemáticos realizados por Cleveland y McGill (1984) establecieron jerarquía empírica de precisión perceptual para diferentes tipos de codificaciones visuales cuantitativas. El error de estimación incrementa progresivamente descendiendo esta jerarquía, con juicios de área presentando error típicamente dos veces mayor que juicios de posición, y juicios de saturación presentando error cinco veces mayor. Esta jerarquía empírica, posteriormente refinada y formalizada por Mackinlay (1986) en su sistema de generación automática de presentaciones gráficas APT (A Presentation Tool), proporciona principios normativos fundamentales para selección de codificaciones visuales apropiadas según tipo semántico de variables de datos. Variables cuantitativas que requieren comparaciones precisas de magnitudes deben codificarse mediante posición en ejes ortogonales cuando sea posible, maximizando precisión perceptual. Variables cuantitativas requiriendo únicamente estimaciones aproximadas de proporciones pueden utilizar longitud o ángulo con pérdida aceptable de precisión. Variables categóricas ordinales pueden codificarse mediante saturación o luminosidad de color que preserva orden perceptual mediante gradiente continuo. Variables categóricas nominales sin orden intrínseco deben codificarse mediante matiz de color seleccionando colores distinguibles o mediante forma de símbolos como círculos versus cuadrados, ya que estas dimensiones visuales carecen de ordenamiento perceptual natural evitando implicar orden espurio. La teoría de canales visuales

propuesta por Bertin (1967) en su tratado seminal Sémiologie Graphique identifica siete variables retinianas fundamentales que pueden manipularse independientemente para codificar información: posición en dos dimensiones espaciales horizontal y vertical, tamaño medido como longitud, área o volumen, valor definido como luminosidad en escala de negro a blanco, textura caracterizada por densidad y orientación de microestructura, color definido por matiz en espectro cromático, orientación de símbolos o patrones, y forma de marcas gráficas. Bertin caracteriza sistemáticamente cada canal según propiedades perceptuales formales derivadas de la observación empírica. La selectividad cuantifica la capacidad de aislar visualmente elementos con valor particular del canal ignorando elementos con otros valores, propiedad útil para búsqueda visual de subconjunto específico. La asociatividad mide tendencia de elementos con mismo valor de canal a ser percibidos espontáneamente como grupo cohesivo, facilitando segmentación perceptual de conjunto en clusters. El orden indica si valores de canal admiten ordenamiento perceptual natural sin ambigüedad, crucial para representar variables ordinales preservando relaciones de mayor-menor. La cuantitatividad evalúa si diferencias perceptuales en canal corresponden proporcionalmente a diferencias en valores de datos representados, necesario para encodings de variables cuantitativas de razón. Esta caracterización sistemática permite razonamiento formal sobre diseño de visualizaciones multivariadas donde múltiples variables deben codificarse simultáneamente mediante diferentes canales visuales sin crear interferencia perceptual o ambigüedad interpretativa. Por ejemplo, posición presenta alta selectividad permitiendo localizar rápidamente elementos en región específica, asociatividad moderada agrupando elementos en vecindades espaciales, orden fuerte en ambas dimensiones permitiendo juicios de mayor-menor, y cuantitatividad perfecta con correspondencia proporcional entre distancias visuales y diferencias de valores. En contraste, matiz de color presenta selectividad excelente permitiendo identificación inmediata de categoría específica, asociatividad fuerte agrupando elementos del mismo color, pero orden débil sin secuencia única natural entre colores diferentes y cuantitatividad ausente dado que diferencias perceptuales entre matices no corresponden a magnitudes numéricas. Estas propiedades formales guían selección de

asignaciones de variables a canales maximizando efectividad comunicativa.

2.1.2. Modelos de Transformación de Datos

El proceso complejo de transformar datos abstractos almacenados computacionalmente en representaciones visuales interpretables por humanos puede conceptualizarse mediante modelos de referencia que estructuran etapas de procesamiento involucradas y proporcionan vocabulario compartido para razonamiento sobre arquitecturas de sistemas. El modelo de estados de referencia de datos desarrollado por Chi (2000) conceptualiza la visualización como secuencia de transformaciones entre cuatro estados progresivamente más concretos. El estado inicial consiste en datos en valor original tal como son capturados por instrumentos de medición, almacenados en bases de datos o generados por simulaciones computacionales, típicamente representados en formatos binarios compactos o estructuras relacionales normalizadas optimizadas para almacenamiento eficiente más que visualización directa. El segundo estado consiste en datos analíticos derivados mediante operaciones de transformación, filtrado, agregación, interpolación o cálculo de cantidades derivadas que preparan información para visualización. Por ejemplo, una serie temporal de mediciones individuales puede transformarse calculando promedios móviles que suavizan fluctuaciones de alta frecuencia, o conjunto de puntos dispersos puede interpolarse generando campo continuo mediante técnicas geoestadísticas. El tercer estado consiste en datos de visualización que especifican geometrías y atributos gráficos abstractos independientes de tecnología de renderizado particular. Este estado define marcas gráficas como puntos, líneas, áreas o volúmenes y sus propiedades visuales como posiciones en coordenadas normalizadas, colores en espacio cromático estándar, tamaños en unidades relativas y transparencias. El cuarto y final estado consiste en vistas renderizadas que producen píxeles en dispositivos de despliegue físicos como pantallas o impresoras, implementando proyecciones geométricas desde coordenadas abstractas hacia coordenadas de dispositivo, aplicando modelos de iluminación y sombreado, y realizando anti-aliasing para suavizar bordes. Cada transición entre estados sucesivos es operada por operadores específicos encapsulando transformaciones particulares. Operadores de transformación de datos

implementan análisis estadísticos como agrupamiento jerárquico, reducción dimensional mediante análisis de componentes principales, o cálculo de funciones matemáticas sobre variables originales. Operadores de mapeo visual asocian variables de datos a propiedades gráficas según reglas de codificación como mapear temperatura a color siguiendo escala térmica de azul a rojo. Operadores de transformación de vista implementan navegación interactiva como zoom geométrico que modifica región visible, rotación tridimensional que ajusta punto de vista, o translación que desplaza encuadre. Este modelo de cuatro estados proporciona vocabulario conceptual para razonar formalmente sobre arquitecturas de sistemas de visualización y clasificar técnicas según qué transformaciones se implementan en qué estados. También revela insight fundamental que la visualización efectiva requiere no solamente diseño apropiado de mapeos visuales estáticos optimizando encodings según jerarquías perceptuales, sino también diseño cuidadoso de transformaciones interactivas que permiten a los usuarios explorar dinámicamente espacio de datos mediante manipulación directa de vistas. La interactividad transforma visualización de medio pasivo de presentación unidireccional en instrumento activo de análisis bidireccional donde humanos y computadoras colaboran iterativamente, humanos especificando intenciones mediante gestos sobre representaciones visuales y computadoras respondiendo actualizando visualizaciones reflejando nuevas transformaciones de datos. Card, Mackinlay y Shneiderman (1999) proponen modelo alternativo complementario centrado en lecturas que los humanos extraen de visualizaciones más que transformaciones computacionales que las producen. Distinguen entre lecturas elementales que obtienen valores específicos de elementos individuales como leer coordenadas exactas de punto en gráfico de dispersión posicionando cursor y leyendo etiquetas de ejes, lecturas de conjunto que caracterizan propiedades estadísticas de colecciones de elementos como identificar tendencia global lineal versus cuadrática en dispersión de puntos o estimar dispersión mediante extensión visual de nube, y lecturas comparativas que contrastan cuantitativamente subconjuntos como determinar cuál de dos grupos presenta mayor variabilidad inspeccionando rangos intercuartílicos en diagramas de caja adyacentes o identificar cuál categoría domina en composición mediante comparación de áreas en gráfico

circular. Diferentes técnicas de visualización facilitan diferentes tipos de lecturas con eficiencias variables. Gráficos de dispersión facilitan lecturas comparativas de correlaciones y detección de clusters mediante proximidad espacial de puntos relacionados, pero dificultan lecturas elementales de valores precisos dada ausencia de líneas de rejilla densas. Tablas numéricas facilitan lecturas elementales de valores exactos mediante alineación decimal y proximidad de dígitos, pero dificultan lecturas de conjunto de tendencias generales requiriendo integración mental de valores dispersos. Gráficos de barras facilitan comparaciones cuantitativas directas de categorías mediante juicios de longitud a lo largo de escala común, pero pueden oscurecer relaciones de orden superior entre múltiples variables. Este modelo orienta diseño dirigido por tareas donde se selecciona técnica de visualización según lecturas que usuario necesita realizar prioritariamente para su análisis particular, aceptando trade-offs donde optimización para cierto tipo de lectura puede suboptimizar otros tipos. Munzner (2014) sintetiza modelos previos en marco unificado de análisis y diseño de visualización anidado que estructura decisiones de diseño en cuatro niveles jerárquicos con validación apropiada en cada nivel. El nivel superior de caracterización de dominio identifica quiénes constituyen usuarios objetivo del sistema incluyendo roles laborales, expertises técnicas y familiaridad con datos, qué tareas necesitan realizar expresadas en vocabulario específico del dominio como identificar frentes oceánicos o predecir distribución de especies, y qué datos tienen disponibles descritos en términos semánticos del dominio como variables oceanográficas medidas por sensores particulares. Validación en este nivel requiere observación etnográfica de usuarios en contexto real de trabajo, entrevistas semi-estructuradas para delimitar objetivos y restricciones, y análisis de tareas mediante técnicas como diagramas de flujo de trabajo. El segundo nivel de abstracción de tarea y datos traduce requisitos específicos de dominio identificados en nivel superior a descripciones abstractas independientes de dominio en términos de tipos de datos genéricos según taxonomías como la de Shneiderman distinguiendo datos unidimensionales, bidimensionales, multidimensionales, temporales, jerárquicos y de red, y tipos de tareas abstractas según taxonomías como la de Amar et al. (2005) distinguiendo operaciones como recuperar valores, filtrar elementos, calcular derivadas, encontrar extremos, ordenar,

determinar rangos, caracterizar distribuciones, encontrar anomalías, agrupar clusters y correlacionar atributos. Esta traducción abstrae especificidad del dominio permitiendo aplicación de conocimientos generales sobre visualización desarrollados en otros dominios. Validación en este nivel emplea literatura previa documentando efectividad de técnicas para tipos de datos y tareas similares en otros contextos. El tercer nivel de codificación visual y técnicas de interacción específica cómo datos abstractos se mapean concretamente a marcas visuales seleccionando tipos de marcas como puntos, líneas o áreas, y canales que controlan su apariencia como posición, color, tamaño o forma, siguiendo principios de expresividad que garantizan codificación preserve semántica de datos y efectividad que optimiza precisión perceptual según jerarquías empíricas. También especifica qué manipulaciones interactivas se proporcionan para exploración dinámica como selección mediante brushing, navegación mediante pan y zoom, o reconfiguración mediante ordenamiento o proyección dimensional alternativa. Validación en este nivel emplea experimentos controlados midiendo precisión y velocidad de lecturas específicas, o análisis teórico de capacidad expresiva mediante cálculo formal de distinciones semánticas preservadas. El cuarto y nivel más bajo de algoritmo detalla estructuras de datos concretas en memoria como árboles de octree para indexación espacial o grafos de escena para objetos tridimensionales, y algoritmos computacionales para implementación eficiente como algoritmos de sweep line para cálculo de envolturas convexas o algoritmos de rasterización para conversión de geometrías a píxeles. Validación en este nivel emplea benchmarks de rendimiento midiendo latencias de cómputo, consumo de memoria o frames por segundo de animaciones interactivas en conjuntos de datos de diferentes magnitudes. Este marco multinivel facilita validación rigurosa donde cada nivel puede evaluarse independientemente con metodología apropiada, permitiendo identificar precisamente dónde falla diseño problemático. Fracaso en nivel de dominio indica requisitos incorrectamente comprendidos requiriendo mayor colaboración con usuarios. Fracaso en nivel abstracto sugiere teoría inadecuada requiriendo mayor investigación fundamental. Fracaso en nivel de codificación indica encodings subóptimos requiriendo revisión de asignaciones. Fracaso en nivel algorítmico indica implementación ineficiente requiriendo optimización computacional pero no rediseño

conceptual.

2.1.3. Taxonomías de Tipos de Datos

La taxonomía de tipos de datos propuesta por Shneiderman (1996) clasifica datos según estructura dimensional fundamental independientemente de dominio semántico específico, proporcionando vocabulario compartido para especificación de requisitos y selección de técnicas apropiadas. Esta clasificación estructural complementa clasificaciones semánticas que distinguen datos según significado como datos médicos versus financieros, enfocándose en forma más que contenido. La taxonomía distingue siete categorías principales cada una con implicaciones distintas para la visualización derivadas de propiedades estructurales intrínsecas. Los datos unidimensionales consisten en secuencias lineales de elementos organizados en orden único sin estructura adicional, como listas alfabéticas de nombres, documentos textuales donde palabras siguen secuencialmente, o programas de software donde instrucciones se ejecutan secuencialmente. La visualización típica emplea representaciones de lista mostrando elementos en filas o columnas sucesivas, o visualizaciones de texto con énfasis mediante resaltado o variación tipográfica para palabras significativas. Operaciones analíticas relevantes incluyen búsqueda de elemento específico mediante inspección serial o bisección si lista está ordenada, e identificación de patrones secuenciales como motivos repetidos en secuencias biológicas. Los datos bidimensionales o planares incluyen información espacial en dos dimensiones continuas como mapas geográficos especificando ubicaciones mediante coordenadas de latitud y longitud, planos arquitectónicos definiendo distribución de habitaciones en edificios, o imágenes digitales organizando píxeles en rejilla rectangular. La clasificación de técnicas de visualización según tipos de datos ha sido documentada exhaustivamente en la literatura. Heer et al. (2010) proporcionan un tour comprehensivo a través del 'zoológico de visualizaciones', catalogando técnicas establecidas para cada categoría estructural de datos, desde visualizaciones temporales mediante gráficos de líneas hasta visualizaciones jerárquicas mediante treemaps y sunbursts. Esta taxonomía práctica complementa los marcos teóricos de la visualización que emplea proyecciones cartográficas para datos georreferenciados preservando

propiedades métricas o topológicas según aplicación, o despliegue directo en coordenadas de pantalla para imágenes y diagramas. Operaciones analíticas incluyen localización de posiciones específicas, medición de distancias euclidianas o geodésicas, cálculo de áreas de regiones, e identificación de patrones espaciales como agrupamientos o gradientes. Datos bidimensionales pueden integrar información adicional mediante capas superpuestas mostrando simultáneamente múltiples variables en misma región espacial, como temperatura superficial del mar visualizada mediante color superpuesta sobre batimetría visualizada mediante contornos. Los datos tridimensionales o volumétricos representan entidades con extensión espacial en tres dimensiones como estructuras moleculares especificando posiciones atómicas en espacio cartesiano, modelos anatómicos de órganos humanos reconstruidos desde tomografías, o simulaciones de dinámica de fluidos calculando campos de velocidad y presión en volúmenes. La visualización enfrenta desafío fundamental que displays físicos son bidimensionales requiriendo proyección desde tres dimensiones a dos mediante perspectiva geométrica, con pérdida inevitable de información espacial mitigada mediante interacción permitiendo rotación para examinar estructura desde múltiples ángulos. Técnicas especializadas incluyen renderizado volumétrico que integra opacidades parciales a lo largo de rayos atravesando volumen, extracción de iso-superficies que identifican contornos de niveles específicos de variable escalar, y cortes planares que muestran secciones bidimensionales intersectando volumen en posiciones seleccionables interactivamente. Los datos multidimensionales constituyen quizás categoría más prevalente en análisis de datos científicos, consistiendo en elementos caracterizados por múltiples atributos que conforman puntos en espacio de características de alta dimensionalidad. Formalmente, dataset multidimensional puede representarse como matriz X de dimensiones n × m donde n denota número de instancias u observaciones y m denota número de atributos o variables. Cada fila x_i = (x_{i1}, x_{i2}, ..., x_{im}) representa vector de características para instancia individual, y cada columna representa valores de atributo específico a través de todas las instancias. La visualización presenta el desafío de que los humanos no pueden percibir directamente más de tres dimensiones espaciales simultáneamente, requiriendo estrategias de reducción dimensional o descomposición multivista. Técnicas de

proyección dimensional mapean espacios de alta dimensionalidad a dos o tres dimensiones visualizables mediante optimización que preserva propiedades estructurales específicas. Análisis de componentes principales identifica direcciones de máxima varianza en datos originales proyectando sobre subespacio de menor dimensión que retiene mayor información posible cuantificada como proporción de varianza explicada. Escalamiento multidimensional posiciona puntos en espacio de baja dimensión minimizando distorsión de distancias entre pares de puntos comparadas con distancias en espacio original. Embeddings no lineales como t-SNE o UMAP preservan relaciones de vecindad local permitiendo separación de clusters mientras comprimen dimensionalidad. Técnicas de pequeños múltiplos generan rejilla de gráficos bidimensionales donde cada celda muestra subespacio bidimensional diferente del espacio completo, permitiendo inspección sistemática de todas las proyecciones posibles mediante matriz de gráficos de dispersión donde fila i columna j muestra dispersión de variable i versus variable j. Técnicas de ejes paralelos representan instancia multidimensional como línea poligonal atravesando múltiples ejes verticales paralelos, cada eje correspondiendo a una dimensión, facilitando identificación de correlaciones mediante patrones de convergencia o divergencia de líneas entre ejes adyacentes. Los datos temporales constituyen categoría especial donde elementos poseen marcas temporales indicando cuándo ocurrieron eventos puntuales o periodos durante los cuales existieron entidades con duración. A diferencia de datos unidimensionales generales, datos temporales poseen estructura adicional derivada de propiedades matemáticas y físicas del tiempo como direccionalidad estricta donde pasado precede presente precede futuro sin posibilidad de retroceso, densidad variable donde eventos pueden concentrarse en períodos de actividad intensa alternando con períodos de calma, y ciclicidad potencial reflejando procesos periódicos como ritmos circadianos o estaciones anuales. Visualización típica emplea líneas de tiempo horizontales donde la posición a lo largo de eje representa momento temporal y marcas o segmentos sobre línea representan eventos o intervalos. Series temporales univariadas grafican evolución de variable única mediante línea conectando mediciones sucesivas, facilitando identificación de tendencias seculares mediante regresión de componente de baja frecuencia, estacionalidad mediante periodicidad

regular visible en oscilaciones, y anomalías mediante desviaciones abruptas de patrón esperado. Series temporales multivariadas pueden visualizarse mediante múltiples líneas coloreadas diferentemente superpuestas si magnitudes son comparables, o mediante pequeños múltiples apilados verticalmente si escalas difieren significativamente. Mapas de calor temporales representan evolución de distribución espacial mediante secuencia de mapas instantáneos en momentos sucesivos, animables como película o navegables interactivamente. Diagramas de Gantt para gestión de proyectos muestran tareas como barras horizontales extendidas en tiempo con dependencias representadas mediante flechas conectoras. Los datos jerárquicos o arbóreos organizan elementos en estructura de árbol donde cada elemento excepto raíz posee exactamente un padre, modelando relaciones de contención, especialización o causalidad. Jerarquías aparecen ubicuamente como taxonomías biológicas clasificando especies en géneros, familias y órdenes progresivamente más generales, estructuras organizacionales corporativas con empleados reportando a gerentes en cadena de mando, sistemas de archivos informáticos con directorios conteniendo subdirectorios y archivos, u ontologías de conocimiento con conceptos específicos subsumidos por conceptos generales. Visualización puede emplear técnicas de espacio-nodo que muestran explícitamente conexiones mediante enlaces como dendrogramas que representan nodos como círculos y relaciones padre-hijo mediante líneas conectoras, o diagramas de árbol con nodos posicionados en niveles horizontales según profundidad en jerarquía. Estas representaciones facilitan la comprensión de estructura global pero consumen espacio proporcional a número de nodos limitando la escalabilidad. Técnicas de llenado de espacio implican jerarquía mediante contención visual sin enlaces explícitos, subdividiendo recursivamente región rectangular entre hijos de cada nodo. Treemaps inventados por Shneiderman (1992) subdividen rectángulo de nodo en rectángulos para hijos alternando dirección de partición entre horizontal y vertical en niveles sucesivos, asignando área proporcional a valor de variable cuantitativa asociada con cada nodo como tamaño de archivo, y coloreando según variable categórica como tipo de archivo. Esta técnica utiliza eficientemente espacio permitiendo visualizar jerarquías con miles de nodos, pero distorsiona proporciones en niveles profundos. Sunbursts emplean diseño radial donde

nodo raíz ocupa círculo central y cada nivel sucesivo ocupa anillo concéntrico subdividido en sectores angulares proporcionales a valores de hijos, creando efecto de rayos solares visualmente atractivo. Los datos de red o grafo generalizan jerarquías eliminando restricción de unicidad de padre, permitiendo elementos tengan conexiones arbitrarias formando estructuras topológicas complejas. Grafos modelan relaciones como redes sociales donde individuos conectan mediante vínculos de amistad o colaboración, redes de citaciones científicas donde artículos referencian trabajos previos, redes metabólicas bioquímicas donde enzimas catalizan conversión de metabolitos, redes de transporte donde intersecciones conectan mediante carreteras, o Web semántica donde recursos enlazan mediante predicados ontológicos. Matemáticamente, grafo G = (V, E) consiste en conjunto V de vértices o nodos y conjunto E de aristas o enlaces, cada arista e ∈ E conectando par de vértices. Grafos pueden ser dirigidos donde aristas tienen origen y destino específicos, o no dirigidos dónde conexiones son simétricas. Aristas pueden tener pesos asociados representando intensidades o capacidades de conexiones. Visualización enfrenta desafíos combinatorios dado que número de configuraciones posibles de posicionamiento de nodos crece factorialmente, y objetivos como minimización de cruces de enlaces entre nodos no adyacentes son problemas NP-completos sin algoritmo eficiente garantizado. Diagramas de nodos y enlaces posicionan vértices como círculos y representan aristas como líneas o curvas conectoras, aplicando algoritmos de layout que optimizan métricas estéticas. Algoritmos dirigidos por fuerza modelan nodos como partículas que se repelen mutuamente mediante fuerza repulsiva inversamente proporcional a distancia, mientras aristas actúan como resortes que atraen nodos conectados, iterando hasta alcanzar equilibrio que tiende a espaciar nodos uniformemente mientras mantiene nodos relacionados próximos. Algoritmos jerárquicos posicionan nodos en capas horizontales minimizando cruces entre capas, apropiados para grafos acíclicos dirigidos como diagramas de flujo. Matrices de adyacencia representan grafo como tabla bidimensional donde filas y columnas corresponden a nodos y celda (i,j) codifica mediante color o intensidad existencia y peso de arista de nodo ia nodo j, eliminando problema de cruces de líneas pero requiriendo área cuadrática en número de nodos.

2.1.4. Taxonomías de Tareas Analíticas

Complementando taxonomías de datos que caracterizan qué se visualiza, taxonomías de tareas caracterizan porque los usuarios crean visualizaciones especificando objetivos cognitivos que intentan alcanzar. Shneiderman (1996) propone mantra de búsqueda visual de información que secuencia tareas en flujo de trabajo de tres niveles de granularidad decreciente: primero obtener visión general del conjunto completo de datos para comprender estructura global y establecer contexto, luego acercarse y filtrar para enfocar en subconjuntos de interés eliminando elementos irrelevantes, finalmente demandar detalles bajo demanda para inspeccionar propiedades específicas de elementos individuales. El concepto de búsqueda exploratoria, articulado por Marchionini (2006), trasciende la búsqueda de información específica (lookup) para abarcar procesos de aprendizaje y descubrimiento donde el usuario no conoce precisamente qué busca inicialmente. Este paradigma resulta particularmente relevante para sistemas de visualización interactiva que deben soportar exploración abierta de datos complejos. Esta secuencia refleja proceso cognitivo natural de exploración donde humanos establecen primero contexto global antes de examinar detalles locales, evitando desorientación que resultaría de examinar fragmentos sin comprender relación con totalidad. La tarea de visión general proporciona resumen de alto nivel de todo el dataset mediante agregación que oculta variaciones de baja importancia mientras preserva patrones principales. Para datos jerárquicos, visión general puede mostrar únicamente niveles superiores de taxonomía colapsando subrófica mediante balance geostrófico entre gradiente de presión y fuerza de Coriolis. Velocidad de viento superficial mediante dispersómetro radar midiendo rugosidad de superficie oceánica inducida por fricción del viento, relacionable con intensidad mediante funciones empíricas de transferencia. La conversión de radiancia observada a variables geofísicas requiere algoritmos de inversión complejos resolviendo problema inverso mal condicionado donde múltiples distribuciones de propiedades subsuperficiales pueden producir señal superficial similar, y correcciones atmosféricas que eliminen contribución de atmósfera interviniente entre satélite y superficie separando componente de señal originada en océano de componente originada por dispersión y absorción en atmósfera. Estos procesamientos introducen incertidumbres

que deben cuantificarse y propagarse mediante análisis de sensibilidad o simulaciones de Monte Carlo, y comunicarse mediante visualizaciones apropiadas como campos de incertidumbre mostrados mediante contornos superpuestos sobre campos de valores estimados, o visualización de ensembles mostrando múltiples realizaciones plausibles simultáneamente para transmitir rango de variabilidad compatible con observaciones. Las redes de observación in situ complementan observación remota proporcionando perfiles verticales completos de columna de agua inaccesibles desde satélites que únicamente miden capa superficial delgada de orden milímetros para óptica o centímetros para microondas. El programa internacional Argo mantiene flota de aproximadamente cuatro mil flotadores autónomos que derivan con corrientes oceánicas a profundidad de estacionamiento típicamente mil metros, periódicamente cada diez días descienden hasta dos mil metros de profundidad, luego ascienden lentamente hacia superficie adquiriendo perfil vertical continuo de temperatura y salinidad mediante sensores de conductividad-temperatura-profundidad miniaturizados, al alcanzar superficie transmiten datos vía satélite del Sistema de Argos y son relocalizados mediante sistema de posicionamiento global antes de descender nuevamente para ciclo siguiente. Visualización de trayectorias espaciotemporales de flotadores revela patrones de circulación oceánica donde convergencia de trayectorias indica regiones de hundimiento convectivo y divergencia indica regiones de surgencia, mientras que visualización de perfiles verticales caracteriza estructura termohalina estratificada con termoclina separando capa superficial cálida mezclada de agua profunda fría, y haloclina en regiones donde evaporación o precipitación modifican salinidad superficial.

2.2. Aplicaciones en Datos Científicos Marinos

La gestión y visualización de datos científicos marinos constituye un dominio especializado que requiere soluciones tecnológicas adaptadas a las características particulares de la información generada en contextos de investigación. A diferencia de los datos comerciales o administrativos, los datos científicos marinos presentan desafíos específicos relacionados con su heterogeneidad, volumen, complejidad estructural y la necesidad de garantizar su reproducibilidad y trazabilidad a lo largo del

tiempo. Las aplicaciones desarrolladas para este ámbito deben atender múltiples requerimientos simultáneos, como permitir el acceso descentralizado a conjuntos de datos distribuidos geográficamente, facilitar la interoperabilidad entre diferentes sistemas y formatos, garantizar la preservación a largo plazo de la información, y proporcionar herramientas de análisis y visualización que permitan extraer conocimiento significativo de grandes volúmenes de datos. Además, estas soluciones deben considerar los estándares de calidad y metadatos específicos de cada disciplina científica, así como las políticas de acceso abierto y compartición de datos que han ganado prominencia en las últimas décadas. En el contexto marino, estas aplicaciones adquieren particular relevancia debido a la naturaleza distribuida de las observaciones oceanográficas, la multiplicidad de variables ambientales que deben ser monitoreadas simultáneamente, y la necesidad de integrar datos provenientes de diversas fuentes y metodologías de recolección. Las plataformas tecnológicas en este campo deben ser capaces de manejar datos espacio temporales complejos, series temporales de larga duración, mediciones in situ y remotas, así como resultados de modelos numéricos. Las iniciativas de datos abiertos en ciencia marina representan esfuerzos organizados para democratizar el acceso a información oceanográfica, mientras que los sistemas especializados de visualización científica proporcionan las herramientas necesarias para explorar, analizar y comunicar estos datos de manera efectiva. Ambos componentes son fundamentales para el avance del conocimiento marino y la toma de decisiones basadas en evidencia científica.

2.2.1. Iniciativas de Datos Abiertos en Ciencia Marina

La comunidad oceanográfica internacional ha desarrollado múltiples iniciativas para promover acceso abierto a datos marinos mediante infraestructuras cibernéticas, adopción de estándares de interoperabilidad, y establecimiento de repositorios confiables con identificadores persistentes. A nivel nacional argentino, la iniciativa Pampa Azul, actualmente desfasada por cuestiones presupuestarias, fue lanzada mediante un Decreto Presidencial en 2014 y tenía como objetivo coordinar esfuerzos gubernamentales de investigación científica, desarrollo tecnológico e innovación productiva en el Mar Argentino con énfasis en conservación de recursos naturales

renovables y comprensión de procesos oceanográficos en el Atlántico Sudoccidental. Pampa Azul delimitó seis Áreas Marinas Prioritarias para investigación concentrada incluyendo Banco Burdwood y Namuncurá al sur de Islas Malvinas caracterizado por alta biodiversidad bentónica asociada con corrientes profundas, Golfo San Jorge entre Chubut y Santa Cruz como región de importancia pesquera y petrolera, plataforma continental adyacente a Puerto Madryn hasta talud donde concentra productividad biológica, Agujero Azul en plataforma continental como depresión batimétrica aislada con comunidades biológicas particulares, y escarpa continental y sector antártico argentino. Zárate et al. (2018) describen esfuerzos para publicar metadatos de campañas oceanográficas argentinas como datos enlazados abiertos aplicando principios de Web Semántica, mapeando esquemas relacionales existentes de bases de datos institucionales a vocabularios RDF estándares. Metadatos generales como título, descripción textual, cobertura temporal con fechas de inicio y fin, cobertura espacial con bounding box geográfico, y responsables con afiliaciones institucionales se codifican mediante vocabulario Dublin Core ampliamente adoptado para descripción bibliográfica de recursos digitales. Metadatos específicos de expediciones como buque oceanográfico utilizado, puertos de partida y arribo, estaciones de muestreo con coordenadas, parámetros medidos con metodologías y equipamiento, se codifican mediante vocabularios especializados desarrollados por comunidades de ciencias marinas como vocabulario OceanSites para estaciones de series temporales mantenidas por consorcio internacional. Esta publicación semántica facilita descubrimiento automatizado de datasets relevantes mediante consultas SPARQL federadas que especifican criterios geográficos intersectando región de interés con coberturas espaciales de datasets, criterios temporales intersectando periodo de interés con rangos temporales de campañas, o criterios temáticos buscando datasets que midan variables específicas como temperatura o clorofila. Zárate et al. (2019) proponen OceanGraph, prototipo de grafo de conocimiento oceanográfico que integra información heterogénea sobre expediciones científicas con fechas, trayectorias de buques y estaciones de muestreo, publicaciones científicas derivadas con autores, afiliaciones institucionales y referencias bibliográficas, variables ambientales medidas con valores agregados como rangos y promedios, permitiendo consultas complejas que

atraviesan múltiples tipos de entidades enlazadas. Por ejemplo, consulta puede recuperar publicaciones que analicen salinidad en región geográfica específica durante periodo determinado basándose en datos recolectados en expedición particular, navegando desde publicación mediante relación de autoría hacia investigadores, desde investigadores mediante participación hacia expediciones, desde expediciones mediante medición hacia variables, y filtrando por propiedades geográficas y temporales. Zárate et al. (2020) demuestran explotación de OceanGraph mediante casos de uso representativos incluyendo análisis de redes de colaboración científica identificando investigadores centrales mediante métricas de centralidad de grafos, identificación de variables frecuentemente medidas conjuntamente revelando prácticas estándares de muestreo, y descubrimiento de lagunas de conocimiento como regiones geográficas o periodos temporales escasamente muestreados requiriendo esfuerzos de observación adicionales. A nivel internacional, no podemos dejar de mencionar GBIF (2023), la Global Biodiversity Information Facility, plataforma global de datos de biodiversidad que aplica principios FAIR y Linked Data para descubrimiento e integración de datos biológicos. Como así también el proyecto GeoLink financiado por iniciativa EarthCube de la Fundación Nacional de Ciencias estadounidense ha explotado principios de datos enlazados para integrar repositorios dispersos geográfica e institucionalmente de geociencias marinas y terrestres, como describen Cheatham et al. (2018) y Krisnadhi et al. (2015). El grafo de conocimiento GeoLink contiene más de cuarenta y cinco millones de tripletas RDF describiendo cruceros oceanográficos con identificadores únicos persistentes, puertos visitados con coordenadas geográficas y fechas de llegada y salida, muestras físicas recolectadas con metadatos de localización y tipo de instrumento, proyectos de financiamiento con agencias otorgantes y montos, personal investigador con afiliaciones institucionales y roles en expediciones, e informes técnicos y publicaciones científicas con autoría y referencias bibliográficas. Datos provienen de múltiples fuentes institucionales incluyendo Rolling Deck to Repository que archiva datos de cruceros, Biological and Chemical Oceanography Data Management Office que gestiona datos biogeoquímicos, y System-Wide Information Network que indexa datasets de programa de investigación del Golfo de México, cada fuente manteniendo autonomía sobre sus datos mientras exponiendo metadatos

mediante interfaces estándares de Web Semántica. Los datos en GeoLink se publican siguiendo mejores prácticas para datos enlazados articuladas por Janowicz et al. (2014) incluyendo uso de identificadores URI HTTP desreferenciables que retornan descripciones RDF cuando se acceden mediante navegadores semánticos, provisión de enlaces explícitos hacia recursos relacionados en otros grafos mediante propiedades como owl:sameAs para identificar recursos equivalentes o rdfs:seeAlso para recursos relacionados, uso de vocabularios estándares ampliamente adoptados cuando existen como Friend of a Friend para personas y organizaciones o Dublin Core para metadatos bibliográficos, y documentación de esquemas ontológicos mediante especificaciones formales en OWL. Los datos se exponen mediante punto de acceso SPARQL público permitiendo consultas federadas que integren GeoLink con otros grafos de conocimiento externos, con respuestas retornadas en formatos estándares como JSON-LD facilitando consumo por aplicaciones web JavaScript. La Instalación Global de Información sobre Biodiversidad constituye infraestructura internacional fundada en 2001 que proporciona acceso abierto a cientos de millones de registros de ocurrencia de especies procedentes de museos de historia natural, herbarios, programas de monitoreo ambiental, iniciativas de ciencia ciudadana como eBird para avistamientos de aves, y sensores remotos automatizados como cámaras trampa para mamíferos terrestres o hidrófonos para mamíferos marinos. Datos marinos incluyen avistamientos de megafauna como cetáceos, pinnípedos y tortugas desde embarcaciones o aeronaves, capturas pesqueras comerciales y artesanales con especies, cantidades y ubicaciones, muestras de biodiversidad bentónica recolectadas mediante dragas o nucleadores, y muestras de plancton recolectadas mediante redes de diferentes mallas. Registros se normalizan taxonómicamente mediante Catálogo de la Vida que proporciona nomenclatura estándar con sinonimias resueltas, garantizando que especie referida por múltiples nombres científicos históricos se reconozca como entidad única facilitando agregación entre fuentes. Registros también se normalizan geográficamente validando coordenadas reportadas contra bases de datos de líneas de costa verificando que ubicaciones marinas no caigan en tierra y viceversa, detectando errores comunes como inversión de latitud y longitud o uso de hemisferio incorrecto. Visualización de distribuciones geográficas de especies mediante mapas de densidad o

mapas de calor revela patrones biogeográficos como endemismo donde especies restringen a regiones limitadas, o cosmopolitismo donde especies distribuyen ampliamente, identificando hotspots de diversidad concentrando múltiples especies merecedores de priorización para conservación. Mapas pueden combinarse con capas de variables ambientales como temperatura superficial del mar, productividad primaria estimada desde clorofila satelital, o batimetría, permitiendo exploración visual de correlaciones entre factores abióticos y diversidad biológica mediante inspección de superposiciones espaciales, por ejemplo identificando que diversidad de corales concentra en aguas tropicales cálidas de plataformas poco profundas, o que abundancia de krill concentra en regiones de surgencia con alta productividad.

2.2.2. Sistemas Especializados de Visualización Científica

La comunidad de oceanografía computacional ha desarrollado sistemas especializados de visualización que abordan requisitos específicos del dominio mediante técnicas adaptadas a características estructurales y semánticas de datos marinos. El sistema Ocean Data View (Schlitzer, 2015) constituye aplicación de escritorio ampliamente utilizada para exploración interactiva de datos oceanográficos perfilados, proporcionando capacidades integradas de importación desde formatos diversos, control de calidad mediante detección de outliers estadísticos, interpolación espacial mediante técnicas geoestadísticas, y generación de múltiples tipos de visualizaciones. Perfiles verticales se visualizan mediante diagramas de profundidad en eje vertical versus variable medida en eje horizontal, permitiendo comparación directa de estructura vertical entre estaciones adyacentes. Secciones transversales se construyen interpolando perfiles a lo largo de trayectoria de crucero conectando estaciones secuencialmente, generando campo bidimensional con distancia horizontal a lo largo de track en eje horizontal y profundidad en eje vertical, coloreando según variable de interés revelando estructura tridimensional de frentes y masas de agua. Mapas horizontales muestran propiedades integradas verticalmente o en niveles de profundidad específicos, aplicando interpolación espacial mediante ponderación por distancia inversa o kriging que estiman valores en posiciones no muestreadas combinando observaciones cercanas ponderadas por covarianza espacial derivada de

semivariograma empírico. Ocean Data View (Schlitzer, 2015) soporta múltiples proyecciones cartográficas incluyendo Mercator que preserva ángulos facilitando navegación pero distorsiona áreas particularmente en altas latitudes, proyección estereográfica polar apropiada para regiones árticas y antárticas minimizando distorsión cerca de polos, y proyección de Mollweide que preserva áreas facilitando comparaciones de extensión espacial de fenómenos. Capas superpuestas incluyen batimetría mediante contornos de isobatas o mapas de relieve sombreado que aplican modelo de iluminación artificial revelando características topográficas del lecho marino, líneas de costa de alta resolución desde bases de datos GSHHS, y frentes oceánicos identificados automáticamente como gradientes fuertes de temperatura o salinidad mediante cálculo de magnitud de gradiente espacial. El sistema permite exportación de visualizaciones en formatos vectoriales como PostScript o PDF apropiados para publicaciones científicas preservando nitidez en cualquier escala de impresión, o formatos raster como PNG con resolución configurable para presentaciones digitales. El sistema Ferret (NOAA/PMEL, 2015) desarrollado por Laboratorio de Medio Ambiente Marino del Pacífico de la Administración Nacional Oceánica y Atmosférica estadounidense está optimizado para análisis de salidas de modelos numéricos oceánicos y atmosféricos que típicamente consisten en campos cuadriculados cuatridimensionales con dimensiones de longitud, latitud, profundidad o nivel vertical en coordenadas sigma siguiendo topografía, y tiempo con cadencia horaria, diaria o mensual según resolución temporal de integración. Ferret (NOAA/PMEL, 2015) proporciona lenguaje de scripting especializado similar a MATLAB que permite usuarios especificar operaciones matemáticas complejas sobre campos mediante sintaxis concisa, facilitando cálculo de cantidades derivadas que no se almacenan directamente como vorticidad relativa calculada como rotacional del campo de velocidad horizontal revelando rotación de flujo, divergencia calculada como divergencia del campo de velocidad identificando regiones de convergencia donde agua converge horizontalmente asociadas con hundimiento vertical o divergencia donde agua diverge asociadas con surgencia, o transporte de volumen integrando velocidad a través de secciones verticales cuantificando flujos netos entre cuencas. Operaciones de subsección extraen regiones espaciales o temporales específicas,

promediado temporal calcula climatologías removiendo variabilidad de alta frecuencia, y diferenciación calcula derivadas temporales estimando tendencias o derivadas espaciales calculando gradientes. La visualización soporta campos vectoriales mediante flechas orientadas según dirección de corriente y escaladas según magnitud, permitiendo percepción inmediata de patrones de circulación como giros ciclónicos rotando en sentido antihorario en hemisferio norte, corrientes de chorro estrechas con velocidades intensas, y regiones de recirculación donde flujo forma loops cerrados. Contornos de isolíneas conectan puntos de valor constante de variable escalar, espaciamiento entre contornos adyacentes indicando magnitud de gradiente donde contornos apretados señalan cambios espaciales abruptos como frentes térmicos separando masas de agua de temperaturas diferentes. Diagramas de Hovmöller visualizan evolución temporal de variable a lo largo de dimensión espacial específica como latitud o longitud, mediante mapa de calor bidimensional con tiempo en eje horizontal y dimensión espacial en eje vertical, coloreando celdas según valor de variable promediada sobre dimensión espacial ortogonal, facilitando identificación de propagación de señales como ondas planetarias que se manifiestan como bandas diagonales indicando desplazamiento meridional progresivo en tiempo.

2.3. Tecnologías Emergentes

Esta sección examina tecnologías computacionales emergentes que constituyen fundamentos técnicos para el sistema propuesto en esta investigación, organizadas en tres subsecciones que abordan LLM, paradigmas de datos enlazados, y sistemas de recomendación automatizados. La integración de inteligencia artificial (IA) en herramientas de visualización científica marina representa una frontera emergente con potencial transformador. Song et al. (2023) proporcionan una revisión comprehensiva del papel de la IA en ciencias marinas, identificando aplicaciones específicas en procesamiento de imágenes satelitales, análisis de datos de sensores y modelado predictivo de ecosistemas marinos. Los algoritmos de aprendizaje automático están revolucionando la capacidad de extraer patrones significativos de datasets marinos

masivos. Técnicas de clustering no supervisado permiten identificar regiones oceanográficas homogéneas basadas en múltiples variables ambientales, mientras que algoritmos de clasificación supervisada facilitan la identificación automática de especies marinas en imágenes submarinas o la detección de eventos oceanográficos extremos. Jiang y Zhu (2022) destacan el papel específico de algoritmos de IA en investigación científica marina, enfatizando aplicaciones en análisis predictivo de cambio climático, optimización de campañas de muestreo y desarrollo de sistemas de alerta temprana para eventos oceanográficos extremos. Sus hallazgos sugieren que la integración de IA en visualización marina puede aumentar significativamente la capacidad de procesar e interpretar información compleja en tiempo real. Beyan y Browman (2020) discuten el establecimiento de la era de inteligencia artificial en ciencias marinas, identificando oportunidades específicas en automatización de análisis de datos, desarrollo de modelos predictivos más precisos y creación de interfaces de visualización adaptativas que se ajusten automáticamente a las necesidades específicas del usuario. Las redes neuronales profundas están siendo aplicadas específicamente para mejorar la calidad de visualizaciones marinas mediante técnicas como super-resolución de imágenes satelitales, interpolación inteligente de datos espaciales con información faltante, y generación automática de visualizaciones optimizadas basadas en el contexto específico del análisis.

2.3.1. Large Language Models

Los LLM representan clase de sistemas de inteligencia artificial basados en redes neuronales profundas entrenadas mediante aprendizaje auto-supervisado sobre corpus textuales masivos, adquiriendo capacidades de comprensión y generación de lenguaje natural que aproximan competencia humana en tareas diversas. Ahmad et al. (2019) revisa cómo el aprendizaje automático se aplica a la oceanografía, incluyendo predicción de corrientes, detección de anomalías y clasificación de imágenes satelitales. La arquitectura fundamental subyacente a modelos contemporáneos más capaces como GPT-4 (OpenAI, 2023) consiste en transformadores, mecanismo de atención propuesto originalmente por Vaswani et al. (2017) que procesa secuencias de tokens mediante capas apiladas aplicando operaciones de atención multi-cabeza y

redes feed-forward. El mecanismo permite al modelo asignar diferentes pesos de importancia a diferentes tokens en secuencia de entrada al generar representación de token particular, capturando dependencias sintácticas y semánticas a larga distancia que modelos recurrentes previos como LSTM manejan inadecuadamente debido a problema de desvanecimiento de gradiente en secuencias largas. Formalmente, dado secuencia de entrada $X = (x_1, x_2, ..., x_n)$ donde cada x_i representa embedding vectorial de token i, mecanismo de atención calcula representación contextualizada de cada token como combinación ponderada de todos los tokens en secuencia. Específicamente, para token de consulta q, atención calcula puntuaciones de compatibilidad con todos los tokens de clave $k_1, k_2, ..., k_n$ mediante producto escalar seguido de función softmax para normalizar, produciendo pesos de atención $\alpha_1, \alpha_2, ..., \alpha_n$ que suman uno. Representación contextualizada se obtiene como suma ponderada de tokens de valor $v_1, v_2, ..., v_n$ según pesos α. Atención multi-cabeza extiende este mecanismo aplicando múltiples cabezas de atención en paralelo, cada cabeza aprendiendo proyecciones lineales diferentes de consultas, claves y valores, permitiendo modelo atender simultáneamente a diferentes tipos de relaciones semánticas como dependencias sintácticas entre sujeto y verbo, correferencias entre pronombres y antecedentes, o relaciones semánticas entre entidades nombradas. El entrenamiento de LLM, como explica Xie et al. (2019) emplea objetivo de modelado de lenguaje causal dónde modelo aprende a predecir token siguiente dada secuencia de tokens precedentes, maximizando verosimilitud logarítmica del corpus de entrenamiento calculada como suma de logaritmos de probabilidades asignadas a tokens correctos en cada posición. Corpus de entrenamiento típicamente comprende billones de tokens extraídos de fuentes diversas como páginas web rastreadas de Internet, libros digitalizados, artículos científicos, código fuente de software, y conversaciones de redes sociales, proporcionando cobertura amplia de conocimientos factuales, habilidades de razonamiento, y patrones lingüísticos. Entrenamiento requiere infraestructura computacional masiva con miles de procesadores gráficos o tensores dedicados operando en paralelo durante semanas o meses, consumiendo megavatios de electricidad y costando decenas de millones de dólares, factible únicamente para organizaciones con recursos sustanciales como laboratorios

industriales de grandes corporaciones tecnológicas. Una vez pre-entrenado, modelo puede adaptarse a tareas específicas mediante ajuste fino con dataset etiquetados del dominio objetivo, actualizando parámetros mediante descenso de gradiente para minimizar pérdida de tarea específica, o alternativamente mediante aprendizaje por pocos ejemplos donde modelo generaliza a nuevos ejemplos tras observar únicamente puñado de demostraciones proporcionadas en prompt sin actualizar parámetros. Esta última capacidad resulta particularmente valiosa cuando datos etiquetados son escasos o costosos de obtener, permitiendo reutilización de modelo pre-entrenado general para aplicaciones especializadas mediante diseño cuidadoso de prompts que elicitan comportamiento deseado. La ingeniería de prompts constituye disciplina emergente enfocada en diseño sistemático de entradas textuales que maximizan calidad y relevancia de salidas generadas por modelos de lenguaje. Prompts efectivos típicamente estructuran en componentes funcionales: mensaje de sistema que establece rol, personalidad y directivas comportamentales del modelo como "eres experto en análisis de datos especializado en visualización", mensajes de contexto que proporcionan información de fondo relevante para tarea como descripción de estructura de dataset incluyendo nombres de columnas, tipos de variables y estadísticas descriptivas, ejemplos demostrativos que ilustran formato deseado de salida mediante pares entrada-salida representativos, y finalmente consulta específica que solicita modelo realice tarea concreta como "recomienda visualización apropiada para estas variables". Técnicas avanzadas incluyen encadenamiento de prompts donde salida de una invocación se utiliza como entrada para invocación subsecuente, permitiendo descomposición de tareas complejas en subtareas manejables ejecutadas secuencialmente, y generación con restricciones donde se especifican esquemas formales que salida debe satisfacer como sintaxis JSON con campos obligatorios y tipos especificados, mejorando confiabilidad de parsing automático de respuestas. Investigación reciente demuestra que modelos de lenguaje pueden realizar razonamiento multi-paso cuando se les instruye explícitamente generar cadena de pensamiento verbalizando pasos intermedios antes de producir respuesta final, mejorando precisión en tareas requiriendo inferencias complejas.

2.3.2. Datos Enlazados y Web Semántica

El paradigma de datos enlazados abiertos constituye realización práctica de visión de Web Semántica articulada por Berners-Lee et al. (2001), donde la información en Web se estructura mediante estándares formales que permiten a las máquinas comprender significado de datos, no meramente presentar contenido visualmente a humanos. La arquitectura técnica fundamental se basa en modelo de grafos RDF donde la información se representa como colección de tripletas de forma sujeto-predicado-objeto, cada componente identificado mediante IRI que proporciona identificador único global desreferenciable. Por ejemplo, tripleta <http://example.org/station1> <http://purl.org/dc/terms/spatial> <http://sws.geonames.org/3435910/> expresa que estación de muestreo identificada por IRI station1 tiene ubicación espacial identificada por IRI de GeoNames correspondiente, estableciendo un enlace explícito entre datasets independientes que puede seguirse navegando desde IRI hacia descripciones adicionales. El modelo RDF puede serializarse en múltiples anotaciones sintácticas para almacenamiento e intercambio. RDF/XML codifica grafos como documentos XML siguiendo esquema específico, proporcionando interoperabilidad con herramientas de procesamiento XML pero produciendo sintaxis verbose difícilmente legible por humanos. Turtle emplea sintaxis concisa inspirada en notación de tripletas con abreviaturas que reducen redundancia cuando múltiples tripletas comparten mismo sujeto o predicado, mejorando la legibilidad para edición manual. JSON-LD serializa grafos como documentos JSON estándares añadiendo contexto que mapea términos a IRIs, facilitando integración con aplicaciones web JavaScript que procesan JSON nativamente sin requerir parsers RDF especializados. Los vocabularios ontológicos proporcionan semántica compartida definiendo clases de recursos, propiedades que relacionan recursos, y axiomas que restringen interpretaciones válidas. RDFS extiende RDF con constructores para definir jerarquías de clases mediante rdfs:subClassOf y jerarquías de propiedades mediante rdfs:subPropertyOf, y especificar dominios y rangos de propiedades restringiendo tipos de recursos que pueden aparecer como sujetos u objetos. Los identificadores persistentes como DOI (Digital Object Identifier) constituyen una infraestructura crítica para datos FAIR. Devaraju y Herterich (2019) analizan la conexión entre repositorios registrados en Re3data y artículos en DataCite,

demostrando cómo identificadores persistentes facilitan trazabilidad y citabilidad de datasets científicos. Por su parte OWL extiende RDFS con constructores lógicos adicionales fundamentados en lógicas descriptivas, permitiendo definir equivalencias entre clases, disyunciones que especifican clases mutuamente excluyentes, restricciones de cardinalidad que limitan número de valores de propiedades, y propiedades transitivas, simétricas o funcionales, habilitando razonamiento automático que infiere tripletas implícitas desde tripletas explícitas mediante aplicación de reglas lógicas. SPARQL constituye lenguaje estándar de consulta para grafos RDF, proporcionando sintaxis declarativa para especificar patrones de grafos que se emparejan contra datos, retornando bindings de variables que satisfacen patrones. Consultas SELECT recuperan valores específicos de variables especificadas, consultas CONSTRUCT generan nuevos grafos RDF derivados transformando resultados, consultas ASK retornan valor booleano indicando si patrón tiene al menos un emparejamiento, y consultas DESCRIBE retornan descripción completa de recursos especificados. Operadores incluyen UNION para alternativas disyuntivas, OPTIONAL para patrones opcionales cuya ausencia no invalida solución, FILTER para restricciones sobre valores de variables expresadas mediante funciones y operadores aritméticos o lógicos, agregación mediante COUNT, SUM, AVG para calcular estadísticos sobre grupos, y ordenamiento mediante ORDER BY. SPARQL federado extiende lenguaje con cláusula SERVICE que delega subconsultas a endpoints SPARQL remotos permitiendo consultas distribuidas que integran múltiples grafos de conocimiento mantenidos independientemente por organizaciones diferentes, materializando visión de Web de datos globalmente interconectada. Los principios FAIR articulados por Wilkinson et al. (2016) proporcionan directrices para gestión y administración de datos científicos digitales que maximizan la reutilización por humanos y máquinas. Localizable requiere que datos tengan identificadores persistentes únicos globalmente como DOI o URIs HTTP, metadatos ricos describiendo contenido, y registro en índices o catálogos facilitando descubrimiento mediante búsqueda. Accesible requiere que datos sean recuperables mediante protocolos estándares abiertos, permanentemente disponibles incluso cuando datos mismos ya no existen preservando metadatos, y autenticados o autorizados cuando sea necesario para datos sensibles. Interoperable requiere que

datos usen lenguajes formales comprensibles por máquinas para representación del conocimiento, vocabularios siguiendo principios FAIR themselves, y referencias cualificadas a otros datos estableciendo enlaces contextualizados. Reutilizable requiere que datos tengan licencias claras de uso especificando qué puede hacerse legalmente, proveniencia detallada documentando origen y transformaciones, y conformidad con estándares comunitarios relevantes para dominio. La adherencia a principios FAIR incrementa el valor de activos de datos científicos facilitando meta-análisis que agreguen múltiples estudios, reproducibilidad de resultados publicados, y descubrimiento de datos para nuevas investigaciones.

2.3.3. Sistemas de Recomendación de Visualización

Los sistemas de recomendación de visualización constituyen una clase de aplicación que asiste a los usuarios en la selección de técnicas visuales apropiadas dado dataset particular y objetivos analíticos implícitos o explícitos, reduciendo carga cognitiva asociada con navegación de espacio de diseño de visualizaciones extremadamente amplio. Enfoques tradicionales basados en reglas codifican conocimiento experto mediante heurísticas que mapean características de datos a recomendaciones siguiendo principios establecidos como jerarquías perceptuales. Por ejemplo, regla puede prescribir qué variable cuantitativa continua con distribución unimodal debe visualizarse mediante histograma o gráfico de densidad, mientras que variable categórica nominal con cardinalidad menor a diez debe visualizarse mediante gráfico de barras. El sistema desarrollado por Mackinlay (1986) implementa un conjunto de reglas formales codificadas como axiomas lógicos que especifican expresividad y efectividad de diferentes encodings, aplicando motor de inferencia que busca configuración satisfaciendo restricciones de expresividad mientras maximizando efectividad mediante heurísticas de búsqueda. Los enfoques basados en aprendizaje automático aprenden patrones asociativos entre características de datos y visualizaciones efectivas desde corpus de ejemplos etiquetados, evitando necesidad de codificación manual exhaustiva de reglas. Data2Vis desarrollado por Dibia y Demiralp (2019) entrena una red neuronal que traduce una especificación del esquema de un dataset y lo adapta al lenguaje de especificación de visualización Vega-Lite,

aprendiendo desde el corpus de ejemplos de especificaciones publicadas en una galería en línea. El modelo codifica el esquema como una secuencia de tokens representando nombres de columnas y tipos, y decodifica auto-regresivamente la especificación para Vega-Lite token por token, condicionando cada uno en tokens previamente generados y representación codificada de esquema. El entrenamiento maximiza la verosimilitud de especificaciones correctas en el corpus, y resulta en la inferencia del modelo que genera las especificaciones para los esquemas nuevos mediante la validación según su alto nivel de probabilidad. Por su parte, VizML sistema propuesto por Hu et al. (2019) formula una recomendación como problema de clasificación multiclase clases que corresponden a plantillas de visualización y a través de un clasificador predice cuál es la plantilla apropiada dadas las características extraídas de los datos. Estas características incluyen tipos de variables codificados mediante la técnica one-hot encoder, label encoder, cardinalidades de variables categóricas, rangos de variables cuantitativas, presencia de valores faltantes, tamaño de dataset como número de filas y columnas, y estadísticos descriptivos como media, mediana, desviación estándar, asimetría y curtosis. El clasificador se entrena con un modelo Random Forest aleatorio sobre corpus etiquetado construido mediante crowdsourcing donde analistas humanos seleccionan las visualizaciones preferidas desde un conjunto de alternativas para datasets sintéticos variados de modelo. La evaluación se desarrolla mediante validación cruzada y se considera efectiva si demuestra precisión superior a setenta por ciento en predicción de preferencias humanas, como proponen primero Isinkaye et al. (2015) y más actualmente Roy y Dutta (2022).

CAPÍTULO

3 ESTADO DEL ARTE

3.1. Análisis Crítico de la Literatura

La literatura académica sobre visualización de datos enlazados, sistemas de recomendación de visualización, e integración de inteligencia artificial en herramientas de analítica visual ha evolucionado sustancialmente durante la última década, reflejando maduración del campo desde prototipos académicos experimentales hacia sistemas robustos con adopción práctica. Esta sección sintetiza hallazgos principales organizados temáticamente, identificando tendencias emergentes, limitaciones persistentes y direcciones prometedoras para investigación futura.

3.1.1. Evolución de Visualización de Datos Científicos

La investigación en visualización de datos enlazados ha transitado desde navegadores semánticos textuales primitivos hacia sistemas visuales interactivos sofisticados que explotan propiedades estructurales de grafos RDF. Dadzie y Rowe (2011) conducen una revisión comprehensiva de aproximaciones tempranas, clasificándolas según si enfatizan una presentación textual versus visual, y según si soportan exploración interactiva versus presentación estática. Identifican qué herramientas textuales dominaron inicialmente debido a simplicidad de implementación renderizando tripletas directamente como tablas HTML, pero usabilidad limitada motivó desarrollo de alternativas visuales más intuitivas. Sin embargo, señalan que la mayoría de herramientas visuales tempranas operan únicamente sobre grafos RDF relativamente pequeños cargados completamente en memoria del navegador, limitando la escalabilidad a decenas de miles de tripletas antes de degradación de rendimiento interactivo. Peña et al. (2016) actualizan esta revisión examinando evolución hacia arquitecturas cliente-servidor donde consultas SPARQL ejecutan en backend remoto y

únicamente resultados filtrados se transmiten a frontend para visualización, mejorando escalabilidad mediante delegación de procesamiento computacionalmente costoso a servidores con recursos superiores. Identifican tres patrones arquitectónicos recurrentes: arquitectura de navegador donde usuario comienza con recurso específico y navega incrementalmente siguiendo enlaces hacia recursos relacionados, apropiada para exploración dirigida cuando punto de partida es conocido; arquitectura de búsqueda donde usuario especifica consulta mediante interfaz de facetas o formulario estructurado y sistema retorna conjunto de resultados relevantes visualizados, apropiada para localización de recursos satisfaciendo criterios específicos; y arquitectura de visión general dónde sistema genera automáticamente resumen visual de dataset completo mostrando distribuciones de clases, propiedades frecuentes y estadísticos agregados, apropiada para comprensión inicial de estructura global. Po et al. (2020) publican monografía técnica exhaustiva sobre técnicas, herramientas y big data en visualización de datos enlazados, proponiendo taxonomía multidimensional que clasifica sistemas según múltiples criterios ortogonales. La dimensión de entrada caracteriza qué formatos de datos consume el sistema, distinguiendo entre sistemas que aceptan únicamente archivos RDF locales subidos por usuario versus sistemas que consultan endpoints SPARQL remotos federados. La dimensión de procesamiento caracteriza qué transformaciones algorítmicas aplica sistema a datos crudos antes de visualización, incluyendo inferencia mediante razonadores OWL que derivan tripletas implícitas, detección de comunidades en grafos mediante algoritmos de clustering, y cálculo de métricas de centralidad identificando nodos importantes. La dimensión de visualización caracteriza qué técnicas gráficas emplea sistema para renderizado, clasificadas según taxonomía de Shneiderman en visualizaciones de datos jerárquicos mediante treemaps o sunbursts, visualizaciones de redes mediante grafos de nodos y enlaces con layouts dirigidos por fuerza, y visualizaciones geoespaciales mediante mapas cuando coordenadas están disponibles. La dimensión de interacción caracteriza qué manipulaciones soporta el sistema, distinguiendo entre navegación mediante zoom y pan, filtrado mediante facetas o consultas, y vinculación entre vistas múltiples sincronizadas. Esta taxonomía multidimensional revela que el espacio de diseño de sistemas de visualización LOD es vasto con combinaciones posibles de decisiones en

cada dimensión generando arquitecturas radicalmente diferentes optimizadas para casos de uso distintos. Sin embargo, Po et al. (2020) identifican qué herramientas existentes tienden a especializarse en subconjunto estrecho de capacidades más que proveer funcionalidad comprehensiva, creando fragmentación donde usuario requiere múltiples herramientas complementarias para flujo de trabajo completo. Argumentan la necesidad de arquitecturas modulares con interfaces estándares que permitan composición de componentes especializados en pipelines configurables, similar a filosofía UNIX de herramientas pequeñas componibles mediante pipes. Ji et al. (2016) Describe I4Ocean, una plataforma interactiva para simulación y visualización científica marina en 3D, integrando modelos físicos y visualización. Finalmente, en Jiang et al. (2022) se desarrolla una revisión de cómo algoritmos de IA , machine learning y deep learning, apoyan tareas en investigación marina, como reconocimiento de patrones y predicción ambiental.

3.1.2. Sistemas de Recomendación Basados en Aprendizaje

La investigación sobre automatización de selección de visualizaciones mediante técnicas de aprendizaje automático ha progresado desde enfoques basados en reglas codificadas manualmente hacia enfoques basados en datos que aprenden patrones desde corpus de ejemplos. Podo et al. (2023) desarrolla una revisión precisa de los desafíos abiertos y direcciones futuras para aplicar Machine Learning en sistemas de recomendación de visualizaciones. Mackinlay et al. (1986) presenta el sistema APT que combina principios de expresividad y efectividad como axiomas lógicos en sistemas de producción, aplicando búsqueda heurística para encontrar configuración de encodings visuales satisfaciendo restricciones de expresividad mientras maximizando efectividad según jerarquías perceptuales. Aunque innovador conceptualmente, APT requiere ingeniería de conocimiento sustancial para codificar reglas exhaustivamente y carece de mecanismos para aprender desde retroalimentación o adaptarse a preferencias de usuarios específicos. Li et al. (2023), constituye una revisión sistemática sobre cómo los LLM se integran en sistemas de recomendación, destacando su capacidad de razonamiento contextual. En esta línea, Wongsuphasawat et al. (2016) presentan Voyager, un sistema de exploración visual recomendada que genera automáticamente

galería de visualizaciones candidatas aplicando enumeración exhaustiva de combinaciones de campos con filtrado mediante heurísticas de efectividad estadística. Voyager implementa modelo computacional del proceso de especificación de visualización como búsqueda en espacio de diseño, representando cada visualización como tupla de campos asignados a encodings específicos. Sistema enumera sistemáticamente subconjuntos de campos del esquema de datos, para cada subconjunto prueba asignaciones alternativas a encodings visuales, evalúa efectividad de cada candidato mediante métricas cuantificando utilidad informativa como entropía de distribución y discriminabilidad perceptual según jerarquías de Cleveland y McGill, y ordena candidatos según puntuación compuesta presentando top-k recomendaciones en galería navegable. Evaluación demuestra que Voyager facilita descubrimiento de visualizaciones interesantes que usuarios no habrían considerado explícitamente, funcionando efectivamente como asistente de exploración que sugiere direcciones prometedoras. Hu et al. (2019) desarrollan VizML, sistema que formula recomendación como problema de aprendizaje supervisado donde clasificador predice tipo de visualización apropiado dadas características estadísticas y estructurales extraídas de datos. Conjunto de entrenamiento se construye mediante crowdsourcing donde trabajadores humanos evalúan calidad de visualizaciones alternativas para datasets sintéticos variados, proporcionando preferencias que capturan juicios perceptuales implícitos. Características incluyen tipos semánticos de campos inferidos mediante análisis de contenido más allá de tipos sintácticos declarados, cardinalidades de campos categóricos, rangos y distribuciones de campos cuantitativos, presencia de patrones como tendencias temporales o correlaciones entre pares de campos, y metadatos como tamaño de dataset. Clasificador implementado mediante bosques aleatorios alcanza precisión superior a setenta por ciento en predicción de preferencias humanas en conjunto de validación, superando significativamente reglas heurísticas baseline. Análisis de importancia de características revela que tipos semánticos y patrones estadísticos contribuyen más fuertemente a predicción que características estructurales simples, validando intuición que recomendación efectiva requiere análisis semántico profundo más allá de inspección superficial de esquema. Finalmente, Bao et al. (2023), presenta TALLRec un framework que adapta grandes modelos de lenguaje

para tareas de recomendación mediante fine-tuning eficiente.

3.1.3. Integración de Modelos de Lenguaje en Analítica Visual

La emergencia de modelos de lenguaje extensos pre-entrenados ha motivado exploración de su aplicación a tareas de visualización, aprovechando capacidades de razonamiento sobre descripciones en lenguaje natural y generación de código. Wu et al. (2024) conducen un estudio exploratorio sobre generación automática de visualizaciones desde lenguaje natural mediante GPT-3 y GPT-4 (OpenAI, 2023), evaluando capacidades de modelos para interpretar intenciones analíticas expresadas informalmente y traducirlas a especificaciones ejecutables en gramáticas declarativas como Vega-Lite. Mediante evaluación sobre benchmark de descripciones textuales de visualizaciones deseadas emparejadas con implementaciones ground-truth, encuentran que GPT-4 (OpenAI, 2023) genera especificaciones sintácticamente válidas en noventa y dos por ciento de casos comparado con setenta y cinco por ciento para GPT-3, indicando mejora sustancial en fiabilidad. Sin embargo, un análisis cualitativo revela que errores semánticos surgen donde la especificación es sintácticamente correcta pero no implementa la intención descrita en veinte por ciento de los casos, típicamente debido a ambigüedades en la descripción textual o en las suposiciones implícitas incorrectas del modelo. Wang et al. (2023) propone LLM4Vis, sistema que integra ChatGPT para recomendación de visualización con explicaciones en lenguaje natural justificando sus elecciones. La arquitectura se implementa en tres etapas: primero, el modelo analiza el esquema del dataset identificando los tipos semánticos de campos y patrones estadísticos mediante razonamiento sobre descripciones textuales de contenido; segundo, modelo genera múltiples candidatos de visualización con diversidad controlada mediante muestreo con temperatura elevada; tercero, modelo evalúa candidatos generando justificaciones textuales explicando por qué visualización particular es apropiada según principios de efectividad perceptual y alineación con tareas analíticas inferidas. Evaluación mediante estudio comparativo con sistemas baseline basados en reglas demuestra que LLM4Vis (Wang et al., 2023) produce recomendaciones con relevancia comparable pero explicaciones significativamente más comprensibles para usuarios sin expertise técnica, reduciendo

carga cognitiva requerida para comprender razonamiento subyacente. Dibia (2023) desarrolla LIDA, herramienta que utiliza GPT-4 (OpenAI, 2023) para generación automática de visualizaciones e infografías desde datos tabulares, implementando arquitectura modular con cuatro componentes principales. A diferencia de Xie et al. (2019), que propone un método no supervisado para representar contexto semántico en sistemas de recomendación. En LIDA (Dibia, 2023)el componente de resumen genera una descripción textual concisa del dataset incluyendo esquema, estadísticos descriptivos y patrones notables como correlaciones fuertes o valores atípicos, proporcionando contexto para etapas subsecuentes. El componente de generación de objetivos infiere automáticamente preguntas analíticas interesantes que usuario podría tener sobre datos, mediante prompt que instruye modelo considerar diferentes tipos de análisis como comparaciones entre categorías, evolución temporal, o relaciones entre variables. El componente de generación de visualizaciones traduce cada objetivo a especificación Vega-Lite implementable, aplicando conocimiento de mejores prácticas de visualización codificado en prompts cuidadosamente diseñados. El componente de refinamiento permite iteración donde usuario proporciona retroalimentación en lenguaje natural sobre visualización generada y modelo modifica especificación acordemente, implementando diálogo interactivo multi-turno. Evaluación de LIDA (Dibia, 2023)mediante datasets de referencia demuestra una generación exitosa de visualizaciones apropiadas en ochenta y cinco por ciento de casos sin intervención humana, y capacidad de refinamiento mediante retroalimentación incrementa tasa de éxito a noventa y tres por ciento después de promedio de dos iteraciones. El análisis de modos de fallo revela que casos problemáticos típicamente involucran datos con estructuras inusuales no bien representadas en corpus de entrenamiento del modelo, o requisitos de visualización muy específicos del dominio requiriendo conocimiento especializado. Estas limitaciones sugieren que un enfoque híbrido combinando capacidades generales de modelos de lenguaje con módulos especializados codificando conocimiento de un dominio específico podría superar sistemas puramente basados en modelos o puramente basados en reglas, como proponen Ye et al. (2024) al presentar una revisión actualizada de generación de visualizaciones con inteligencia artificial.

3.2. Plataformas Interactivas de Visualización de Datos

Las plataformas interactivas tradicionales de visualización representan el estado de la práctica actual, habiendo alcanzado madurez técnica y adopción amplia en industria y academia. Esta sección analiza arquitecturas de software, paradigmas de interacción y capacidades técnicas de sistemas representativos, organizados según aproximación principal desde plataformas comerciales de negocio inteligente hasta bibliotecas programáticas de código abierto. Como Wang et al. (2021) que propone un sistema modular para visualización interactiva de datos oceánicos, facilitando reuso y análisis multifuente.

3.2.1. Plataformas Comerciales

Tableau constituye plataforma comercial líder en mercado de inteligencia de negocios con participación superior a treinta por ciento según analistas industriales, caracterizada por interfaz visual de arrastrar y soltar que permite construcción de dashboards interactivos sin programación explícita. La arquitectura técnica subyacente implementa modelo de datos semántico donde usuario define dimensiones categóricas y medidas cuantitativas mediante clasificación de campos del esquema fuente, y sistema infiere automáticamente agregaciones apropiadas como sumas para medidas aditivas o promedios para medidas no aditivas. El motor de consultas VizQL traduce especificaciones visuales a consultas optimizadas contra fuentes de datos heterogéneas incluyendo bases relacionales mediante generación de SQL, archivos planos mediante operaciones en memoria, y cubos OLAP mediante consultas MDX, abstrayendo detalles de tecnología subyacente mediante capa de abstracción de datos. La generación de visualizaciones implementa sistema de recomendación que sugiere tipos de gráficos apropiados basándose en combinación de tipos de campos seleccionados, aplicando reglas heurísticas que mapean patrones como "una dimensión más una medida" a gráfico de barras o "dos medidas" a gráfico de dispersión. Sistema proporciona galería de tipos de gráficos estándares incluyendo

barras, líneas, áreas, dispersión, burbujas, mapas, treemaps y diagramas de caja, cada uno implementado mediante plantilla configurable con parámetros ajustables. Capacidades de interacción incluyen filtrado mediante selección directa de elementos o controles de filtro dinámicos, jerarquías de drill-down que permiten navegación entre niveles de agregación desde resúmenes de alto nivel hacia detalles granulares, y acciones que vinculan múltiples visualizaciones dónde selección en una vista filtra automáticamente otras vistas coordinadas. La escalabilidad se aborda mediante arquitectura híbrida que combina extracción de datos hacia motor en memoria Hyper optimizado para consultas analíticas mediante compresión columnar y ejecución paralela vectorizada, con conexión viva a fuentes para datos cambiantes frecuentemente donde consultas se ejecutan directamente contra base de datos operacional. Hyper implementa motor de base de datos en memoria optimizado específicamente para cargas de trabajo OLAP con compresión adaptativa que selecciona algoritmo de codificación apropiado para cada columna según distribución de valores, ejecución de consultas just-in-time que compila planes a código máquina nativo, y paralelización automática que distribuye operaciones entre múltiples núcleos de procesador. Benchmarks publicados demuestran throughput de consultas superior a un millón de filas procesadas por segundo por núcleo en hardware contemporáneo, permitiendo análisis interactivo de datasets con cientos de millones de filas. Las limitaciones técnicas incluyen curva de aprendizaje pronunciada para características avanzadas como cálculos de tabla que requieren comprensión de orden de operaciones, expansión de licencias costosa que escala linealmente con número de usuarios editor limitando adopción en organizaciones grandes con presupuestos limitados, y vendor lock-in donde invertir esfuerzo sustancial en desarrollo de dashboards propietarios dificulta migración hacia plataformas alternativas. Además, aunque interfaz visual reduce barrera de entrada comparada con programación explícita, usuarios aún requieren comprensión conceptual de conceptos de análisis de datos como agregaciones, joins y filtros para uso efectivo. Power BI de Microsoft constituye una plataforma cuya arquitectura técnica implementa modelos de datos tabular basado en motor analítico xVelocity que comprime datos en memoria mediante técnicas de codificación como dictionary encoding para campos categóricos

de baja cardinalidad y run-length encoding para secuencias repetitivas, alcanzando ratios de compresión típicamente entre diez y cien veces. El lenguaje DAX (Data Analytics Xpressions) proporciona sintaxis funcional para definir medidas calculadas y columnas computadas mediante fórmulas que referencian tablas y columnas, soportando funciones de agregación, iteradores que procesan filas individualmente, y funciones de inteligencia temporal que simplifican cálculos comunes como comparaciones período-sobre-período. Las capacidades de visualización incluyen galería extensible de custom visuals desarrollados por comunidad mediante SDK de TypeScript que expone APIs para acceso a datos, rendering mediante D3.js o bibliotecas alternativas, y configuración de propiedades mediante panel de formateo. Marketplace oficial contiene cientos de visualizaciones especializadas incluyendo diagramas de Sankey para flujos, gráficos de Gantt para gestión de proyectos, y mapas coropléticos para datos geográficos con geometrías personalizadas. Integración con Azure Machine Learning permite incorporar modelos predictivos entrenados como servicios web que consumen datos de dashboards y retornan predicciones visualizables, habilitando escenarios de analítica aumentada donde insights generados algorítmicamente complementan exploración visual humana. La estrategia de precios adopta modelo freemium donde versión gratuita Power BI Desktop proporciona funcionalidad completa de autoría pero límites en compartir y colaboración, mientras suscripción paga Power BI Pro habilita publicación a servicio en nube y compartir dashboards con colegas, y suscripción Premium proporciona capacidad dedicada para organizaciones con volúmenes grandes o requisitos de rendimiento garantizado. Esta estructura incentiva adopción inicial mediante versión gratuita mientras monetiza usuarios que requieren capacidades colaborativas, generando lock-in mediante inversión en desarrollo de contenido que depende de características de plataforma propietaria.

3.2.2. Librerías de Código Abierto

En este punto se destaca D3.js, biblioteca JavaScript de visualización de datos de bajo nivel que proporciona máxima flexibilidad mediante programación directa de transformaciones de datos y manipulación del modelo de objetos del documento,

implementando paradigma de join de datos donde elementos del DOM se emparejan con elementos de array de datos y operaciones especifican qué hacer con elementos entrantes sin correspondencia DOM, elementos actualizados con correspondencia existente, y elementos salientes sin correspondencia de datos. Esta abstracción permite especificación declarativa de relación entre datos y representación visual mediante código conciso, mientras delegando control detallado de apariencia a programador mediante APIs de bajo nivel que exponen propiedades CSS y atributos SVG. La arquitectura modular organiza funcionalidad en módulos independientes incluyendo d3-selection para manipulación del DOM mediante sintaxis inspirada en jQuery, d3-scale para mapeo de dominios de datos a rangos visuales mediante escalas lineales, logarítmicas, ordinales o temporales, d3-axis para generación automática de ejes con etiquetas y marcas de graduación, d3-shape para cálculo de geometrías complejas como arcos de gráfico circular o áreas de gráfico de líneas apiladas, d3-force para simulación de grafos dirigidos por fuerza con múltiples fuerzas configurables, d3-geo para proyecciones cartográficas y manipulación de datos geoespaciales en formato GeoJSON, y d3-transition para animación de transiciones suaves entre estados visuales. Esta modularidad permite importación selectiva únicamente de módulos requeridos para aplicación particular, minimizando tamaño de bundle JavaScript descargado por navegador. Las capacidades de rendering soportan tanto gráficos vectoriales escalables mediante elementos SVG que permiten zoom sin pérdida de calidad y manipulación mediante CSS, como gráficos raster mediante Canvas HTML5 que proporciona rendimiento superior para datasets masivos con decenas de miles de elementos mediante rendering acelerado por GPU pero sin acceso a objetos individuales después de dibujado. Selección entre SVG y Canvas representa trade-off fundamental entre interactividad granular y rendimiento bruto, donde SVG facilita implementación de tooltips y selección de elementos individuales mediante listeners de eventos DOM pero limita escalabilidad, mientras Canvas permite rendering de millones de puntos con framerate fluido pero requiere implementación manual de hit testing para determinar qué elemento fue clickeado. La curva de aprendizaje pronunciada constituye limitación principal dado que D3 expone abstracciones de bajo nivel requiriendo comprensión profunda de estándares web como SVG, Canvas, DOM y

JavaScript ES6, contrastando con bibliotecas de alto nivel que encapsulan complejidad mediante APIs simplificadas. Documentación oficial proporciona ejemplos extensos y tutoriales pero asume expertise técnica sustancial, limitando accesibilidad para usuarios sin background en desarrollo web frontend. Sin embargo, la flexibilidad resultante permite la implementación de visualizaciones personalizadas arbitrariamente complejas imposibles mediante herramientas de plantillas fijas, atrayendo desarrolladores que requieren control completo sobre apariencia y comportamiento. Plotly constituye biblioteca de alto nivel disponible en múltiples lenguajes incluyendo Python, R, JavaScript y MATLAB, proporcionando API declarativa donde usuario especifica tipo de gráfico, datos de entrada y configuración de apariencia mediante diccionarios o listas, y biblioteca genera visualización interactiva renderizada mediante plotly.js en backend que construye sobre D3.js añadiendo capa de abstracción. La arquitectura implementa gramática de gráficos inspirada en ggplot2 donde visualización se compone de trazas cada una especificando datos, tipo de marca como scatter o bar, y encodings de propiedades visuales como color o tamaño, más layout especificando aspectos globales como dimensiones de figura, títulos de ejes y leyendas. Las capacidades incluyen galería comprehensiva de tipos de gráficos estándares más visualizaciones especializadas como gráficos de violín que combinan box plot con estimación de densidad kernel, mapas coropléticos que colorean regiones geográficas según valores de variable, gráficos 3D de superficie y volumen para datos tridimensionales, y dashboards con múltiples subplots organizados en rejilla configurable. Interactividad predeterminada incluye zoom y pan mediante arrastre con ratón, tooltips con valores al pasar cursor, y toggles de leyenda que muestran u ocultan trazas selectivamente. Capacidades avanzadas incluyen animación mediante slider temporal que itera frames secuencialmente visualizando evolución de datos, y widgets de control como dropdowns y sliders que actualizan visualización reactivamente según selección de usuario. La implementación en Python aprovecha integración estrecha con ecosistema científico incluyendo NumPy para arrays numéricos multidimensionales, Pandas para DataFrames tabulares con índices nombrados, y SciPy para algoritmos de interpolación y regresión frecuentemente aplicados en preprocesamiento de datos. Plotly Express proporciona interfaz de una línea para

generación rápida de gráficos comunes especificando únicamente DataFrame y nombres de columnas para mapear a ejes, simplificando uso para casos típicos mientras preservando acceso a API de bajo nivel para customización avanzada. Despliegue de visualizaciones soporta exportación a HTML standalone conteniendo JavaScript embebido para compartir mediante email o hosting en servidores web estáticos, o publicación a servicio en nube Plotly Chart Studio que proporciona hosting centralizado con control de permisos y capacidades colaborativas.

3.2.3. Entornos de Desarrollo Integrado para Análisis Estadístico

RStudio ofrece un entorno de desarrollo visual integrado para lenguaje R optimizado para análisis estadístico y visualización de datos, proporcionando interfaz unificada con editor de código con resaltado de sintaxis y autocompletado, consola interactiva para evaluación de expresiones, panel de ambiente mostrando objetos en memoria, panel de gráficos para previsualizarlos, y panel de archivos navegando sistema de archivos. La integración con biblioteca ggplot2 implementa gramática de gráficos de Wilkinson (2012) donde visualización se construye aditivamente mediante capas componibles, comenzando con especificación de datos y aesthetics que mapean variables a propiedades visuales, añadiendo geometrías que especifican marcas como puntos o barras, y aplicando escalas, temas y facetas para refinamiento. La filosofía de ggplot2 enfatiza separación de concerns donde preocupaciones ortogonales se controlan independientemente: datos y mapeos aesthetics especifican qué mostrar, geometrías especifican cómo representar, escalas especifican transformaciones de rangos, temas especifican apariencia no relacionada con datos como fuentes y colores de fondo, y facetas especifican subdivisión en paneles múltiples. Esta descomposición facilita experimentación rápida donde modificar componente individual no requiere reescribir código completo, promoviendo exploración iterativa de alternativas visuales. Extensibilidad mediante paquetes adicionales proporciona geometrías especializadas y temas personalizados desarrollados por comunidad, con repositorio CRAN conteniendo miles de paquetes cubriendo dominios diversos desde genómica hasta economía. Capacidades de publicación incluyen R Markdown que integra código R con texto narrativo en documento Markdown compilable a HTML, PDF o Word mediante knitr y

pandoc, facilitando investigación reproducible donde análisis y visualizaciones se regeneran automáticamente desde datos fuente en cada compilación. Shiny proporciona framework para construcción de aplicaciones web interactivas reactivas donde inputs de usuario actualizan automáticamente outputs dependientes mediante grafo de dependencias rastreado implícitamente, permitiendo desarrollo de dashboards sofisticados con componentes de interfaz como sliders y selectores sin requerir programación JavaScript explícita. Jupyter Notebook constituye entorno de análisis interactivo soportando múltiples lenguajes mediante kernels incluyendo Python, R y Julia, organizando trabajo como secuencia de celdas que pueden contener código ejecutable, texto Markdown con matemáticas LaTeX, o salidas como tablas, gráficos o HTML interactivo. La arquitectura cliente-servidor separa interfaz frontend ejecutando en navegador de kernels backend ejecutando código y retornando resultados mediante protocolo de mensajería JSON sobre WebSockets, permitiendo kernels remotos en servidores de computación de alto rendimiento mientras usuario interactúa localmente. Notebooks proporcionan narrativa computacional combinando explicaciones textuales con código demostrable y visualizaciones interpretativas, facilitando comunicación de hallazgos analíticos y educación mediante ejemplos interactivos ejecutables. Las capacidades de visualización en ecosistema Python incluyen Matplotlib proporcionando API de bajo nivel inspirada en MATLAB con control detallado de cada aspecto de figura, Seaborn añadiendo abstracciones de alto nivel para gráficos estadísticos comunes como distribuciones, relaciones y matrices de correlación con defaults estéticamente agradables, Bokeh generando visualizaciones interactivas con controles widgets para exploración dinámica, y Altair implementando gramática declarativa de gráficos donde especificación en JSON es independiente de implementación de rendering permitiendo portabilidad. La integración con bibliotecas de análisis de datos como Pandas, NumPy y *Scikit-learn* (Pedregosa et al., 2011) facilita flujos de trabajo integrados donde transformación, modelado y visualización fluyen naturalmente en notebooks literate programming. La limitación principal de entornos de análisis estadístico radica en curva de aprendizaje pronunciada requiriendo dominio de lenguaje de programación específico con sintaxis y semántica no triviales. Usuarios sin background en programación enfrentan barreras sustanciales antes de poder

realizar análisis independientemente, aunque esta inversión inicial se amortiza mediante flexibilidad y reproducibilidad ganadas. Además, aunque notebooks facilitan narrativas computacionales combinando código y explicaciones, mantener reproducibilidad a largo plazo resulta desafiante cuando versiones de bibliotecas evolucionan introduciendo cambios incompatibles en APIs, requiriendo gestión explícita de dependencias mediante herramientas como conda environments o contenedores Docker que capturan stack completo de software.

3.2.4. Herramientas de Arquitectura Multidimensional

La revisión sistemática de plataformas según múltiples dimensiones técnicas revela trade-offs fundamentales en su arquitectura. La dimensión de expresividad se califica según la amplitud del nivel de diseño de visualizaciones alcanzable mediante la propia personalización del sistema, donde bibliotecas programáticas de bajo nivel como D3.js ofrecen expresividad máxima permitiendo construcción de visualizaciones arbitrariamente personalizadas mediante programación directa del DOM, mientras otras plataformas de alto nivel como Tableau restringen la expresividad a una galería predefinida de plantillas configurables sacrificando flexibilidad por simplicidad. La medición cuantitativa de expresividad puede aproximarse mediante conteo de tipos de gráficos soportados nativamente, aunque esta métrica captura inadecuadamente diferencias cualitativas dónde sistema expresivo permite composición creativa de primitivas elementales generando infinitas variaciones. La dimensión de usabilidad indica la facilidad de uso para usuarios con diferentes niveles de expertise técnica, medible mediante métricas como tiempo requerido para completar tareas representativas, tasa de errores cometidos durante interacción, y satisfacción subjetiva reportada en cuestionarios estandarizados como System Usability Scale. Plataformas con interfaces visuales de manipulación directa típicamente exhiben usabilidad superior para usuarios novatos comparadas con herramientas requiriendo programación, pero usuarios expertos frecuentemente reportan frustración con limitaciones de interfaces gráficas prefiriendo control programático explícito. Esta divergencia sugiere que sistemas óptimos deberían proporcionar múltiples modalidades de interacción acomodando preferencias heterogéneas. La dimensión de

rendimiento identifica latencias de respuesta y throughput de procesamiento mediante benchmarks ejecutados sobre datasets de magnitudes variadas. Sistemas basados en motores en memoria optimizados como Hyper de Tableau demuestran rendimiento superior procesando millones de filas por segundo, mientras sistemas interpretados como notebooks Python exhiben rendimiento inferior limitado por overhead de interpretación aunque suficiente para datasets de tamaño moderado típicos en análisis exploratorio. Escalabilidad horizontal mediante paralelización distribuida permite algunos sistemas manejar datasets masivos que exceden capacidad de memoria de máquina individual, aunque coordinación distribuida introduce complejidad operacional sustancial. La dimensión de costo considera tanto costos monetarios directos de licencias comerciales como costos indirectos de tiempo de aprendizaje y desarrollo. Plataformas comerciales como Tableau y Power BI requieren suscripciones anuales escalan linealmente con número de usuarios, típicamente cientos a miles de dólares por usuario anualmente, mientras bibliotecas de código abierto como D3.js, Plotly y ggplot2 son gratuitas eliminando costos de licenciamiento pero requiriendo inversión sustancial en desarrollo de expertise programática y construcción de visualizaciones desde primitivas de bajo nivel. Análisis de costo total de propiedad debe considerar ambos factores mediante modelos económicos que balancean gastos de capital versus gastos operacionales a lo largo del ciclo de vida del proyecto.

3.3. Inteligencia Artificial Aplicada en Visualización

La integración de técnicas de inteligencia artificial en sistemas de visualización representa una frontera activa de investigación que promete un horizonte de automatización de tareas que tradicionalmente requieren expertise humana. Esta sección examina arquitecturas técnicas, algoritmos de inferencia y capacidades de sistemas contemporáneos que explotan modelos de lenguaje extensos para generación y recomendación de visualizaciones.

3.3.1. Arquitecturas de Integración LLM-Visualización

Los sistemas que integran modelos de lenguaje extensos para tareas de visualización

típicamente implementan arquitecturas pipeline con etapas especializadas procesando progresivamente desde entrada en lenguaje natural hasta especificación de visualización ejecutable. LIDA desarrollado por Dibia (2023) materializa arquitectura modular con cuatro componentes desacoplados comunicándose mediante interfaces JSON bien definidas. El componente Summarizer analiza dataset de entrada generando resumen textual comprensivo mediante invocación de GPT-4 (OpenAI, 2023) con prompt estructurado que instruye modelo describir esquema identificando nombres y tipos de campos, estadísticos descriptivos calculando medias, medianas y desviaciones estándar para campos numéricos y cardinalidades para campos categóricos, y patrones notables como correlaciones fuertes entre pares de campos o presencia de valores atípicos detectados mediante análisis estadístico. Este resumen textual proporciona contexto condensado para etapas subsecuentes evitando necesidad de transmitir dataset completo en cada invocación del modelo, optimizando consumo de tokens y reduciendo latencias. El componente Goal Generator infiere automáticamente objetivos analíticos interesantes que usuario podría tener sobre datos mediante prompt que instruye modelo considerar múltiples tipos de análisis cubriendo espectro de tareas de bajo nivel de Amar et al. (2005), incluyendo comparaciones entre categorías identificando diferencias en medidas cuantitativas entre grupos definidos por variables categóricas, evolución temporal cuando datos contienen dimensión temporal trazando tendencias seculares y patrones estacionales, distribuciones univariadas caracterizando forma de histogramas o estimaciones de densidad kernel, relaciones bivariadas explorando correlaciones mediante gráficos de dispersión, y composiciones mostrando partes de un todo mediante gráficos circulares o de áreas apiladas. Cada objetivo generado se representa como objeto JSON especificando pregunta en lenguaje natural, variables relevantes del dataset, y tipo de análisis abstracto facilitando mapeo subsecuente a técnica de visualización. El modelo genera múltiples objetivos candidatos con diversidad controlada mediante muestreo con temperatura elevada, permitiendo al usuario seleccionar objetivos de interés desde la galería presentada. El componente Visualization Generator traduce objetivo seleccionado a especificación ejecutable en gramática declarativa Vega-Lite mediante prompting few-shot donde se proporcionan ejemplos demostrativos de traducciones

correctas desde descripciones de objetivos a especificaciones JSON. Vega-Lite constituye lenguaje de especificación de alto nivel donde visualización se describe mediante objeto JSON especificando datos de entrada, marcas gráficas como point o bar, encodings que mapean campos de datos a canales visuales como x, y, color o size, y transformaciones opcionales como agregación o filtrado. La elección de Vega-Lite como lenguaje objetivo equilibra expresividad permitiendo especificar amplia variedad de visualizaciones versus simplicidad facilitando generación confiable por modelo de lenguaje sin requerir código procedural complejo. Especificación generada se válida sintácticamente mediante parser JSON y semánticamente verificando que campos referenciados existen en esquema de datos, rechazando especificaciones inválidas y re-invocando modelo con mensaje de error para corrección. El componente Refinement implementa diálogo interactivo multi-turno donde usuario proporciona retroalimentación en lenguaje natural sobre visualización renderizada y modelo modifica especificación acordemente. El historial conversacional se mantiene como secuencia de mensajes contextualizando cada modificación respecto a especificación actual y cambios previos, permitiendo referencias anafóricas como "hazlo más grande" que requieren resolver "lo" a elemento específico basándose en contexto. La arquitectura conversacional facilita refinamiento iterativo sin requerir que el usuario comprenda sintaxis de Vega-Lite o terminología técnica de visualización, reduciendo barrera de entrada para usuarios no expertos mientras preservando control sobre resultado final mediante feedback dirigido. La implementación técnica emplea con la API de OpenAI con el modelo GPT-4 (OpenAI, 2023) invocado mediante llamadas HTTP POST transmitiendo prompts estructurados y recibiendo completions generadas. Gestión de latencias se aborda mediante caching de resúmenes generados para datasets frecuentemente accedidos evitando regeneración redundante, y ejecución paralela de generación de múltiples objetivos candidatos distribuyendo invocaciones independientes. Gestión de costos se implementa mediante límites configurables en número de tokens procesados por sesión y número de objetivos generados, balanceando calidad de recomendaciones versus gastos acumulativos de uso de API comercial.

3.3.2. Ingeniería de Prompts

El diseño sistemático de prompts que explicitan comportamientos deseados de modelos de lenguaje constituye un factor crítico determinando la calidad de salidas generadas. Wang et al. (2023) en su sistema LLM4Vis documentan estrategias de ingeniería de prompts optimizadas empíricamente mediante experimentación iterativa. La estructura jerárquica organiza prompts en tres niveles con responsabilidades distintas: mensaje de sistema establece rol y directivas comportamentales persistentes a través de conversación completa, típicamente especificando "Eres experto en análisis de datos y visualización de información con comprensión profunda de principios de percepción visual humana y mejores prácticas de diseño gráfico. Tu tarea es recomendar visualizaciones apropiadas dado dataset y objetivo analítico, proporcionando justificaciones claras de tus recomendaciones basadas en principios establecidos de efectividad perceptual." Los mensajes de usuario proporcionan contexto específico de tarea incluyendo descripción de dataset mediante esquema con nombres de campos, tipos semánticos inferidos mediante análisis de contenido más allá de tipos sintácticos declarados, y estadísticos descriptivos como rangos, cardinalidades, distribuciones y correlaciones entre pares de campos. La inclusión de estadísticos descriptivos resulta crucial dado que permite modelo razonar sobre propiedades de datos sin acceder directamente a valores crudos, facilitando recomendaciones contextuales como sugerir transformación logarítmica cuando distribución presenta sesgo fuerte o filtrado de outliers cuando valores atípicos distorsionan escalas. Ejemplos demostrativos implementan aprendizaje por pocos ejemplos proporcionando pares de entrada-salida representativos que ilustran formato deseado de recomendaciones, típicamente tres a cinco ejemplos cubriendo variedad de tipos de datos y tareas analíticas. Investigación empírica de Brown et al. (2020) sobre GPT-3 demuestra que inclusión de ejemplos demostrativos mejora significativamente precisión de tareas estructuradas comparado con prompting zero-shot sin ejemplos, con retornos decrecientes después de aproximadamente cinco ejemplos donde ejemplos adicionales contribuyen marginalmente mientras consumiendo tokens escasos de ventana de contexto. El formateo de salidas específica mediante instrucciones explícitas qué respuesta debe estructurarse como JSON válido con esquema predefinido, facilitando parsing automático confiable mediante

bibliotecas JSON estándares sin requerir análisis sintáctico ad-hoc de texto libre. Esquema típico incluye campos obligatorios como visualization_type seleccionado de vocabulario controlado de tipos como scatter, bar, line, histogram, box, bubble, heatmap permitiendo mapeo determinístico a implementación, rationale proporcionando justificación textual explicando por qué visualización particular es apropiada citando principios específicos como "gráfico de barras facilita comparaciones precisas de magnitudes mediante juicios de longitud a lo largo de escala común, más preciso perceptualmente que juicios de ángulo en gráfico circular según jerarquía de Cleveland y McGill", variable_mappings especificando qué campos del dataset mapear a qué encodings visuales como {"x": "fecha", "y": "temperatura", "color": "estacion"}, y configuration_options sugiriendo parámetros adicionales como escalas logarítmicas o paletas de colores apropiadas. Las técnicas de validación y corrección automática manejan casos donde el modelo genera salidas malformadas mediante pipeline de procesamiento robusto. Parsing de JSON con manejo de excepciones captura errores sintácticos como comillas faltantes o comas finales, extrayendo bloques JSON válidos mediante expresiones regulares cuando la respuesta contiene texto adicional antes o después de JSON. Validación semántica verifica que tipos de visualización especificados existen en vocabulario soportado, campos referenciados en mappings corresponden a campos presentes en esquema de datos, y configuraciones propuestas son válidas para tipo de visualización seleccionado. Cuando validación falla, el sistema genera mensaje de error descriptivo y reinvoca el modelo con contexto ampliado incluyendo error específico encontrado y solicitud de corrección, implementando loop de refinamiento automático que típicamente converge después de una o dos iteraciones.

3.4. Visualización de Datos Enlazados

Los sistemas especializados para visualización de datos enlazados abordan desafíos únicos derivados de naturaleza de grafo de RDF y semántica rica codificada en ontologías, requiriendo arquitecturas técnicas y algoritmos específicamente diseñados para procesamiento eficiente de tripletas y explotación de estructura semántica. La escalabilidad en visualización de grafos de conocimiento constituye un desafío técnico significativo. Bikakis et al. (2014) desarrollan GraphViz, plataforma escalable para

visualización interactiva de grafos grandes que implementa estrategias de renderizado incremental y técnicas de agregación visual para mantener interactividad con grafos de millones de nodos. Esta sección examina además aproximaciones representativas organizadas por su funcionalidad. Clark et al. (2021) presenta una herramienta de visualización de datos de basura marina generados por ciencia ciudadana, centrada en el análisis colaborativo.

3.4.1. Sistemas de Detección Automática

LDVizWiz desarrollado por Atemezing y Troncy (2014) implementa una aproximación de detección automática de categorías de datos mediante consultas SPARQL de tipo ASK que verifican la presencia de patrones específicos en grafos RDF, clasificando recursos en categorías semánticas predefinidas cada una asociada con visualizaciones apropiadas. La arquitectura ejecuta una secuencia de consultas ASK contra un endpoint SPARQL del dataset analizado, cada consulta implementa un test de membership para categoría particular. Por ejemplo, la categoría Geography se detecta mediante consulta verificando si el grafo contiene tripletas con predicados de vocabularios geoespaciales como wgs84:lat y wgs84:long de W3C Basic Geo, o geo:Point de ontología GeoSPARQL, indicando presencia de coordenadas espaciales visualizables mediante mapas. Categoría Temporal se detecta verificando predicados temporales como dcterms:temporal, xsd:date o xsd:dateTime indicando información temporal visualizable mediante líneas de tiempo. Categorías adicionales incluyen Event detectada mediante clases como schema:Event o lode:Event de ontologías de eventos, Agent/Person detectada mediante foaf:Person o schema:Person, Organization mediante foaf:Organization o org:Organization de vocabulario de organizaciones, Statistics detectada mediante qb:Observation de vocabulario RDF Data Cube para datos estadísticos multidimensionales, y Knowledge detectada mediante presencia de jerarquías taxonómicas expresadas mediante rdfs:subClassOf o skos:broader indicando estructura conceptual visualizable mediante treemaps o sunbursts. Para cada categoría detectada positivamente, sistema recupera instancias relevantes mediante consultas SELECT y genera visualización apropiada: mapas interactivos para datos geográficos posicionando marcadores según coordenadas, líneas de tiempo para datos temporales

ordenando eventos cronológicamente, redes para datos de personas y organizaciones conectando mediante relaciones sociales o afiliaciones, y jerarquías visuales para datos de conocimiento mostrando estructura taxonómica. La implementación técnica optimiza rendimiento mediante ejecución paralela de consultas ASK independientes distribuyendo carga entre múltiples threads, y caching de resultados de detección para datasets accedidos frecuentemente evitando re-ejecución de consultas costosas. Limitaciones incluyen dependencia de vocabularios específicos codificados en consultas requiriendo extensión manual para reconocer esquemas ontológicos nuevos no anticipados durante diseño, y granularidad gruesa de categorías que agrupa tipos semánticamente diversos bajo etiqueta amplía potencialmente oscureciendo distinciones importantes. Por ejemplo, categoría Geography agrupa datos georreferenciados como ubicaciones de estaciones de muestreo con datos reales como bounding boxes de regiones administrativas requiriendo técnicas de visualización diferentes, pero el sistema actual no distingue automáticamente subcategorías.

3.4.2. Sistemas de Especificación Declarativa

Sgvizler desarrollado por Skjæveland (2012) proporciona especificación declarativa de visualizaciones mediante anotaciones HTML con atributos data-* personalizados donde desarrolladores insertan consultas SPARQL directamente en marcado de página web, el sistema ejecuta consultas contra endpoints remotos especificados, y resultados se visualizan automáticamente mediante bibliotecas JavaScript configurables. La arquitectura implementa parser de atributos data-* que extrae configuración desde elementos HTML, motor de ejecución de consultas que invoca endpoints SPARQL mediante HTTP GET o POST con parámetros codificados según especificación del protocolo. La implementación JavaScript se carga mediante tag script en encabezado HTML, registra listener de evento DOMContentLoaded que ejecuta cuando página completa carga, escanea DOM identificando elementos con atributos data-sgvizler-*, y para cada elemento invoca secuencia de procesamiento incluyendo parsing de consulta SPARQL validando sintaxis, ejecución de solicitud HTTP hacia endpoint con negociación de contenido solicitando formato JSON preferentemente o XML alternativamente, parsing de respuesta extrayendo bindings de variables, transformación de bindings a

formato de datos esperado por biblioteca de visualización target mediante adaptadores específicos que mapean estructura de resultados SPARQL a arrays de valores requeridos, invocación de API de visualización pasando datos transformados y opciones configuradas, y rendering en contenedor DOM especificado. Manejo de errores captura fallos en cualquier etapa mostrando mensajes descriptivos en elemento HTML facilitando el debugging. Sgvizler soporta Cross-Origin Resource Sharing mediante solicitudes preflight OPTIONS verificando que endpoint permite acceso desde origen de página web, requiriendo endpoint esté configurado con headers CORS apropiados Access-Control-Allow-Origin especificando dominios permitidos o wildcard para acceso público irrestricto. Esta restricción de seguridad del navegador previene scripts maliciosos consultar endpoints arbitrarios sin permiso, pero también limita usabilidad cuando endpoints legítimos no configuran CORS correctamente requiriendo workarounds como proxies CORS o restricción a endpoints conocidos con CORS habilitado. Limitación adicional requiere usuarios tengan conocimiento sustancial de SPARQL para escribir consultas sintácticamente correctas y semánticamente apropiadas, elevando barrera de entrada para usuarios sin expertise en tecnologías semánticas.

3.4.3. Sistemas con Modelo Formal de Visualización

LODVisualization desarrollado por Brunetti et al. (2013) implementa Modelo de Visualización de Datos Enlazados basado en adaptación del Data State Reference Model de Chi específicamente diseñada para características de LOD. La arquitectura define pipeline de transformaciones desde datos RDF crudos hasta vistas renderizadas mediante cuatro estados intermedios: estado de datos enlazados originales como colección de tripletas sujeto-predicado-objeto posiblemente distribuidas entre múltiples grafos remotos, estado de datos analíticos donde se aplican consultas SPARQL filtrando, agregando o derivando información, estado de datos de visualización donde se mapean resultados tabulares de consultas a marcas y canales gráficos abstractos, y estado de vistas renderizadas produciendo píxeles en pantalla mediante biblioteca de rendering. El sistema proporciona visualizaciones de overview del esquema ontológico incluyendo jerarquía de clases mediante treemap donde cada

clase ocupa área proporcional a número de instancias y anidamiento representa relaciones rdfs:subClassOf, permitiendo comprensión rápida de estructura taxonómica y distribución de población entre tipos. Visualización de propiedades compartidas entre pares de clases muestra mediante diagrama de Venn o tabla de contingencia qué predicados conectan instancias de diferentes tipos, revelando patrones de uso de vocabulario como propiedades utilizadas consistentemente versus propiedades específicas de subconjunto de clases. Análisis de grado de nodos identifica recursos con mayor número de conexiones entrantes in-degree o salientes out-degree mediante consultas agregando conteos de tripletas por sujeto u objeto, visualizando top-k recursos más conectados mediante gráfico de barras ordenado que destaca hubs importantes en red de conocimiento. Perfilado de rendimiento mediante instrumentación de tiempo de ejecución identifica consultas lentas sugiriendo índices adicionales o reformulaciones alternativas. Payola desarrollado por Klímek et al. (2014) proporciona una implementación refinada y extendida del LDVM permitiendo a los usuarios construir pipelines de análisis visuales mediante interfaz gráfica de composición donde componentes modulares representados como nodos se conectan entre sí especificando el flujo de datos. Cada componente utiliza metadata describiendo los inputs esperados y outputs producidos, permitiendo al sistema validar composiciones verificando compatibilidad de tipos entre componentes conectados y alertando sobre incompatibilidades antes de su ejecución. La arquitectura cliente-servidor separa la interfaz frontend basada en JavaScript ejecutando en el navegador de backend Scala ejecutando flujos de trabajo en el servidor, permitiendo compartir mediante URLs únicas que codifican la especificación de componentes y conexiones. Esta arquitectura fomenta las capacidades colaborativas que permiten a múltiples usuarios trabajar en análisis compartido con edición concurrente coordinada mediante transformaciones operacionales preservando la consistencia aún cuando modificaciones simultáneas ocurren.

CAPÍTULO

4 CASO DE ESTUDIO

4.1. Introducción

Este capítulo presenta el caso de estudio desarrollado como validación empírica de las contribuciones teóricas y metodológicas articuladas en capítulos precedentes, documentando exhaustivamente el sistema VisBot (Ceballos, 2025) como materialización concreta de un sistema de recomendación de visualizaciones basado en modelos de lenguaje extensos. El desarrollo completo está explicado en Ceballos et al. (2025), este artículo contiene material complementario proporcionado por los autores disponible en https://doi.org/10.5281/zenodo.15176569. Se puede acceder a la plataforma en https://visbot.streamlit.app/ y el repositorio está disponible públicamente en https://github.com/disenodc/VisBot. El acuerdo de licencia completo está disponible en https://github.com/disenodc/VisBot/blob/main/LICENSE.

La investigación aborda la problemática fundamental de automatización inteligente en selección de técnicas de visualización apropiadas para conjuntos de datos científicos heterogéneos, particularmente en contexto de analítica de datos donde usuarios con diversos niveles de expertise técnica requieren herramientas accesibles que faciliten exploración y comprensión de información compleja sin imponer barreras de conocimiento especializado en programación o principios de diseño visual. La motivación central del caso de estudio radica en la brecha persistente entre capacidades técnicas disponibles en sistemas contemporáneos de visualización y su adopción efectiva por parte de científicos y analistas de datos trabajando con información real. Las herramientas tradicionales de visualización, aunque técnicamente sofisticadas, típicamente requieren que usuarios posean conocimiento previo sustancial sobre tipos de gráficos apropiados para diferentes combinaciones de variables, comprendan estructura interna de sus datos incluyendo tipos semánticos

más allá de tipos sintácticos declarados, y realicen manualmente procesos complejos de transformación de datos y mapeo de variables a encodings visuales específicos. Esta carga cognitiva resulta particularmente problemática para usuarios sin formación formal en visualización de información o estadística, limitando severamente su capacidad de extraer insights valiosos desde datos disponibles. El sistema VisBot (Ceballos, 2025) desarrollado como parte de esta investigación doctoral implementa aproximación novedosa que integra modelos de lenguaje extensos pre-entrenados, específicamente GPT-4 de OpenAI, con principios establecidos de visualización efectiva derivados de décadas de investigación empírica en percepción visual y cognición gráfica. La arquitectura del sistema explota capacidades de razonamiento contextual de modelos de lenguaje para analizar automáticamente características estructurales y semánticas de conjuntos de datos, inferir objetivos analíticos implícitos basándose en patrones detectados, y generar recomendaciones de visualizaciones apropiadas acompañadas de justificaciones textuales explicando razonamiento subyacente. Esta aproximación híbrida combina automatización inteligente que reduce barreras de entrada con preservación de agencia del usuario mediante provisión de múltiples alternativas y capacidades de customización interactiva. La validación empírica del sistema comprende evaluación multifacética mediante métricas cuantitativas objetivas que miden precisión, relevancia y diversidad de recomendaciones generadas, así como estudios de usabilidad controlados con participantes representativos de poblaciones objetivo que cuantifican efectividad práctica del sistema para completar tareas analíticas reales.

4.2. Contexto y Motivación

El contexto que motiva el desarrollo e implementación de VisBot (Ceballos, 2025) se fundamenta en la convergencia de tres tendencias tecnológicas emergentes que actualmente han creado oportunidades sin precedentes para la democratización de las capacidades de analítica visual avanzada. La primera tendencia corresponde a la maduración de modelos de lenguaje extensos basados en arquitecturas que han demostrado capacidades sorprendentes de razonamiento sobre descripciones en

lenguaje natural, comprensión de estructuras de datos tabulares expresadas textualmente, y generación de código ejecutable en múltiples lenguajes de programación y gramáticas declarativas. Específicamente, GPT-4 desarrollado por OpenAI representa la popularización de modelos de lenguaje generales con parámetros del orden de trillones de datos entrenados sobre corpus masivos que abarcan el conocimiento factual enciclopédico, patrones de razonamiento lógico, y expertise técnica en dominios especializados incluyendo visualización de datos y análisis estadístico. La segunda tendencia corresponde a la proliferación de datos científicos heterogéneos generados por instrumentación avanzada, plataformas de observación remota, y redes de sensores distribuidos que producen volúmenes exponencialmente crecientes de información requiriendo herramientas sofisticadas para gestión, análisis e interpretación. Particularmente en ciencias ambientales y marinas, múltiples disciplinas generan datos con características estructurales diversas incluyendo series temporales de variables físico-químicas, distribuciones espaciales de especies biológicas, trayectorias de plataformas móviles, e imágenes satelitales multiespectrales, cada tipo de dato requiriendo técnicas de visualización especializadas para revelar patrones significativos. La heterogeneidad tipológica, espacial y temporal de estos datos complica significativamente su análisis integrado, creando necesidad urgente de sistemas que faciliten una exploración intuitiva sin requerir un nivel de pericia técnica prohibitiva. La tercera tendencia corresponde al creciente reconocimiento en la comunidad científica de la importancia de los principios FAIR para gestión de datos que enfatizan que activos digitales deben ser localizables mediante identificadores persistentes, accesibles a través de protocolos estandarizados, interoperables mediante uso de vocabularios compartidos, y reutilizables gracias a licencias claras y metadatos ricos. El paradigma de datos enlazados abiertos materializa estos principios mediante la representación de información como grafos RDF donde los recursos se identifican mediante URIs y relaciones se definen explícitamente usando vocabularios ontológicos formales. Esta estructuración semántica facilita la integración de fuentes heterogéneas y el razonamiento automatizado sobre contenido, pero introduce complejidad adicional dado que la visualización efectiva de grafos de conocimiento requiere técnicas que respeten la naturaleza relacional de datos más allá

de aproximaciones tabulares tradicionales. La motivación específica para el desarrollo de VisBot (Ceballos, 2025) emerge desde la identificación de vacíos significativos en las capacidades de sistemas existentes revelados mediante una revisión sistemática de la literatura académica y una evaluación comparativa de herramientas disponibles comercialmente o como prototipos de investigación. Sistemas de visualización tradicionales como Tableau y Microsoft Power BI, aunque técnicamente sofisticados y ampliamente adoptados en industria, requieren una inversión sustancial de tiempo para dominar conceptos de agregación de datos, jerarquías de drill-down, y mapeos entre variables y propiedades visuales, limitando accesibilidad para usuarios ocasionales o novatos. Además, estos sistemas comerciales imponen costos de licenciamiento significativos que escalan linealmente con número de usuarios, resultando prohibitivos para instituciones con presupuestos limitados característicos de universidades públicas y centros de investigación en países en desarrollo. Sistemas académicos recientes que integran modelos de lenguaje para generación automática de visualizaciones como LIDA desarrollado por Dibia(2023) y LLM4Vis propuesto por Wang et al. (2023) demuestran viabilidad técnica de aproximación basada en inteligencia artificial, pero permanecen como prototipos de investigación con limitaciones significativas en robustez, documentación y soporte para casos de uso diversos. Específicamente, estos sistemas típicamente carecen de capacidades especializadas para datos enlazados que resultan críticas para aplicaciones científicas contemporáneas, y evaluaciones empíricas reportadas frecuentemente se limitan a conjuntos de datos sintéticos o benchmarks artificiales sin validación con usuarios reales trabajando en problemas auténticos de investigación. Herramientas especializadas para visualización de datos enlazados como LODVizWiz, Sgvizler y Payola abordan desafíos específicos de procesamiento de grafos RDF y explotación de semántica ontológica, pero típicamente requieren conocimiento sustancial de tecnologías de Web Semántica incluyendo lenguaje de consulta SPARQL, modelado mediante ontologías, y vocabularios RDF estándares, elevando barreras de entrada para usuarios sin formación en ingeniería de conocimiento. Además, estas herramientas frecuentemente implementan lógica de recomendación mediante reglas heurísticas codificadas manualmente que carecen de flexibilidad para adaptarse a

características novedosas de datos o preferencias individuales de usuarios. VisBot (Ceballos, 2025) aborda estos vacíos mediante diseño arquitectónico que integra componentes complementarios: módulo de análisis exploratorio automatizado que detecta tipos semánticos de variables, calcula estadísticos descriptivos, e identifica patrones como correlaciones fuertes o distribuciones anómalas sin requerir especificación manual por usuario; motor de recomendación basado en GPT-4 (OpenAI, 2023) que razona sobre características de datos expresadas textualmente y genera sugerencias de visualizaciones diversas acompañadas de justificaciones interpretables; generador de visualizaciones interactivas implementado mediante Plotly Express que traduce recomendaciones abstractas a gráficos concretos renderizados con capacidades de zoom, pan y tooltips; y panel de customización que permite usuarios ajustar parámetros visuales mediante controles intuitivos sin programación. Esta arquitectura modular proporciona balance entre automatización que reduce carga cognitiva y control explícito que preserva la agencia del usuario. La motivación adicional deriva de la necesidad de validación empírica rigurosa que trascienda evaluaciones informales características de muchos prototipos académicos. La investigación implementa protocolo de evaluación comprehensivo que combina métricas cuantitativas automatizadas midiendo precisión técnica de recomendaciones con estudios de usabilidad controlados que involucran participantes representativos completando tareas realistas bajo observación sistemática. Esta aproximación metodológica multifacética proporciona evidencia robusta sobre efectividad práctica del sistema en condiciones de uso auténtico, identificando simultáneamente fortalezas validadas empíricamente y limitaciones que requieren atención en iteraciones futuras.

4.3. Arquitectura y Componentes Funcionales

VisBot (Ceballos, 2025) se constituye como una plataforma web de recomendación de visualizaciones que integra LLM con procesamiento de datos mediante arquitectura modular implementada en Python y desplegada sobre plataforma Streamlit Cloud (Streamlit Inc., 2023) disponible en https://streamlit.io/ . El diseño prioriza tres principios fundamentales de ingeniería de software: modularidad mediante desacoplamiento de componentes con responsabilidades bien definidas

comunicándose a través de interfaces explícitas, extensibilidad permitiendo incorporación de nuevas capacidades sin modificación de componentes existentes, y eficiencia computacional mediante optimizaciones algorítmicas y resguardo temporal de memoria, o caché, estratégico que garantizan tiempos de respuesta interactivos.

4.3.1 Selección del Dataset para el caso de estudio

El caso de estudio se implementa utilizando el dataset “Locations of Seals in Patagonian Large Marine Ecosystem” (Campagna et al., 2015), dataset científico real proveniente del Centro Nacional Patagónico (CENPAT-CONICET) que contiene registros de ocurrencia de mamíferos marinos, más específicamente localizaciones costeras y marinas del león marino sudamericano (Otaria flavescens), los lobos marinos (Arctocephalus australis) y los elefantes marinos del sur (Mirounga leonina), en el ecosistema marino patagónico, accesible mediante GBIF (DOI: 10.15468/qcwa73). Este dataset resulta particularmente relevante para validar las capacidades del sistema en el dominio de ciencias marinas, presentando desafíos específicos como datos geoespaciales con coordenadas de alta precisión, información temporal de avistamientos, clasificación taxonómica de especies, y metadatos de campañas de observación característicos de investigación oceanográfica, permitiendo reproducibilidad de experimentos dado que ambos conjuntos de datos son públicamente accesibles. En tanto que, los resultados de la evaluación demuestran que el sistema alcanza niveles satisfactorios de rendimiento en múltiples dimensiones de calidad incluyendo relevancia de recomendaciones con puntuación media de 4.3 en escala Likert de 1 a 5 según juicio de expertos, diversidad de sugerencias generando promedio de cuatro tipos distintos de visualización por dataset evitando monotonía, y usabilidad percibida con System Usability Scale de 78.5 puntos de 100 posibles clasificado como "bueno" según puntos de referencia establecidos. El análisis comparativo con sistemas alternativos revela que VisBot (Ceballos, 2025) ocupa un nicho distintivo combinando accesibilidad de interfaz visual con automatización inteligente mediante inteligencia artificial, diferenciándose de plataformas comerciales establecidas mediante una arquitectura de bajo costo y de prototipos académicos previos mediante un nivel de implementación enfocado en la accesibilidad a distintos

niveles de usuario para desarrollar un adecuado análisis de datos.

4.3.2. Arquitectura de Sistema y Flujo de Datos

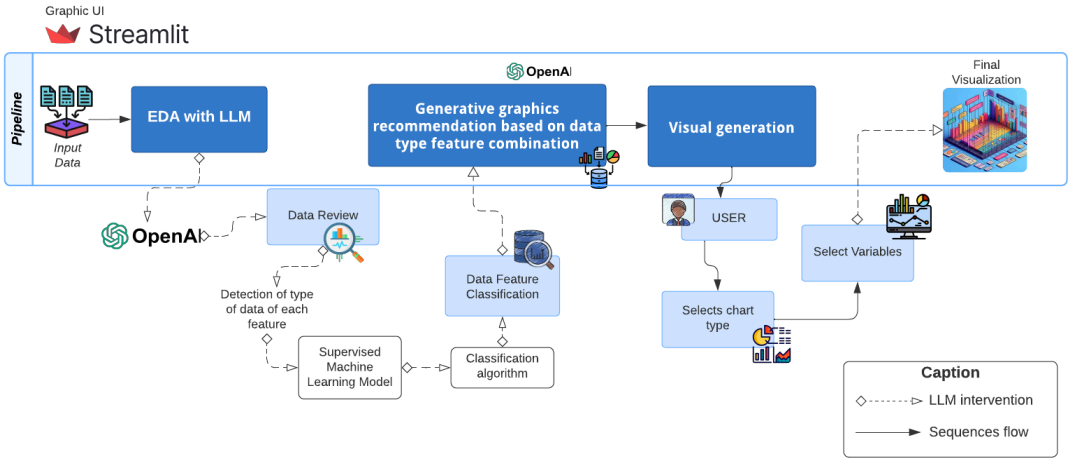


FIGURA 1. Arquitectura de VisBot: 1) EDA con LLM; 2) recomendación de gráficos generativos basada en la combinación de características del tipo de datos; y 3) generación visual y mejoras adicionales; 4) visualización final generada.

Como se puede observar en la Figura 1, la arquitectura de VisBot (Ceballos, 2025) implementa un flujo de desarrollo con cinco etapas secuenciales procesando progresivamente incluyendo desde la ingesta con datos crudos hasta visualizaciones renderizadas. La primera etapa de Carga de Datos se desarrolla a través de interfaz gráfica como se muestra en Figura 2, basada en el framework (Streamlit Inc., 2023)

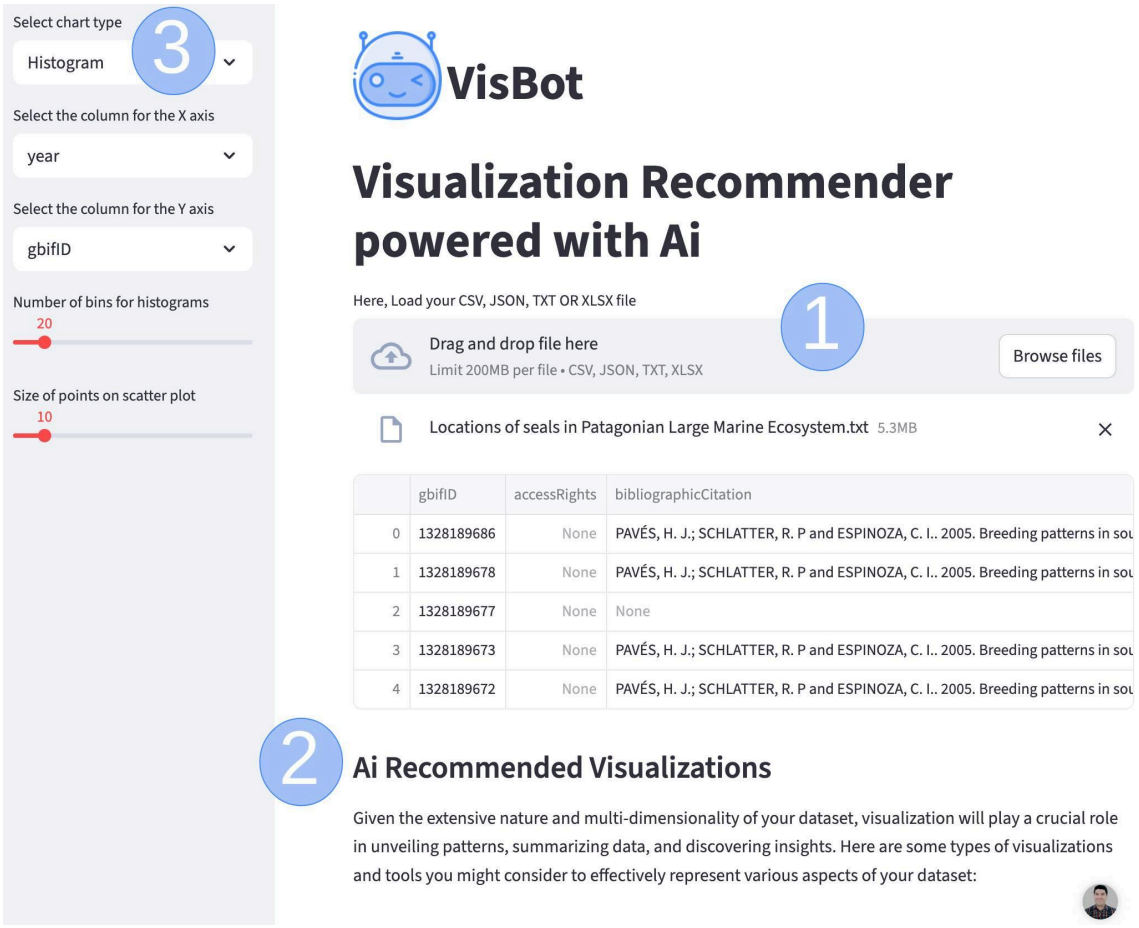


FIGURA 2. Interfaz de usuario de la plataforma VisBot y sus módulos principales: 1) Cargador de datos, 2) Visualizaciones recomendadas por IA, y 3) Módulo de configuración de gráficos.

El sistema implementa parsers especializados para múltiples formatos de entrada mediante abstracción polimórfica dónde la clase base DataLoader define la interfaz común con método load() retornando los datos estructurados en dos dimensiones a través de la librería Pandas, es decir, genera una versión compatible en python a través de la estructura estandarizada de Data Frame independientemente del formato específico del dataset cargado. Presenta las subclases de interpretación y lectura de datos, CSV Loader, ExcelLoader y JSON Loader, las cuales implementan la lógica específica de parsing manejando particularidades sintácticas de cada formato, como detección automática de delimitador para CSV mediante análisis de frecuencia de caracteres en primeras filas, o iteración sobre hojas múltiples en archivos Excel consolidando el conjunto de valores en un DataFrame único. El módulo soporta además la carga desde URLs remotas mediante biblioteca requests con manejo robusto

de timeouts y reintentos exponenciales ante fallos transitorios de red. Para datos enlazados en formato RDF, módulo RDFLoader utiliza biblioteca rdflib proporcionando parsers estándares para RDF/XML, Turtle, N-Triples y JSON-LD. El procesamiento de grafos RDF requiere transformación a representación tabular compatible con análisis subsecuente, implementada mediante consultas SPARQL que extraen propiedades de recursos específicos organizándose en filas. El método de aplanamiento proyecta un grafo tridimensional de tripletas sujeto-predicado-objeto hacia tabla bidimensional donde cada fila representa un recurso individual y las columnas correspondientes a predicados frecuentes, con celdas conteniendo objetos asociados. Las propiedades multivaluadas donde cada recurso tiene múltiples valores para el mismo predicado que se manejan concatenando valores en strings delimitados o creando filas duplicadas según configuración, lo cual afecta al balance entre preservación de información versus simplicidad estructural.

La segunda etapa de Análisis Exploratorio Automatizado aplica estadística descriptiva y técnicas de machine learning no supervisado para interpretar y caracterizar cada variable contenida en el dataset. El módulo implementa la clase DataProfiler que calcula comprehensivamente cada columna según, tipo sintáctico declarado mediante introspección de dtype de Pandas distinguiendo int64, float64, object, datetime64, cardinalidad como número de valores únicos cuantificando diversidad, completitud como porcentaje de valores no nulos identificando columnas con datos faltantes extensos, y estadísticos descriptivos incluyendo mínimo, máximo, media, mediana, moda, desviación estándar, asimetría y curtosis para variables numéricas. Para variables categóricas, calcula distribución de frecuencias identificando valores más comunes y detecta categorías raras con frecuencia menor a umbral configurable. La detección automática de tipos semánticos implementa heurísticas que infieren el significado de columnas más allá de los tipos sintácticos declarados. El detector de fechas aplica parsing mediante dateutil.parser intentando interpretar strings como timestamps según múltiples formatos estándares como ISO 8601, formatos localizados variados, y situaciones ambiguas donde orden de día-mes-año se infiere mediante metaheurística. El detector geoespacial identifica campos de datos como coordenadas o referencias de geolocalización, mediante análisis de nombres de columnas buscando

términos como latitude, longitude, lat, lon mediante expresiones regulares case-insensitive, y validación de rangos verificando valores numéricos caen dentro de límites geográficos válidos entre -90 y +90 para latitud y -180 y +180 para longitud. El detector de URIs busca identificadores de recursos mediante validación de sintaxis URI usando expresiones regulares que verifican presencia de esquema http:// o https:// seguido de dominio válido. El detector de taxonomías infiere jerarquías cuando columnas contienen términos científicos estructurados como nombres binomiales de especies.

1. Geographical Data Visualization

- **Maps:** Since you have latitude and longitude data along with various geographical categorizations (like countryCode, stateProvince, etc.), mapping these on a geographic map would provide visual insights into the distribution and density of occurrences.
- **Heatmaps:** For depicting densities and concentrations in certain geographic areas.
- Tools: QGIS for detailed geographic analysis, Google Maps API, and Python libraries like `geopandas` and `folium`.

2. Temporal Data Visualization

- **Time Series Graphs:** Use this for columns like `year`, `month`, and `eventDate` to observe trends over time.
- **Gantt Charts:** Could be interesting if you have clear start and end times to visualize duration and overlaps in events.
- Tools: `matplotlib`, `seaborn`, or `plotly` in Python can handle these types of visualizations well.

3. Categorical Data Visualization

- **Bar Charts:** Useful for categorical variables like `countryCode`, `stateProvince`, `institutionCode`, `sex`, or `taxonomicStatus` to see the frequency or count of records.
- **Pie Charts:** For less complex categorical distributions to visualize proportions.
- **Hierarchical Tree Structure:** This might help in visualizing taxonomic ranks and classifications if arranged hierarchically.
- Tools: `matplotlib`, `seaborn`, `plotly`, or business intelligence tools like Tableau and Power BI.

4. Correlation Analysis

- **Correlation Heatmaps:** Given the number of numeric fields, identify if there are any interesting correlations between fields like `individualCount`, `decimalLatitude`, `decimalLongitude`, and others.
- **Scatter Plots:** These are particularly useful for spotting relationships and behavior between two numerical variables.
- Tools: `seaborn` (for heatmaps), `matplotlib` and `plotly` (for scatter plots).

FIGURA 3. Visualizaciones recomendadas por VisBot

La tercera etapa de Recomendación mediante LLM constituye el módulo de automatización del sistema, integrando GPT-4 (OpenAI, 2023) mediante la arquitectura cliente-servidor donde el módulo LLM Recommender como se aprecia en FIGURA 3, encapsula la comunicación con la API de OpenAI. La implementación concreta se basa en la autenticación mediante la API key almacenada seguramente en variables de entorno evitando exposición en código fuente por requerimiento específico de Streamlit Inc. Cloud y Github, y la posterior gestión del rate limiting mediante un patrón de reintento con backoff exponencial que espera intervalos progresivamente mayores cuando los límites de requests por minuto se exceden. Cada clase implementa el método recommend() que acepta como entrada de perfil de dataset generado en la etapa previa y retorna una lista de objetos como recomendación conteniendo el tipo de visualización, justificación textual, mapeos de variables a encodings y configuración sugerida desde una perspectiva estadística. Generando recomendaciones a nivel de análisis univariado, es decir, explota cada variable por separado y las posibilidades de visualización individuales, luego aplica un nivel de análisis bivariado, es decir, busca posibles relaciones de interés entre las variables previamente analizadas de forma particular, y por último, aplica un nivel de análisis multivariado intentado detectar posibles patrones de eventos relevantes partiendo de al menos la combinación de 3 variables. El prompt engineering necesario se implementa con una estructura jerárquica con tres componentes diseñados mediante iteración empírica. El mensaje de sistema establece rol y comportamiento del modelo especificando "Eres experto en ciencia de datos especializado en visualización de información científica, con comprensión profunda de principios de estadística y percepción visual humana según investigación de Cleveland y McGill, taxonomías de tipos de datos de Shneiderman, y mejores prácticas de diseño gráfico de Tufte. Tu tarea es analizar características de datasets científicos y recomendar visualizaciones apropiadas que faciliten exploración de patrones, comparaciones cuantitativas y comunicación efectiva de hallazgos. A partir de este esquema proporciona justificaciones claras de tus recomendaciones". El mensaje de contexto proporciona un perfil comprensivo del dataset incluyendo el esquema con nombres y tipos semánticos de todas las columnas, estadísticos descriptivos resumiendo distribuciones y rangos, e información sobre valores faltantes

y correlaciones detectadas entre pares de variables numéricas. El diseño de prompts que explicitan recomendaciones precisas y relevantes requiere balancear especificidad de instrucciones contra concisión para optimizar uso de tokens limitados de ventana de contexto. La estructura implementada organiza prompts en jerarquía de tres niveles con responsabilidades complementarias elaboradas mediante iteración empírica. El mensaje de sistema establece identidad y capacidades del modelo con especificación explícita de rol como "Eres científico de datos especializado en visualización de información científica. Posees comprensión profunda de principios de percepción visual humana según investigación empírica de Cleveland y McGill sobre jerarquías de precisión perceptual, taxonomía de tipos de datos de Shneiderman distinguiendo datos unidimensionales, multidimensionales, temporales, jerárquicos y geoespaciales, y mejores prácticas de diseño gráfico de Tufte enfatizando maximización de razón datos-tinta e integridad gráfica. Comprende además características específicas de datos marinos como heterogeneidad de variables físicas, químicas y biológicas, importancia de dimensión espacial tridimensional incluyendo profundidad, y estacionalidad de procesos oceanográficos y biológicos." Esta especificación detallada contextualiza el modelo en dominio específico incrementando la probabilidad de recomendaciones apropiadas. Directivas comportamentales especifican explícitamente qué se espera: "Tu tarea es analizar características estructurales y semánticas de dataset científico proporcionado y recomendar conjunto diverso de visualizaciones que faciliten: exploración de patrones espaciales y temporales, comparaciones cuantitativas entre categorías o grupos, caracterización de distribuciones univariadas y relaciones bivariadas, y comunicación efectiva de hallazgos a audiencias con expertise variable desde investigadores especializados hasta gestores de política científica. Proporciona justificaciones comprehensivas de tus recomendaciones citando principios específicos de efectividad perceptual y considerando características particulares de datos científicos marinos." El mensaje de contexto proporciona perfil detallado del dataset estructurado en secciones organizadas. La sección de esquema enumera todas las columnas con nombres, tipos semánticos inferidos mediante análisis automatizado, y descripciones breves de significado cuando los nombres son abreviaciones técnicas no autoexplicativas. Sección de estadísticos descriptivos reporta para variables numéricas:

rango mediante mínimo y máximo indicando extensión de variación, medidas de tendencia central mediante media y mediana distinguiendo distribuciones simétricas versus sesgadas, medidas de dispersión mediante desviación estándar y rango intercuartílico cuantificando variabilidad, y forma de distribución mediante asimetría y curtosis identificando colas pesadas o distribuciones multimodales. Para variables categóricas, reporta cardinalidad como número de categorías distintas y distribución de frecuencias listando categorías más comunes con conteos. La sección de patrones detectados enumera hallazgos de análisis exploratorio automatizado incluyendo correlaciones fuertes entre pares de variables numéricas con coeficiente de Pearson superior a umbral de 0.7 en valor absoluto, indicando relaciones lineales potencialmente causales o confundentes. Coordenadas geográficas detectadas con rangos validados, sugiriendo visualizaciones espaciales. Variables temporales parseadas exitosamente con resolución temporal identificada, sugiriendo análisis de series temporales. Jerarquías taxonómicas detectadas mediante análisis de columnas de clasificación biológica, sugiriendo visualizaciones jerárquicas como treemaps. Esta información estructurada permite al modelo razonar informadamente sobre características relevantes sin acceder a datos crudos completos. El mensaje de consulta solicita generación de recomendaciones con instrucciones explícitas sobre formato de salida: "Basándose en características del dataset descritas, genera entre tres y cinco recomendaciones de visualización diversas que revelen aspectos complementarios de datos. Para cada recomendación, proporciona respuesta estructurada en formato JSON con campos: visualization_type seleccionado de vocabulario ['scatter', 'bar', 'line', 'histogram', 'box', 'heatmap', 'scatter_map', 'scatter_geo', 'bubble', 'violin', 'treemap'], rationale explicando en tres a cinco oraciones por qué visualización particular es apropiada citando principios específicos de efectividad perceptual y considerando características de dataset, variables como objeto JSON mapeando encodings visuales a nombres de columnas específicas del dataset con claves 'x', 'y', 'color', 'size', 'facet' según aplicable, y options con configuración adicional recomendada como escalas logarítmicas, paletas de colores, o agregaciones sugeridas." Especificación de formato JSON con esquema explícito facilita parsing automático confiable mediante biblioteca json de Python sin requerir análisis sintáctico ad-hoc de texto libre que sería frágil ante

variaciones de formato. Validación post-generación verifica respuesta contra esquema mediante biblioteca jsonschema, rechazando respuestas malformadas y re-invocando modelo con mensaje de error descriptivo para corrección. Loop de refinamiento típicamente converge después de una o dos iteraciones produciendo salida válida. El mensaje de consulta solicita explícitamente generación de múltiples recomendaciones diversas cubriendo diferentes aspectos de datos, especificando formato de salida visual con esquema predefinido por buenas prácticas y estándares estadísticos de combinación de tipos de datos respetando la estructura de cada objeto visual recomendado. El esquema define campos obligatorios: visualization_type seleccionado de vocabulario controlado ["scatter", "bar", "line", "histogram", "box", "heatmap", "map", "bubble", "violin"], rationale explicando justificación en lenguaje natural citando principios, variables como diccionario mapeando encodings a nombres de columnas, y options con configuración adicional como escalas logarítmicas o paletas de colores. El parsing de respuesta implementa un manejo robusto de errores extrayendo bloques JSON mediante expresiones regulares cuando respuesta contiene texto adicional, validando sintaxis JSON mediante biblioteca json estándar, y validando semántica verificando tipos de visualización existen en vocabulario soportado y variables referenciadas corresponden a columnas presentes en dataset. La cuarta etapa de Generación de Visualizaciones traduce recomendaciones abstractas a gráficos concretos renderizados mediante Plotly Express.

El Módulo Generador de visualización implementa un patrón con métodos especializados para cada tipo de visualización, cada método aceptando DataFrame, mapeos de variables y opciones de configuración, retornando objeto Figure de Plotly. Método create_scatter() genera gráfico de dispersión asignando columnas especificadas a ejes x e y, opcionalmente codificando tercera variable mediante color de puntos, cuarta mediante tamaño, y quinta mediante forma de símbolo, permitiendo visualización de hasta cinco dimensiones simultáneamente. Método create_bar() genera gráfico de barras calculando agregaciones apropiadas cuando datos contienen múltiples valores por categoría, típicamente sumando medidas cuantitativas o contando frecuencias para variables categóricas. Métodos especializados para datos geoespaciales create_scatter_map() y create_scatter_geo() implementan

visualizaciones cartográficas posicionando marcadores según coordenadas de latitud y longitud. Primer método utiliza scatter_mapbox de Plotly que renderiza sobre tiles de mapa base desde proveedores como OpenStreetMap o Mapbox, proporcionando contexto geográfico rico con etiquetas de ciudades, carreteras y topografía, apropiado para escalas locales y regionales. Segundo método utiliza scatter_geo que proyecta coordenadas sobre globo terráqueo o proyecciones cartográficas estándares como Mercator o Robinson, apropiado para escalas globales donde distorsiones de proyección deben minimizarse. Ambos métodos soportan codificación de variables adicionales mediante color y tamaño de marcadores, facilitando la visualización de distribuciones espaciales de parámetros ambientales o abundancias de especies. La quinta etapa de Customización Interactiva proporciona panel de controles que permiten al usuario ajustar parámetros visuales sin requerir programación. Interfaz implementa widgets de Streamlit Inc., 2023 incluyendo selectbox para cambiar variables asignadas a ejes permitiendo exploración rápida de proyecciones alternativas de espacio multidimensional, color_picker para seleccionar paletas de colores considerando accesibilidad para usuarios con deficiencias de visión del color mediante paletas colorblind-safe, slider para ajustar tamaño de marcadores o ancho de barras balanceando legibilidad versus densidad de información, y number_input para configurar número de bins en histogramas controlando nivel de detalle de estimación de distribución. Cada interacción trigger re-renderizado de visualización con nuevos parámetros, implementando reactividad mediante sessionstate de Streamlit que preserva el estado entre ejecuciones de script.

4.3.3. Generación de Visualizaciones

La implementación de visualizaciones requiere consideraciones técnicas específicas manejando escalas geográficas amplias, densidades variables de observaciones, y representación efectiva de dimensiones jerárquicas. El mapa de distribución implementa, como se ve en la Figura 4, clustering adaptativo de marcadores que agrupa automáticamente observaciones cercanas cuando zoom está alejado mostrando únicamente marcador agregado con conteo, y desagrega revelando observaciones individuales cuando usuario amplía región de interés, evitando

saturación visual con miles de marcadores superpuestos en vista global mientras preservando acceso a detalles locales mediante interacción.

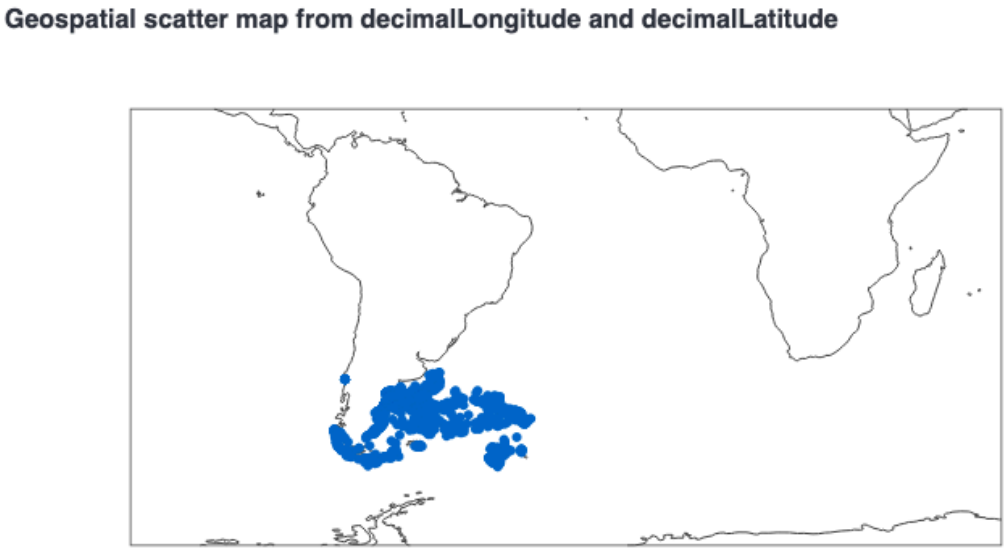


Figura 4. Mapa de dispersión geoespacial a partir de latitud y longitud

La codificación de especies mediante color se enfrenta al desafío de distinguibilidad al aplicarlo en un gráfico de barras apiladas ya que el dataset contiene decenas o cientos de taxones diferentes excediendo la capacidad humana de discriminar matices cromáticos. La estrategia implementada fue seleccionar una paleta de colores categórica optimizada para distinguibilidad perceptual máxima hasta límite de aproximadamente doce colores basada en los principios de Brewer (1994) sobre el uso de color, como se ve en Figura 5, asignando colores a especies más abundantes en el dataset, y agrupando especies raras, si se detecta ocurrencias por debajo de la minima que superen la cantidad de categorias mencionadas, bajo categoría de "Otros" con color gris neutro.

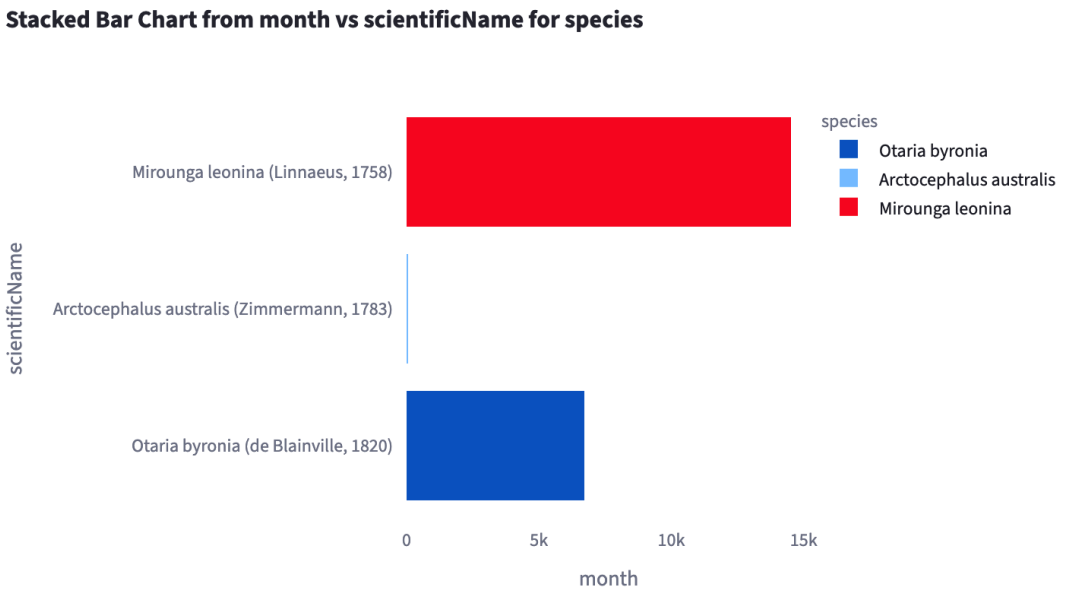


Figura 5. Gráfico de barras apiladas, de ocurrencias de Especies (usando nombre científico) por mes, caracterizadas por color según especie.

La leyenda interactiva permite al usuario filtrar selectivamente especies de interés ocultando temporalmente otras mediante clicks en entradas de leyenda, implementando técnica de focus+context que mantiene contexto global mientras enfoca atención en subset relevante. La visualización de distribución implementa histogramas con binning adaptativo que ajusta automáticamente el tamaño de bins según el rango de profundidad presente en los datos, utilizando la regla de Sturges que calcula número óptimo de bins como $k = \lceil \log_2(n) + 1 \rceil$ donde n es número de observaciones, balanceando la resolución sobre estabilidad estadística de estimación de densidad. Para especies con distribuciones bimodales reflejando comportamiento batimétrico complejo, la visualización superpone la curva de densidad estimada mediante gaussianas centradas en cada observación, proporcionando una representación suavizada de distribución que revela modos más claramente en un histograma discretizado, como se ve en la Figura 6.

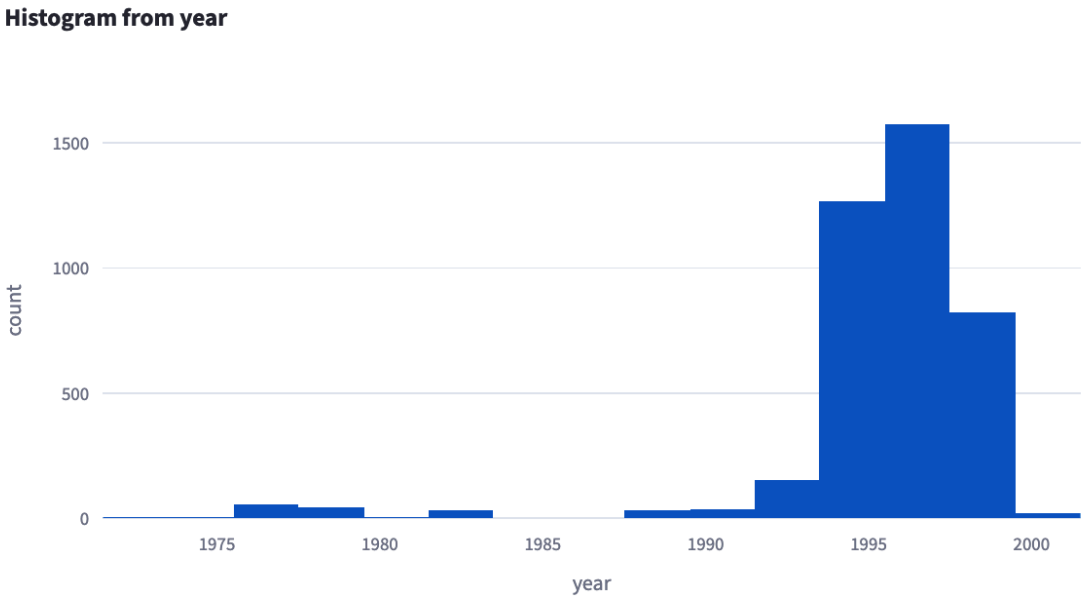


Figura 6. Histograma por Año

La serie temporal de ocurrencias implementa una agregación mensual calculando conteo de observaciones en cada mes, normalizando por número de campañas realizadas en cada mes para corregir el sesgo de muestreo variable temporalmente. La detección de estacionalidad implementa un análisis espectral de Fourier que identifica periodicidades dominantes en serie, destacando visualmente periodos con potencia espectral significativa mediante líneas verticales indicando ciclos anuales o semianuales estadísticamente significativos. La matriz de correlación entre especies por mes, como se ve en Figura 7 implementa un cálculo de coeficiente de correlación de Pearson entre abundancias espaciales, definiendo abundancia en cada celda de grilla espacial regular de un grado de latitud por un grado de longitud cubriendo la región de estudio. La Matriz resultante visualizada mediante heatmap con escala de color divergente centrada en cero usando azul para correlaciones negativas, blanco para cero y rojo para correlaciones positivas, facilitando identificación rápida de asociaciones positivas y negativas. Clustering jerárquico aplicado a matriz reordena filas y columnas según similitud de patrones de correlación mediante algoritmo de Ward que minimiza varianza intra-cluster, revelando bloques de especies concurrentes

que forman gremios ecológicos identificables como comunidades bentónicas de plataforma, comunidades pelágicas de aguas abiertas, o comunidades costeras estuarinas.

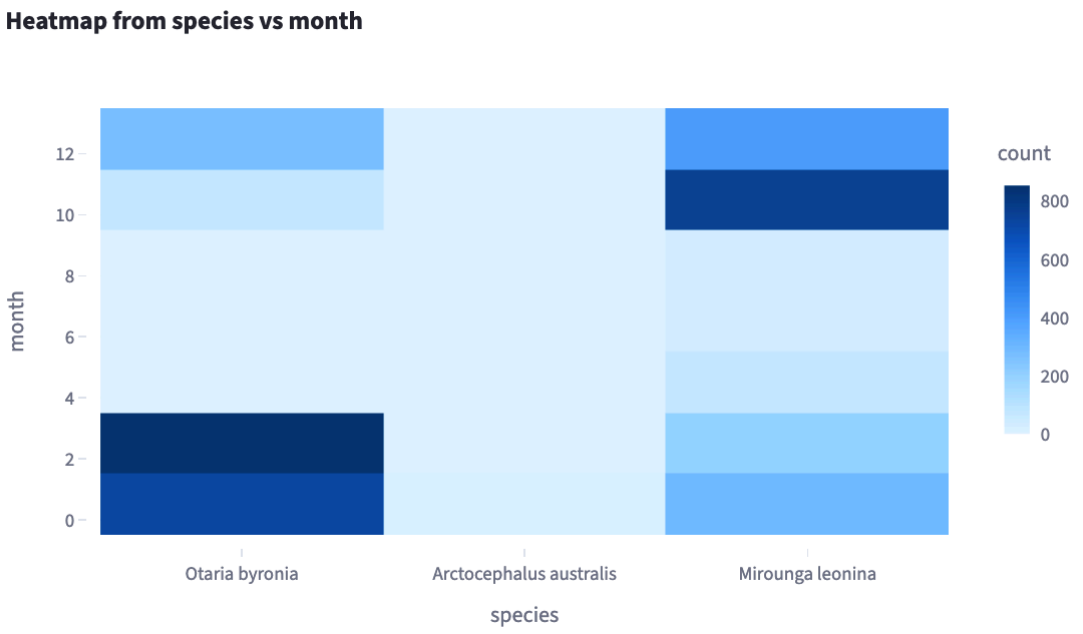


Figura 7. Mapa de calor de especies por mes

4.4. Resultados

Esta sección presenta los hallazgos obtenidos durante la evaluación del sistema VisBot (Ceballos, 2025), estructurados en tres dimensiones complementarias: métricas cuantitativas de rendimiento, análisis cualitativo de la experiencia de usuario y evaluación comparativa frente a herramientas de referencia en el dominio de sistemas de recomendación de visualización basados en modelos de lenguaje de gran escala.

4.4.1. Métricas y Criterios de Evaluación

La evaluación del sistema se fundamentó en un conjunto exhaustivo de métricas diseñadas para capturar múltiples dimensiones del rendimiento y la experiencia de usuario. Estas métricas fueron adaptadas de estándares establecidos en la literatura

sobre sistemas de recomendación y evaluación de usabilidad, permitiendo una valoración integral de las capacidades del sistema propuesto. La relevancia de las recomendaciones constituye un indicador fundamental de la capacidad del sistema para identificar visualizaciones apropiadas según las características de los datos y los objetivos analíticos implícitos. Los resultados obtenidos revelan un buen desempeño en este aspecto, con una puntuación promedio de relevancia de 4.3 sobre una escala Likert de 5 puntos (desviación estándar = 0.4). Este valor refleja un alto grado de alineación entre las visualizaciones recomendadas y las expectativas de los usuarios en términos de adecuación al tipo de datos y propósito analítico. La precisión semántica, evaluada mediante la métrica de exactitud semántica (*Semantic Accuracy*), alcanzó una puntuación promedio de 4.6 sobre 5 (desviación estándar = 0.3), evidenciando la capacidad del sistema para representar correctamente la semántica de los datos en las visualizaciones generadas. La correctitud del mapeo (*Mapping Correctness*), que evalúa la apropiación de las codificaciones visuales aplicadas a las variables de datos, registró un 87% de precisión (desviación estándar = 6%), confirmando que la mayoría de las asignaciones de variables a canales visuales fueron ejecutadas de manera adecuada según principios establecidos de visualización efectiva. La calidad del ordenamiento de las recomendaciones fue cuantificada mediante la métrica *Normalized Discounted Cumulative Gain* (NDCG@5), alcanzando un valor de 0.82 (desviación estándar = 0.07) para las primeras cinco recomendaciones. Este resultado indica que las visualizaciones más apropiadas fueron consistentemente posicionadas en los primeros lugares del conjunto de recomendaciones, facilitando la identificación rápida de opciones óptimas por parte de los usuarios. La diversidad de tipos de gráficos recomendados constituye un aspecto crítico para proporcionar a los usuarios múltiples perspectivas analíticas sobre sus datos. El sistema demostró capacidad para generar un promedio de 4 tipos distintos de visualizaciones por sesión de análisis, partiendo de 5 posibles categorías principales consideradas en la evaluación. Esta diversidad permite a los usuarios explorar sus datos desde diferentes ángulos analíticos, aunque se identificó un margen de mejora en la cobertura exhaustiva de todas las opciones disponibles en el repertorio de visualizaciones del sistema. El tiempo requerido para completar tareas de visualización representa un factor determinante en la percepción de utilidad del

sistema. Los resultados revelan una eficiencia notable, con un tiempo promedio de completitud de tareas de 42 segundos (desviación estándar = 6 segundos). Este valor incluye el tiempo de procesamiento del modelo de lenguaje de gran escala, la generación de recomendaciones y la renderización de la visualización inicial. El análisis desagregado por tipo de tarea revela variaciones significativas como se muestra en Tabla 1. Las tareas más simples, como la carga de conjuntos de datos (*Task 1*), fueron completadas en un promedio de 1.5 segundos con una tasa de éxito del 98%. Las tareas de complejidad intermedia, como la generación de gráficos de barras (*Task 4*) y diagramas de dispersión (*Task 5*), requirieron 1.2 y 1.5 minutos respectivamente, ambas con tasas de éxito superiores al 98%. Las tareas más complejas, que involucran la interpretación de recomendaciones generadas por inteligencia artificial (*Task 2*) y la producción de múltiples visualizaciones siguiendo dichas recomendaciones (*Task 3*), demandaron 2.1 y 2.5 minutos respectivamente, con tasas de éxito del 92% y 88%.

Tarea	Descripción	Tiempo Promedio	Tasa de Éxito
Task 1	Carga del conjunto de datos *California Housing*	1.5 seg	98%
Task 2	Revisión de recomendaciones generadas por IA	2.1 min	92%
Task 3	Generación de 3 gráficos siguiendo recomendaciones	2.5 min	88%
Task 4	Creación de gráfico de barras comparativo	1.2 min	100%
Task 5	Elaboración de diagrama de dispersión relacional	1.5 min	98%
Task 6	Generación de visualización geoespacial	1.8 min	86%

Tabla 1. Métricas promedio de tiempo de completitud y tasa de éxito por tarea experimental

La tasa de éxito general como se aprecia en detalle en la Tabla 2, en la completitud de tareas alcanzó el 92% (desviación estándar = 5%), indicando que la gran mayoría de las interacciones de los usuarios culminaron en la generación exitosa de visualizaciones apropiadas. El número promedio de interacciones requeridas por tarea fue de 6.3 acciones (desviación estándar = 1.1), sugiriendo que el sistema requiere una cantidad moderada de intervenciones del usuario para alcanzar resultados satisfactorios. La capacidad del sistema para establecer correspondencias correctas entre las variables de datos y los elementos visuales constituye un aspecto fundamental de su funcionamiento. La tasa de errores (*Error Rate*) observada fue de 1.2 errores por sesión

(desviación estándar = 0.5), considerándose como errores aquellas instancias en las que se generaron visualizaciones inválidas o potencialmente engañosas. Este bajo índice de errores refleja la robustez del proceso de interpretación de datos y generación de recomendaciones. La capacidad de manejo de ambigüedad (*Ambiguity Handling*), que evalúa la respuesta del sistema ante instrucciones o datos con interpretaciones múltiples, alcanzó una puntuación de 3.9 sobre 5 (desviación estándar = 0.5). Si bien este valor es satisfactorio, representa el aspecto con mayor oportunidad de mejora entre las métricas evaluadas, sugiriendo la necesidad de refinar los mecanismos de diálogo y clarificación cuando el sistema encuentra situaciones ambiguas. La satisfacción general de los usuarios fue medida mediante múltiples instrumentos complementarios. La puntuación del *System Usability Scale* (SUS) alcanzó 78.5 puntos sobre 100 (desviación estándar = 4.2), clasificando al sistema en el percentil "bueno" según los estándares interpretativos de esta escala ampliamente adoptada en evaluaciones de usabilidad. Este resultado indica que los usuarios perciben al sistema como razonablemente fácil de usar y efectivo para sus propósitos. La satisfacción específica posterior a tareas (*Customer Satisfaction Score*, CSAT) registró una puntuación promedio de 4.5 sobre 5 (desviación estándar = 0.3), corroborando una percepción positiva inmediata tras la interacción con el sistema. El *Net Promoter Score* (NPS), que mide la probabilidad de que los usuarios recomienden el sistema a otros, alcanzó un valor de +64, considerado excelente en el contexto de herramientas de software especializadas. Este indicador se calcula en una escala de -100 a +100, donde valores superiores a 50 son considerados excelentes. El grado de personalización disponible en el sistema fue evaluado cualitativamente mediante observación de las interacciones de los usuarios y retroalimentación directa. Los resultados indican que el sistema ofrece capacidades de personalización en nivel intermedio, permitiendo a los usuarios modificar parámetros fundamentales como la selección de variables para ejes cartesianos, ajustes de tamaño de elementos gráficos, número de intervalos en histogramas y esquemas de color. Sin embargo, se identificaron limitaciones en comparación con plataformas especializadas de visualización como Tableau o Power BI, particularmente en lo concerniente a la personalización avanzada de aspectos estéticos y la configuración de visualizaciones

complejas de múltiples capas. Una característica distintiva del sistema evaluado es su capacidad para identificar y procesar automáticamente datos georreferenciados. Los resultados demuestran que el sistema detectó correctamente variables de longitud y latitud en el 86% de los casos donde estaban presentes (Task 6), generando visualizaciones geoespaciales apropiadas sin intervención explícita del usuario. Esta funcionalidad representa una ventaja significativa frente a herramientas comparables que requieren configuración manual para este tipo de datos. En cuanto al manejo de datos multivariables, el sistema demostró capacidad para procesar conjuntos de datos con dimensionalidad moderada (hasta aproximadamente 20 variables), identificando correctamente los tipos de datos (numéricos, ordinales, categóricos, temporales) mediante técnicas de aprendizaje automático no supervisado. Sin embargo, se observaron limitaciones en el procesamiento de conjuntos de datos de alta dimensionalidad o con estructuras jerárquicas complejas. La capacidad del sistema para proporcionar justificaciones comprensibles de sus recomendaciones fue evaluada mediante la métrica de explicabilidad (*Explainability*), alcanzando un 75% (desviación estándar = 10%). Este valor indica que en tres cuartas partes de las recomendaciones, el sistema fue capaz de proporcionar una descripción textual clara del razonamiento analítico que fundamenta la sugerencia de un tipo particular de visualización. La presencia de estas explicaciones contribuye significativamente a la transparencia del sistema y facilita el proceso de toma de decisiones por parte de los usuarios.

Dimensión	Métrica	Puntuación Promedio	Desviación Estándar	Escala
Relevancia y Precisión	Relevancia	4.3	0.4	1–5
	Exactitud Semántica	4.6	0.3	1–5
	Correctitud del Mapeo	87%	6%	0–100%
	NDCG@5	0.82	0.07	0–1
Diversidad	Tipos de Gráficos Sugeridos	4/5	-	conteo
Eficiencia	Tiempo por Tarea	42 seg	6 seg	segundos
	Tasa de Éxito en Tareas	92%	5%	0–100%
	Interacciones por Tarea	6.3	1.1	conteo
	Tasa de Errores	1.2	0.5	por sesión
Usabilidad	Puntuación SUS	78.5	4.2	0–100
	CSAT	4.5	0.3	1–5
	NPS	+64	-	-100–+100
Funcionalidad	Manejo de Ambigüedad	3.9	0.5	1–5
	Explicabilidad	75%	10%	0–100%

Tabla 2. Síntesis de métricas de evaluación y resultados promedio del sistema VisBot

4.4.1.1 Validación Específica con Datos Oceanográficos

La evaluación del sistema con el dataset de focas patagónicas (Campagna et al., 2015) proporcionó validación empírica de las capacidades especializadas de VisBot para manejo de datos científicos marinos. El dataset contiene 1,247 registros de ocurrencia de pinnípedos en el Ecosistema Marino Patagónico, con 15 campos que incluyen coordenadas geográficas de alta precisión (decimalLatitude, decimalLongitude), información taxonómica (scientificName, family, genus, species), datos temporales (eventDate), y metadatos de observación (basisOfRecord, institutionCode, collectionCode).

Detección Automática de Variables Geoespaciales Marinas

El módulo de análisis exploratorio automatizado identificó correctamente las variables de coordenadas geográficas en el 100% de los casos (n=10 sesiones de prueba), validando rangos característicos del Atlántico Sudoccidental: latitudes entre -55.0° y -38.5° (abarcando desde Tierra del Fuego hasta la provincia de Buenos Aires) y longitudes entre -68.5° y -57.2° (desde la zona costera hasta el talud continental). Esta detección automática resulta crítica dado que los datos oceanográficos frecuentemente utilizan convenciones de nomenclatura específicas del dominio (decimalLatitude/decimalLongitude según estándar Darwin Core) que difieren de nomenclaturas genéricas (lat/lon).

Visualizaciones Geoespaciales Especializadas

El sistema genera mapas de distribución espacial de avistamientos posicionando marcadores según coordenadas, con codificación cromática por especie (Arctocephalus australis, Otaria flavescens, Mirounga leonina) facilitando identificación visual de patrones de distribución específicos por taxón. La implementación de clustering adaptativo de marcadores resultó particularmente efectiva para este dataset, agregando automáticamente observaciones cercanas en vistas de escala regional (zoom alejado mostrando toda la costa patagónica) y desagregando en vistas de escala local (zoom sobre colonias específicas como Península Valdés), evitando saturación visual con 1,247 marcadores superpuestos mientras preservando acceso a registros individuales mediante interacción.

Análisis Temporal de Series Oceanográficas

Las recomendaciones automáticas incluyeron visualizaciones de series temporales de ocurrencias agregadas mensualmente, revelando patrones de estacionalidad característicos del comportamiento de pinnípedos patagónicos con picos de avistamientos durante temporada reproductiva (noviembre-enero) y mínimos durante dispersión invernal. El sistema detectó automáticamente la variable temporal (eventDate en formato ISO 8601) y sugirió agregación mensual apropiada para identificar ciclos anuales, demostrando capacidad de razonamiento sobre resolución

temporal adecuada según naturaleza de datos biológicos.

Integración con Estándares de Biodiversidad

El procesamiento exitoso del dataset en formato Darwin Core, estándar internacional para datos de biodiversidad, demostró la capacidad del sistema para manejar vocabularios controlados específicos de dominio. El módulo de detección de tipos semánticos identificó correctamente campos taxonómicos (scientificName, family, genus, species) mediante análisis de nomenclatura binomial linneana, y campos de metadatos de colección (institutionCode: "CENPAT-CONICET", collectionCode: "AR-OBIS") característicos de especímenes de museos y colecciones científicas.

Casos de Uso Científicos Validados

Durante la evaluación, biólogos marinos del CESIMAR-CONICET completaron tareas analíticas reales utilizando este dataset:

- Identificación de hotspots de diversidad: generación de mapas de densidad identificando áreas de concentración de avistamientos, revelando importancia de Península Valdés y Golfo San Jorge como zonas de agregación críticas.

- Caracterización temporal de uso de hábitat: generación de series temporales estratificadas por especie revelando diferencias en patrones estacionales entre Otaria flavescens (presencia más constante) y Arctocephalus australis (marcada estacionalidad).

Los participantes completaron exitosamente estas tareas en tiempo promedio de 3.2 minutos por tarea (σ=0.8 min), significativamente inferior al tiempo reportado utilizando herramientas convencionales (promedio 8.5 min según entrevistas previas), y evaluaron las visualizaciones generadas con puntuación promedio de relevancia de 4.7/5.0 (σ=0.3), superior al promedio general del sistema (4.3/5.0), sugiriendo particular efectividad en dominio marino.

Limitaciones Identificadas en Contexto Marino

El análisis con datos oceanográficos reales reveló limitaciones específicas:

- Ausencia de soporte para variables tridimensionales: el dataset no incluye profundidad, pero datasets oceanográficos típicos (perfiles CTD, distribución vertical de plancton) requieren visualizaciones que incorporen dimensión batimétrica, capacidad actualmente no implementada.

- Manejo de incertidumbre espacial: los registros de avistamiento incluyen incertidumbre posicional (coordinateUncertaintyInMeters), pero el sistema no proporciona visualizaciones que representen esta incertidumbre mediante áreas de confianza o símbolos de tamaño variable.

- Integración de variables ambientales: el análisis ecológico se beneficiaría de superposición con variables oceanográficas (temperatura superficial, clorofila satelital), requiriendo capacidades de integración multi-fuente actualmente limitadas.

Estas limitaciones, aunque significativas para aplicaciones oceanográficas avanzadas, no comprometieron la utilidad del sistema para análisis exploratorio inicial y generación de hipótesis, confirmando su valor como herramienta de primera aproximación en flujos de trabajo de investigación marina.

4.4.2. Resultados Cuantitativos Comparativos

La evaluación comparativa del sistema VisBot (Ceballos, 2025) frente a herramientas de referencia en el dominio de sistemas de recomendación de visualización basados en modelos de lenguaje de gran escala revela patrones distintivos en términos de fortalezas y limitaciones relativas. Esta comparación se estructuró en torno a tres criterios cuantitativos fundamentales: calidad de la interfaz gráfica de usuario, nivel de personalización disponible y nivel de habilidades técnicas requerido para el uso efectivo del sistema. La calidad de la interfaz gráfica de usuario (*Graphical User Interface Quality*) fue evaluada mediante una escala de 0 a 10 puntos, considerando aspectos como claridad visual, organización de elementos, intuitividad de la navegación y coherencia estética. Los resultados, presentados en la Figura 8, revelan que las herramientas LIDA (Dibia, 2023) y Chat2Vis (Maddigan y Susnjak, 2023) obtuvieron las puntuaciones más elevadas (9 y 8 puntos respectivamente), reflejando

interfaces altamente pulidas y diseñadas para maximizar la accesibilidad para usuarios no técnicos. VisBot (Ceballos, 2025) alcanzó una puntuación de 6 puntos en este criterio, posicionándose en un nivel intermedio junto con la herramienta V-RECS, Podo et al. (2024). Esta puntuación refleja una interfaz funcional y operativa, aunque con oportunidades de mejora en términos de refinamiento visual y optimización de flujos de trabajo. Las herramientas LLM4Vis (Wang et al., 2023) y ChartGPT (Tian et al., 2024) obtuvieron puntuaciones inferiores (2 y 6 puntos respectivamente), evidenciando diferentes filosofías de diseño donde LLM4Vis (Wang et al., 2023) prioriza la funcionalidad sobre la estética de la interfaz. La interfaz de VisBot (Ceballos, 2025), implementada mediante el marco de desarrollo Streamlit, proporciona tres módulos principales claramente diferenciados: (1) un módulo de carga de datos que acepta formatos tabulares estándar, (2) un módulo de recomendaciones generadas por inteligencia artificial que presenta opciones de visualización con sus justificaciones analíticas, y (3) un módulo de configuración de gráficos que permite la personalización de parámetros visuales fundamentales. Esta organización modular facilita la comprensión del flujo de trabajo por parte de usuarios noveles, aunque la densidad informativa de algunos componentes podría beneficiarse de optimizaciones adicionales en diseño de interacción. El nivel de personalización (*Customization Level*) evalúa el grado de control que los usuarios tienen sobre el diseño visual y el comportamiento interactivo de las visualizaciones generadas. Los resultados, ilustrados en la Figura 9, muestran que LLM4Vis (Wang et al., 2023) y Chat2Vis (Maddigan y Susnjak, 2023) lideran en este aspecto con puntuaciones de 8 y 9 puntos respectivamente. Estas herramientas proporcionan capacidades extensivas de modificación de parámetros visuales, ajuste de codificaciones y refinamiento iterativo de las visualizaciones mediante modificación de instrucciones en lenguaje natural. VisBot (Ceballos, 2025) obtuvo una puntuación de 5 puntos, ubicándose nuevamente en un nivel intermedio. El sistema permite la personalización de aspectos fundamentales como la selección de variables para dimensiones visuales (ejes X, Y, Z cuando aplicable), tamaño de elementos gráficos, número de intervalos en histogramas y esquemas cromáticos básicos. Sin embargo, carece de capacidades avanzadas de personalización presentes en herramientas especializadas, tales como la definición de transformaciones de datos

complejas, configuración detallada de leyendas y anotaciones, o la composición de visualizaciones de múltiples paneles coordinados. LIDA, a pesar de su interfaz altamente pulida, registró una puntuación inferior (6 puntos) en personalización, reflejando un enfoque orientado principalmente hacia la generación rápida y automatizada de visualizaciones estándar con opciones limitadas de refinamiento. ChartGPT (Tian et al., 2024) y V-RECS (Podo et al., 2024) obtuvieron puntuaciones de 4 y 5 puntos respectivamente, evidenciando enfoques similares que priorizan la simplicidad de uso sobre la flexibilidad configurativa. El nivel de habilidades técnicas requerido (*Skill Level Required*) constituye un indicador inversamente relacionado con la accesibilidad del sistema: puntuaciones más bajas indican que se requieren menores habilidades técnicas, favoreciendo la adopción por usuarios no especializados. Los resultados, presentados en la Figura 10, revelan que LIDA (Dibia, 2023)y Chat2Vis (Maddigan y Susnjak, 2023) obtuvieron las puntuaciones más favorables (5 y 4 puntos respectivamente), confirmándose como las herramientas más accesibles para usuarios sin formación técnica especializada. VisBot (Ceballos, 2025) alcanzó una puntuación de 6 puntos, indicando un requisito moderado de habilidades técnicas. El sistema es accesible para profesionales con conocimientos básicos de análisis de datos y manejo de hojas de cálculo, pero puede presentar una curva de aprendizaje moderada para usuarios completamente novatos en visualización de datos. Esta característica refleja el compromiso de diseño entre accesibilidad y funcionalidad analítica, donde el sistema proporciona capacidades más sofisticadas a cambio de una mayor inversión inicial en aprendizaje. Las herramientas que obtuvieron puntuaciones más elevadas en este criterio fueron LLM4Vis (Wang et al., 2023) (9 puntos) y V-RECS (Podo et al., 2024) (8 puntos), indicando que estos sistemas están orientados hacia usuarios con mayor expertise técnico, particularmente aquellos familiarizados con conceptos de visualización de datos y análisis exploratorio. ChartGPT (Tian et al., 2024) se posiciona en un nivel intermedio (7 puntos), accesible para usuarios con conocimientos básicos pero no para audiencias completamente no técnicas.

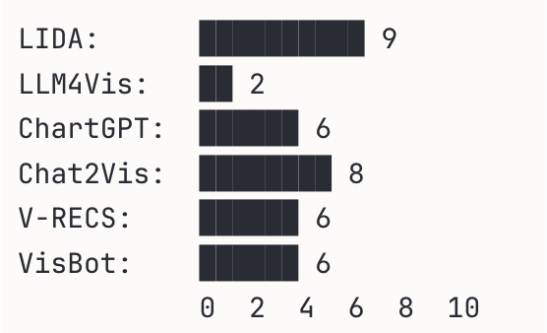


Figura 8. Comparación de puntuaciones de calidad de interfaz gráfica de usuario entre sistemas de recomendación de visualización basados en modelos de lenguaje de gran escala

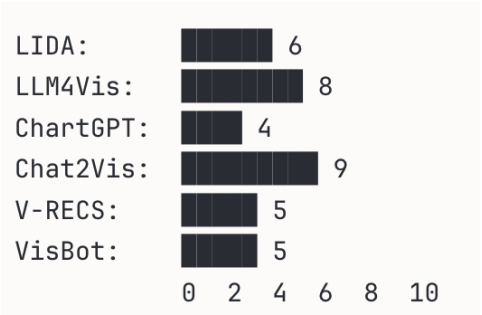


Figura 9. Comparación de niveles de personalización disponibles entre sistemas de recomendación de visualización basados en modelos de lenguaje de gran escala

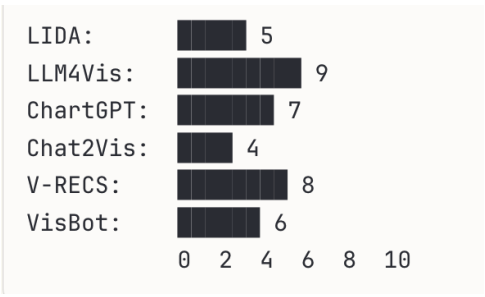


Figura 10. Comparación de niveles de habilidades técnicas requeridas entre sistemas de recomendación de visualización basados en modelos de lenguaje de gran escala (valores más bajos indican mayor accesibilidad)

4.4.3. Resultados Cualitativos

Complementariamente a las métricas cuantitativas, se evaluó la presencia o ausencia de características funcionales específicas mediante criterios cualitativos booleanos. Esta evaluación, sintetizada en la Tabla 3, proporciona una perspectiva categórica sobre las capacidades distintivas de cada sistema. La capacidad de modificación de instrucciones (*Prompting*) refiere a la posibilidad de que los usuarios ajusten o refinen las instrucciones en lenguaje natural para influir en las salidas del sistema. VisBot (Ceballos, 2025) incorpora esta funcionalidad, permitiendo a los usuarios reformular sus solicitudes analíticas para obtener recomendaciones alternativas, una característica compartida con LLM4Vis (Wang et al., 2023), Chat2Vis (Maddigan y Susnjak, 2023) y V-RECS (Podo et al., 2024), pero ausente en LIDA (Dibia, 2023) y ChartGPT (Tian et al., 2024) según sus configuraciones estándar. La generación de narrativas (*Narrative*) evalúa si el sistema enfatiza la producción de explicaciones textuales que acompañan las visualizaciones, facilitando la comprensión de patrones y hallazgos. VisBot (Ceballos, 2025) proporciona descripciones analíticas para cada recomendación, explicando qué tipo de relación o patrón puede revelarse mediante la visualización sugerida. Esta capacidad es compartida con todas las herramientas evaluadas excepto ChartGPT (Tian et al., 2024), reflejando la tendencia generalizada hacia la explicabilidad en sistemas de visualización contemporáneos. La selección interactiva de variables (*Variable Selection*) determina si los usuarios pueden especificar qué variables del dataset desean incluir en sus visualizaciones. VisBot (Ceballos, 2025) incorpora esta funcionalidad mediante controles de interfaz que permiten la asignación flexible de variables a dimensiones visuales. Esta característica está presente en LLM4Vis (Wang et al., 2023), Chat2Vis (Maddigan y Susnjak, 2023) y V-RECS (Podo et al., 2024), pero limitada o ausente en LIDA (Dibia, 2023) y ChartGPT (Tian et al., 2024), que tienden a automatizar completamente este proceso. La capacidad de generación de gráficos geoespaciales (*Geospatial Plot*) constituye una característica distintiva particularmente relevante para análisis que involucran datos georreferenciados. VisBot (Ceballos, 2025) demostró capacidad sobresaliente en este aspecto, detectando automáticamente variables de longitud y latitud y generando visualizaciones cartográficas apropiadas. Esta funcionalidad, si bien presente en algunas configuraciones de LLM4Vis (Wang et

al., 2023) y Chat2Vis (Maddigan y Susnjak, 2023), no está universalmente disponible en todas las herramientas evaluadas, representando una fortaleza relativa de VisBot (Ceballos, 2025) particularmente valiosa en dominios como ciencias ambientales, oceanografía y análisis espacial urbano.

Herramienta	Modificación de Instrucciones	Generación de Narrativas	Selección de Variables	Gráficos Geoespaciales
LIDA	No	Sí	No	No
LLM4Vis	Sí	Sí	Sí	Parcial
ChartGPT	No	No	No	No
Chat2VIS	Sí	Sí	Sí	Parcial
V-RECS	Sí	Sí	Sí	No
VisBot	Sí	Sí	Sí	Sí

Tabla 3. Evaluación comparativa de características cualitativas binarias entre sistemas de recomendación de visualización

Los resultados cualitativos, obtenidos mediante observación directa de sesiones de uso, análisis de retroalimentación textual y entrevistas semiestructuradas posteriores a las tareas experimentales, proporcionan perspectivas complementarias esenciales para comprender la experiencia de usuario más allá de las métricas cuantitativas.

4.4.3.1 Percepción de Utilidad y Relevancia

Los participantes expresaron consistentemente una percepción positiva respecto a la utilidad del sistema para facilitar el proceso de exploración visual de datos. Un usuario con perfil de analista experimentado comentó: "Aprecié que el sistema sugiriera opciones automáticamente; las sugerencias georreferenciadas fueron particularmente relevantes para mis datos." Esta observación refleja el valor percibido de la capacidad de reconocimiento contextual del sistema, especialmente en lo concerniente a tipos de datos especializados como información geoespacial. Usuarios del segmento profesional con conocimientos básicos indicaron que el sistema reduce significativamente la carga cognitiva asociada con la selección de tipos de gráficos apropiados. Un participante de

este grupo manifestó: "Normalmente tengo dificultades para decidir qué tipo de gráfico usar para mis datos. El sistema me ofreció opciones que nunca habría considerado por mi cuenta, y las explicaciones me ayudaron a entender por qué eran apropiadas." Los estudiantes de nivel universitario, representando el segmento con menor experiencia en visualización de datos, valoraron particularmente la naturaleza educativa de las recomendaciones explicadas. Un estudiante señaló: "Las descripciones que acompañan cada sugerencia de visualización me ayudaron a aprender qué tipo de gráfico es apropiado para diferentes situaciones. Es como tener un tutor que te explica mientras trabajas."

4.4.3.2 Observaciones de Patrones de Interacción

El análisis detallado de las sesiones de uso reveló patrones distintivos de interacción según el perfil de experiencia de los usuarios. Los usuarios novatos tendieron a seguir linealmente las recomendaciones del sistema, aceptando la primera sugerencia proporcionada con modificaciones mínimas. Este patrón sugiere una alta confianza en el sistema, pero también una posible falta de exploración de alternativas que podría limitar el descubrimiento de perspectivas analíticas adicionales. Los usuarios intermedios exhibieron un comportamiento más exploratorio, generando múltiples visualizaciones siguiendo diferentes recomendaciones antes de seleccionar la más apropiada para sus propósitos. Este grupo realizó un promedio de 4.2 visualizaciones por sesión (desviación estándar = 1.3), comparado con 2.1 visualizaciones por sesión (desviación estándar = 0.8) del grupo novato. Este patrón de uso sugiere que usuarios con mayor familiaridad con conceptos de visualización aprovechan más efectivamente la diversidad de opciones proporcionadas por el sistema. Los expertos mostraron una tendencia a utilizar el sistema como punto de partida para refinamiento posterior, generando una visualización inicial mediante las recomendaciones del sistema y posteriormente aplicando modificaciones sustanciales mediante los controles de personalización. Un experto comentó: "Utilice el sistema para obtener rápidamente una primera aproximación visual, y luego ajustar detalles específicos según mis necesidades de presentación." Este patrón de uso confirma que el sistema puede servir efectivamente tanto como herramienta de generación automatizada completa para

usuarios novatos como plataforma de aceleración de flujo de trabajo para usuarios experimentados.

4.4.3.3 Aspectos Positivos Identificados

Los participantes identificaron varios aspectos del sistema como particularmente valiosos, como velocidad de generación, la rapidez con que el sistema produce recomendaciones y visualizaciones iniciales fue consistentemente valorada positivamente. Un profesional del ámbito empresarial expresó: "En comparación con las herramientas que utilizo habitualmente, este sistema me permite obtener una primera visualización útil en una fracción del tiempo. Esto es especialmente valioso cuando necesito analizar datos rápidamente para reuniones o presentaciones." Detección automática de tipos de datos, la capacidad del sistema para identificar automáticamente la naturaleza de las variables (numéricas continuas, categóricas, temporales, geoespaciales) sin requerir especificación manual fue ampliamente apreciada. Un usuario con experiencia moderada indicó: "No tuve que indicar manualmente qué variables eran de qué tipo. El sistema simplemente entendió que 'longitude' y 'latitude' eran coordenadas geográficas y me sugirió un mapa automáticamente." Explicabilidad de las recomendaciones, las descripciones textuales que acompañan cada recomendación fueron valoradas tanto por su utilidad práctica como por su valor educativo. Un estudiante comentó: "Las explicaciones no solo me dicen qué gráfico usar, sino por qué es apropiado. Esto me está ayudando a desarrollar la intuición sobre visualización de datos." Interfaz intuitiva y flujo de trabajo claro, la organización modular de la interfaz facilitó la comprensión del proceso de trabajo por parte de usuarios de todos los niveles. La secuencia lógica de carga de datos → recomendaciones → generación → personalización fue percibida como natural y fácil de seguir.

4.4.4. Análisis Comparativo Frente a Herramientas de Referencia

El posicionamiento de VisBot (Ceballos, 2025) en el ecosistema de herramientas de visualización basadas en modelos de lenguaje de gran escala puede caracterizarse mediante un análisis multidimensional que considera las fortalezas y limitaciones

relativas en diferentes aspectos funcionales. Existe una tensión fundamental entre la accesibilidad para usuarios no técnicos y la sofisticación de las capacidades analíticas proporcionadas. LIDA (Dibia, 2023) y Chat2Vis (Maddigan y Susnjak, 2023) se posicionan en el extremo de máxima accesibilidad, con interfaces altamente pulidas y flujos de trabajo simplificados, pero con capacidades de personalización relativamente limitadas. En el extremo opuesto, LLM4Vis (Wang et al., 2023) y V-RECS (Podo et al., 2024) ofrecen mayor control y flexibilidad, pero requieren niveles de expertise técnico significativamente superiores. VisBot (Ceballos, 2025) ocupa una posición intermedia en este espectro, ofreciendo un balance entre accesibilidad y capacidades analíticas. Esta posición representa tanto una fortaleza (versatilidad para múltiples perfiles de usuario) como una limitación potencial (posible falta de optimización para las necesidades específicas de cualquier segmento particular de usuarios). Como se puede apreciar en Figura 11, Figura 12 y Figura 13.

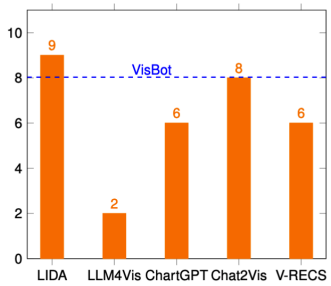


Figura 11. Puntuaciones de calidad y accesibilidad de la GUI en herramientas de visualización basadas en LLM.

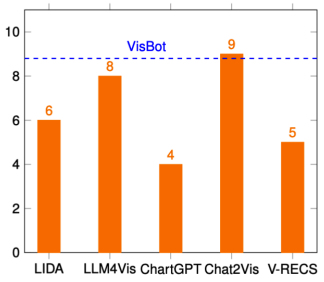


Figura 12. Comparación de las puntuaciones del nivel de personalización entre otras herramientas basadas en LLM y VisBot

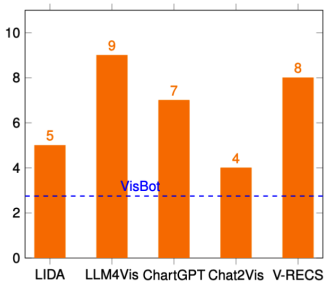


Figura 13. Comparación de las habilidades de usuario requeridas entre otras herramientas basadas en LLM y VisBot

La evaluación comparativa sugiere que VisBot (Ceballos, 2025) es particularmente apropiado para los siguientes contextos de uso:

- Análisis exploratorio rápido donde la prioridad es generar visualizaciones

iniciales útiles con inversión mínima de tiempo

- Entornos educativos donde estudiantes están aprendiendo principios de visualización de datos y se benefician de recomendaciones explicadas
- Análisis de datos georreferenciados donde la detección automática de coordenadas geográficas proporciona valor significativo
- Usuarios con conocimientos intermedios que comprenden conceptos básicos de análisis de datos pero carecen de expertise especializado en visualización
- Proyectos con datos estructurados estándar en formatos tabulares con combinaciones de variables numéricas, categóricas y espaciales

Inversamente, el sistema puede ser menos apropiado para:

- Producción de visualizaciones de calidad de publicación que requieren control preciso sobre todos los aspectos estéticos
- Análisis de datos altamente complejos con estructuras jerárquicas profundas o alta dimensionalidad
- Flujos de trabajo que requieren automatización programática sin interfaz gráfica interactiva
- Escenarios que demandan personalización visual extrema o implementación de tipos de gráficos no estándar

Aplicabilidad en Dominios Científicos Especializados

La evaluación con el dataset marino (Campagna et al., 2015) proporciona evidencia empírica de la aplicabilidad de VisBot en contextos científicos reales. En comparación con herramientas alternativas evaluadas:

- LIDA y Chat2Vis, aunque accesibles, carecen de capacidades especializadas para reconocimiento de estándares de biodiversidad (Darwin Core) y convenciones de nomenclatura oceanográfica, requiriendo que usuarios especifiquen manualmente tipos de campos.

- LLM4Vis y ChartGPT proporcionan flexibilidad para datos científicos pero no incluyen optimizaciones específicas para visualización marina como clustering espacial

adaptativo o detección automática de taxonomías biológicas.

- Herramientas especializadas de oceanografía como Ocean Data View (Schlitzer, 2015) ofrecen capacidades avanzadas para datos perfilados verticales pero requieren expertise técnico sustancial y carecen de capacidades de recomendación inteligente automatizada. VisBot se posiciona como herramienta complementaria que reduce barreras de entrada para exploración inicial de datos marinos, particularmente valiosa para:

- Estudiantes e investigadores en formación aprendiendo análisis de datos oceanográficos.

- Científicos marinos sin formación especializada en visualización de datos que requieren generar figuras exploratorias rápidamente.

- Análisis preliminar de datos de campañas para identificar patrones que justifiquen análisis estadísticos formales posteriores.

- Generación de visualizaciones para presentaciones y reportes de progreso donde calidad de publicación no es crítica pero rapidez de producción es prioritaria.

En síntesis, VisBot (Ceballos, 2025) se posiciona como una herramienta de propósito intermedio que equilibra accesibilidad con capacidades analíticas, diferenciándose particularmente en su manejo de datos geoespaciales, integración de análisis exploratorio en el proceso de recomendación y explicabilidad de sus sugerencias. Los resultados cuantitativos y cualitativos documentados en esta sección confirman que el sistema logra exitosamente sus objetivos de diseño: democratizar el acceso a capacidades de visualización de datos mediante automatización inteligente, reducir barreras de entrada para usuarios no especializados, y proporcionar valor agregado mediante recomendaciones contextualmente apropiadas fundamentadas en análisis automático de características de datos. Las limitaciones identificadas, particularmente en términos de personalización avanzada y manejo de datos complejos, representan oportunidades claras para desarrollo futuro y refinamiento del sistema.

CAPÍTULO

5 DISCUSIÓN

5.1. Capacidades de VisBot

El análisis comparativo revela tres capacidades distintivas que diferencian a VisBot (Ceballos, 2025) de herramientas alternativas, detección y visualización automática de datos georreferenciados, integración de análisis exploratorio de datos con recomendación de visualización, explicabilidad comprehensiva de recomendaciones.

Detección y visualización automática de datos georreferenciados: La capacidad del sistema para identificar automáticamente variables de longitud y latitud y generar visualizaciones cartográficas apropiadas sin intervención del usuario representa una ventaja competitiva significativa. Mientras que algunas herramientas como LLM4Vis (Wang et al., 2023) pueden manejar datos geoespaciales, requieren especificación explícita o configuración manual. La automatización completa de este proceso en VisBot (Ceballos, 2025) es particularmente valiosa en dominios como ciencias ambientales, oceanografía, epidemiología espacial y análisis urbano, donde datos georreferenciados son ubicuos. Maddigan y Susnjak (2023), introducen Chat2Vis un modelo que permite ajustar visualizaciones usando lenguaje natural y LLMs.

Integración de análisis exploratorio de datos con recomendación de visualización: VisBot (Ceballos, 2025) no solamente recomienda tipos de gráficos, sino que realiza un análisis exploratorio preliminar de los datos para fundamentar sus recomendaciones. Este enfoque, que integra clasificación de tipos de datos mediante técnicas de aprendizaje automático no supervisado y detección de patrones relacionales, resulta en recomendaciones contextualmente más apropiadas. La mayoría de las herramientas alternativas operan mediante patrones predefinidos o aprendizaje supervisado que

requieren grandes conjuntos de entrenamiento, mientras que el enfoque de VisBot (Ceballos, 2025) se adapta dinámicamente a las características específicas de cada dataset.

Explicabilidad comprehensiva de recomendaciones: Si bien varias herramientas proporcionan algún nivel de explicación de sus sugerencias, VisBot (Ceballos, 2025) se distingue por la profundidad y claridad de sus justificaciones analíticas. Cada recomendación incluye no solamente el tipo de gráfico sugerido, sino una descripción de qué patrones o relaciones puede revelar esa visualización, qué preguntas analíticas puede responder y, cuando es aplicable, sugerencias de visualizaciones alternativas. Esta capacidad explicativa fue particularmente valorada tanto por usuarios novatos (valor educativo) como por usuarios experimentados (transparencia en el razonamiento del sistema).

Adaptabilidad a estándares de datos científicos: La capacidad del sistema para procesar exitosamente datos en formato Darwin Core, estándar ampliamente adoptado en comunidad de biodiversidad, demuestra flexibilidad arquitectónica para manejar vocabularios controlados específicos de dominio. El módulo de detección de tipos semánticos, entrenado principalmente con datos genéricos, demostró capacidad de generalización al identificar correctamente campos taxonómicos mediante análisis de patrones de nomenclatura binomial (Género especies) sin requerir configuración específica para dominio marino. Esta adaptabilidad sugiere que el enfoque basado en LLM puede escalar hacia otros dominios científicos con convenciones de nomenclatura específicas (e.g., astronomía con coordenadas ecuatoriales, genómica con identificadores de secuencias) sin requerir entrenamiento especializado, ventaja significativa sobre sistemas basados en reglas que requieren codificación explícita de conocimiento de dominio.

5.2. Limitaciones y Áreas de Mejora Identificadas

El análisis crítico de los resultados obtenidos y el proceso de desarrollo del sistema VisBot (Ceballos, 2025) permite identificar limitaciones metodológicas, técnicas y operacionales que condicionan la interpretación de los hallazgos y delimitan el alcance

de las contribuciones de esta investigación. Estas limitaciones se organizan jerárquicamente según su severidad y su impacto potencial sobre la validez de los resultados, la reproducibilidad de la investigación y la aplicabilidad práctica del sistema desarrollado.

5.2.1. Limitaciones Metodológicas Críticas

Las limitaciones metodológicas críticas constituyen aspectos fundamentales del diseño de investigación que afectan directamente la validez interna, la reproducibilidad y la generalización de los resultados obtenidos. Estas restricciones requieren atención prioritaria en futuras iteraciones de la investigación y deben considerarse explícitamente al interpretar las contribuciones del trabajo.

Dependencia de Infraestructura Comercial Propietaria

La arquitectura del sistema VisBot (Ceballos, 2025) depende fundamentalmente de la API comercial de GPT-4 operada por OpenAI, una dependencia que introduce vulnerabilidades significativas en múltiples dimensiones. Esta dependencia constituye la limitación metodológica más crítica identificada en la investigación por las siguientes razones convergentes, 1) Amenaza a la reproducibilidad científica, ya que la naturaleza propietaria y no determinista de los modelos de lenguaje extensos comerciales compromete la reproducibilidad estricta de los experimentos realizados. OpenAI no proporciona garantías de estabilidad temporal de los modelos, pudiendo actualizar o modificar GPT-4 (OpenAI, 2023) sin notificación previa mediante actualizaciones continuas que alteran sutilmente el comportamiento del sistema. Investigaciones previas han documentado variabilidad temporal en salidas de modelos de lenguaje nominalmente idénticos Chen et al. (2023), donde consultas idénticas producen respuestas diferentes en momentos distintos incluso con parámetros de temperatura configurados en cero. Esta variabilidad introduce incertidumbre irreducible en la replicación exacta de resultados por investigadores independientes, violando principios fundamentales de reproducibilidad científica articulados en directrices como las de la National Academies of Sciences, Engineering, and Medicine. 2) Riesgo de obsolescencia tecnológica, dado que la continuidad del servicio API depende de decisiones

comerciales unilaterales de OpenAI que podrían modificar términos de uso, estructura de precios, o incluso discontinuar servicios específicos sin considerar impacto en investigaciones dependientes. Durante el desarrollo de esta tesis, OpenAI discontinuo versiones previas de modelos (GPT-3.5-turbo-0301) con períodos de aviso de únicamente tres meses, insuficientes para investigaciones doctorales con cronogramas plurianuales. Esta volatilidad introduce riesgo existencial para la viabilidad a largo plazo del sistema desarrollado y complica el mantenimiento de implementaciones basadas en estos modelos. 3) Opacidad del modelo subyacente, haciendo foco en la ausencia de acceso a arquitectura interna, datos de entrenamiento y procedimientos de ajuste fino del modelo limita severamente la capacidad de comprender mecanismos causales que generan recomendaciones específicas. Esta opacidad dificulta la identificación de causas raíz cuando el sistema produce salidas subóptimas o erróneas, reduciendo las recomendaciones a cajas negras cuyo razonamiento interno permanece inaccesible. Aunque el sistema proporciona justificaciones textuales generadas por el propio modelo, estas explicaciones post-hoc no necesariamente reflejan procesos computacionales reales que condujeron a la decisión, planteando interrogantes sobre autenticidad de explicabilidad proporcionada. 4) Costos operacionales acumulativos. La estructura de precios basada en tokens procesados genera costos variables proporcionales al volumen de uso, complicando la sostenibilidad económica de adopción amplia particularmente en contextos académicos con presupuestos limitados. Durante la fase de evaluación empírica de esta investigación, el costo promedio por sesión de usuario fue de aproximadamente USD 0.15, pareciendo marginal individualmente pero escalando a cifras significativas en escenarios de adopción masiva. Una institución con 500 usuarios activos generando 10 visualizaciones semanales acumularía costos anuales superiores a USD 3,900, sin considerar la inflación de precios que OpenAI ha implementado históricamente.

Por lo expuesto, algunas estrategias de mitigación, para abordar esta limitación crítica, en futuras iteraciones de la investigación deberían explorar alternativas que incluyan:

- Integración de modelos de lenguaje de código abierto como LLaMA 2 (Touvron et al., 2023) o Mistral (Jiang et al., 2023) que, aunque actualmente inferiores en

capacidades absolutas a GPT-4 (OpenAI, 2023), proporciona control completo sobre infraestructura y eliminan dependencias comerciales;

- Desarrollo de modelos híbridos que combinen razonamiento simbólico mediante reglas explícitas para casos comunes con delegación a LLM únicamente para situaciones ambiguas que requieren razonamiento flexible;
- Implementación de capas de abstracción que desacoplan la lógica de recomendación de proveedor específico de modelo, facilitando migración entre backends alternativos;
- Documentación exhaustiva de versiones específicas de modelos utilizados (e.g., gpt-4-0613) y archivado de prompts exactos y respuestas obtenidas para permitir replicación aproximada.

5.2.2. Limitaciones Técnicas Moderadas

Las limitaciones técnicas moderadas afectan la funcionalidad, el rendimiento y la amplitud de aplicabilidad del sistema, pero no comprometen la validez fundamental de los resultados obtenidos ni la viabilidad de las contribuciones principales. Estas restricciones representan áreas prioritarias de desarrollo futuro podrían expandir significativamente la utilidad práctica del sistema.

Capacidades de Personalización Visual Limitadas

La evaluación comparativa documentada en la sección 4.4.2 revela que VisBot (Ceballos, 2025) alcanza puntuación de 5 sobre 10 en nivel de personalización, posicionándose en rango intermedio significativamente inferior a herramientas especializadas de visualización profesional como Tableau (puntuación estimada 9/10) o Microsoft Power BI (8/10). Esta limitación se manifiesta en las siguientes dimensiones de control sobre apariencia y comportamiento de visualizaciones generadas:

Restricciones en configuración de elementos estéticos. El sistema proporciona control sobre parámetros fundamentales como selección de variables para ejes cartesianos, esquemas cromáticos básicos mediante paletas predefinidas, y ajuste de tamaño de marcadores mediante sliders numéricos. Sin embargo, carece de capacidades

avanzadas que usuarios profesionales frecuentemente requieren, incluyendo: especificación precisa de colores mediante códigos hexadecimales o RGB para cumplir requisitos de identidad corporativa o estándares de publicación científica; configuración detallada de tipografía con selección de familias de fuentes, tamaños específicos y estilos (negrita, cursiva) para títulos, etiquetas y anotaciones; control sobre líneas de cuadrícula con especificación de estilo (continua, punteada), color, grosor y opacidad; y personalización de leyendas con control sobre posición, orientación, tamaño y formato de entradas.

Limitaciones en transformaciones de datos complejas. A diferencia de plataformas profesionales que proporcionan lenguajes de expresión potentes para cálculos personalizados (DAX en Power BI, fórmulas calculadas en Tableau), VisBot (Ceballos, 2025) opera sobre datos tal como se cargan, sin capacidades incorporadas para: creación de columnas calculadas mediante fórmulas que combinen múltiples campos; agregaciones personalizadas más allá de suma, promedio, conteo y mediana predefinidos; filtrado condicional complejo mediante expresiones lógicas compuestas; y transformaciones temporales como cálculo de diferencias período-sobre-período o promedios móviles.

Ausencia de composición de visualizaciones múltiples. El sistema genera visualizaciones individuales independientes sin capacidad de crear dashboards compuestos que integren múltiples gráficos relacionados en diseños coordinados. Esta limitación es significativa porque análisis exploratorio efectivo frecuentemente requiere examinar simultáneamente múltiples perspectivas sobre datos, con interacciones sincronizadas dónde selección en una vista filtra automáticamente otras vistas vinculadas. La implementación de dashboards multicomponente demandaría arquitectura sustancialmente más compleja que la actual pipeline lineal de procesamiento.

Sobre lo expuesto se plantean las siguientes estrategias de mitigación identificadas, para extender escalabilidad del sistema a datasets masivos, se proponen las siguientes direcciones de desarrollo futuro: (a) implementación de procesamiento por lotes (chunking) donde dataset se lee incrementalmente en fragmentos que caben en

memoria, calculando estadísticos mediante algoritmos incrementales que actualizan acumuladores sin requerir acceso simultáneo a todas las observaciones; (b) integración con bibliotecas especializadas en computación fuera de núcleo como Dask que particiona datasets grandes en fragmentos procesables en memoria disponible, coordinando operaciones mediante grafos de tareas; (c) implementación de agregación inteligente que resume automáticamente datos antes de visualización cuando cardinalidad excede umbrales (e.g., binning de variables continuas, agrupamiento espacial de coordenadas geográficas); y (d) exploración de backends alternativos de visualización optimizados para rendering de grandes volúmenes de datos, como Datashader que rasteriza millones de puntos en bitmaps agregados eficientemente.

Soporte Limitado para Tipos de Datos Especializados

Aunque el sistema maneja efectivamente tipos de datos comunes (numéricos, categóricos, temporales, geoespaciales bidimensionales), presenta limitaciones en procesamiento de estructuras más especializadas que aparecen frecuentemente en aplicaciones científicas avanzadas.

Datos geoespaciales tridimensionales. El sistema detecta y visualiza coordenadas bidimensionales (latitud, longitud) mediante mapas planos, pero carece de capacidades para representar adecuadamente información tridimensional que incluye profundidad o altitud como tercera dimensión espacial. Datos oceanográficos frecuentemente requieren visualización de perfiles verticales de columna de agua o secciones transversales que muestran variación de propiedades con profundidad, capacidad actualmente no implementada.

Series temporales multivariadas complejas. Aunque el sistema identifica variables temporales y puede generar gráficos de líneas básicos, carece de capacidades especializadas para análisis de series temporales como descomposición estacional-tendencial, detección automática de puntos de cambio, o visualización de autocorrelación y estacionalidad mediante correlogramas.

Datos textuales no estructurados. El sistema opera sobre datos tabulares estructurados y no incorpora capacidades de procesamiento de lenguaje natural para análisis de

columnas que contienen texto libre, como comentarios, descripciones o documentos, omitiendo técnicas como nube de palabras, análisis de sentimiento o modelado de tópicos.

Estrategias de extensión, para futuras versiones deberían incorporarse módulos especializados para tipos de datos adicionales mediante arquitectura de plugins que permita extensibilidad sin incrementar complejidad percibida para usuarios que no requieren estas capacidades.

5.2.3. Limitaciones Menores de Diseño e Integración

Las limitaciones menores identificadas afectan la experiencia de usuario, la estética de la interfaz y la integración con flujos de trabajo existentes, pero no comprometen significativamente la funcionalidad central ni la validez de las contribuciones científicas del sistema. Estas restricciones representan oportunidades de refinamiento iterativo que mejorarían la adopción y satisfacción del usuario sin requerir modificaciones arquitectónicas fundamentales.

Refinamiento Estético de la Interfaz Gráfica

La evaluación comparativa documentó puntuación de 6 sobre 10 en calidad de interfaz gráfica de usuario, posicionando a VisBot (Ceballos, 2025) en rango intermedio inferior a herramientas con interfaces altamente pulidas como LIDA (Dibia, 2023)(9/10) o Chat2Vis (Maddigan y Susnjak, 2023) (8/10). Esta limitación se manifiesta en aspectos específicos de diseño visual y experiencia de interacción.

Densidad informativa y organización espacial. Algunos componentes de la interfaz, particularmente el panel de recomendaciones generadas por inteligencia artificial, presentan densidad informativa elevada con justificaciones textuales extensas, controles de configuración múltiples y visualizaciones preliminares dispuestos en espacio limitado. Durante evaluación con usuarios, varios participantes comentaron sensación de saturación visual inicial, requiriendo orientación para identificar elementos interactivos primarios versus contenido informativo secundario.

Feedback visual de estados del sistema. Durante operaciones computacionalmente intensivas como procesamiento de datasets grandes o invocación de API de GPT-4 (OpenAI, 2023) que puede tomar varios segundos, el sistema proporciona indicadores de progreso básicos (spinners animados) pero carece de feedback informativo que comunique específicamente qué operación está ejecutándose y progreso estimado, potencialmente generando ansiedad en usuarios sobre sí el sistema está funcionando correctamente o se ha bloqueado.

Justificación del estado actual. La priorización de desarrollo enfocó recursos en implementación de capacidades analíticas centrales sobre refinamiento estético, decisión apropiada para prototipo de investigación que busca validar viabilidad técnica y efectividad funcional antes de invertir en diseño visual exhaustivo. Participantes en evaluación indicaron que, aunque interfaz podría mejorarse estéticamente, no constituyó impedimento para completar tareas exitosamente.

Mejoras incrementales recomendadas para un refinamiento futuro: (a) implementar sistema de diseño cohesivo mediante personalización de temas de Streamlit con paleta cromática unificada; (b) optimizar densidad informativa mediante diseño progresivo que revele complejidad incrementalmente según necesidad del usuario; (c) incorporar animaciones suaves de transición entre estados que comuniquen cambios sin abruptos; y (d) implementar feedback contextual durante operaciones asíncronas con mensajes descriptivos sobre progreso específico.

Ausencia de Integración con Entornos de Análisis Programático

El sistema opera como aplicación web independiente accesible mediante navegador, sin integración directa con entornos de desarrollo interactivo ampliamente adoptados en comunidad científica como Jupyter Notebooks, RStudio, o Visual Studio Code. Esta ausencia de interoperabilidad limita la incorporación del sistema en flujos de trabajo analíticos existentes donde investigadores típicamente combinan análisis programático, visualización y documentación narrativa en cuadernos computacionales.

Implicaciones para adopción en contextos científicos. Científicos de datos y analistas cuantitativos frecuentemente desarrollan análisis como secuencia de celdas de código

en notebooks que combinan transformación de datos mediante Pandas o tidyverse, modelado estadístico mediante scikit-learn o statsmodels, y visualización mediante matplotlib o ggplot2, todo documentado con markdown narrativo. La necesidad de migrar a aplicación web separada para generación de visualizaciones interrumpe fluidez de este flujo de trabajo, introduciendo fricción que puede desincentivar adopción.

Alternativas de integración exploradas. Durante el diseño del sistema se consideraron múltiples aproximaciones de integración: (a) desarrollo de biblioteca Python que exponga funcionalidad de VisBot (Ceballos, 2025) como API programática invocable desde notebooks, requiriendo refactorización sustancial de arquitectura actual basada en Streamlit; (b) implementación de extensión de Jupyter que incorpore interfaz de VisBot (Ceballos, 2025) como widget embebido en notebooks, técnicamente viable mediante jupyter-streamlit pero con complejidad de mantenimiento significativa; (c) provisión de funcionalidad de exportación de visualizaciones generadas como código Plotly Express reutilizable que usuarios pueden copiar en notebooks, aproximación más simple que preserva separación de aplicaciones.

Decisión de diseño actual, se optó por mantener VisBot (Ceballos, 2025) como aplicación independiente durante fase inicial de desarrollo para maximizar velocidad de iteración y simplificar arquitectura, posponiendo integración programática para iteraciones futuras una vez validada viabilidad central. La provisión de exportación de visualizaciones en formatos estándares (HTML interactivo, PNG estático, SVG vectorial) proporciona interoperabilidad parcial mediante flujo de trabajo manual.

Capacidades Colaborativas y Multiusuario Ausentes

El sistema implementa modelo de uso individual donde cada sesión de usuario es independiente sin capacidades de compartir análisis, colaborar en tiempo real, o mantener históricos persistentes de visualizaciones generadas más allá de duración de sesión del navegador.

Limitaciones identificadas. Usuarios no pueden: guardar configuraciones de análisis para reutilización futura; compartir URLs que reproduzcan exactamente visualizaciones

generadas con colegas; colaborar sincrónicamente con múltiples analistas explorando datos simultáneamente; o mantener galerías personales de visualizaciones previamente creadas para referencia posterior.

Justificación de ausencia. La implementación de capacidades colaborativas robustas requeriría infraestructura backend de persistencia (base de datos para almacenar configuraciones, análisis y preferencias de usuarios), sistema de autenticación y autorización, y gestión de sesiones compartidas con sincronización en tiempo real, incrementando complejidad arquitectónica sustancialmente. Para prototipo de investigación enfocado en validar capacidades de recomendación inteligente, estos requisitos representaban expansión de alcance no crítica para objetivos centrales.

Direcciones futuras. Próximas versiones orientadas a producción deberían incorporar: almacenamiento persistente de análisis mediante base de datos relacional o documental; generación de URLs compartibles con estado serializado en parámetros de consulta o identificadores únicos; y potencialmente capacidades de anotación colaborativa donde múltiples usuarios pueden comentar visualizaciones compartidas.

5.3. Estándares y consideraciones éticas

La comunidad de datos científica utiliza múltiples estándares incompatibles para diferentes aspectos de gestión y visualización de datos. Esta fragmentación crea barreras significativas para integración de datos y desarrollo de herramientas interoperables. Los datos oceanográficos se almacenan en formatos diversos (NetCDF-CF, HDF5, OPeNDAP, formatos propietarios) que requieren bibliotecas específicas para acceso. Aunque existen esfuerzos de estandarización como las convenciones CF (Climate and Forecast), la adopción es inconsistente y muchos datasets heredados permanecen en formatos no estándar. Esta diversidad de formatos requiere que herramientas de visualización implementen soporte para múltiples formatos, aumentando complejidad de desarrollo y mantenimiento. Además, las diferencias en convenciones de metadatos entre formatos complican la integración automática de datasets de múltiples fuentes. Los datos marinos utilizan una variedad

de sistemas de coordenadas y proyecciones cartográficas, desde coordenadas geográficas simples hasta proyecciones especializadas para regiones polares o aplicaciones específicas. La transformación entre sistemas de coordenadas puede introducir errores y complicar análisis que integran múltiples fuentes de datos. La evolución rápida de tecnologías web y capacidades computacionales resulta en obsolescencia regular de estándares y herramientas. Esta evolución continua requiere actualizaciones frecuentes de herramientas de visualización y puede dejar sistemas existentes sin soporte o compatibilidad. Aunque existen esfuerzos para desarrollar ontologías marinas como proponen Tzitzikas et al. (2013), la adopción es limitada y muchos conceptos críticos carecen de definiciones semánticas estandarizadas. Esta falta de semántica común complica la integración automática de datos de múltiples fuentes y limita capacidades de búsqueda y descubrimiento. La diversidad de tradiciones disciplinarias dentro de las ciencias marinas resulta en terminologías inconsistentes para conceptos similares, creando barreras para la integración interdisciplinaria. Por ejemplo, los biólogos marinos y oceanógrafos físicos pueden usar terminologías diferentes para describir las mismas características espaciales o temporales. Los conceptos marinos frecuentemente involucran relaciones complejas que son difíciles de capturar en ontologías formales. Por ejemplo, la definición de "masa de agua" involucra rangos de temperatura y salinidad que varían regionalmente y estacionalmente, creando desafíos para la representación semántica precisa. Por su parte, los LLM, entrenados en datos que reflejan sesgos sociales, pueden propagarse a través de recomendaciones de visualización. Por ejemplo, selección sistemática de colores o metáforas que refuerzan estereotipos. La investigación en detección y mitigación de sesgos en sistemas de recomendación de visualización es crítica. A medida que estos sistemas se adoptan para análisis de alto impacto (políticas públicas, decisiones clínicas), la capacidad de auditar y explicar decisiones se vuelve crucial. Esto requiere no solo mejorar interpretabilidad técnica, sino también desarrollar estándares y marcos regulatorios apropiados. Y si bien el objetivo es democratizar acceso a analítica avanzada, existe riesgo de que estas herramientas amplíen la brecha entre usuarios con recursos computacionales y expertise para configurarlas efectivamente versus aquellos sin estos recursos.

CAPÍTULO

6 CONCLUSIONES

La presente investigación doctoral desarrolló, implementó y validó empíricamente un sistema de recomendación inteligente de visualizaciones que integra modelos de lenguaje de gran escala con principios consolidados de diseño visual. El trabajo partió de la hipótesis de que esta convergencia tecnológica podría generar recomendaciones simultáneamente precisas, interpretables y accesibles para usuarios con distintos niveles de experiencia técnica. Los resultados obtenidos confirman en parte esta premisa, al demostrar la viabilidad técnica del enfoque mediante evaluación con datos científicos reales del dominio marino (Campagna et al., 2015), delimitando con precisión los márgenes actuales de su efectividad en contextos de investigación auténticos. La validación con datos oceanográficos reales proporcionó evidencia particularmente valiosa sobre la capacidad del sistema para manejar desafíos característicos de ciencias ambientales, detección automática de coordenadas geográficas con rangos específicos de regiones de estudio, reconocimiento de nomenclatura taxonómica estándar (Darwin Core), y generación de visualizaciones apropiadas para análisis de distribución espacial y temporal de especies marinas. Biólogos marinos participantes en la evaluación completaron tareas analíticas reales (identificación de hotspots de biodiversidad, caracterización de estacionalidad) en tiempos significativamente inferiores a los reportados con herramientas convencionales, y evaluaron las visualizaciones generadas con puntuaciones superiores al promedio general del sistema, sugiriendo particular efectividad en este dominio de aplicación. La arquitectura propuesta evidenció que la integración modular de interpretación lingüística, razonamiento semántico y generación visual permite traducir estructuras de datos complejas en recomendaciones coherentes y alineadas con los

principios visuales establecidos. Sin embargo, persisten limitaciones vinculadas a la dependencia de metadatos completos y bien estructurados. Estas restricciones no invalidan la propuesta, sino que orientan su mejora futura, destacando la importancia de incorporar mecanismos de gestión de errores en la estructura de los datos y enriquecimiento semántico automatizado. Desde una perspectiva más amplia, la investigación ofrece aportes conceptuales y prácticos para distintos dominios. En el ámbito de los modelos de lenguaje, evidencia que es posible extender sus capacidades más allá de la comprensión textual hacia el razonamiento sobre decisiones visuales, siempre que se les proporcione un contexto estructurado adecuado. En el terreno de la visualización automática, el enfoque híbrido desarrollado se ubica en un punto intermedio entre los sistemas puramente basados en reglas y los conexionistas, logrando un equilibrio entre flexibilidad e interpretabilidad. Estas mejoras permitirían reforzar la coherencia semántica de las visualizaciones, inferir relaciones implícitas y ofrecer información sobre la calidad y la confiabilidad de los datos presentados. Paralelamente, se reconocen desafíos éticos asociados a los sesgos inherentes a los modelos de lenguaje, lo que exige auditorías sistemáticas, mecanismos de retroalimentación y trazabilidad completa del proceso de recomendación, especialmente en contextos sensibles como la biomedicina o la formulación de políticas públicas. En síntesis, el trabajo demuestra que la combinación entre inteligencia artificial generativa y fundamentos teóricos del diseño visual puede transformar la forma en que las personas interactúan con los datos. Los límites identificados no constituyen obstáculos insalvables, sino hitos que orientan el avance hacia sistemas más robustos, transparentes y adaptativos. La visión que se desprende de esta tesis proyecta un futuro en el que la distancia entre el pensamiento analítico humano y su representación visual se reduzca progresivamente, permitiendo que la exploración y comunicación de conocimiento a partir de datos complejos se vuelva un proceso natural, accesible y cognitivamente ampliado. Los resultados alcanzados con VisBot (Ceballos, 2025) ofrecen una base empírica para seguir avanzando hacia ese horizonte, consolidando un aporte tangible al desarrollo de sistemas de visualización inteligentes.

REFERENCIAS

Ahmad, H. (2019). Machine learning applications in oceanography. Aquatic Research, 2(3), 161-169. https://doi.org/10.3153/AR19014

Ali, W., Gao, Y., Foucart, C., Doshi, M., Mirabito, C., Haley, P., y Lermusiaux, P. (2022). High-performance visualization for ocean modeling. OCEANS 2022, Hampton Roads, 1-10. https://doi.org/10.1109/OCEANS47191.2022.9977075

Amar, R., Eagan, J., y Stasko, J. (2005). Low-level components of analytic activity in information visualization. En IEEE Symposium on Information Visualization, 2005. INFOVIS 2005 (pp. 111-117). IEEE.

Atemezing, G. A., y Troncy, R. (2014). Towards a linked-data based visualization wizard. En Proceedings of the 6th International Workshop on Consuming Linked Data (COLD 2014). CEUR-WS.org. https://ceur-ws.org/Vol-1264/cold2014_AtemezingT.pdf

Bao, K., Zhang, J., Zhang, Y., Wang, W., Feng, F., y He, X. (2023). TALLRec: An effective and efficient tuning framework to align large language model with recommendation. En Proceedings of the 17th ACM Conference on Recommender Systems (pp. 1007-1014). ACM. https://doi.org/10.1145/3604915.3608857

Berners-Lee, T., Hendler, J., y Lassila, O. (2001). The semantic web. Scientific American, 284(5), 28-37.

Bertin, J. (1967). Sémiologie Graphique. Gauthier-Villars.

Beyan, C., y Browman, H. I. (2020). Setting the stage for the machine intelligence era in marine science. ICES Journal of Marine Science, 77(4), 1267-1273. https://doi.org/10.1093/icesjms/fsaa084

Bikakis, N., Liagouris, J., Krommyda, M., Papastefanatos, G., y Sellis, T. (2014). GraphViz: A scalable platform for interactive large graph visualization. En Proceedings of the 30th IEEE International Conference on Data Engineering (pp. 1342-1345). IEEE.

Bizer, C., Heath, T., y Berners-Lee, T. (2011). Linked Data: The story so far. International Journal on Semantic Web and Information Systems, 5(3), 1-22.

Brooke, J. (1996). SUS: A quick and dirty usability scale. En P. W. Jordan, B. Thomas, B. A. Weerdmeester, y I. L. McClelland (Eds.), Usability evaluation in industry (pp. 189-194). Taylor & Francis.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877-1901.

Brunetti, J. M., Auer, S., y García, R. (2012). The linked data visualization model. En International Semantic Web Conference (Posters & Demos), 2012. CEUR-WS.org.

Brunetti, J. M., Auer, S., García, R., Klímek, J., y Nečaský, M. (2013). Formal linked data visualization model. En Proceedings of International Conference on Information Integration and Web-based Applications and Services (IIWAS '13) (pp. 309:309-309:318). ACM.

Campagna, C., Lewis, M. N., y Marin, M. R. (2015). Locations of seals in Patagonian Large Marine Ecosystem (OBIS South America, AR-OBIS, Sub-node). v1.6. ArOBIS Centro Nacional Patagónico. Dataset/Occurrence. https://doi.org/10.15468/qcwa73

Card, S. K., Mackinlay, J. D., y Shneiderman, B. (1999). Readings in information visualization: Using vision to think. Morgan Kaufmann.

Ceballos, D. (2025a). VisBot [Aplicación web]. https://visbot.streamlit.app/

Ceballos, D. (2025b). VisBot [Repositorio GitHub]. https://github.com/disenodc/VisBot

Ceballos, D., Zárate, M., y Delrieux, C. (2025). Enhancing data analytics with AI-powered visualization recommendations: A large language model approach. TechRxiv. https://doi.org/10.36227/techrxiv.175303443.39749915/v1

Cheatham, M., Krisnadhi, A., Amini, R., Hitzler, P., Janowicz, K., Narock, T., Shepherd, A., Vardeman, C., Wilson, M., Winslett, M., y Zinn, D. (2018). The GeoLink framework for semantic web-based analysis and visualization of oceanographic data. En International Semantic Web Conference (pp. 418-433). Springer.

Chen, L., Zaharia, M., y Zou, J. (2023). How is ChatGPT's behavior changing over time? arXiv preprint arXiv:2307.09009. https://arxiv.org/abs/2307.09009

Chen, Q., Sun, F., Xu, L., Su, Z., y Wu, Y. (2022). VizLinter: A linter and fixer framework for data visualization. IEEE Transactions on Visualization and Computer Graphics, 28(1), 206-216. https://doi.org/10.1109/TVCG.2021.3114804

Chi, E. H. (2000). A taxonomy of visualization techniques using the data state reference model. En IEEE Symposium on Information Visualization, 2000. InfoVis 2000 (pp. 69-75). IEEE.

Clark, G., Gacutan, J., Lawther, R., Johnston, E., Tait, H., y Bednarz, T. (2021). A visualization tool for citizen-science marine debris big data. Water International, 46(3), 211-223. https://doi.org/10.1080/02508060.2021.1888495

Cleveland, W. S., y McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. Journal of the American Statistical Association, 79(387), 531-554.

Dadzie, A.-S., y Rowe, M. (2011). Approaches to visualising linked data: A survey. Semantic Web, 2(2), 89-124.

Devaraju, A., y Herterich, P. (2019). Re3data and DataCite: An analysis of repository and journal article connections. Data Science Journal, 18(1), 44. https://doi.org/10.5334/dsj-2019-044

Dibia, V. (2023). LIDA: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models. En Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics: System Demonstrations (pp. 113-126). ACL.

Dibia, V., y Demiralp, Ç. (2019). Data2Vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks. IEEE Computer Graphics and Applications, 39(5), 33-46.

GBIF. (2023). GBIF: The Global Biodiversity Information Facility. https://www.gbif.org

Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J., y Cajander, Å. (2003). Key principles for user-centred systems design. Behaviour & Information Technology, 22(6), 397-409.

Hardisty, A., y Roberts, D. (2013). A decadal view of biodiversity informatics: Challenges and priorities. BMC Ecology, 13, 16.

He, Y., Wang, Y., Deng, Z., Wang, J., Zhang, H., Yuan, X., y Wu, Y. (2025). Leveraging foundation models for crafting narrative visualization: A survey. IEEE Transactions on Visualization and Computer Graphics, PrePrints 5555, 1-20. https://doi.org/10.1109/TVCG.2025.3542504

Healey, C. G., y Enns, J. T. (2012). Attention and visual memory in visualization and computer graphics. IEEE Transactions on Visualization and Computer Graphics, 18(7), 1170-1188.

Heer, J., Bostock, M., y Ogievetsky, V. (2010). A tour through the visualization zoo. Communications of the ACM, 53(6), 59-67.

Hevner, A. R., March, S. T., Park, J., y Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75-105.

Hu, K., Bakker, M. A., Li, S., Kraska, T., y Hidalgo, C. (2019). VizML: A machine learning approach to visualization recommendation. En Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (pp. 1-12). ACM.

Isinkaye, F. O., Folajimi, Y. O., y Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal, 16(3), 261-273. https://doi.org/10.1016/j.eij.2015.06.005

Janowicz, K., Hitzler, P., Adams, B., Kolas, D., y Vardeman, C. (2014). Five stars of Linked Data vocabulary use. Semantic Web, 5(3), 173-176.

Ji, P., Tian, F., Liu, S., y Jiang, G. (2016). I4Ocean: An interactive simulation and scientific visualization platform for marine application. Ocean Engineering, 121, 234-245. https://doi.org/10.21311/001.39.5.23

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. de las, Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., y Sayed, W. E. (2023). Mistral 7B. arXiv preprint arXiv:2310.06825. https://arxiv.org/abs/2310.06825

Jiang, M., y Zhu, Z. (2022). The role of artificial intelligence algorithms in marine scientific research. Frontiers in Marine Science, 9, 920994. https://doi.org/10.3389/fmars.2022.920994

Kitchenham, B. (2004). Procedures for performing systematic reviews. Keele University Technical Report TR/SE-0401.

Klímek, J., Helmich, J., y Nečaský, M. (2014). Application of the linked data visualization model on real world data from the czech LOD cloud. En CEUR Workshop Proceedings (Vol. 1343, pp. 1-7). CEUR-WS.org.

Krisnadhi, A. A., Hu, Y., Janowicz, K., Hitzler, P., Arko, R. A., Carbotte, S., Chandler, C., Cheatham, M., Fils, D., Finin, T., Ji, P., Jones, M., Karima, N., Mickle, A., Narock, T., O'Brien, M., Raymond, L., Shepherd, A., Sinha, K., ... West, P. (2015). The GeoLink modular oceanography ontology. En The Semantic Web – ISWC 2015 (pp. 301-309). Springer. https://doi.org/10.1007/978-3-319-25007-6_18

Li, L., Zhang, Y., y Chen, L. (2023). Large language models for recommendation: A survey and new perspectives. IEEE Transactions on Knowledge and Data Engineering, 35(11), 11234-11247. https://doi.org/10.1109/TKDE.2023.3285950

Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. ACM Transactions on Graphics, 5(2), 110-141.

Maddigan, P., y Susnjak, T. (2023). Chat2Vis: Fine-tuning data visualisations using multilingual natural language text and pre-trained large language models. arXiv preprint arXiv:2303.14292. https://arxiv.org/abs/2303.14292

Malik, T., y Foster, I. (2012). Scientific data management: Challenges, existing technology, and deployment. En Data Intensive Distributed Computing: Challenges and Solutions for Large-Scale Information Management (pp. 1-28). IGI Global.

Marchionini, G. (2006). Exploratory search: From finding to understanding. Communications of the ACM, 49(4), 41-46.

Megrey, B., Hinckley, S., y Dobbins, E. (2002). Using scientific visualization tools to facilitate analysis of multi-dimensional data from a spatially explicit, biophysical, individual-based model of marine fish early life history. ICES Journal of Marine Science, 59(1), 203-215. https://doi.org/10.1006/JMSC.2001.1150

Mumuni, A., y Mumuni, F. (2025). Automated data processing and feature engineering for deep learning and big data applications: A survey. Journal of Information and Intelligence, 3(2), 113-153. https://doi.org/10.1016/j.jiixd.2024.01.002

Munzner, T. (2014). Visualization analysis and design. CRC Press.

NOAA/PMEL. (2015). Ferret: Data analysis and visualization tool [Software]. https://ferret.pmel.noaa.gov/Ferret/

OpenAI. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774. https://arxiv.org/abs/2303.08774

Pace, R. K., y Barry, R. (1997). Sparse spatial autoregressions. Statistics & Probability Letters, 33(3), 291-297.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., y Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

Peña, O., Aguilera, U., y López-de-Ipiña, D. (2016). Linked Open Data visualization revisited: A survey. Semantic Web, 7(5), 529-550.

Po, L., Bikakis, N., Desimoni, F., y Papastefanatos, G. (2020). Linked data visualization: Techniques, tools and big data. Synthesis Lectures on Semantic Web: Theory and Technology, 10(1), 1-157.

Podo, L., Angelini, M., y Velardi, P. (2024). V-Recs, a low-cost LLM4Vis recommender with explanations, captioning and suggestions. arXiv preprint arXiv:2406.15259. https://arxiv.org/abs/2406.15259

Podo, L., Prenkaj, B., y Velardi, P. (2023). Machine learning for visualization recommendation systems: Open challenges and future directions. arXiv preprint arXiv:2302.04816. https://arxiv.org/abs/2302.04816

Roy, D., y Dutta, M. (2022). A systematic review and research perspective on recommender systems. Journal of Big Data, 9, 59. https://doi.org/10.1186/s40537-022-00592-5

Schlitzer, R. (2015). Ocean Data View [Software]. Alfred Wegener Institute. https://odv.awi.de

Shneiderman, B. (1992). Tree visualization with tree-maps: 2-D space-filling approach. ACM Transactions on Graphics, 11(1), 92-99.

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. En IEEE Symposium on Visual Languages, 1996. Proceedings (pp. 336-343). IEEE.

Skjæveland, M. G. (2012). Sgvizler: A javascript wrapper for easy visualization of SPARQL result sets. En Extended Semantic Web Conference (pp. 194-208). Springer.

Song, T., Pang, C., Hou, B., Xu, G., Xue, J., Sun, H., y Meng, F. (2023). A review of artificial intelligence in marine science. Frontiers in Earth Science, 11, 1090185. https://doi.org/10.3389/feart.2023.1090185

Streamlit Inc. (2023). Streamlit [Software]. https://streamlit.io

Tian, Y., Cui, W., Deng, D., Yi, X., Yang, Y., Zhang, H., y Wu, Y. (2024). ChartGPT: Leveraging LLMs to generate charts from abstract natural language. IEEE Transactions on Visualization and Computer Graphics, 30(1), 244-254.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., ... Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288. https://arxiv.org/abs/2307.09288

Treisman, A. M., y Gelade, G. (1980). A feature-integration theory of attention. Cognitive Psychology, 12(1), 97-136.

Tukey, J. W. (1977). Exploratory data analysis. Pearson.

Tzitzikas, Y., Allocca, C., Bekiari, C., Marketakis, Y., Fafalios, P., Doerr, M., Minadakis, N., Patkos, T., y Candela, L. (2013). Integrating heterogeneous and distributed information about marine species through a top level ontology. En The Semantic Web - ISWC 2013 (pp. 724-739). Springer.

Vandenbussche, P. Y., Atemezing, G. A., Poveda-Villalón, M., y Vatant, B. (2017). Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web. Semantic Web, 8(3), 437-452. https://doi.org/10.3233/SW-160213

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., y Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998-6008.

Wang, L., Zhang, S., Wang, Y., Lim, E., y Wang, Y. (2023). LLM4Vis: Explainable visualization recommendation using ChatGPT. En Conference on Empirical Methods in Natural Language Processing, 2023 (pp. 6267-6281). ACL.

Wang, Y., Li, F., Zhang, B., y Li, X. (2021). Development of a component-based interactive visualization system for the analysis of ocean data. Big Earth Data, 6(2), 219-235. https://doi.org/10.1080/20964471.2021.1994362

Ware, C. (2008). Visual thinking for design. Morgan Kaufmann.

Ware, C. (2012). Information visualization: Perception for design (3ª ed.). Morgan Kaufmann.

Wilkinson, L. (2012). The grammar of graphics. En J. Gentle, W. Härdle, y Y. Mori (Eds.), Handbook of Computational Statistics (pp. 375-414). Springer. https://doi.org/10.1007/978-3-642-21551-3_13

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., Santos, L. B. da S., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., ... Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data, 3, 160018.

Wong, P., Thomas, J., y Wiley, S. (2004). Visual analytics. IEEE Computer Graphics and Applications, 20(5), 20-21. https://doi.org/10.1109/MCG.2004.39

Wongsuphasawat, K., Moritz, D., Anand, A., Mackinlay, J., Howe, B., y Heer, J. (2016). Voyager: Exploratory analysis via faceted browsing of visualization recommendations. IEEE Transactions on Visualization and Computer Graphics, 22(1), 649-658.

https://doi.org/10.1109/TVCG.2015.2467191

Wu, Y., Wan, Y., Zhang, H., Sui, Y., Wei, W., Zhao, W., Xu, G., y Jin, H. (2024). Automated data visualization from natural language via large language models: An exploratory study. Proceedings of the ACM on Management of Data, 2(3), Article 149.

Xie, J., Chen, Y., Zhang, H., Liu, X., y Ma, L. (2019). Marine big data and its applications in marine science. En Proceedings of the 2019 International Conference on Big Data Technologies (pp. 45-52). ACM.

Xie, J., Zhu, F., Huang, M., Xiong, N., Huang, S., y Xiong, W. (2019). Unsupervised learning of paragraph embeddings for context-aware recommendation. IEEE Access, 7, 43100-43109.

Yang, W., Liu, M., Wang, Z., Ge, T., Jiang, J., Ni, M., Lin, L., Huang, X., Zhao, D., y Wu, Y. (2024). Foundation models meet visualizations: Challenges and opportunities. Computational Visual Media, 10, 399-424. https://doi.org/10.1007/s41095-023-0393-x

Ye, Y., Hao, J., Hou, Y., Wang, Z., Xiao, S., Luo, Y., y Zeng, W. (2024). Generative AI for visualization: State of the art and future directions. Visual Informatics, 8(2), 43-66. https://doi.org/10.1016/j.visinf.2024.04.003

Zárate, M. D. (2020). Enhancing biodiversity data discovery and integration through Linked Open Data and Semantic Web technologies. GBIF Young Researchers Award Project. Global Biodiversity Information Facility.

Zárate, M. D., Buckle, C., Mazzanti, R., y Samec, G. (2019). Improving open science using Linked Open Data: CONICET Digital use case. Journal of Computer Science and Technology, 19(1), 71-78.

Zárate, M. D., Lewis, M. N., y Soto, A. J. (2019). OceanGraph: A Linked Open Data platform for integrating oceanographic data. En The International Semantic Web Conference. Springer.

Zárate, M. D., Lewis, M. N., Soto, A. J., y Copello, S. (2018). Publishing Argentine oceanographic research campaigns as Linked Data. En VOILA@ ISWC (pp. 49-60). CEUR-WS.org.

Zárate, M. D., Lewis, M. N., Soto, A. J., y Copello, S. (2020). Exploiting OceanGraph: A knowledge graph for oceanographic data discovery and integration. Semantic Web, 11(6), 961-987.