PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

DESARROLLO DE UNA HERRAMIENTA PARA LA PREDICCIÓN DE ESTRUCTURAS TERCIARIAS DE PROTEÍNAS REPETIDAS A PARTIR DE SU ESTRUCTURA PRIMARIA

Tesis para obtener el título profesional de Ingeniera Informática

AUTORA:

Solange Estrella Palomino Chahua

ASESOR:

Dra. Layla Hirsh Martinez

Lima, noviembre, 2021

Declaración jurada de autenticidad

Yo, Layla Hirsh Martinez, docente de la Facultad de Ciencias e Ingenieria de la Pontificia Universidad Católica del Perú, asesor(a) de la tesis/el trabajo de investigación titulado Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria de la autora: Solange Estrella Palomino Chahua

dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 13%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 26/11/2022.
- He revisado con detalle dicho reporte y confirmo que cada una de las coincidencias detectadas no constituyen plagio alguno.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha: .....28 diciembre 2022.....

Apellidos y nombres de la asesora: Hirsh Martinez Layla	
DNI: 40329232	Firma
ORCID: 0000-0002-8215-6716	

Dedicatoria

A mi papá, Abraham Palomino, quien siempre me inculcó la semilla de la educación y la perseverancia. Quien seguro es el más feliz al saber que voy acabando mi vida universitaria con este proyecto y seguro me acompaña en la celebración desde el cielo.

A mi mamá, Dolores Chahua, quien siempre entendió y me apoyó en los días y noches de dedicación a este proyecto y todas mis labores, desde el colegio. Por enseñarme el valor del esfuerzo y siempre ser mi inspiración.

A mis hermanos, Brigitte, Estephany y Juan Roberth; mis cuñados; y mis amigas y amigos; quienes siempre son la fuente donde recargo mis energías, por todas las risas y porque me motivan a continuar planteándome retos.

A mis tíos, Emerson y Mariela, por confiar en mí y apoyar mis decisiones.

A mi asesora, Dra. Layla Hirsh, por abrirme la puerta a un mundo de oportunidades, por confiar en mi talento, por apoyarme en los momentos difíciles y por guiarme en mi realización profesional.

A mí, porque aprendo todos los días, por continuar a pesar de las circunstancias complicadas y por seguir esforzándome para ser un orgullo para mi familia.

Resumen

La predicción de estructuras de proteínas es uno de los retos más importantes de la biología y la bioinformática (Lopes et al., 2019). Esta última es el campo de investigación que se apoya en la computación para analizar la información relacionada a las macromoléculas biológicas como las proteínas (Xiong, 2006). Las proteínas son moléculas esenciales compuestas por varios cientos o miles de aminoácidos configurados de forma secuencial, lo cual se conoce como estructura primaria (Xiong, 2006). Esta organización se va plegando espontáneamente hasta resultar en una conformación tridimensional diferente una de otra denominada como estructura terciaria, la cual es fundamental para determinar la función de la proteína y realizarla de forma exitosa (Xiong, 2006).

Hay muchas razones por las cuales la predicción de estructuras proteicas sigue siendo una problemática vigente. Una de ellas es que, actualmente, es mucho más complicado obtener estructuras tridimensionales que secuencias de proteínas, por lo cual existe una brecha cuantitativa entre ellas, que crece exponencialmente (Deng et al., 2018). Además, la determinación de las estructuras tridimensionales sigue siendo una tarea que requiere muchos recursos económicos, computacionales y algunos no renovables, como el tiempo (Lopes et al., 2019). En adición, se ha evidenciado una significativa ausencia de criterios de usabilidad en el desarrollo de muchas herramientas informáticas relacionadas a la predicción de las proteínas (Paixão-Cortes et al., 2018). Esto conlleva al gasto innecesario de tiempo y esfuerzo de los usuarios que deben interactuar con interfaces difíciles de entender (Bolchini et al., 2009).

Esta situación se replica en proteínas específicas como las proteínas repetidas, las cuales son grupos de familias de proteínas que tienen propiedades particulares como la existencia de unidades de repetición en su estructura (Hirsh et al., 2016). Estas proteínas son importantes dado que se sabe que se relacionan con muchas enfermedades humanas en su proceso de diagnóstico y porque dan pie al desarrollo de nueva medicina (Burley et al., 2021; Kajava & Steven, 2006). No obstante, debido a su complejidad, aún se requieren esfuerzos para estudiarlas en temas como la predicción de sus estructuras (MSCA & RISE, 2018).

Por todo ello, este proyecto de tesis busca proponer el desarrollo de una herramienta dedicada a la predicción de estructuras terciarias de proteínas repetidas a partir de sus estructuras primarias, la cual deberá cumplir con lineamientos de usabilidad. Se espera

responder a la problemática planteando una plataforma web que sea amigable para el usuario, que permita obtener resultados en tiempos aceptables y que utilice un algoritmo de predicción que aplique inteligencia artificial y sea eficaz respecto a la evaluación de alineamientos estructurales.

En primera instancia, se evaluarán distintos algoritmos de predicción de proteínas en general, para luego seleccionar uno y adaptarlo a los requerimientos de los especialistas en proteínas repetidas. Con ello, se crearán servicios y rutinas de ejecución que permitirán predecir estructuras terciarias de proteínas a partir de diversos tipos de datos de entrada. Posteriormente, se construirá la interfaz gráfica de la herramienta, partiendo de la definición de estándares y el desarrollo de un prototipo de alta fidelidad. Finalmente, se integrarán ambos componentes para conformar la herramienta completa, la cual será valorada a través de diversas pruebas funcionales y una evaluación de usabilidad. Cabe mencionar que esta última se realizará utilizando una herramienta enfocada a la evaluación de herramientas bioinformáticas.

FACULTAD DE CIENCIAS E INGENIERÍA

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

TEMA DE TESIS

PARA OPTAR : Título Profesional de Ingeniera Informática

TEMA : Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

ÁREA : Ciencias de la Computación e Ingeniería de Software

ASESOR : Dra. Layla Hirsh Martinez

ALUMNO : Solange Estrella Palomino Chahua - 20140104

FECHA : 08/02/2022

DESCRIPCIÓN Y OBJETIVOS:

La predicción de estructuras de proteínas es uno de los retos más importantes de la biología y la bioinformática (Lopes et al., 2019). Esta última es el campo de investigación que se apoya en la computación para analizar la información relacionada a las macromoléculas biológicas como las proteínas (Xiong, 2006). Las proteínas son moléculas esenciales compuestas por varios cientos o miles de aminoácidos configurados de forma secuencial, lo cual se conoce como estructura primaria (Xiong, 2006). Esta organización se va plegando espontáneamente hasta resultar en una conformación tridimensional diferente una de otra denominada como estructura terciaria, la cual es fundamental para determinar la función de la proteína y realizarla de forma exitosa (Xiong, 2006).

Hay muchas razones por las cuales la predicción de estructuras proteicas sigue siendo una problemática vigente. Una de ellas es que, actualmente, es mucho más complicado obtener estructuras tridimensionales que secuencias de proteínas, por lo cual existe una brecha cuantitativa entre ellas, que crece exponencialmente (Deng et al., 2018). Además, la determinación de las estructuras tridimensionales sigue siendo una tarea que requiere muchos recursos económicos, computacionales y algunos no renovables, como el tiempo (Lopes et al., 2019). En adición, se ha evidenciado una significativa ausencia de criterios de usabilidad en el desarrollo de muchas herramientas informáticas relacionadas a la predicción de las proteínas (Paixão-Cortes et al., 2018). Esto conlleva al gasto innecesario de tiempo y esfuerzo de los usuarios que deben interactuar con interfaces difíciles de entender (Bolchini et al., 2009).

Esta situación se replica en proteínas específicas como las proteínas repetidas, las cuales son grupos de familias de proteínas que tienen propiedades particulares como la existencia de unidades de repetición en su estructura (Hirsh et al., 2016). Estas proteínas son importantes dado que se sabe que se relacionan con muchas enfermedades humanas en su proceso de diagnóstico y porque dan pie al desarrollo de nueva medicina (Burley et al., 2021; Kajava & Steven, 2006). No obstante, debido a su complejidad, aún se requieren esfuerzos para estudiarlas en temas como la predicción de sus estructuras (MSCA & RISE, 2018).

Por todo ello, este proyecto de tesis busca proponer el desarrollo de una herramienta dedicada a la predicción de estructuras terciarias de proteínas repetidas a partir de sus estructuras primarias, la cual deberá cumplir con lineamientos de usabilidad. Se espera responder a la problemática planteando una plataforma web que sea amigable para el

usuario, que permita obtener resultados en tiempos aceptables y que utilice un algoritmo de predicción que aplique inteligencia artificial y sea eficaz respecto a la evaluación de alineamientos estructurales.

En primera instancia, se evaluarán distintos algoritmos de predicción de proteínas en general, para luego seleccionar uno y adaptarlo a los requerimientos de los especialistas en proteínas repetidas. Con ello, se crearán servicios y rutinas de ejecución que permitirán predecir estructuras terciarias de proteínas a partir de diversos tipos de datos de entrada. Posteriormente, se construirá la interfaz gráfica de la herramienta, partiendo de la definición de estándares y el desarrollo de un prototipo de alta fidelidad. Finalmente, se integrarán ambos componentes para conformar la herramienta completa, la cual será valorada a través de diversas pruebas funcionales y una evaluación de usabilidad. Cabe mencionar que esta última se realizará utilizando una herramienta enfocada a la evaluación de herramientas bioinformáticas.

Objetivo General y Objetivos Específicos:

En base a la problemática seleccionada, el objetivo general del presente proyecto de tesis es desarrollar una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria que cumpla con lineamientos de usabilidad.

En el mismo sentido, los objetivos específicos de esta investigación son los siguientes:

O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.

O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

IMPORTANTE

1. Usted debe adjuntar un archivo conteniendo el tema de tesis en formato PDF con el visto bueno (firma) de su asesor (o asesores).
2. Usted no debe contar con un Tema de tesis asignado anteriormente. De darse el caso, deberá efectuar el trámite de cambio del tema de tesis en la Facultad.
3. Usted debe encontrarse matriculado o haber aprobado el primer curso de Tesis de su especialidad.
4. En caso de que el tema de tesis mencione a una organización, deberá adjuntar la autorización del representante legal de dicha organización.
5. Se recomienda que la extensión del documento final de tesis, incluyendo los anexos, esté comprendida entre 75 y 150 páginas. Asimismo, el archivo del documento final de tesis no deberá exceder los 15 MB. Revisar el instructivo para la elaboración de documentos académicos https://drive.google.com/open?id=15XqAM1J4YDk4wi_EAgVUQEJbGfaZihUr

En caso de alguna consulta adicional, puede contactarnos a la cuenta: titulacion-fci@pucp.edu.pe

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERU
Departamento de Ingeniería
Dra. LAYLA HIRSH M.
Coordinadora de Especialidad
Ingeniería Informática

Tabla de Contenido

Índice de Figuras

Índice de Tablas

Capítulo 1.Generalidades

1.1 Problemática

En este capítulo se presentarán los problemas causa, los problemas efecto y el problema central que sirvieron de motivación y dieron pie al desarrollo del presente proyecto de tesis. Se desarrollará una contextualización de la problemática a través de la metodología del árbol de problemas, la cual se justificará en base a los estudios primarios obtenidos a través de la revisión sistemática del Capítulo 3 Estado del Arte.

1.1.1 Árbol de Problemas

A continuación, en la Figura 1, se muestra el árbol de problemas perteneciente al proyecto de investigación. En la sección superior de la imagen se plantean los tres problemas efecto; en la zona intermedia, el problema central; y en la zona inferior, los tres problemas causa identificados en relación al tema de tesis.

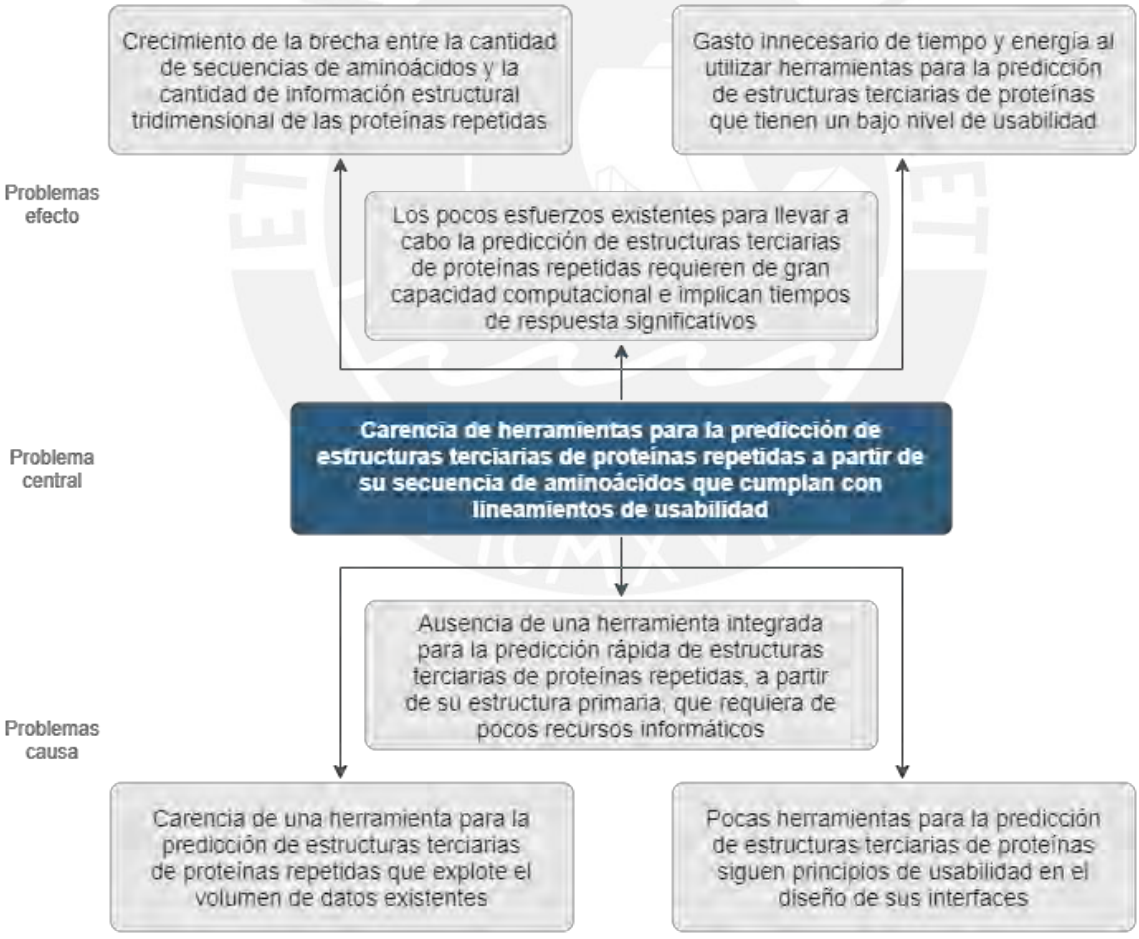

Figura 1. *Árbol de problemas: área superior - problemas efectos, área central - problema central, área inferior - problemas causas. (Elaboración propia).*

1.1.2 Descripción

Las proteínas son macromoléculas fundamentales que realizan las funciones químicas y biológicas más esenciales en una célula (Xiong, 2006). Un claro e impresionante ejemplo es la Hemoglobina (Hb), una proteína que se encuentra en los glóbulos rojos de nuestra sangre (Nelson & Cox, 2017). Numerosos estudios han determinado la existencia de un promedio de quinientos cincuenta y cinco gramos de moléculas de Hemoglobina en una mujer y ochocientos tres gramos en un hombre adulto (SJÖSTRAND, 1949). Así, en base a un cálculo simple y teniendo en cuenta que una molécula de Hemoglobina pesa aproximadamente sesenta y cuatro mil quinientos daltons[1] (Billett, 1990), una mujer tiene aproximadamente cinco mil trillones de moléculas de Hemoglobina en su sangre, responsables del transporte de oxígeno desde sus pulmones hasta su cerebro en cada respiración (Nelson & Cox, 2017). En el caso de un hombre, serán aproximadamente siete mil trillones de moléculas de Hemoglobina las que intervengan.

En base a lo anterior, resulta sorprendente imaginar que en una acción tan habitual como la respiración se encuentre involucrada tal vasta cantidad de macromoléculas desenvolviendo un rol tan esencial y a gran escala (Nelson & Cox, 2017). En este punto, cabe mencionar que cada uno de estos polipéptidos está constituido por una secuencia de aminoácidos o residuos, también conocida como estructura primaria (Nelson & Cox, 2017). Este primer nivel de organización es el más simple pero el más importante puesto que, al desplegarse espontáneamente, da pie a una única estructura 3D para cada una de las proteínas (Lopes et al., 2019). Esta estructura tridimensional define la función de la proteína y es por eso que conocerla resulta tan importante y, a veces, tan complejo (Deng et al., 2018; Gao et al., 2019). Gran parte de los datos conocidos actualmente con relación a las estructuras de las proteínas han sido obtenidos por medio de métodos experimentales (Lopes et al., 2019). No obstante, la significativa inversión financiera y temporal que implican conllevó a la necesidad de soluciones más tecnológicas (Lopes et al., 2019; Makigaki & Ishida, 2020).

La predicción de las estructuras terciarias de las proteínas consiste en determinar la posición relativa de los átomos de la proteína en un espacio tridimensional a partir de su secuencia de aminoácidos (Lopes et al., 2019). Esta predicción es de vital importancia, además de que es uno de los más grandes desafíos dentro de la

[1] La unidad de medida molecular Dalton equivale a 1,66 x 10^{-24} gramos.

bioinformática estructural, que aún queda por resolver (Deng et al., 2018; Gao et al., 2019; Lopes et al., 2019; Machado et al., 2018).

Esta problemática ha capturado el interés de muchos investigadores pertenecientes a distintas áreas (Lopes et al., 2019) y, desde que fue propuesta en 1960, se ha invertido mucho esfuerzo para poder resolverla (Gao et al., 2019). Los consorcios internacionales y multidisciplinarios como REFRACT[2] son un ejemplo de ello (Marie Sklodowska-Curie Actions [MSCA] & Research and Innovation Staff Exchange [RISE], 2018). Dicho proyecto está conformado por diversos especialistas que colaboran para estudiar, en particular, a las proteínas repetidas, TRP, por sus siglas en inglés (MSCA & RISE, 2018), un grupo de familias de proteínas muy extendidas en la naturaleza destacadas por estar constituidas por múltiples copias de una unidad estructural (Brunette et al., 2015).

Hoy por hoy, a diferencia de las más de doscientas catorce millones de secuencias de proteínas albergadas en UniProt (UniProt Consortium, 2021b), tan solo se cuenta con información estructural 3D de aproximadamente ciento ochenta mil de ellas (~1.2%), alojadas en el Banco de Datos de Proteínas (Burley et al., 2021). RepeatsDB es una base de datos que alberga información estructural de las unidades de repetición en proteínas repetidas (Paladin et al., 2021). En ese sentido y respecto a estas últimas, cabe mencionar que, hace unos años, en RepeatsDB se contaba con información estructural detallada de solo trescientas de estas macromoléculas (di Domenico et al., 2014), un número que hoy en día asciende a alrededor de ocho mil (Paladin et al., 2021). Estas conformaciones estructurales han sido obtenidas a partir de las bases de datos ya mencionadas, aplicando un método de predicción enfocado a la información y alineamiento estructural de las proteínas repetidas (Paladin et al., 2017). Este método se basó en la superposición de estructuras, un modo de alineamiento extremadamente costoso que, en su momento, representaba la única solución posible debido a la baja cantidad de datos con la que se contaba (Hirsh et al., 2016). Posteriormente, el consorcio REFRACT ha centralizado los esfuerzos para caracterizar la función y evolución de las proteínas repetidas (MSCA & RISE, 2018). Sin embargo, no han considerado la predicción de la estructura terciaria debido a la complejidad de la misma (MSCA & RISE, 2018).

Con todo lo anterior, se evidencia la necesidad de una herramienta que sea capaz de explotar la extensa cantidad de datos respecto a las secuencias de las proteínas, un

[2] Repeat protein Function, Refinement, Annotation and Classification of Topologies, REFRACT.

número que crece exponencialmente (Deng et al., 2018; Kuhlman & Bradley, 2019a; Nelson & Cox, 2017), para transformarlos en información estructural tridimensional que conducirá al entendimiento de la función de las proteínas.

Pero como se mencionó previamente, utilizar métodos experimentales para este fin implica un costo significativo en el plano financiero y temporal, lo cual conllevó a la necesidad de soluciones tecnológicas más innovadoras (Lopes et al., 2019; Makigaki & Ishida, 2020). Tal como se muestra en la revisión sistemática, dentro de este tipo de propuestas de solución tenemos el uso de inteligencia artificial, aprendizaje de máquina y aprendizaje profundo, cuyas aplicaciones eran impensables hasta hace unos pocos años debido al bajo volumen de datos (Hirsh et al., 2016), pero que hoy podemos aplicar gracias a esfuerzos como los de REFRACT.

Sin embargo, incluso con métodos computacionales como éstos, cabe la necesidad de asegurar la eficiencia y la rapidez de esas predicciones (Lopes et al., 2019). De acuerdo con la revisión sistemática, la situación se torna aún más complicada si nos enfocamos en la búsqueda de herramientas que requieran de pocos recursos informáticos; en realidad, muchas veces la capacidad necesaria supera la habilidad tecnológica existente (Deng et al., 2018). Esto último conduce a que los pocos esfuerzos desarrollados actualmente requieran de gran esfuerzo computacional (Lopes et al., 2019). Es así que, en septiembre del 2020, se propone la investigación de AlphaFold2, un proyecto que utiliza aprendizaje profundo para producir estructuras de proteínas, en base a la información autogenerada de sus ángulos de torsión, las distancias entre sus residuos, las coordenadas de sus átomos, entre otros (Protein Structure Prediction Center, 2020). Este método devuelve resultados después de cierta cantidad de días (DeepMind, 2021). Con ello se demuestra que arquitecturas basadas en inteligencia artificial son factibles, aunque no siempre eficientes, para lograr la predicción requerida. No obstante, algo importante de mencionar es que este método se aplica a proteínas no repetidas. En el caso particular de las proteínas TRP, y debido a su divergencia en la secuencia (Hirsh et al., 2016; Parmeggiani & Huang, 2017), este tipo de métodos resultan poco precisos.

Por otro lado, se ha identificado que la usabilidad no parece ser un factor crucial en los servidores de predicción de estructuras de proteínas, por lo que pocas de estas herramientas siguen sus principios en el diseño de sus interfaces (Paixão-Cortes et al., 2018). Esta ausencia de la aplicación de lineamientos de usabilidad implica un desarrollo de interfaces difíciles de usar significando, así, un gasto innecesario de tiempo y esfuerzo por parte de sus usuarios (Bolchini et al., 2009; Paixão-Cortes et al., 2018).

Por consiguiente a lo explicado, cabe reconocer que aún prevalece la necesidad de una herramienta dedicada a la predicción de la configuración espacial de las proteínas repetidas a partir de su secuencia de residuos (Lopes et al., 2019). Esto último teniendo en cuenta que, tal como lo demuestra REFRACT, conocer a detalle la relación entre la composición y las propiedades estructurales de las proteínas repetidas significa el conocimiento de la función de las mismas, lo cual es de crucial importancia para las investigaciones y demás aplicaciones en torno a la medicina y biotecnología (MSCA & RISE, 2018), considerando que este tipo particular de proteínas se relaciona con muchas enfermedades humanas (Kajava & Steven, 2006).

1.1.3 Problema seleccionado

Conforme al árbol de problemas y a lo descrito en la sección previa se ha identificado que el problema central del presente proyecto de tesis es la carencia de herramientas dedicadas a la predicción de estructuras terciarias[3] de proteínas repetidas a partir de su secuencia de aminoácidos que cumplan con lineamientos de usabilidad.

1.2 Objetivos

En esta sección se presenta el objetivo general de la investigación, además de los objetivos específicos y los resultados esperados de los mismos.

1.2.1 Objetivo general

En base al problema seleccionado, el objetivo general del presente proyecto de tesis es desarrollar una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria que cumpla con lineamientos de usabilidad.

1.2.2 Objetivos específicos

A continuación, se presentan los objetivos específicos del proyecto de tesis:

- O 1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.
- O 2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

[3] La definición de los términos con relación a la estructura de las proteínas se encuentra especificada en el Capítulo 2. Marco Conceptual.

O 3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

1.2.3 Resultados esperados

O 1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.

R 1. Lista de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general.

R 2. Algoritmo seleccionado y planteamiento de las modificaciones necesarias al mismo.

R 3. Implementación del algoritmo adaptado.

O 2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

R 1. Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta.

R 2. Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

R 3. Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas.

O 3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

R 1. Interfaz que permite utilizar el algoritmo modificado.

R 2. Pruebas funcionales de la herramienta de predicción con el algoritmo integrado.

R 3. Evaluación de usabilidad de la herramienta implementada.

1.2.4 Mapeo de objetivos, resultados y verificación

Los objetivos específicos, sus resultados y sus medios de verificación, así como los indicadores objetivamente verificables por cada uno de ellos, se presentan en la Tabla 1, Tabla 2 y Tabla 3. Estas últimas corresponden a los objetivos específicos O1, O2 y O3, respectivamente.

Objetivo: O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R1. Lista de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general	• Reporte que contiene el detalle de los algoritmos identificados	El reporte debe contener al menos dos algoritmos capaces de explotar datos de estructuras terciarias de proteínas a partir de sus estructuras primarias, todos con su respectivo detalle referente a sus datos de entrada y de salida, modo de procesamiento, recurso computacional utilizado y tiempo requerido.
R2. Algoritmo seleccionado y planteamiento de las modificaciones necesarias al mismo	• Documento que contenga la comparación detallada de los algoritmos identificados	El documento debe contener una comparación del tiempo de ejecución y cantidad de recursos utilizados de los algoritmos identificados
	• Documento que justifique la elección y describa las modificaciones propuestas al algoritmo seleccionado	El entregable debe incluir el análisis que justifique la elección del algoritmo y el detalle de al menos dos modificaciones, además de la mejora esperada respecto a sus datos de entrada y de salida, modo de procesamiento, recurso computacional utilizado y/o tiempo requerido
R3. Implementación del algoritmo adaptado	• Documento que contiene el diagrama del algoritmo adaptado con su respectiva descripción funcional y código fuente correspondiente	El documento debe abarcar la descripción del algoritmo adaptado e información de la evaluación realizada detallando si las modificaciones ayudan a explotar los datos existentes de las estructuras primarias para predecir estructuras terciarias. El 100% de las funciones del código fuente estarán documentadas.

	• Reporte con los resultados de las pruebas del algoritmo adaptado	Documento que contiene la descripción del 100% de las pruebas realizadas y documento detallando el cumplimiento del 100% de ellas. Documento de aceptación del algoritmo por parte de un experto basado en los resultados de las pruebas anteriores.

Tabla 1. *Mapeo de O1. Modificar un algoritmo capaz de explotar los datos existentes de estructuras primarias para predecir estructuras terciarias de las proteínas repetidas.*

Objetivo: O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R1. Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta.	• Documento con la especificación de los requisitos funcionales y no funcionales de la herramienta	El documento debe incluir la especificación de al menos diez requisitos funcionales y no funcionales de la herramienta como el lenguaje de programación del desarrollo, tipos de mensajes de error informativos, capacidad computacional requerida, entre otros.
		Aprobación del 100% de los requisitos funcionales y no funcionales de la herramienta por parte de un experto en bioinformática.
R2. Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	• Reporte con el mockup de interfaz de la herramienta propuesta incluyendo pantallas y navegación	El reporte debe contener el diagrama de flujo que describen los pasos a seguir para llevar a cabo la predicción de la estructura terciaria a partir de la secuencia de aminoácidos de una proteína repetida. Este reporte debe ser aprobado por juicio experto.
R3. Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas	• Reporte descriptivo de la interfaz desarrollada • Video de la navegación de la interfaz	El reporte debe asegurar que la interfaz lleve a cabo satisfactoriamente el flujo completo especificado en el reporte planteado en el resultado esperado R2, además de incluir los requisitos funcionales y no funcionales del resultado esperado R1.

Tabla 2. *Mapeo de O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su secuencia de aminoácidos.*

Objetivo: O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R1. Interfaz que permite utilizar el algoritmo adaptado	• Reporte de funcionamiento de la herramienta implementada	El reporte debe demostrar que los resultados del algoritmo adaptado integrado con la interfaz retornan los mismos resultados obtenidos en las pruebas del objetivo específico O1. El 100% de los requisitos funcionales y no funcionales del resultado esperado R2 del O2 objetivo deben estar implementados.
	• Herramienta implementada	
R2. Pruebas funcionales de la herramienta de predicción con el algoritmo integrado	• Documento de especificación y resultado de pruebas	El documento debe contener información detallada de la ejecución de pruebas de proteínas repetidas y la comparación de los resultados obtenidos con las estructuras tridimensionales conocidas de los datos de entrada
	• Video de la herramienta en ejecución	Se deben cumplir el 100% de las pruebas funcionales
		Aprobación del documento de especificación y resultado de pruebas por parte de un experto en bioinformática.
R3. Evaluación de usabilidad de la herramienta implementada	• Reporte de la evaluación de usabilidad de la herramienta implementada basada en los lineamientos de usabilidad para herramientas bioinformáticas	El reporte debe comprender el detalle de las pruebas y resultados de la evaluación de usabilidad realizada. Estos resultados deben ser satisfactorios para la herramienta. Aceptación de la prueba realizada por juicio experto en usabilidad.

Tabla 3. *Mapeo de O3. Integrar el algoritmo y la interfaz desarrollados para crear la herramienta para realizar la predicción de estructuras terciarias de proteínas repetidas a partir de su secuencia de aminoácidos.*

1.3 Métodos y Procedimientos

Los herramientas, métodos y procedimientos[4] a utilizar en los resultados pertenecientes a los objetivos específicos O1, O2 y O3 serán presentados en la Tabla 4, Tabla 5 y Tabla 6, respectivamente.

Objetivo: O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas	
Resultado	**Herramientas, métodos y procedimientos**
R1. Lista de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general	• Algoritmos de predicción de proteínas: Con apoyo de la revisión sistemática realizada se obtendrán algoritmos para posteriormente adaptarlos al problema.
R2. Algoritmo seleccionado y planteamiento de las modificaciones necesarias al mismo	• GitHub: Se manejará un repositorio que aloje a los algoritmos en evaluación. • Jupyter Notebook: Aplicación para la programación de los algoritmos. • Python: Lenguaje de programación • Pytorch: Librería en versión 1.9.0 • Numpy: Librería en versión 1.21.2
R3. Implementación del algoritmo adaptado	• LucidChart: Aplicación para construir los diagramas que expliquen el funcionamiento del algoritmo. • Jupyter Notebook: Aplicación para la programación de los algoritmos. • Python: Lenguaje de programación • Pytorch: Librería en versión 1.9.0 • Numpy: Librería en versión 1.21.2 • Protein Data Bank: Base de datos de estructuras de proteínas • RepeatsDB: Base de datos de estructuras de proteínas repetidas

Tabla 4. *Herramientas, métodos y procedimientos de los resultados esperados del objetivo O1.*

[4] Se ha evitado mencionar al editor de texto puesto que es una herramienta que se usará de forma frecuente en la mayoría de los resultados esperados de todos los objetivos específicos planteados.

Objetivo: O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	
Resultado	**Herramientas, métodos y procedimientos**
R1. Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta.	No aplica.
R2. Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	• Figma: Herramienta de prototipado • LucidChart: Aplicación para construir el diagrama de flujo de la interfaz. • Zoom: La interacción del experto se organizará y se guiará a través de reuniones online. • Noun Project: Proyecto web para la obtención de iconos a utilizar.
R3. Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas	• Visual Studio Code: Editor de código fuente. • JavaScript: Lenguaje de programación para la lógica funcional de la interfaz. • HTML: Lenguaje de marcas de hipertexto para la programación de las vistas de la interfaz. • CSS: Lenguaje para la definición de estilos de la interfaz. • Noun Project: Proyecto web para la obtención de iconos a utilizar. • Vue.js: Framework para la construcción de la interfaz. • GitHub: Se manejará un repositorio que aloje al proyecto de la interfaz desarrollada. • Xbox Game Bar: Herramienta para videograbar la navegación de la interfaz.

Tabla 5. *Herramientas, métodos y procedimientos de los resultados esperados del objetivo O2.*

Objetivo: O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	
Resultado	**Herramientas, métodos y procedimientos**
R1. Interfaz que permite utilizar el algoritmo adaptado	• GitHub: Se manejará un repositorio que aloje al proyecto completo. • Visual Studio Code: Editor de código fuente. • Python: Lenguaje de programación para el algoritmo. • JavaScript: Lenguaje de programación para la lógica de la interfaz. • Flask: Framework de Python para integrar el algoritmo con la interfaz. • Nginx: Servidor web para el despliegue de la herramienta desarrollada. • Amazon Web Services: Se utilizarán algunos servicios para poder instanciar y potenciar la herramienta en la nube.
R2. Pruebas funcionales de la herramienta de predicción con el algoritmo integrado	• Protein Data Bank: Base de datos de estructuras de proteínas. • RepeatsDB: Base de datos de estructuras de proteínas repetidas. • Xbox Game Bar: Herramienta para videograbar el funcionamiento de la herramienta.
R3. Evaluación de usabilidad de la herramienta implementada	• Herramienta de evaluación de usabilidad: Se considera conveniente usar una herramienta innovadora y reciente que se adapte a la herramienta desarrollada y a su contexto.

Tabla 6. *Herramientas, métodos y procedimientos de los resultados esperados del objetivo O3.*

Capítulo 2.Marco Conceptual

2.1 Introducción

El marco teórico se define como el eje integrador de interpretaciones hipotéticas que pueden ser tanto verdaderas como falsas (Daros, 2002). Este da pie a la especificación de los supuestos en los que un investigador se basará para justificar y realizar su estudio (Daros, 2002). A diferencia de esto, el marco conceptual es un bloque de soporte a la investigación, que está constituido por conceptos básicos que han sido seleccionados y luego organizados para dar sentido a la temática de un proyecto (Daros, 2002).

En el presente capítulo se llevará a cabo el desarrollo del marco conceptual que plantea la definición de términos identificados que se consideran fundamentales para poder entender este proyecto. En ese sentido, su objetivo principal es brindar nociones básicas de los conceptos principales a tener en cuenta en la revisión de esta investigación.

2.2 Desarrollo del marco

Se considera de gran importancia tener en cuenta los siguientes conceptos para poder entender a profundidad tanto la problemática como el planteamiento de la solución de la misma en el presente proyecto. Con el objetivo de ayudar a entender las definiciones de los ítems presentados a continuación se procurará ejemplificarlos a través de ilustraciones.

C1. Aminoácido

La definición de un aminoácido se determina por su conformación, ya que es una pequeña molécula constituida por un grupo amino (NH_2) y un grupo carboxilo (COOH) (Xiong, 2006). Estas dos agrupaciones se unen con ayuda de un átomo central de carbono (C$\boldsymbol{\alpha}$), al cual se adhieren también un hidrógeno y una cadena lateral R como se muestra en la Figura 2 . (Xiong, 2006). Lo que diferencia a un aminoácido del otro es el grupo R, el cual puede ser reemplazado por una de las veinte cadenas laterales existentes, formando así uno de los veinte aminoácidos estándares de la biología y bioquímica (Xiong, 2006), tales como la alanina (Ala), la valina (Val) y la leucina (Leu).

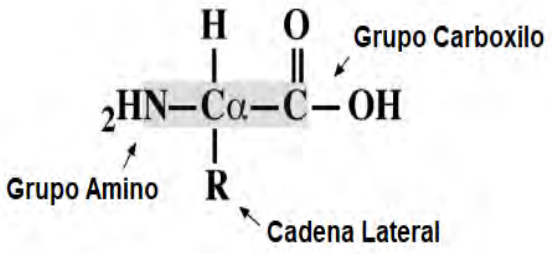

Figura 2. *Estructura general de un aminoácido, la Prolina es un aminoácido excepción con una estructura cíclica. El grupo R puede ser reemplazado por una de las veinte cadenas laterales existentes. Adaptado de* (Xiong, 2006).

C2. Proteína

La ligación de dos aminoácidos a través de un enlace peptídico da lugar a una molécula más grande denominada dipéptido (Xiong, 2006). Análogamente, al incorporar más de cincuenta aminoácidos a una cadena se obtiene un polipéptido: una macromolécula también conocida como proteína (Xiong, 2006). La variabilidad que puedan tener esas secuencias recae en el orden en el que están dispuestos los aminoácidos, con el cual, además, se puede explicar la vasta cantidad de proteínas conocidas en la actualidad, entre los que se encuentran la hemoglobina, la insulina, las inmunoglobulinas y el colágeno, tal como se muestra en la Figura 3. Cabe destacar que las proteínas desempeñan un rol muy importante dentro de las células, siendo parte de procesos esenciales como los estructurales, los enzimáticos, los de transporte y los reguladores (Xiong, 2006).

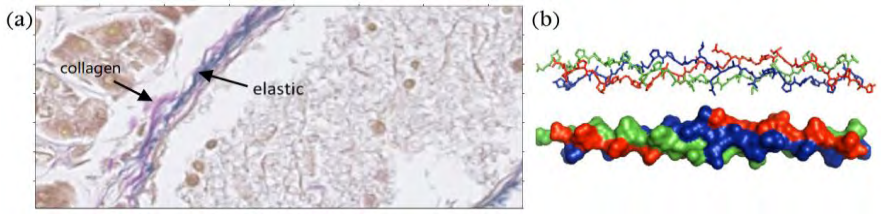

Figura 3. *(a) Fibras elásticas y de colágeno identificadas a través de imágenes hiperespectrales con el método de tinción EVG. Adaptado de* (Septiana et al., 2019). *(b) Estructura de un modelo de colágeno corto en barras y presentaciones de superficie; PDB 1BKV. Adaptado de* (Guo et al., 2006).

C3. Proteína repetida

Un grupo más específico y extendido dentro de la naturaleza y el mundo de la proteómica, una subárea clave de la bioinformática, es el que conforman las proteínas repetidas[5] (Parmeggiani & Huang, 2017). Este grupo de familias de macromoléculas destacan por estar conformadas por una serie de estructuras básicas que se repiten (Parmeggiani & Huang, 2017). En la Figura 4 podemos observar el diseño de una proteína repetida constituida por diez unidades básicas, agrupadas de tal manera que se consigue una forma toroidal. Cabe mencionar que una característica adicional de este tipo de proteínas es su alta tasa de degeneración, con lo cual dos fragmentos con un plegamiento estructural similar pueden tener secuencias de aminoácidos no necesariamente iguales (Deng et al., 2018; Hirsh et al., 2016; Parmeggiani & Huang,

[5] También conocidas como proteínas repetidas en tándem, TRP, por sus siglas en inglés (MSCA & RISE, 2018).

2017), esto quiere decir que la estructura tridimensional se mantiene más ante los efectos de la evolución que la propia secuencia.

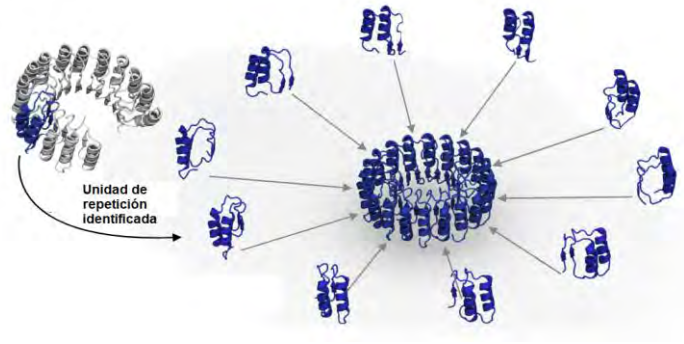

Figura 4. *Estructura de una proteína repetida obtenida a través del diseño basado en plantillas. Se identifica una unidad básica que se repite 10 veces, secuencias ricas en leucina del inhibidor de ribonucleasa porcina; PDB: 2BNH. Adaptado de (Parmeggiani & Huang, 2017).*

C4. Estructuras proteicas

Las proteínas se pueden dimensionar dentro de una jerarquía de cuatro niveles: estructura primaria, secundaria, terciaria y cuaternaria como se observa en la Figura 5 (Xiong, 2006). La primera categoría hace referencia a una secuencia de aminoácidos en forma lineal tipo cadena; mientras que la segunda se denomina como un arreglo local y regular, donde los diversos residuos (o grupos R) de los aminoácidos se estabilizan por enlaces de hidrógeno entre los átomos de los grupos CO y NH ubicados de forma no adyacente en la cadena principal (Xiong, 2006). Hay tres tipos principales de estructuras secundarias: las hélices α, las láminas plegadas β y los bucles (Lopes et al., 2019). En el mismo sentido, la tercera y la cuarta categoría están relacionadas a estructuras tridimensionales más complejas, incluyendo, para esta última, a la asociación de varias cadenas de polipéptidos para construir una proteína más elaborada (Xiong, 2006). Cabe mencionar que en la presente investigación se ahondará más en las estructuras primarias y terciarias para llevar a cabo los objetivos de la misma.

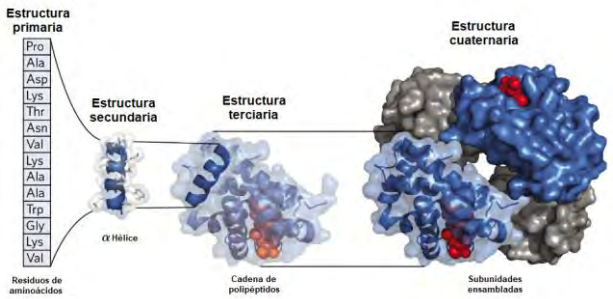

Figura 5. *Niveles de estructura de las proteínas: primaria, secundaria, terciaria y cuaternaria. En este caso de la Hemoglobina; PDB 1HGA. Adaptado de (Nelson & Cox, 2017).*

C5. Estructura primaria de proteínas

Una manera de concebir la idea de la estructura primaria de una proteína es imaginar una cadena de dos, tres o miles de aminoácidos conectados unos a otros (Nelson & Cox, 2017). En esta estructura recae una noción básica que nos permite conocer qué elementos (a nivel de aminoácidos) conforman una molécula mucho más compleja, pero sin la cual no se lograría entenderla (Nelson & Cox, 2017).

De manera análoga a la que, según la morfología, podemos descomponer una palabra en letras, también se puede desglosar a una proteína en sus aminoácidos. Todos los aminoácidos tienen una abreviación y una letra que los representa, conformando así un alfabeto (Xiong, 2006). Por tanto, siguiendo la organización dispuesta en la Tabla 7, encontrar una 'A' en una secuencia se interpretará como la presencia del aminoácido Alanina; una 'T' se traduce a una Treonina; y una 'R', a una Arginina.

Catálogo de aminoácidos			
Aminoácido	**Aminoácido (En)**	**Abreviación**	**Símbolo**
Glicina	Glycine	Gly	G
Alanina	Alanine	Ala	A
Prolina	Proline	Pro	P
Valina	Valine	Val	V
Leucina	Leucine	Leu	L
Isoleucina	Isoleucine	Ile	I
Metionina	Methionine	Met	M
Fenilalanina	Phenylalanine	Phe	F
Tirosina	Tyrosine	Tyr	Y
Triptófano	Tryptophan	Trp	W
Serina	Serine	Ser	S
Treonina	Threonine	Thr	T
Cisteína	Cysteine	Cys	C
Asparagina	Asparagine	Asn	N
Glutamina	Glutamine	Gln	Q
Lisina	Lysine	Lys	K

Histidina	Histidine	His	H
Arginina	Arginine	Arg	R
Aspartato	Aspartate	Asp	D
Glutamato	Glutamate	Glu	E

Tabla 7. *Catálogo de aminoácidos, traducción en inglés, abreviatura y simbología. Adaptado de (Nelson & Cox, 2017).*

C6. Estructura terciaria de proteínas

El nivel de complejidad de las proteínas se incrementa cuando se analizan respecto a su estructura terciaria, tal como puede observarse en la Figura 5. Esto debido a que toman relevancia otros aspectos como las interacciones electromagnéticas entre sus residuos, los ángulos de torsión y el medio (Xiong, 2006).

La estructura terciaria está constituida por varias unidades de estructuras secundarias, tales como hélices α (Figura 5), las láminas plegadas β o los bucles (Lopes et al., 2019; Nelson & Cox, 2017). Ciertamente, este nivel estructural describe todos los aspectos del plegamiento tridimensional de un polipéptido; no obstante, es indudable la dependencia entre esta organización 3D y su secuencia de aminoácidos, es decir, de nuevo, la estructura primaria (Nelson & Cox, 2017). De esa manera, los pequeños aminoácidos que la conforman determinan, también, su función (Nelson & Cox, 2017). En ese sentido, si una proteína tiene una única cadena de aminoácidos que determina una estructura terciaria única, esta estructura también le atribuye una función única (Nelson & Cox, 2017).

C7. Bases de datos de estructuras de proteínas

El desarrollo de bases de datos que permitan manejar una vasta cantidad de datos biológicos moleculares es una tarea imprescindible de la bioinformática (Xiong, 2006). Dependiendo de su contenido, estos archivos computarizados pueden ser de tres tipos: bases de datos primarias, secundarias y especializadas (Xiong, 2006). Las bases de datos primarias almacenan información de secuencias o datos estructurales no procesados (Xiong, 2006). Las bases de datos secundarias contienen conocimiento biológico más sofisticado, que previamente se alojaban en bases de datos del primer tipo; estos recursos pueden haber sido procesados computacional o manualmente (Xiong, 2006). Por último, las bases de datos especializadas se concentran en campos

de investigación específicos, incluso se pueden enfocar en algunos organismos en particular (Xiong, 2006).

Cabe resaltar que el contenido del último tipo de repositorio mencionado es diverso, ya que puede incorporar tanto datos de secuencias como información específica del elemento de investigación. Así, para efectos del presente proyecto con enfoque en proteínas repetidas se utilizarán ese tipo de bases de datos, algunos de los cuales son: RepeatsDB, una iniciativa del consorcio internacional REFRACT[6] (Paladin et al., 2021) y UniprotKB, perteneciente a UniProt Consortium (UniProt Consortium, 2021a).

C8. Usabilidad

El concepto de usabilidad no tiene una definición completamente aceptada, a pesar de haber ido ganando mayor protagonismo y de que la cantidad de esfuerzos para tenerlo en cuenta en el desarrollo de software tenga una tendencia creciente (Bevan et al., 1991).

La usabilidad se define como el conjunto de atributos de software que influyen en el esfuerzo requerido para usarlo y en su evaluación en general (International Organization for Standarization [ISO], 1991, como se citó en Bevan et al., 1991). No obstante, esta especificación depende también del enfoque desde el cual se realice: la vista orientada en el producto determina la medida de la usabilidad con relación a los atributos ergonómicos del componente; y la vista orientada al usuario la engloba en torno al esfuerzo mental y la actitud de la persona (Bevan et al., 1991). En el mismo sentido, la vista del rendimiento del usuario tiene una perspectiva que se basa en la relación entre el usuario y el producto, dándole mayor énfasis a la facilidad de uso y a la aceptabilidad (Bevan et al., 1991).

C9. Métodos computacionales

Actualmente existen distintos métodos computacionales cada uno con un nivel de complejidad más definido, tales como la inteligencia artificial, machine learning y deep learning (SESAR Joint Undertaking, 2021). Cada técnica surge como una especialidad de la otra, dando paso al crecimiento de la disciplina que los engloba: las ciencias de la computación (SESAR Joint Undertaking, 2021). Esta es un área de investigación que ha logrado mayor acogida a lo largo de los años, la cual se concentra en el estudio de algoritmos con un sólido sustento en las matemáticas y la programación (ACM et al.,

[6] Repeat protein Function, Refinement, Annotation and Classification of Topologies, REFRACT.

2005). Así, a través del uso de tecnología como redes neuronales y la adición de capas internas, logran obtener patrones dentro de grandes volúmenes de datos (Kuhlman & Bradley, 2019b). Su popularidad aumenta con los años, incluso mientras los problemas se vuelven más y más complejos, puesto que el transcurrir del tiempo da paso al incremento de datos, con lo cual este tipo de algoritmos mejoran su performance (Zhao & Gong, 2019).

C10. Algoritmo de predicción de estructuras de proteínas

En general, un algoritmo se define como una secuencia de pasos finitos que describen cómo llevar a cabo un proceso con el fin de encontrar la solución a un problema (Lambert & Osborne, 2018). Actualmente, los algoritmos de predicción de estructuras de proteínas se desarrollan en torno a dos tendencias (Xiong, 2006). Por un lado, los métodos con un enfoque libre de plantillas basan su predicción en la propia secuencia de aminoácidos, además de algunas otras propiedades del mismo polipéptido (Deng et al., 2018). Los algoritmos basados en redes neuronales artificiales son las tecnologías más comunes dentro de este campo de investigación (Xiong, 2006). Por otro lado, los métodos con enfoque basado en plantillas concentran su esfuerzo principalmente en la búsqueda de otras proteínas, que cumplan cierto grado de homología, con las cuales se pueda tener un punto de partida en la predicción (Deng et al., 2018).

Capítulo 3.Estado del Arte

A continuación, se desarrollan los acápites relacionados al Estado del Arte.

3.1 Introducción

Para que un proyecto se convierta en un recurso con credibilidad respecto a lo que propone y concluye es importante, mas no definitivo, tener en consideración y apoyarse, en algunos proyectos que diversos investigadores reconocidos han realizado respecto al tema planteado. Esto, de alguna manera, se traduce en un respaldo para el trabajo que se está llevando a cabo y es un sustento para los resultados de este.

Debido a ello y acorde a los objetivos del estudio, en el presente capítulo se desarrollará una revisión sistemática cualitativa de documentos relacionados a los tópicos involucrados en el proyecto, es decir, al uso de algoritmos para la predicción de estructuras terciarias de proteínas repetidas y al desarrollo de software teniendo en cuenta sus requerimientos, además de las diversas metodologías de usabilidad

disponibles en la actualidad. Se ha considerado la aplicación de este método con el fin de hacerlo transparente y replicable (Olivares, 2016).

3.2 Objetivos de revisión

Nos encontramos en un contexto en el cual, como lo afirma el microbiólogo Mathias Uhlén, “es más fácil generar datos que obtener conocimientos de ellos” (2015, como se citó en Savage, 2015). Esta realidad no es ajena a la bioinformática, en específico a la proteómica, donde se ha logrado obtener información necesaria de más de ciento setenta mil estructuras de proteínas, de las más de doscientas millones existentes en la naturaleza (Pérez, 2020).

Sin embargo, a nuestro conocimiento y luego de una revisión cualitativa, aún hace falta una herramienta que sea capaz de explotar esa data conocida, es decir, las estructuras primarias de las proteínas, con el fin de lograr un conocimiento más fructífero, como lo es su estructura tridimensional, y, por ende, su función. Así, “los intentos de los últimos 50 años no han tenido éxito debido a la complejidad del problema, aunque ha habido algunos avances” (Jordan, 2021).

Teniendo en cuenta lo expuesto, el objetivo de este capítulo es efectuar una revisión sistemática cualitativa de investigaciones previas que se hayan realizado en torno al uso de algoritmos para la predicción de estructuras terciarias de las proteínas, en particular de las proteínas repetidas. Asimismo, se llevará a cabo una búsqueda de material relacionado a las herramientas propuestas recientemente con respecto a la predicción de la conformación espacial de las proteínas para identificar las características funcionales y no funcionales que engloban las necesidades generales. Además, se busca conocer las características de los métodos utilizados para evaluar ese tipo de proyectos en torno a la usabilidad.

3.3 Preguntas de revisión

Para poder cumplir con los objetivos de revisión establecidos es necesario plantearnos las siguientes preguntas:

- P1- ¿Cuáles son los algoritmos capaces de explotar datos de estructura primaria para predecir estructuras terciarias de proteínas, cuáles son sus requisitos y cómo funcionan?[7]
- P2- ¿Cuáles son los requisitos necesarios para realizar el diseño de una interfaz que permita explotar datos de estructura primaria para predecir estructura terciaria y cuáles son sus especificaciones?
- P3- ¿Qué técnicas y métodos existen para la evaluación de usabilidad en herramientas bioinformáticas y cómo se llevan a cabo?

3.4 Estrategia de búsqueda

A continuación, se describe la estrategia de búsqueda aplicada para la obtención de recursos relacionados a los objetivos de la revisión.

3.4.1 Motores de búsqueda a usar

Los motores de búsqueda son herramientas web que localizan información y recursos deseados existentes en Internet de forma rápida y eficiente (Vaquero, 1997). Estos servicios son importantes para el desarrollo de un proyecto de investigación ya que, en un ciberespacio inmenso y desorganizado, se requiere emplear mucho tiempo para obtener resultados relevantes (Oller, 2003). Así, un 85% de los usuarios de la red los utilizan para obtener una fracción de la información disponible y relacionada a las palabras clave ingresadas y cotejadas con el índice del propio buscador (Oller, 2003).

Acorde a ello, en esta sección se presentarán dos motores de búsqueda que se utilizarán para cumplir los objetivos de la investigación documental.

SCOPUS	Es una herramienta para la búsqueda de información, conformada principalmente por una base de datos de citas y resúmenes tratada por expertos (Elsevier, 2021). Tal como se menciona en su página web, Scopus es usada por más de 5 mil instituciones pertenecientes a diferentes ámbitos como el académico, corporativo y gubernamental (Elsevier, 2021).

[7] Considerar que la solución tendrá un enfoque en torno a las proteínas repetidas. No obstante, se requiere de la obtención de información relacionada a todo tipo de proteínas puesto que será de aporte crucial para el propósito del presente proyecto.

	En este motor de búsqueda se pueden obtener datos confiables y relevantes de forma rápida (Elsevier, 2021). Además, se puede contar con diversas métricas a disposición del usuario para identificar expertos en diversas materias y acceder a material bibliográfico de forma fiable (Elsevier, 2021). En ese sentido, posee una vasta variedad de artículos de revistas y permite el acceso a las referencias incluidas en ellos (Elsevier, 2021).
ACM Digital Library	La biblioteca digital ACM es reconocida como una de las colecciones más completas respecto a las áreas de Computación, Informática y Electrónica (ACM, 2021). Esto se debe a que fue desarrollada por la Asociación para Maquinaria de Computación, ACM, por sus siglas en inglés (ACM, 2021). En ese sentido, es una plataforma de investigación, descubrimiento y networking (ACM, 2021). Este motor de búsqueda pone a disposición de sus usuarios una base de datos bibliográfica integral centrada exclusivamente en el campo de la informática (ACM, 2021). De esta manera, el investigador puede encontrar todas las publicaciones de la asociación, además de actas de congresos, libros y artículos de revistas de autores seleccionados, entre otros (ACM, 2021).

Teniendo en cuenta la descripción, los recursos disponibles y el reconocimiento de fiabilidad de los motores de búsqueda descritos, se decide su elección para llevar a cabo la obtención de investigaciones relacionadas al presente proyecto.

3.4.2 Cadenas de búsqueda a usar

En esta sección se presentarán las cadenas de búsqueda usadas para poder obtener investigaciones acordes a los objetivos de la revisión. Estas consultas se realizarán en el idioma inglés debido a que así se podrá extender el alcance de las mismas.

Previo a ello, es necesario identificar algunas subcadenas para poder definir con claridad el enfoque de nuestra búsqueda documentaria y, en consecuencia, poder maximizar la eficiencia en la obtención de recursos de interés y utilidad.

En ese sentido, se definieron las siguientes subcadenas, en sintaxis de Scopus:

SC1. ("protein fold" OR "protein structure prediction") AND sequence AND protein AND (aminoacid OR "amino acid") AND (3d OR tertiary)	Son las palabras clave que tienen mayor relevancia y relación con la investigación. Se podrían utilizar para todas las preguntas de revisión, sin embargo, el uso de todas en la misma consulta podría afectar negativamente a la búsqueda.
SC2. (alphafold AND AUTHOR-NAME (senior))	Es una subcadena que nos va a permitir incluir uno de los estudios más recientes e importantes en el mundo de la proteómica respecto a la predicción de estructuras terciarias.
SC3. NOT secondary AND NOT "wet lab" AND NOT experimental AND NOT coronavirus AND NOT "cov-2" AND NOT covid AND NOT (RNA OR DNA)	Son sentencias que nos permitirán centrar nuestro foco de atención fuera de los temas de coyuntura, el nivel estructural o moléculas que no se relacionan con el objetivo del proyecto.
SC4. interface AND (application OR web OR serv* OR tool OR source)	Términos y sinónimos principales relacionados a la segunda pregunta de revisión respecto al software. Se utiliza el símbolo comodín (*) para referirse tanto a “services” como a “servers”.
SC5. usability OR user OR intuitive OR easy	Palabras relevantes relacionadas a la tercera pregunta de revisión con referencia a las metodologías de usabilidad disponibles en la actualidad.
SC6. SUBJAREA “ENGI” OR “MATH” OR “BIOC” OR “COMP” OR “MULT”	Estas son las expresiones que determinan la relación entre los recursos encontrados y su área de investigación.

SC7. PUBYEAR 2021 OR 2020 OR 2019 OR 2018 OR 2017	Son términos que ayudarán a limitar los resultados respecto al año de publicación.
SC8. DOCTYPE (ar)	Es una subcadena alternativa que nos ayudará a depurar los documentos respecto al tipo de artículo.

Se han logrado determinar ocho subcadenas que serán útiles para realizar la búsqueda de información. Luego de identificarlas, se procedió a formular cadenas de búsqueda más complejas que las incluyan, obteniendo entonces, tres sentencias globales por cada uno de los motores de búsqueda.

Es necesario tener en cuenta que el proyecto tiene tres enfoques distintos, por lo mismo que se han planteado tres preguntas de revisión; con lo cual, se destinará una cadena de búsqueda a cada una de ellas.

En la Tabla 8 y en la Tabla 9 se especifican las cadenas de búsqueda mencionadas. Asimismo, se detalla el número de documentos obtenidos a partir de la aplicación de estas sentencias en Scopus y en ACM DL, respectivamente. Finalmente se identifica el total de documentos resultantes de aplicar las tres cadenas de búsqueda.

Sintaxis SCOPUS	
Cadena de búsqueda 1	**N° resultados obtenidos**
TITLE-ABS-KEY (("protein fold" OR "protein structure prediction") AND sequence AND protein AND (aminoacid OR "amino acid") AND (3d OR tertiary) AND NOT secondary AND NOT "wet lab" AND NOT experimental AND NOT coronavirus AND NOT "cov-2" AND NOT covid) OR (alphafold AND AUTHOR-NAME (senior))	434
Cadena de búsqueda 2	**N° resultados obtenidos**
TITLE-ABS-KEY ("protein structure prediction" AND interface AND (application OR web OR serv* OR tool OR source))	106

Cadena de búsqueda 3	N° resultados obtenidos
TITLE-ABS-KEY ((usability OR user OR intuitive OR easy) AND interface AND web AND "protein structure prediction" AND NOT (rna OR dna))	33
Total	**573**

Tabla 8. *Cadenas de búsqueda con sintaxis del motor de búsqueda Scopus y cantidad de resultados obtenidos a partir de la ejecución de consulta. Hasta el momento no se ha utilizado ningún criterio de inclusión o exclusión*

Sintaxis ACM Digital Library	
Cadena de búsqueda 1	**N° resultados obtenidos**
AllField:((("protein fold" OR "protein structure prediction") AND sequence AND protein AND (aminoacid OR "amino acid") AND (3d or tertiary) AND NOT secondary AND NOT "wet lab" AND NOT experimental AND NOT coronavirus AND NOT "cov-2" AND NOT covid)) OR (alphafold AND ContribAuthor:(senior))	19
Cadena de búsqueda 2	**N° resultados obtenidos**
AllField:("protein structure prediction" AND interface AND (application OR web OR serv* OR tool OR source))	93
Cadena de búsqueda 3	**N° resultados obtenidos**
AllField:((usability OR user OR intuitive OR easy) AND interface AND web AND "protein structure prediction" AND NOT (RNA OR DNA))	15
Total	**127**

Tabla 9. *Cadenas de búsqueda con sintaxis del motor de búsqueda ACM DL y cantidad de resultados obtenidos a partir de la ejecución de consulta. Hasta el momento no se ha utilizado ningún criterio de inclusión o exclusión.*

Tanto en la Tabla 8 como en la Tabla 9 se observan que las cadenas de búsqueda aplicadas proporcionan una serie de publicaciones interesantes por cada uno de los motores utilizados, pero se identificaron algunas publicaciones repetidas (resultantes en más de una cadena de búsqueda utilizando el mismo motor).

Respecto a Scopus, se obtuvo un total de quinientas setenta y tres publicaciones, pero, al compilar los resultados de las tres cadenas, se identificó cuarenta y nueve duplicados. Asimismo, se obtuvo un total de ciento veintisiete publicaciones en ACM Digital Library, pero se detectaron diecinueve duplicadas.

Posteriormente se procedió a eliminar los sesenta y ocho duplicados encontrados en los diversos motores de búsqueda, con lo cual la cantidad de documentos se redujo de setecientos a seiscientos treinta y dos. En este punto se interceptaron los resultados de los dos motores de búsqueda y se identificó un documento en común. Quedando finalmente, seiscientos treinta y un artículos.

En la Tabla 10 se puede observar un resumen cuantitativo de los resultados obtenidos en base a las cadenas de búsqueda descritas anteriormente. Se muestra la cantidad de documentos obtenidos con cada cadena de búsqueda y en cada motor, además del total de documentos duplicados y no duplicados. Estos últimos serán los que continuarán en la presente revisión.

Resumen cuantitativo de publicaciones obtenidas			
N° cadena de búsqueda	**SCOPUS**	**ACM Digital Library**	**Totales**
1	434	19	453
2	106	93	199
3	33	15	48
Totales con duplicados	573	127	700
Duplicados	49+1[8]	19	69
Totales sin duplicados	**523**	**108**	**631**

Tabla 10. *Resumen cuantitativo de las publicaciones obtenidas por cadena y motor de búsqueda.*

3.4.3 Criterios de inclusión/exclusión

En la sección anterior se obtuvieron seiscientas treinta y un publicaciones en total, considerando ambos motores de búsqueda. Sin embargo, consideramos inviable la evaluación de esta cantidad de artículos debido a la naturaleza del proyecto de tesis, razón por la cual se procedió a aplicar algunos criterios de inclusión o exclusión, los

[8] Se hace referencia al documento común hallado al interceptar los dos motores de búsqueda.

cuales serán detallados enseguida. Esto se realiza con la finalidad de acotar los resultados a los que estrictamente tienen una vinculación con el presente proyecto, y, con ello, obtener la mayor utilidad para resolver las preguntas de revisión.

3.4.3.1 Criterios de exclusión

Los criterios de exclusión fueron definidos a partir del análisis general de los documentos obtenidos a partir de las consultas de la sección anterior. Cada uno de estos criterios se identificará a través de un código único con la siguiente estructura: CE[número].

A continuación, se detallan los criterios de exclusión que se aplicarán en la presente revisión:

CE1. Excluir aquellos documentos que superen los cinco años de antigüedad a partir de su publicación. Esto debido a que, al contar con suficientes recursos disponibles, se considera conveniente valorar mucho más a las publicaciones que hayan sido presentadas recientemente.

CE2. Descartar aquellos documentos que no tengan relación directa con las especialidades descritas en la Tabla 11. Las áreas de investigación a tener en cuenta varían dependiendo de la pregunta de revisión a la que está destinado el recurso encontrado, ya que cada pregunta tiene un enfoque diferente de acuerdo a los objetivos del presente estudio.

Especificación del criterio de exclusión CE2	
Precondición	**Área de investigación a relacionar**
La publicación se relaciona con al menos una pregunta de revisión.	Ingeniería (ENGI) y Ciencias de la Computación (COMP).
La publicación se relaciona con la pregunta de revisión P1.	Biomedicina (BIOC), Matemáticas (MATH) y Multidisciplinario (MULT).

Tabla 11. *Relación entre el criterio de exclusión CE2 y las áreas de investigación. Esta relación depende de la pregunta de revisión a la que está destinada la publicación encontrada.*

CE3. Excluir los recursos que estén relacionados a la pregunta de revisión N°1 y que sean de tipo “Artículos de conferencia”. Esto a consecuencia de que el factor de impacto de la procedencia de las investigaciones relacionadas a la bioinformática es alto. De esta manera, a través de este criterio se busca

una mayor tasa de confiabilidad y credibilidad en los recursos a tomar en cuenta.

CE4. Separar aquellos resultados que estén relacionados a la pregunta de revisión P2 en ACM DL y que sean de tipo “Libro” o “Sección de Libro”. Será importante tener en consideración este criterio en vista de que esa pregunta de revisión está relacionada a la evaluación de proyectos de software ya implementados, por lo cual se ve muy conveniente enfocar el análisis de documentos en los casos de aplicación, en lugar de en teoría concreta.

3.4.3.2 Criterios de inclusión

Los criterios de inclusión fueron definidos a partir del análisis general de los recursos obtenidos luego de la ejecución de las cadenas de búsqueda en cada uno de los motores seleccionados. Cada criterio se identificará a través de un código único con la siguiente estructura: CI[número].

A continuación, se detallan los criterios de inclusión que se aplicarán en la presente revisión:

CI1. Incluir aquellos resultados que estén relacionados a más de una pregunta de revisión a la vez. Esto debido a que las cadenas de búsqueda utilizadas han tenido un enfoque independiente, sin embargo, algunos documentos se obtuvieron por duplicado. Lo cual quiere decir que un mismo documento podría ser de utilidad para responder a más de una pregunta de revisión a la vez.

CI2. Incluir aquellos estudios que estén desarrollados en su totalidad en el idioma inglés o español. Se han considerado estos dos idiomas debido a tres factores: en primer lugar, la mayoría de los recursos se encuentran elaborados en inglés; en segundo lugar, porque las cadenas de búsqueda se han evaluado de esta manera para poder mejorar el alcance de las consultas; y, en tercer lugar, porque se dispone de conocimiento avanzado de ambos idiomas.

CI3. Incluir aquellos resultados que estén relacionados con alguna de las otras preguntas de revisión. Se ha identificado información de interés que apoya la resolución de las otras preguntas de revisión. Es por ello, que se deciden incluir dichos resultados, a pesar de haber sido obtenidos en el proceso de

otra pregunta de revisión. Este criterio de inclusión surge a partir del análisis que se explicará en el siguiente apartado[9].

3.4.4 Aplicación de criterios de exclusión/inclusión

Una vez identificados los criterios de exclusión e inclusión, éstos se aplicarán a las publicaciones obtenidas por las diversas consultas en los dos motores de búsqueda seleccionados. De esa manera se obtendrán las investigaciones que superen los criterios de exclusión y que cumplan enteramente con los criterios de inclusión.

Cabe destacar que, si bien hasta este punto no se utilizaron todas las subcadenas de búsqueda planteadas en el apartado anterior, en esta sección sí se tendrán en cuenta, ya que están directamente relacionados con los criterios de exclusión e inclusión. En otras palabras, aquellas subcadenas cumplirán un rol de soporte para evaluar dichos criterios. Hasta este momento se contaba con seiscientos treinta y un documentos. A partir de ello, se procedió a organizar, de forma preliminar, los datos principales del total de resultados en un formulario de extracción, el cual se presentará a detalle en la sección 3.5 Formulario de extracción de datos. Dicho formulario se encuentra adjunto en el Anexo B del presente documento. Se hizo uso de este, particularmente, para la aplicación de los criterios de inclusión y exclusión de la estrategia planteada. De modo concreto, se presenta el detalle de los documentos que van siendo descartados o incluidos por cada uno de los criterios.

La primera instancia de la estrategia consistió en la ejecución de los criterios de exclusión. En la Tabla 12 se muestra el código identificador y la descripción de cada uno de ellos, así como la cantidad de documentos que se excluyeron de la revisión luego de aplicarlos. Se detalla, también, el total de recursos restantes que serán evaluados en una segunda instancia.

Resumen cuantitativo de resultados por criterio de exclusión		
Código de criterio	**Criterios de exclusión**	**Número de resultados**
CE1	Excluir aquellos documentos que superen los cinco años de antigüedad a partir de su publicación.	519
CE2	Descartar aquellos documentos que no tengan relación directa con las especialidades descritas en la Tabla 11.	32

[9] Esto se desarrolla en la sección 3.4.4 del capítulo Estrategia de búsqueda, Tabla 15.

CE3	Excluir los recursos que estén relacionados a la pregunta de revisión P1 y que sean de tipo “Artículos de conferencia”.	8
CE4	Separar aquellos resultados que estén relacionados a la pregunta de revisión P2 en ACM DL y que sean de tipo “Libro” o “Sección de Libro”.	10
Resultados que no son excluidos		62

Tabla 12. *Resumen cuantitativo de resultados por criterios de exclusión.*

Como resultado de aplicar los criterios de exclusión a las seiscientas treinta y un investigaciones, se han podido extraer quinientas sesenta y nueve, quedando, así, sesenta y dos documentos no excluidos. Posteriormente procedemos a filtrarlos respecto a los criterios de inclusión.

No obstante, previo a este paso, se determina como acción necesaria el análisis de los resúmenes de las investigaciones obtenidas hasta el momento. Esto debido a que se está asumiendo que, por ejemplo, los recursos obtenidos en las consultas N°1[10] ayudarán a resolver la pregunta de revisión

P1, y que, si el mismo documento fue obtenido, por ejemplo, en la consulta N°2 destinada a la pregunta P2, entonces tal documento ayudará a responder ambas preguntas de revisión. Acorde a ello, la totalidad de recursos, así como su distribución por pregunta de revisión, hasta este momento, se encuentran detallados en la Tabla 13.

Distribución de documentos por pregunta de revisión	
Pregunta de revisión	**Número de resultados**
P1	39
P2	26
P3	9
Total de documentos	62

Tabla 13. *Distribución cuantitativa de documentos respecto a preguntas de revisión antes del análisis de resúmenes.*

[10] Tener en cuenta que las consultas se realizaron en dos motores de búsqueda diferentes.

Ante ello, se reconoce que al partir de una suposición no es plausible continuar con la aplicación de los criterios de inclusión, los cuales analizan mucho más el beneficio del recurso en cuanto a su utilidad dentro de las preguntas planteadas. Así, se tomará conocimiento de los abstractos del total de investigaciones obtenidas hasta ahora. Esto con el fin de verificar o, según sea conveniente, corregir la relación entre el recurso y la pregunta de revisión a la que se dirigirá. Para poder llevar a cabo este análisis se recurrió a completar con más detalle el formulario de extracción adjunto en el Anexo B. En específico, se completaron las columnas del resumen de los sesenta y dos documentos, y de las anotaciones resultantes de su lectura. En ese sentido, se identificó que algunos resúmenes presentaban un enfoque enteramente biomédico, biológico o de otra índole. Esto en tanto que se incluían términos como virus, biological insights, diabetes, hepatitis, alzheimer y demás terminologías médicas que dieron a entrever que el enfoque de esas investigaciones no era la predicción de estructuras ni el desarrollo de software usable.

Dado que, en algunos casos, al revisar el resumen de los documentos no se encontró relación directa con los temas planteados, se procedió a la lectura rápida del propio recurso. Gracias a ello se comprendió el motivo por el cual se habían obtenido en la consulta, y ello fue debido a que sí incluyen nuestras cadenas de búsqueda; sin embargo, solo ocurren como menciones puntuales. A consecuencia de lo explicado, estos recursos no fueron asignados a ninguna pregunta de revisión. En el mismo sentido, se identificó que algunas otras podían ser de utilidad para otras preguntas, por lo cual se actualizaron la relación entre los documentos y las preguntas de revisión. Este cambio puede observarse también en el formulario de extracción del Anexo B. El resultado de este análisis se refleja en la nueva distribución de documentos respecto a las preguntas de revisión de la Tabla 14.

Distribución de documentos por pregunta de revisión	
Pregunta de revisión	**Número de resultados**
P1	37
P2	22
P3	7
No relacionados	18
Total de documentos	62

Tabla 14. *Distribución cuantitativa de documentos respecto a preguntas de revisión luego del análisis de abstractos.*

Una vez realizada esta validación, se observa que existen dieciocho documentos que no tienen relación directa a ninguna pregunta de revisión, por lo cual se optará por considerar un criterio de inclusión adicional CI3, el cual tendrá que aplicarse antes que los criterios definidos previamente (CI1 y CI2)[11]. Tras definir el nuevo criterio, se procede a continuar con la evaluación de los documentos. Solo los resultados que cumplan fielmente con la totalidad de los criterios de inclusión serán los que se tendrán en cuenta para resolver las preguntas de revisión. La segunda instancia de la estrategia consistió en la ejecución de los criterios de inclusión. En la Tabla 15 se muestra el código identificador y la descripción de cada uno de ellos, así como la cantidad de documentos que los superaron, por lo que seguirían siendo parte de la revisión. Se detalla, también, el total de recursos que finalmente no fueron excluidos y que cumplieron con todos los criterios aplicados en ambas instancias de la estrategia de búsqueda.

Resumen cuantitativo de resultados por criterio de inclusión		
Código de criterio	**Criterios de inclusión**	**Número de resultados**
CI3	Incluir aquellos resultados que estén relacionados con al menos una pregunta de revisión.	44
CI1	Incluir aquellos resultados que estén relacionados a más de una pregunta de revisión a la vez.	18
CI2	Incluir aquellos estudios que estén desarrollados en el idioma inglés o español.	17
Resultados que superaron los criterios de exclusión y que cumplen con los criterios de inclusión		17

Tabla 15. *Resumen cuantitativo de resultados por criterios de inclusión.*

3.4.5 Documentos encontrados

A partir de la aplicación de la estrategia de extracción de datos en base a criterios de inclusión y exclusión, se ha obtenido un total de diecisiete publicaciones relacionadas a la presente revisión. A continuación, se listan los documentos y su código identificador, obtenidos a partir de lo mencionado y ordenados por el ID de los mismos, de acuerdo al formulario de extracción del Anexo B:

[11] Ver sección 3.4.3.2 del capítulo Estrategia de Búsqueda.

[E18] Machado, V. S., Tanus, M. S. S., Paixão-Cortes, W. R., de Souza, O. N., Campos, M. B., & Silveira, M. S. (2018). wCReF – a web server for the cref protein structure predictor. *Advances in Intelligent Systems and Computing, 558*, 831–838. https://doi.org/10.1007/978-3-319-54978-1_103

[E44] Makigaki, S., & Ishida, T. (2020). Sequence alignment using machine learning for accurate template-based protein structure prediction. *Bioinformatics*, *36*(1), 104–111. https://doi.org/10.1093/bioinformatics/btz483

[E148] Deng, H., Jia, Y., & Zhang, Y. (2018). Protein structure prediction. *International Journal of Modern Physics B, 32*(18). https://doi.org/10.1142/S021797921840009X

[E255] Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W. R., Bridgland, A., Penedones, H., Petersen, S., Simonyan, K., Crossan, S., Kohli, P., Jones, D. T., Silver, D., Kavukcuoglu, K., & Hassabis, D. (2020). Improved protein structure prediction using potentials from deep learning. *Nature, 577*(7792), 706–710. https://doi.org/10.1038/s41586-019-1923-7

[E418] Gao, M., Zhou, H., & Skolnick, J. (2019). DESTINI: A deep-learning approach to contact-driven protein structure prediction. *Scientific Reports, 9*(1). https://doi.org/10.1038/s41598-019-40314-1

[E442] Huang, Y., Li, H., & Xiao, Y. (2018). 3dRPC: A web server for 3D RNA-protein structure prediction. *Bioinformatics, 34*(7), 1238–1240. https://doi.org/10.1093/bioinformatics/btx742

[E451] Aguirre-Plans, J., Meseguer, A., Molina-Fernandez, R., Marín-López, M. A., Jumde, G., Casanova, K., Bonet, J., Fornes, O., Fernandez-Fuentes, N., & Oliva, B. (2021). SPServer: split-statistical potentials for the analysis of protein structures and protein–protein interactions. *BMC Bioinformatics, 22*(1). https://doi.org/10.1186/s12859-020-03770-5

[E452] Adhikari, B., & Cheng, J. (2018). CONFOLD2: Improved contact-driven ab initio protein structure modeling. *BMC Bioinformatics, 19*(1). https://doi.org/10.1186/s12859-018-2032-6

[E470] Paixão-Cortes, V. S. M., Tanus, M. S. S., Paixão-Cortes, W. R., de Souza, O. N., Campos, M. B., & Silveira, M. S. (2018). Usability as the key factor to the design of a web server for the CReF protein structure predictor: The wCReF. *Information (Switzerland), 9*(1). https://doi.org/10.3390/info9010020

[E482] Lopes, G. R., de Souza, P. S. L., & Delbem, A. C. B. (2019). A Systematic Mapping on High-Performance Computing for Protein Structure Prediction. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 11333 LNCS*. https://doi.org/10.1007/978-3-030-15996-2_6

[E484] Hou, J., Wu, T., Guo, Z., Quadir, F., & Cheng, J. (2020). The MULTICOM Protein Structure Prediction Server Empowered by Deep Learning and Contact Distance Prediction. *Methods in Molecular Biology, 2165*, 13–26. https://doi.org/10.1007/978-1-0716-0708-4_2

[E490] Kuhlman, B., & Bradley, P. (2019). Advances in protein structure prediction and

design. *Nature Reviews Molecular Cell Biology, 20*(11), 681–697. https://doi.org/10.1038/s41580-019-0163-x

[E492] Jin, S., Contessoto, V. G., Chen, M., Schafer, N. P., Lu, W., Chen, X., Bueno, C., Hajitaheri, A., Sirovetz, B. J., Davtyan, A., Papoian, G. A., Tsai, M.-Y., & Wolynes, P. G. (2020). AWSEM-Suite: A protein structure prediction server based on template-guided, coevolutionary-enhanced optimized folding landscapes. *Nucleic Acids Research, 48*(W1), W25–W30. https://doi.org/10.1093/NAR/GKAA356

[E493] Zheng, W., Zhang, C., Wuyun, Q., Pearce, R., Li, Y., & Zhang, Y. (2019). LOMETS2: improved meta-threading server for fold-recognition and structure-based function annotation for distant-homology proteins. *Nucleic Acids Research, 47*(W1), W429–W436. https://doi.org/10.1093/nar/gkz384

[E514] McGehee, A. J., Bhattacharya, S., Roche, R., & Bhattacharya, D. (2020). PolyFold: An interactive visual simulator for distance-based protein folding. *PLoS ONE, 15*(12 December). https://doi.org/10.1371/journal.pone.0243331

[E598] Kordic, B., Popovic, M., Popovic, M., Goldstein, M., Amitay, M., & Dayan, D. (2019). A Protein Structure Prediction Program Architecture Based on a Software Transactional Memory. *Proceedings of the 6th Conference on the Engineering of Computer Based Systems.* https://doi.org/10.1145/3352700.3352701

[E621] Abeysinghe, E., Brylinski, M., Christie, M., Marru, S., & Pierce, M. (2019). LSU computational system biology gateway for education. *ACM International Conference Proceeding Series.* https://doi.org/10.1145/3332186.3333259

A modo de síntesis, en la Tabla 16 se presenta el total de documentos encontrados, así como su última distribución cuantitativa por pregunta de revisión.

Distribución de documentos por pregunta de revisión	
Pregunta de revisión	**Número de resultados**
P1	14
P2	17
P3	7
Total de documentos	17

Tabla 16. *Distribución cuantitativa de documentos respecto a preguntas de revisión.*

3.5 Formulario de extracción de datos

Con el propósito de mantener organizada la información general recopilada de los documentos identificados, se ha preparado un formulario de extracción de datos. En la Tabla 17, se muestran los encabezados de dicho formulario. En particular, se detalla el nombre del campo, una breve descripción del mismo y la pregunta con la que se relaciona. El formulario de extracción se adjunta al presente documento como parte del

Anexo B. En él podemos encontrar dos pestañas: la primera llamada “Formulario de extracción” donde se encuentran detallados todos los recursos obtenidos desde las primeras consultas, y se especifica, además, los resultados de la aplicación de cada uno de los criterios de extracción e inclusión, con el fin de permitir replicar la revisión sistemática de forma transparente; la segunda pestaña llamada “Estudios primarios” contiene solo los recursos que superaron todos los criterios mencionados anteriormente. Ambas pestañas tienen el mismo formato, de acuerdo con la Tabla 17.

Campo	Descripción	Pregunta
ID	Identificador del documento encontrado. La estructura es la siguiente: [E{número}]. Por ejemplo: [E39].	General
Motor de búsqueda	Nombre del motor de búsqueda por el cual se obtuvo el recurso. Puede ser Scopus o ACM.	General
Autor(es)	Autor o autores del recurso.	General
Título	Nombre completo del ejemplar encontrado.	General
Fuente	Procedencia del recurso. Puede ser el nombre de una conferencia, un journal, entre otros.	General
Tipo de documento	Tipo de documento encontrado. Por ejemplo: Artículo, Libro, Sección de Libro, entre otros.	General
Año	Año de publicación del documento.	General
DOI	Identificador de un objeto digital (DOI, por sus siglas en inglés).	General
CE0	Afirmación o negación con relación a si es un documento sin duplicados (“SI” en caso sea único).	General
CE1	Afirmación o negación con relación al resultado de la aplicación del CE1 (“SI” en caso de superar el criterio de exclusión).	General
CE2	Afirmación o negación con relación al resultado de la aplicación del CE2 (“SI” en caso de superar el criterio de exclusión).	General
CE3	Afirmación o negación con relación al resultado de la aplicación del CE3 (“SI” en caso de superar el criterio de exclusión).	General

CE4	Afirmación o negación con relación al resultado de la aplicación del CE4 (“SI” en caso de superar el criterio de exclusión).	General
CI1	Afirmación o negación con relación al resultado de la aplicación del CI1 (“SI” en caso de superar el criterio de inclusión).	General
CI2	Afirmación o negación con relación al resultado de la aplicación del CI2 (“SI” en caso de superar el criterio de inclusión).	General
Aceptado	Afirmación o negación con relación al resultado de la aplicación de todos los criterios (“SI” en caso sea parte de los estudios primarios).	General
Resumen	Abstracto de la investigación. Este puede incluir el motivo, la propuesta y los resultados.	General
Anotación	Interpretación del resumen leído, a partir del cual se vuelve a considerar la relación entre el documento y las preguntas de revisión.	General
Pregunta de revisión relacionada por la consulta	Pregunta de revisión a la cual se dirige por su contenido. Esta se obtuvo a partir de la consulta.	General
Pregunta de revisión relacionada por análisis del abstracto	Pregunta de revisión a la cual se dirige por su contenido. Esta se obtuvo a través del análisis del resumen del documento.	General
Idioma	Refiere al idioma en el que está desarrollado el documento. No basta con revisar el idioma del resumen.	General
Citado por	Número de recursos que han citado a este documento.	General
Técnica(s) y/o algoritmos de predicción	Mencionar todas las técnicas y/o algoritmos específicos utilizados para realizar la predicción de estructuras. Se obtiene el detalle de estos y de su procedimiento.	Pregunta 1
Requisitos funcionales y no funcionales	Describir si el recurso presenta un proyecto de software, ofrece sus fuentes, web service desplegado, datos de entrada y salida. Se obtiene el detalle de requisitos.	Pregunta 2
Metodología de usabilidad	Identificar si el documento presenta y/o aplica alguna metodología de usabilidad. Se obtiene el detalle de los pasos seguidos.	Pregunta 3

Tabla 17. *Encabezados del formulario de extracción de las publicaciones obtenidas.*

3.6 Resultados de la revisión

Una vez obtenidos y analizados los estudios primarios recopilados a partir de la estrategia de búsqueda descrita en el apartado 3.4 Estrategia de búsqueda, se procede a elaborar una respuesta a cada pregunta de revisión planteada. Estas respuestas están fundadas en base a la información del formulario de extracción presentado.

3.6.1 Respuesta a pregunta P1: ¿Cuáles son los algoritmos capaces de explotar datos de estructura primaria para predecir estructura terciaria, cuáles son sus requisitos y cómo funcionan?

La revisión sistemática realizada permite entender que para solucionar el problema de predicción de estructuras terciarias de proteínas a partir de su secuencia de aminoácidos se debe seguir una receta general constituida por fases que pertenecen a campos de investigación independientes (Deng et al., 2018). Esta receta consta de cinco pasos fundamentales y se observa en el flujograma de la Figura 6: inicialización de conformación, búsqueda conformacional, selección de estructura, reconstrucción de todos los átomos y el refinamiento de la estructura (Deng et al., 2018).

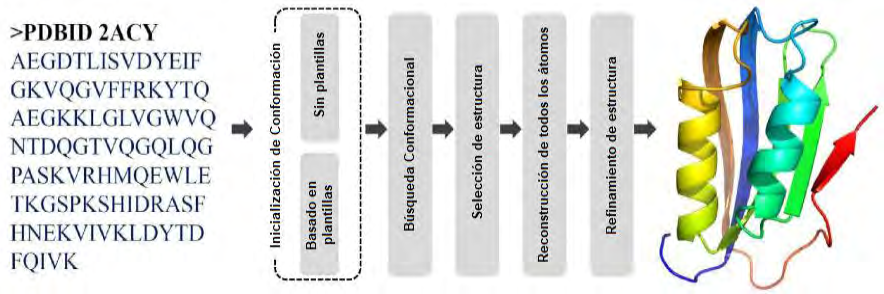

Figura 6. *Flujograma general de predicción de estructuras de proteínas. Inicia a partir de la secuencia de aminoácidos de la proteína y termina en la estructura tridimensional. Adaptado de (Deng et al.,2018).*

Previo a describir cada uno de los pasos del flujograma, cabe mencionar que existe una distinción general de los métodos de predicción de proteínas que los separa en dos tipos: las técnicas basadas en plantillas y las técnicas con un enfoque libre de estas (Kuhlman & Bradley, 2019b). Esta separación se apoya en la primera etapa del flujograma, donde los métodos basados en plantillas obtienen su conformación inicial a partir de proteínas con estructuras previamente identificadas, las cuales son homólogas secuencial o estructuralmente a la proteína objetivo (Deng et al., 2018). Una secuencia será considerada homóloga de otra siempre que cumpla con tener una similitud mínima del 30% en su conformación de aminoácidos (Jin et al., 2020). En este punto, algunos métodos para obtener estructuras primarias desde el Banco de Datos de Proteínas

(PDB, por sus siglas en inglés), para usarlos como plantillas, son BLAST y HHblits denominados como métodos de alineamiento de secuencias (Kuhlman & Bradley, 2019b; Machado et al., 2018). Por otro lado, se destacan otros subtipos denominados como técnicas de reconocimiento de pliegues que se basan en la existencia de proteínas que no son homólogas según sus secuencias, pero sí según sus estructuras 3D obtenidas de forma experimental (Deng et al., 2018). Respecto a estas técnicas, se han encontrado estudios donde se introducen tecnologías de aprendizaje de máquina para poder obtener estructuras relacionadas a la proteína objetivo, logrando mejores resultados en comparación a técnicas como PSI-BLAST, HHsearch, DELTA-BLAST, entre otros, que pertenecen al subtipo anterior basado en el alineamiento de secuencias (Makigaki & Ishida, 2020). Estos mencionados también pueden haber usado machine learning como es el caso de eThread (Abeysinghe et al., 2019). No obstante, no siempre hay garantías de encontrar exitosamente proteínas homólogas en PDB, por lo que se requieren de métodos como Rosseta, QUARK, SCRATCH, PROFESY o FRAGFOLD, que no dependan de las plantillas o la familiaridad de las proteínas y puedan ser aplicados a moléculas que suponen tener un plegado completamente desconocido (Deng et al., 2018). Estos métodos usualmente suponen el alineamiento de estructuras primarias de proteínas relacionadas para predecir sus características locales y no locales (Khulman & Bradley, 2019). Se ha identificado el uso de técnicas computacionales en relación a lo mencionado: minería de datos en CReF para predecir los ángulos de torsión de la columna central de la estructura de proteínas; redes neuronales residuales convolucionales (aprendizaje profundo) en DESTINI o PSICOV para predecir los contactos entre residuos; y redes neuronales profundas en AlphaFold para predecir las distancias entre los residuos (Adhikari & Cheng, 2018; Gao et al., 2019; Hou et al., 2020; Machado et al., 2018; Senior et al., 2020). Respecto al segundo paso del proceso de la Figura 6, la búsqueda conformacional espera encontrar un amplio número de estructuras casi nativas[12] relacionadas a la proteína objetivo, esto con la guía de ciertos campos de fuerza o funciones energéticas dentro de un espacio de posibles conformaciones (Deng et al., 2018) Estas funciones se pueden construir en base a las características locales y no locales mencionadas en el párrafo anterior, por lo cual, las técnicas aplicadas en ese paso también pueden intervenir en esta etapa del proceso (Deng et al., 2018). Una vez que la función energética ha sido determinada, se requiere

[12] Se considera estructura nativa a toda aquella que tiene la tasa más baja de energía libre (Kuhlman & Bradley, 2019b).

establecer la manera de encontrar la conformación con menor energía para lo cual muchas estrategias como *simulating annealing* y *replica exchange* fueron utilizados en programas como DEEPSAM (Kordic et al., 2019). Asimismo, se han utilizado algoritmos genéticos para explorar el espacio de búsqueda, como es el caso de los sistemas informáticos de alto rendimiento, HPC, por sus siglas en inglés (Lopes et al., 2019). Llegado a la etapa de selección de estructura es crucial determinar un método de evaluación de estructuras capaz de distinguir entre las que son nativas o no (Deng et al., 2018). Para este paso se disponen de métodos como 3dRPC-score (Huang et al., 2018). Posteriormente, los procedimientos para el reconocimiento de todos los átomos y el refinamiento de estructuras surgen a partir de que en la etapa de la búsqueda conformacional se adoptan simplificaciones de la representación de las proteínas (Deng et al., 2018). De esa manera, la estructura se redujo de las conformaciones completas de carbono, nitrógeno, oxígeno, el carbono alfa y residuo, a solo estos dos últimos (Deng et al., 2018). Ante lo cual, se requiere de procesos de reconstrucción de todo el esqueleto central de la proteína y de las cadenas laterales con métodos especializados en ello como SABBAC, BBQ, PULCHRA y REMO; al igual que Scwrl, SCATD y RASP (Deng et al., 2018). Del mismo modo, se han implementado algunas técnicas como las de FG-MD basadas en la simulación dinámica de las moléculas, con las cuales se mejora la calidad de la estructura que finalmente es obtenida de la predicción (Deng et al., 2018). Por último, es importante mencionar que, aunque los procesos de predicción de estructuras terciarias de proteínas en detalle estén comprendidos por varios métodos, los pasos fundamentales son los especificados a lo largo de esta respuesta (Deng et al., 2018).

3.6.2 Respuesta a pregunta P2: ¿Cuáles son los requisitos necesarios para realizar el diseño de una interfaz que permita explotar datos de estructura primaria para predecir su estructura terciaria y cuáles son sus especificaciones?

De acuerdo a la documentación obtenida en la revisión sistemática, se ha observado una tendencia en torno al desarrollo de métodos que estarán disponibles a través de la web como medios de solución en el CASP (Paixão-Cortes et al., 2018), tal y como es el caso de wCREF[13] (Machado et al., 2018), 3dRPC[14] (Huang et al., 2018),

[13] https://www.wcref.labio.org/. Al desarrollo de la investigación el servidor no estaba disponible.

[14] http://biophy.hust.edu.cn/3dRPC. Al desarrollo de la investigación el servidor no estaba disponible.

the MULTICOM (Hou et al., 2020), AWSEM-Suite[15] (Jin et al., 2020), Robetta server[16] (Machado et al., 2018), I-TASSER[17] (Machado et al., 2018), QUARK[18] (Machado et al., 2018), entre otros. Teniendo en cuenta estos proyectos desarrollados en torno a la predicción de estructuras y temáticas afines, se han obtenido ciertos requisitos que ayudan a la definición de las necesidades principales y al diseño de las interfaces que permitan explotar datos de estructura primaria de las proteínas para predecir sus conformaciones tridimensionales.

En la Tabla 18 se presentan y especifican los requisitos funcionales de las interfaces identificadas. Dicha tabla posee tres campos: el primero es el campo “Requisitos”, donde se presentan los requisitos encontrados, el segundo campo denominado “Especificación” contiene más detalles respecto a la funcionalidad y el tercero, “Referencia”, hace alusión al registro de la sección 3.4.5 Documentos encontrados del cual se obtuvo la información. Estos registros también se encuentran en el formulario de extracción adjunto en el Anexo B.

De manera similar, en la Tabla 19 se muestran los requisitos no funcionales de las interfaces identificadas en los proyectos capturados en la revisión[19]. Esta tabla posee los mismos campos descritos en el párrafo anterior.

[15] https://awsem.rice.edu

[16] https://robetta.bakerlab.org/login.php?next_url=%2Fsubmit.php

[17] https://zhanglab.ccmb.med.umich.edu/I-TASSER/

[18] https://zhanglab.ccmb.med.umich.edu/QUARK/

[19] De la literatura identificada, no se han podido encontrar requisitos de dominios aplicables a nuestro caso en particular, es decir, a la predicción de estructuras terciarias de proteínas repetidas, pero la identificación de estos se considerará dentro de los objetivos del presente proyecto

Requisitos funcionales		
Requisito	**Especificación**	**Referencia**
El dato de entrada principal es la estructura primaria de una proteína.	Se requiere una secuencia plana de los aminoácidos de las proteínas que esté especificada en sus simbologías, como se muestra en la Tabla 7.	[E18], [E44], [E418], [E484], [E490], [E492]
Verificación de contenido de secuencias	El sistema verifica que la estructura primaria ingresada como dato de entrada solo esté conformado por los veinte aminoácidos fundamentales.	[E484], [E490]
Uso del servidor sin inicio de sesión	Para poder enviar una solicitud de predicción no es necesario el inicio de sesión del usuario.	[E44], [E255], [E490], [E442]
Uso del servidor con inicio de sesión	Para poder solicitar una predicción es necesario el inicio de sesión del usuario.	[E18]
Formatos aceptados de la secuencia de aminoácidos.	El dato de entrada principal será aceptado solo en formato de texto plano, en formato FASTA o en el formato de secuencias del Banco de Datos de Proteínas (PBD, por sus siglas en inglés).	[E18], [E44], [E255], [E418], [E451], [E470], [E490], [E492]
Datos de entrada secundarios: el nombre de la proteína objetivo y el correo electrónico del usuario.	Se solicita el ingreso de estos datos secundarios para poder brindar la respuesta del procesamiento.	[E18], [E442], [E470], [E484]
Identificadores de solicitud de predicción	Cada tarea enviada para la predicción registrará información que permita su identificación como: ID, nombre de la proteína, tamaño (cantidad de aminoácidos) y fecha de solicitud.	[E18], [E470], [E490], [E451]
Envío de resultados.	La estructura tridimensional obtenida será enviada al usuario por correo electrónico.	[E18], [E470], [E484]
Formato de entrega de resultados	La predicción de la estructura terciaria, así como otros resultados obtenidos, se entregarán a través de links dentro del correo o en la interfaz.	[E255], [E484], [E490], [E492]

Consulta de solicitudes	El servidor permite consultar las solicitudes ingresadas.	[E255], [E442], [E470]
Consulta privada de solicitudes	El usuario solo podrá consultar por las solicitudes que haya ingresado.	[E18]
Modelos adicionales de apoyo.	El usuario puede ingresar modelos de apoyo en referencia a la predicción de contacto entre residuos de proteínas o los resultados de las alineaciones de secuencias múltiples (MSA, por sus siglas en inglés). Si no se ingresan, la herramienta los genera.	[E442], [E452], [E492]
Registro de usuarios	Se requiere la creación de una cuenta para el usuario. Se validará su usuario y contraseña para que pueda utilizar servicios adicionales de predicción.	[E18], [E255], [E470]
Sección adicional para usuarios registrados	La visualización del estado general de todas las solicitudes enviadas será visible para todos los usuarios, pero el seguimiento detallado de las solicitudes y la visualización de los resultados solo serán posibles para el usuario registrado.	[E18], [E470]
Perfil de usuario	El usuario registrado podrá disponer de diversas funcionalidades respecto al mantenimiento de su cuenta, como el inicio de sesión, cierre de sesión, modificación del usuario, recuperación de cuenta y la cancelación de su registro.	[E18], [E470]
Estado de solicitudes	En el caso del usuario registrado, se podrá visualizar el progreso de las solicitudes de predicción.	[E18], [E255], [E470]
Eliminar solicitudes	Se cuenta con la opción de eliminar solicitudes de predicción. Solo para los usuarios registrados y con la debida confirmación.	[E18], [E470]
Visualización de resultados	El sistema ofrece un resumen de los datos ingresados, el proceso seguido, los parámetros usados y la predicción como resultado del procesamiento.	[E18], [E442], [E470]
Estructuras adicionales	El sistema ofrece, además, la estructura secundaria generada en el proceso.	[E18], [E484]
Visualización de predicción	Se cuenta con una herramienta de visualización añadida a la propia interfaz.	[E18], [E442], [E470], [E490]

Exportación de resultados	El sistema permite exportar los resultados tanto en modo impresión para PDF como en el formato PDB. Solo para el usuario registrado (en algunos casos).	[E18], [E44], [E470], [E484], [E493], [E514]
Soporte al usuario	El servidor cuenta con herramientas como tutoriales, documentación y ayuda online para guiar al usuario en torno a su uso.	[E18], [E44], [E470]
Alertas y errores	El sistema notifica todo tipo de error en la misma ventana en la que ocurrió; solicita, además, su confirmación para proseguir.	[E18], [E470]
Información adicional	El sistema cuenta con información adicional y links externos sobre la problemática, además de información sobre el equipo de investigación.	[E18]
Seguridad del servidor	Solo el administrador podrá realizar configuraciones generales del servidor.	[E470]
Datos obligatorios y opcionales	Los formularios del proceso de predicción indican qué campos son obligatorios u opcionales.	[E18]
Envío de varias solicitudes a la vez	El usuario puede enviar varias solicitudes de predicción a la vez, sin tener que seguir muchos pasos. Requiere de su confirmación.	[E18], [E490]
Idioma	El usuario podrá cambiar el idioma del servidor.	[E18]
Tamaño de secuencia	El sistema permite un máximo de residuos de aminoácidos dentro de la secuencia de la proteína a predecir. Algunos contemplan un máximo de 500 o 1500 residuos.	[E493], [E514]
Tamaño máximo de archivo de secuencias	El sistema permite la carga de la estructura primaria a través de un archivo de peso máximo de 10 MB	[E451]
Archivos de prueba	El servicio pone a disposición del usuario una serie de archivos de ejemplo para usar en la plataforma.	[E451], [E490]

Tabla 18. *Listado de requisitos funcionales reconocidos en los documentos de la revisión.*

Requisitos no funcionales		
Requisito	**Especificación**	**Referencia**
Lenguajes de programación para el algoritmo	Los lenguajes de programación utilizados son Fortran, C++ y Python	[E44], [E470], [E492], [E598]
Lenguajes de programación para la interfaz	Los lenguajes de programación utilizados para el desarrollo de la interfaz fueron CSS, HTML5 o Java.	[E18], [E470], [E514]
Base de datos	La base de datos del servidor es MySQL	[E18], [E470]
Ejecución concurrente	El servidor puede recibir y ejecutar múltiples solicitudes de predicción por programación de tareas (colas).	[E18], [E598]
Visibilidad del sistema	La información del estado del proceso se refresca cada 30 segundos.	[E18], [E470]
Diseño minimalista	El enfoque del servidor en torno al diseño se centrará en los pasos para la predicción, en particular, al envío de solicitudes de predicción. Las demás opciones se muestran a demanda del usuario.	[E18], [E470]
Visualización de resultados.	El resultado de la predicción y otros complementos podrán ser visualizados a través de herramientas como: JSmol viewer, nextProt protein sequence viewer, NGL viewer, MSAviewer y iView.	[E18], [E470], [E484], [E492]
Compatibilidad con navegadores web.	La herramienta es compatible con los siguientes navegadores: Chrome, Firefox, Safari y Explorer.	[E451], [E470], [E484]
Disponibilidad del servidor	El servicio de predicción web estará disponible las 24 horas de los 7 días de la semana. En caso ocurra algún error del sistema, se permitirá la no disponibilidad de este hasta que se corrija el desperfecto.	[E470]
Performance del servidor	Se requiere que el sistema brinde resultados en una fracción de tiempo aceptablemente rápida. Algunos determinan la entrega en no más de 24 horas.	[E470], [E493]

Compatibilidad de software y hardware	Al ser una plataforma web no se considera ningún tipo de restricción respecto al hardware o software necesarios para utilizar el servidor. Solo debe contar con el navegador e Internet.	[E470]
Algoritmo de predicción	El sistema constituye un ambiente donde el algoritmo se ejecuta sin ningún inconveniente.	[E18]
Fácil de usar	Usar la interfaz implica una curva de aprendizaje baja.	[E18]
Información requerida solo una vez	El sistema solo solicita una vez la información necesaria pero invariable como el correo o la contraseña.	[E18]
Consistencia de la interfaz	Los elementos de la interfaz mostrarán consistencia respecto a sus colores y formas. Se resalta información relevante del proceso como alertas, estados y errores.	[E18], [E470]
Idioma	El sistema está disponible en varios idiomas.	[E18]
Mantenimiento de logs	El servidor mantiene logs de ejecuciones del usuario y log de errores.	[E44], [E470]
Software descargable	El sistema es descargable para usarlo desde un entorno local.	[E418], [E514]
Compatibilidad de sistemas operativos	El sistema se puede desplegar en entornos Windows, Linux/Unix, macOS.	[E514]

Tabla 19. *Listado de requisitos no funcionales reconocidos en los documentos de la revisión.*

3.6.3 Respuesta a pregunta P3: ¿Qué técnicas y métodos existen para la evaluación de usabilidad en herramientas bioinformáticas y cómo se llevan a cabo?

Para comenzar a responder la pregunta P3 es necesario recapitular brevemente el concepto básico en el que esta se concentra: la usabilidad.

La usabilidad es un término que se define como un conjunto de atributos que influyen tanto en el esfuerzo requerido para el uso de un software como en la apreciación general del mismo (International Organization for Standarization [ISO], 1991, como se citó en Bevan et al., 1991). A partir de ello, el usuario ocupa un rol primordial en la evaluación del software, así, los criterios mínimos que un recurso debe cumplir respecto a su facilidad de uso dependerán de las exigencias y necesidades de este (Bevan et al., 1991).

De acuerdo con la revisión sistemática realizada, existen distintos métodos con los cuales se puede efectuar la evaluación de usabilidad en herramientas bioinformáticas. Sin embargo, se dispone también de pasos previos a su aplicación que ayudan a involucrar este aspecto desde las etapas tempranas del desarrollo (Paixão-Cortes et al., 2018).

El primer paso identificado consiste en detectar las necesidades del usuario final, teniendo en cuenta perfiles expertos y no expertos (Paixão-Cortes et al., 2018). Se realizan consultas respecto a las tareas que usualmente se llevan a cabo en una plataforma de predicción a partir de la secuencia primaria de una proteína (Paixão-Cortes et al., 2018). Con ello se logra definir escenarios principales que serán de utilidad para, posteriormente, evaluar la usabilidad, además de identificar procesos primordiales que deberán incluirse en la solución (Paixão-Cortes et al., 2018).

El segundo paso comprende la comparación de soluciones similares a la que se espera proponer (Machado et al., 2018). Esta evaluación no se concentra en la herramienta propia, pero se lleva a cabo con metodologías dedicadas a la usabilidad, las cuales también podrían usarse al final del desarrollo de la misma.

Llegado a este punto, si bien estamos buscando métodos para medir el nivel de usabilidad aplicables a herramientas bioinformáticas, los pasos antes mencionados podrían ayudar a obtener herramientas usables.

De la literatura obtenida se identificaron dos metodologías de interés: la evaluación heurística de Nielsen y la aplicación del cuestionario (o adaptaciones) de Ssemugabi (Machado et al., 2018; Paixão-Cortes et al., 2018).

En relación al primero de ellos, cabe resaltar que Nielsen pone diez heurísticas a disposición del evaluador, con los cuales se otorga una calificación cuantitativa en base a los grados de severidad clasificados desde cero a cuatro (Paixão-Cortes et al., 2018). Las heurísticas son las siguientes:

- Visibilidad del estado del sistema
- Coincidencia entre el sistema y el mundo real
- Control y libertad para el usuario
- Consistencia y estándares
- Prevención de errores
- Reconocimiento antes que recuerdo
- Flexibilidad y eficiencia de uso
- Estética y diseño minimalista
- Ayuda a los usuarios para el reconocimiento, diagnóstico y recuperación de errores
- Ayuda y documentación

La verificación del cumplimiento de la lista de heurísticas se realiza a partir de la navegación de la interfaz (Paixão-Cortes et al., 2018). No se restringe a realizarla una sola vez, al contrario, se anima a usar la primera interacción con la herramienta para recorrerla y familiarizarse con ella (Paixão-Cortes et al., 2018). A partir de la segunda revisión, se pondrá mayor énfasis en los elementos y detalles que se crean convenientes (Paixão-Cortes et al., 2018).

En el mismo sentido, se considera apropiado contar con evaluadores de distintos perfiles, es decir, profesionales con experticia en el campo de interacción humano-computador (HCI, por sus siglas en inglés) además de especialistas afines a la herramienta, en este caso, bioinformáticos, biólogos, programadores, entre otros (Paixão-Cortes et al., 2018). Con ello se asegura la obtención de resultados desde diversas perspectivas.

Respecto a la segunda técnica de evaluación de usabilidad encontrada, está constituida por la interacción del usuario con la herramienta y el llenado de un cuestionario posterior a ella (Machado et al., 2018). Estos son de utilidad para la recolección de información subjetiva y retroalimentación acerca de la interfaz utilizada y el proceso mismo (Paixão-Cortes et al., 2018). Este tipo de evaluaciones se complementan con la observación de las actitudes y el análisis de los sentimientos de las personas que las realizan (Machado et al., 2018).

En particular, los cuestionarios de Ssemugabi se dividen en dos secciones: la primera parte consiste en preguntas referidas a las heurísticas de Nielsen mencionadas previamente; mientras que la segunda, se enfoca en preguntas relacionadas a E-learning (Paixão-Cortes et al., 2018). Debido a esto último, se considera pertinente adaptar esa sección en torno a la usabilidad en servidores dedicados a la predicción de estructuras de proteínas (Paixão-Cortes et al., 2018).

Finalmente, cabe mencionar a un servidor web de predicción de estructuras como caso de éxito de la aplicación de estas dos metodologías: wCReF. El estudio donde se presenta su evaluación de usabilidad define un cuestionario adaptado al contexto (Paixão-Cortes et al., 2018). El cual está constituido por cinco partes que se describen a continuación:

- Reconocimiento del perfil del usuario.
- Evaluación del diseño de interfaz, en base a las preguntas de Ssemugabi.
- Valoración del diseño web, en base a las preguntas de Ssemugabi.
- Calificación del contenido con relación a la bioinformática.
- Preguntas de conclusión con enfoque a aspectos positivos y negativos de la herramienta.

3.7 Conclusiones

Tal como se mencionó al inicio de la revisión sistemática, la predicción de estructuras de proteínas es un problema complejo que hoy en día continúa significando un desafío, a pesar de los avances logrados (Jordan, 2021). En este punto, luego de la revisión realizada a lo largo de todo el capítulo del estado del arte, se ha tomado consciencia de la realidad actual de la problemática y de los esfuerzos llevados a cabo con relación a los algoritmos implementados, el diseño de interfaces de las soluciones que las incluyen y la existencia de técnicas para su evaluación en torno a la usabilidad.

Respecto al primer punto[20], cabe mencionar que para lograr la predicción de estructuras de proteínas intervienen una serie de pasos actualmente identificados: inicialización de conformación, búsqueda conformacional, selección de estructura, reconstrucción de todos los átomos y el refinamiento de la estructura (Deng et al., 2018). No obstante, cada uno de ellos pertenece a campos de investigación totalmente independientes unos de los otros (Deng et al., 2018). Con ello, la solución está abierta a la mezcla de distintos

[20] Algoritmos de predicción de estructuras terciarias de proteínas

métodos destinados a ciertas etapas del proceso, los cuales fueron mencionados en la respuesta de la pregunta P1. Asimismo, se debe reconocer que lo que prima en la valoración de una propuesta no está relacionado a la mejora de esas tantas técnicas o al proceso mismo de selección e inclusión de ellas en una sola, sino que se enfoca en la facilidad de uso, eficiencia y confiabilidad de los resultados (Deng et al., 2018).

Acerca del segundo aspecto investigado[21], se resalta la diversidad de características funcionales y no funcionales que se han tomado en consideración dentro de las herramientas identificadas. Esto teniendo en cuenta que no todas las herramientas resolvían la predicción de estructuras de proteínas, sino que se enfocaron en una técnica perteneciente a una de las etapas que permiten lograrla, tal como se mencionó en el punto anterior. Asimismo, cabe mencionar, que si bien se identificaron algunos servicios web puestos a disposición de todo quien esté interesado, muchos de estos no se encontraron disponibles al momento de la realización de la presente investigación.

En torno al tercer elemento en cuestión[22]: ¿se puede afirmar que la usabilidad forma parte del proceso de desarrollo de las herramientas bioinformáticas? La respuesta a esta pregunta es definitiva y negativa en base a la revisión sistemática realizada. A pesar de que muchos de los casos de estudio se describen a sí mismos como herramientas “easy-to-use” y “user-friendly”, la usabilidad no parece haber sido un factor crucial para los servidores de predicción de estructuras de proteínas (Machado et al., 2018). Esto conlleva que las herramientas planteadas como solución al reto bioinformático estén compuestos por interfaces y reglas de juego que son difíciles de usar o entender, o, incluso, llevar a que los usuarios no puedan obtener resultados (Machado et al., 2018).

Por último, sin ánimo de perder de vista el objetivo del presente proyecto de tesis, es digno de mencionar el factor de aplicación de la predicción de estructuras dentro de un campo específico de la proteómica. Este aspecto se destacó a lo largo de toda la revisión en tanto que no hubo ningún resultado enfocado a la familia de proteínas repetidas. Esto se tomó en consideración a partir de una serie de consultas preliminares en los motores de búsqueda. En consecuencia, se decidió iniciar la revisión sistemática desde un plano más amplio, es decir, desde las proteínas en general, lo cual dio paso a toda la investigación presentada en esta sección.

---

[21] Diseño de interfaces de soluciones que incluyen algoritmos de predicción de estructuras de proteínas.

[22] Técnicas para la evaluación de la usabilidad.

Capítulo 4. Adaptación de algoritmos de predicción de estructuras terciarias de proteínas en general

4.1 Introducción

En este capítulo se expondrán los resultados alcanzados para el primer objetivo específico de este proyecto de tesis[23]. Este primer objetivo busca identificar un algoritmo dedicado a la explotación de los datos existentes de estructuras primarias de proteínas en general con el fin de predecir sus estructuras tridimensionales para, posteriormente, adaptarlo a la predicción de estructuras terciarias de proteínas repetidas. Cabe mencionar que las proteínas repetidas son grupos de familias de proteínas muy extendidas en la naturaleza que cuentan con características particulares (MSCA & RISE, 2018) que se deben tener en cuenta para poder lograr la predicción de sus estructuras.

4.2 Resultados alcanzados

El cumplimiento del primer objetivo específico del presente proyecto de fin de carrera dependerá del logro de tres resultados esperados definidos a detalle en el apartado 1.2 Objetivos. En esta sección, se presenta un resumen de cada uno de ellos y del proceso que se realizó para poder obtenerlos.

4.2.1 Lista de algoritmos de predicción de estructuras terciarias de proteínas en general a partir de su secuencia de aminoácidos

Para proceder con la búsqueda de los algoritmos de predicción de estructuras terciarias, se vio la necesidad de establecer la delimitación de los tipos de algoritmos que deberían incluirse. Con ello se determinó que el alcance de la búsqueda incluiría algoritmos basados en la aplicación de técnicas de aprendizaje profundo y que cuenten con una arquitectura “end-to-end”.

Este primer aspecto está fundamentado en el crecimiento exponencial de la cantidad de datos de secuencias de aminoácidos que se evidencia hoy en día gracias al avance de distintos métodos genómicos (Makigaki & Ishida, 2020). El incremento de estos datos tanto para proteínas en general como para las proteínas repetidas da pie a que se apliquen este tipo de técnicas de inteligencia artificial.

[23] Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.

El segundo aspecto relacionado a la arquitectura del algoritmo corresponde a los datos de entrada y datos de salida de los algoritmos. Una arquitectura “end-to-end” hace referencia a los métodos que logran que a partir de la secuencia de aminoácidos de una proteína como dato de entrada se obtengan las coordenadas 3D de las estructuras terciarias (Laine et al., 2021). Precisamente, esto es lo que se tiene como objetivo trasversal dentro del presente proyecto.

A partir de lo mencionado se capturaron dos algoritmos capaces de explotar los datos existentes de estructuras primarias: DMPfold (Greener et al., 2019) y trRosetta (Yang et al., 2020). Estos dos métodos destacan sobremanera dado que han introducido características novedosas a sus procedimientos, así, los algoritmos que serán analizados son alternativas mejoradas de versiones anteriores. Ambos incluyen redes neuronales profundas para la predicción de mapas de distancias entre los residuos, en adición a los mapas de contacto, que por general son binarios. Asimismo, respecto al primer algoritmo se destaca la aplicación de los ángulos de torsión y la predicción de los enlaces de hidrógeno, mientras que para el segundo algoritmo se resalta el ingreso de las coordenadas de orientación espacial en parejas de residuos de la proteína en cuestión.

El documento con el reporte de los algoritmos identificados capaces de realizar la predicción de estructuras terciarias de proteínas en general a partir de sus secuencias de aminoácidos se encuentra en el Anexo D. Este documento contiene información detallada de la delimitación de la búsqueda, la presentación de los algoritmos identificados y la descripción de su funcionamiento en general, así como la información sobre sus datos de entrada y de salida, modo de procesamiento, recurso computacional y tiempo requerido.

La verificación de este resultado corresponde al cumplimiento de la cantidad de algoritmos identificados, en cuyo caso debe ser mayor o igual a dos. Asimismo, se debe especificar el detalle de cada uno de los algoritmos en base a los aspectos descritos anteriormente. Ambos indicadores objetivamente verificables se cumplen acorde a lo especificado en el Capítulo 1 de este proyecto.

4.2.2 Algoritmo seleccionado y planteamiento de modificaciones necesarias

Una vez que se han identificado los algoritmos capaces de explotar los datos existentes de proteínas en general, se procede a seleccionar el que se considere más ventajoso para poder adaptarlo a las particularidades de las proteínas repetidas. Para ello, se requiere de la comparación de las características y funcionalidades de cada uno de los algoritmos, obtenidas a partir de la verificación de sus funcionamientos. A continuación, en la Tabla 20, se presenta un cuadro comparativo entre las distintas características de los algoritmos seleccionados DMPfold y trRosetta, acorde al presente resultado esperado.

Característica	DMPfold	trRosetta
Técnica de inteligencia artificial	Redes neuronales profundas convolucionales	Redes neuronales profundas residuales
Factor diferenciador	Predice los enlaces de hidrógeno de la cadena principal, los ángulos de torsión y los límites de distancias interatómicas. Utiliza mapas de contacto y datos de covarianza como datos de entrada.	Predice las coordenadas de orientación espacial, los mapas de distancias y los mapas de contacto
Técnica de modelado	CNS	PyRosetta
Datos de entrada	Secuencia de aminoácidos	Secuencia de aminoácidos
Datos de salida	Coordenadas 3D de estructura (archivo PDB)	Coordenadas 3D de estructura (archivo PDB)
Lenguaje de programación	C y Python	Python y Bash
Versión de Python	3.7	3.6
Librería de Aprendizaje de Máquina	PyTorch 0.4 o posterior (Evaluado con PyTorch 1.9)	Tensorflow 1.14 PyTorch 1.4
Complejidad de instalación	Alta	Alta
Recurso computacional requerido	Requiere de la capacidad de una computadora estándar de un núcleo y de memoria suficiente para la descarga de algunas bases de datos	Requiere de servidores con GPU y de memoria suficiente para la descarga de distintas bases de datos

Recursos de libre acceso	El código fuente, la documentación completa y el modelo entrenado de la red neuronal. Los recursos adicionales que se utilizan también son de libre acceso.	Los alineamientos múltiples de secuencias usadas, los archivos fuente para la predicción de geometrías interresiduales y el protocolo para el modelado y la generación de estructuras
Resultado de verificación de predicción	Éxito (Genera 2 estructuras en formato PDB)	Sin éxito (No genera estructuras)
Errores obtenidos	Error no significativo por carencia de la licencia de Modeller	Error en el refinamiento de estructuras. Arrastre de error para la selección de uno final
Tiempo promedio de ejecución	~36 minutos	~35 minutos
Secuencia de proteína de prueba[24]	PF10963 Escherichia coli UPI0006A5DD14	T1078 Trichoderma virens
Cantidad de residuos	82 residuos	138 residuos

Tabla 20. *Cuadro comparativo entre algoritmos seleccionados: DMPfold y trRosetta.*

El reporte comparativo de los algoritmos identificados capaces de realizar la predicción de estructuras terciarias de proteínas en general a partir de sus secuencias de aminoácidos se encuentra en el Anexo E. En ese documento se encuentra una breve descripción de los algoritmos, además de la explicación detallada del proceso de verificación de su funcionamiento y un cuadro comparativo de las características generales obtenidas a partir de esta.

Después de que es posible reconocer las características más resaltantes de cada uno de los algoritmos, validar su funcionamiento y evaluar el procedimiento que siguen para lograr la predicción de estructuras terciarias, se dio paso a la selección de uno de ellos: el algoritmo DMPfold (Greener et al., 2019). Los criterios más influyentes dentro de la toma de decisión fueron: la eficacia del algoritmo luego de seguir las instrucciones para su instalación y uso, el empleo de librerías actualizadas y el recurso computacional requerido. El Anexo F contiene la descripción a detalle de dichos criterios de selección, además describe cuatro modificaciones propuestas para lograr la adaptación del algoritmo hacia el entorno de las proteínas repetidas.

[24] La secuencia de prueba usada para verificar el funcionamiento de cada algoritmo se obtiene de su propio repositorio.

Cabe resaltar que las cuatro modificaciones formuladas aprovechan la particularidad más reconocible de las proteínas repetidas: los patrones de repetición dentro de una misma familia (Kajava, 2012). Es por esto que el algoritmo trabajará, en primera instancia, con fragmentos de secuencias de aminoácidos obtenidas a partir de la base de datos PFAM, reconocida por poseer colecciones de familias de proteínas (Mistry et al., 2021). En segunda instancia, se trabajará con secuencias limitadas de proteínas repetidas. Esta información se obtendrá de la base de datos Uniprot (UniProt Consortium, 2021b).

En particular, una propuesta de mejora se enfoca en los resultados generados para el usuario. Así, se sacará el mayor provecho de las herramientas disponibles en el contexto de las proteínas repetidas, para poder identificar las unidades de repetición de una estructura predicha por el algoritmo. Se propone esta mejora teniendo en cuenta que el presente proyecto tiene principal interés en el modelamiento de la estructura terciaria de los fragmentos pertenecientes a las familias de proteínas repetidas.

La verificación de este resultado consiste en la presentación de la comparación de las características de los algoritmos identificados, así como de los recursos que utiliza, ya sean los datos de entrada, datos de salida o datos intermedios utilizados para cumplir sus objetivos. La información pertinente ha sido registrada de forma resumida en la Tabla 20. En el mismo sentido, como segunda verificación del resultado, se presentó un reporte que contiene el análisis y justificación de la elección del algoritmo, además de proponer al menos dos modificaciones para adaptarlo a las particularidades de las proteínas repetidas. Así, ambos indicadores objetivamente verificables, especificados en el Capítulo 1 de este proyecto, han sido cubiertos.

4.2.3 Adaptación del algoritmo para la predicción de estructuras terciarias de proteínas repetidas

En la sección anterior se propusieron una serie de modificaciones para adaptar el algoritmo previamente seleccionado: DMPfold (Greener et al., 2019). Este fue desarrollado e implementado bajo una perspectiva general, aplicable para todas las proteínas existentes. No obstante, esa universalización reduce la posibilidad de obtener resultados enfocados a los requerimientos de los usuarios interesados en las proteínas repetidas. Estas, en realidad, son grupos familias de proteínas, que cuentan con características particulares. En ese sentido, en este apartado, se explicará el proceso de adaptación del algoritmo de predicción seleccionado para su aplicación en torno a

las proteínas repetidas. Asimismo, se explicará de forma general, el procedimiento de pruebas realizado para evaluar el correcto funcionamiento del algoritmo adaptado.

4.2.3.1 Aplicación de las modificaciones propuestas al algoritmo seleccionado

La aplicación de las cuatro modificaciones propuestas al algoritmo seleccionado se realizará de forma secuencial. De forma general, este proceso consistirá en la creación de scripts que se complementen y consoliden las diversas etapas del algoritmo propuesto, incluyendo al proceso de predicción llevado a cabo por el algoritmo inicialmente seleccionado. Esto para poder obtener los resultados que respondan a los requerimientos de las personas interesadas en las proteínas repetidas.

La primera de las modificaciones se enfoca el dato de entrada del proceso. Originalmente el algoritmo requiere del ingreso de tres archivos (.fasta, .21c y .map) para poder llevar a cabo la predicción. Lo que se está planteando en primer lugar es trabajar con los identificadores de las familias de las proteínas, teniendo en cuenta que es mucho más conveniente analizar a las proteínas repetidas en base a la familia a la que pertenecen. El identificador se obtiene de la base de datos PFAM y, a partir de esta, como parámetro en un servicio de esta plataforma, se obtiene un archivo de texto plano que contiene una serie de fragmentos de proteínas que se agrupan en esa familia, el identificador uniprot de las secuencias donde han sido identificados y los límites de posición de cada fragmento en cada secuencia relacionada.

Al pertenecer a la misma familia se entiende que comparten características similares. Esta similitud es más significativa respecto a la estructura debido a la alta tasa de degeneración en la secuencia de las proteínas repetidas, es decir, los efectos de la evolución respecto a la estructura de la proteína, ya que es sabido que la conformación tridimensional de este grupo de familias de proteínas se conserva más que la propia secuencia (Deng et al., 2018; Paladin et al., 2021).

Una vez que se cuenta con el archivo de fragmentos se procede a depurarlo, discriminando los fragmentos que hayan sido identificados dentro de una secuencia de proteína con una estructura terciaria predicha de los que no. Estos últimos serán los fragmentos que continuarán en el proceso. Para realizar esta tarea se utilizará un servicio puesto a disposición general por parte del Banco de Datos de Proteínas en Europa (EBSC PDB, por sus siglas en inglés), el cual requiere del ingreso del identificador de acceso de la secuencia relacionada al fragmento. No obstante, solo se

cuenta con el identificador Uniprot, obtenido del archivo de la familia de proteínas del inicio.

La base de datos Swiss-Prot del consorcio Uniprot es un repositorio de información biológica de secuencias de proteínas de donde se puede obtener la relación entre el identificador Uniprot y el identificador de acceso. Por ello, se decidió utilizar sus datos como una especie de traductor de identificadores. Así, a modo de incrementar la eficiencia de esta actividad, se realizó un filtrado de la información de esa base de datos, para eliminar todos los datos adicionales no concernientes al objetivo del presente proyecto, y se generó un nuevo fichero con dos tipos de datos: los datos requeridos en la tarea descrita en el párrafo anterior.

Utilizando el fichero mencionado, se conseguirá la traducción del identificador de la secuencia, cuyo resultado será ingresado al servicio de RSCB PDB. Este api brinda información estructural e información afín por cada secuencia de proteínas. Es por ello que, ingresando el dato mencionado, se puede validar si se cuenta con estructuras terciarias relacionadas a esa secuencia o no. Posterior a esta primera etapa se seguirán dos caminos, uno de los cuales constituye a una segunda modificación propuesta.

El primer camino implica la generación de un archivo en formato fasta por cada uno de los fragmentos sin estructura identificados en el paso anterior. El algoritmo adaptado los generará y les asignará el prefijo *'is'* en su nombre ya que se trata de lo que se ha denominado como subsecuencias independientes, '*independent subsequences'*. Por otro lado, el segundo camino conlleva dos pasos adicionales: la búsqueda del menor límite inferior y el mayor límite superior de la posición de los fragmentos de las proteínas repetidas en cada secuencia en la que han sido identificados. Estos datos se encuentran en el archivo PFAM del inicio del proceso.

Posterior a ello, utilizando un servicio de Uniprot se obtendrán las secuencias completas en las que se encuentran dichos fragmentos, las cuales se reducirán a una subcadena representativa de la proteína repetida, en base a los límites capturados previamente. El algoritmo generará un archivo en formato fasta por cada subcadena obtenida y les asignará el prefijo *'nr'* en su nombre dado que se trata de lo que se ha denominado en este proyecto como una nueva secuencia representativa, *'new representative sequences'*.

Finalmente, para cada uno de los archivos fasta generados se ejecutará el script de conversión de secuencias a mapas de contacto. Este genera dos archivos intermedios

con extensión 21c y map, los cuales corresponden al mapa de contacto y a los datos de covarianza de la secuencia de aminoácidos ingresada. En este punto se cuentan con tres ficheros por cada fragmento “is” y por cada secuencia representativa “nr”: los dos últimos archivos generados más el fichero en formato fasta, los cuales funcionarán como input para el algoritmo inicialmente seleccionado.

La tercera y cuarta modificación propuestas corresponden a un proceso de generación de resultados a partir de la predicción y a la evaluación de la misma, respectivamente. Estas, junto a las anteriores, se explican a detalle en el documento del Anexo I. Este documento contiene, además, el diagrama del algoritmo adaptado con su respectiva descripción funcional y código fuente. Con ello, se verifica el cumplimiento de la primera sección de indicadores propuestos para el tercer resultado esperado del primer objetivo específico de este proyecto de tesis.

4.2.3.2 Procedimiento para la ejecución de pruebas del algoritmo adaptado

Llegado a este punto, el algoritmo se encuentra listo para realizar la predicción de las estructuras terciarias de las proteínas repetidas a partir de sus estructuras primarias. En tal sentido, para proceder a evaluar el performance del algoritmo adaptado, se requiere de la captura de datos de entrada que permitan llevar a cabo un ‘*benchmarking’* entre los resultados obtenidos y los resultados esperados.

Se debe tener en cuenta que, acorde a las características de las proteínas repetidas, se considera muy significativo para la evaluación de una predicción al resultado de la comparación entre una estructura predicha por el algoritmo y una estructura previamente almacenada en el Banco de Datos de Proteínas, la cual contenga a uno o más fragmentos de la misma familia de proteínas repetidas que la secuencia ingresada en el algoritmo adaptado. Ese será el lineamiento que se seguirá para realizar la evaluación del performance del algoritmo adaptado. Con ello se plantea al alineamiento estructural como la actividad base de la evaluación. Así, esta comparación entre lo predicho y lo esperado se realizará de forma cualitativa y cuantitativa, en base a herramientas de visualización PyMol (PyMOL, 2021) y la herramienta de alineamiento estructural TM-align (Zhang & Skolnick, 2005). Cabe mencionar que ambas son herramientas disponibles para su descarga a través de la web. No obstante, en el caso de PyMol, se requiere de una licencia para poder activarla. Para este proyecto se obtuvo una licencia académica bajo una solicitud enviada a los propietarios de la herramienta.

Para obtener la información con la que se ejecutará el algoritmo adaptado, se creó un procedimiento que permite capturar los datos requeridos para la evaluación. Este procedimiento se explica de forma detallada en el documento de reporte de pruebas del algoritmo adaptado, el cual puede encontrarse en el Anexo I. En forma general, el procedimiento de captura de datos comprende el uso de una serie de recursos web disponibles gratuitamente para cualquier interesado. Involucra varios servicios de diversas fuentes como RepeatsDB, Uniprot, Pfam y RCSB Protein Data Bank. El objetivo es capturar datos de proteínas repetidas pertenecientes a una categoría y una topología específica, así como la información de las familias a las cuales pertenecen sus unidades de repetición. Adicionalmente, el procedimiento comprende la captura de estructuras terciarias previamente predichas. Estas estructuras terciarias serán seleccionadas en base a las secuencias obtenidas en los primeros pasos de este procedimiento. Para la presente evaluación, se seleccionaron diez familias de proteínas repetidas. Cada una de estas familias cuenta con distinta cantidad de secuencias o, como se están denominando en este proyecto, fragmentos. En la Tabla 21, se puede observar el código de acceso PFAM, la descripción, el tipo, el número de secuencias por cada tipo de generación y la longitud promedio de las secuencias pertenecientes a las familias de proteínas repetidas seleccionadas para la evaluación del algoritmo adaptado.

Catálogo de familias de proteínas seleccionadas para pruebas					
Código PFAM	Descripción	Tipo	Número de secuencias		Longitud promedio de cada secuencia
			Semilla	Completo	
PF00023	Ankyrin repeat	Repeat	1062	23676	33.60
PF00514	Armadillo/beta-catenin-like repeat	Repeat	197	85142	40.70
PF16186	Atypical Arm repeat	Repeat	139	5093	52.50
PF18770	Armadillo tether-repeat of vescicular transport factor	Repeat	29	700	60.30
PF18773	Importin 13 repeat	Repeat	13	679	39.30
PF08569	Mo25-like	Repeat	117	2902	286.90
PF01239	Protein prenyltransferase alpha subunit repeat	Repeat	619	16557	31.70
PF00806	Pumilio-family RNA binding repeat	Repeat	1115	57203	33.10

PF08238	Sel1 repeat	Repeat	167	133325	35.30
PF03377	TAL effector repeat	Repeat	41	373	33.50

Tabla 21. Catálogo de familias de proteínas seleccionadas para realizar las pruebas del algoritmo adaptado

Para cada familia de proteínas se identificará una serie de cinco estructuras tridimensionales alojadas en el Banco de Datos de Proteínas. Estas estructuras están relacionadas a una o más familias seleccionadas, dado que los fragmentos de proteínas repetidas o unidades de repetición pertenecientes a la familia en cuestión deberán estar presentes en la secuencia de la cual se ha obtenido la estructura terciaria a comparar.

Se realizará la predicción de las estructuras de las proteínas repetidas o las unidades de repetición de las familias de proteínas seleccionadas y se seleccionarán al menos tres predicciones de forma aleatoria por cada una de estas familias. Estas predicciones serán comparadas con las cinco estructuras seleccionadas por cada familia.

Las pruebas fueron enteramente cuantitativas y su veredicto se basó en los resultados del alineamiento estructural realizado con TM-align. Esta herramienta retorna un valor entre 0 y 1 que califica el alineamiento. Un valor superior o igual a 0.5 indica generalmente un mismo plegado estructural entre las proteínas ingresadas. Por otro lado, las pruebas contarán con una sección cualitativa que corresponde más a una verificación de los datos obtenidos en la sección anterior. Esta prueba será realizada en base a la similaridad visual que se pueda reconocer entre dos estructuras. Para ello se utilizará la herramienta PyMol.

Los resultados de la evaluación realizada fueron positivos en tanto se logró evaluar una cantidad significativa de las familias seleccionadas. La explicación a detalle y el análisis de las pruebas realizadas se encuentran en el Anexo I. Con ello, se verifica el cumplimiento del indicador planteado para esta segunda parte del resultado esperado número tres del primer objetivo del presente proyecto de tesis. Por último, cabe mencionar que el proceso presentado en este capítulo ha sido validado por juicio experto en bioinformática y el documento del reporte de las pruebas realizadas al algoritmo adaptado ha sido verificado y aprobado al 100% por este mismo experto. El acta de validación se encuentra como un apéndice dentro del Anexo I.

4.3 Discusión

A modo de síntesis, en el presente capítulo se llevó a cabo el cumplimiento del primer objetivo específico de este proyecto de fin de carrera. Este objetivo corresponde a la

adaptación de un algoritmo de predicción de estructuras terciarias de proteínas en general a partir de su estructura primaria. Esto con el fin de aplicarlo en proteínas repetidas. En esta primera etapa se alcanzaron los tres resultados esperados. En primer lugar, se presenta una lista de algoritmos identificados, los cuales son capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general. En segundo lugar, se prosiguió a la selección justificada de uno de los algoritmos identificados y al planteamiento de diversas modificaciones que permitan aprovechar las características de las proteínas repetidas. Por último, se inició la aplicación de las modificaciones propuestas con lo cual se obtuvo un algoritmo adaptado a las cualidades de las proteínas repetidas. El proceso de adaptación del algoritmo seleccionado se fundamenta primordialmente en la afinidad de las funciones y características entre las proteínas repetidas que comparten la misma familia. En este punto, se destaca el alineamiento casi perfecto de sus unidades de repetición. Un segundo aspecto tomado en cuenta para la proposición de modificaciones es la alta tasa de degeneración en su secuencia, por el cual, a pesar de la variación de los aminoácidos en una proteína, la estructura de esta se mantiene (Deng et al., 2018; Hirsh et al., 2016; Parmeggiani & Huang, 2017). Con esto último se justifica el agrupamiento de las proteínas en familias, dado que la evolución y diversos factores internos conllevan la variación de las secuencias y de su estructura, aunque este último no se ve sometido a grandes cambios en comparación. Consecutivamente, la existencia de estructuras terciarias similares dará pie a la ejecución de funciones también semejantes.

Adicionalmente, cabe mencionar que en los resultados alcanzados se ha considerado a la adición de actividades dentro del flujo de procesamiento de predicción como una modificación válida. Esto en base a que, en concordancia al marco conceptual desarrollado en el Capítulo 2. Marco Conceptual, se entiende el término *'algoritmo'* como una secuencia de pasos finitos que describen cómo llevar a cabo un proceso con el fin de encontrar la solución a un problema (Lambert & Osborne, 2018). Por último, es importante resaltar que el cumplimiento de este primer objetivo representa un avance significativo en el desarrollo del presente proyecto de tesis, debido a que constituye un alto porcentaje de esfuerzo, además de ser clave tanto para el cumplimiento de los requisitos definidos en el segundo objetivo como para el inicio del tercero.

Capítulo 5. Diseño e implementación de una interfaz para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

5.1 Introducción

El presente capítulo abarcará la presentación de los resultados pertenecientes al segundo objetivo específico de este proyecto de tesis[25] y que, finalmente, fueron alcanzados con éxito. Dicho objetivo busca implementar una interfaz que contenga un flujo de actividades directo y efectivo para poder realizar la predicción de estructuras terciarias de proteínas repetidas a partir de sus secuencias de aminoácidos. Para poder llevarlo a cabo se seguirán buenas prácticas del desarrollo de software, así como la creación de documentación pertinente que funcione como apoyo para lograr los objetivos. Se espera que, posteriormente, la interfaz se integre al algoritmo adaptado en el capítulo anterior para así conformar una herramienta bioinformática de predicción de estructuras 3D de proteínas repetidas.

5.2 Resultados alcanzados

El cumplimiento del segundo objetivo específico del presente proyecto de fin de carrera dependerá del logro de tres resultados esperados definidos a detalle en el apartado 1.2 Objetivos. A continuación, se presentará un resumen de cada uno de ellos y del proceso que se realizó para poder obtenerlos.

5.2.1 Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta

La etapa de análisis del ciclo de vida de software es primordial y constituye la definición de aspectos importantes para el propio desarrollo. Una de las actividades incondicionales a llevar a cabo en esta sección corresponde al establecimiento de los requisitos funcionales y no funcionales de la herramienta propuesta. En el mismo sentido, cabe rescatar que el segundo resultado de la revisión sistemática realizada en el Capítulo 3. Estado del Arte correspondía a la identificación de servicios relacionados a la herramienta que se está proponiendo en el presente proyecto de fin de carrera. Con ello se esperaba reconocer aquellas funcionalidades que son necesarias para llevar a

[25] Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

cabo la predicción de estructuras de proteínas y que actualmente están siendo aceptadas por los usuarios bioinformáticos. Se hallaron distintos requisitos; muchos de ellos se complementaban y muchos otros se contradecían. Es a partir de esa captura de información que se han definido un total de cuarenta y un requisitos tanto funcionales como no funcionales, que serán contemplados en la herramienta de predicción que propone el presente proyecto. El documento que contiene la lista de requisitos, el tipo de requisito (Funcional o No Funcional) y el tipo de exigencia (Deseable o Exigible) se encuentra en el Anexo G. Asimismo, en el apéndice del anexo, se presenta el acta entregada por el experto en bioinformática, quien brindó la aprobación de dicho documento. Este resultado cuenta con dos indicadores objetivamente verificables: la inclusión de al menos diez requisitos funcionales y no funcionales de la herramienta, y la aprobación del 100% de estos a través del juicio experto de un investigador en el área de bioinformática.

5.2.2 Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

El segundo resultado alcanzado perteneciente al segundo objetivo específico del presente proyecto corresponde al prototipo de la interfaz de la herramienta de predicción. Este cuenta con un indicador que lo verifica: el reporte con el mockup de la interfaz de la herramienta. El reporte incluye las pantallas del prototipo de alta fidelidad del proyecto web y la descripción de su navegación, es decir, el detalle de la relación entre ellas. Además, contiene el diagrama de flujo de la herramienta, con el cual se especifican los pasos que un usuario deberá seguir para poder interactuar con la plataforma y obtener la predicción de la estructura terciaria de la proteína repetida deseada. A continuación, en la Figura 7, se presenta la pantalla principal del prototipo de la interfaz propuesta desarrollada con la herramienta Figma. Esta será la primera pantalla que el usuario visualizará al ingresar a la plataforma web y desde la cual se podrá tener acceso al algoritmo de predicción adaptado correspondiente al primer objetivo del proyecto. La aplicación de dicho algoritmo es su función más importante, por ello, la vista principal permite su uso desde la primera interacción. Así, esta pantalla contiene los campos requeridos para poder enviar una solicitud de predicción de una proteína repetida. Desde este formulario se ingresarán los datos de entrada de la predicción: el identificador PFAM de una familia de proteínas repetidas o la secuencia de aminoácidos de una proteína repetida, esta última ya sea directamente desde el campo de entrada de texto o desde un archivo. En este caso, la herramienta solo

permitirá el ingreso de archivos con formato Fasta, Stockholm o un archivo de texto simple con extensión .fasta, .sto o .txt, respectivamente. Asimismo, esta primera pantalla pondrá a disposición del usuario dos datos de entrada de ejemplo, correspondientes a los dos tipos de datos permitidos para realizar la solicitud de predicción, es decir, un identificador PFAM o la secuencia de una proteína. De esta manera, con pocos *clicks*, el usuario podrá hacer uso del servicio de predicción denominado como DeepReSPred, Deep Repeat protein Structure Predictor.

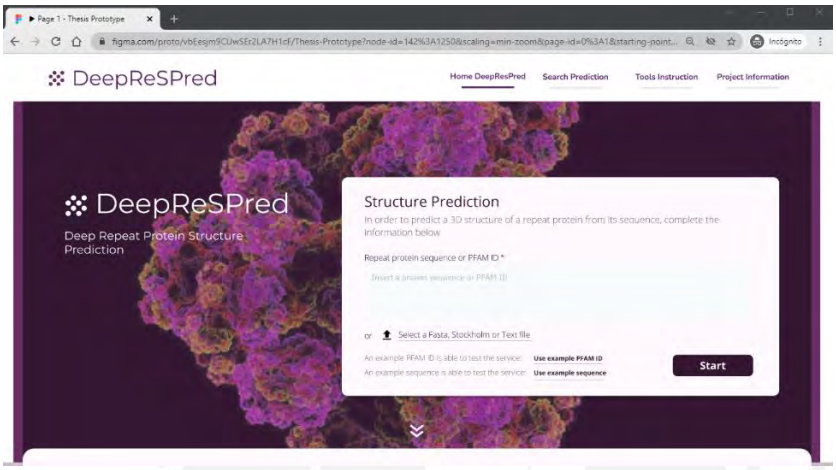

Figura 7. *Pantalla principal del prototipo de alta fidelidad de la interfaz de la herramienta propuesta. (Elaboración propia).*

El prototipo de la interfaz también incluye ciertas interacciones con el usuario como los mensajes de validación de los datos ingresados. Estos mensajes se muestran cuando se verifica que los datos de entrada no corresponden a lo admitido por el algoritmo. Por ejemplo, se verificará que los caracteres ingresados en el campo de entrada de texto, en caso de corresponder a la secuencia de una proteína, estén acorde a la simbología de los aminoácidos existentes. Se puede revisar la simbología de los aminoácidos permitidos en el apartado de la descripción del término *estructura primaria de proteína* del Capítulo 2. Marco Conceptual.

El prototipo muestra cuatro divisiones secundarias que conformarán a la interfaz de la plataforma: Estas se ubicarán en la zona inferior de la pantalla principal y son las siguientes: la pestaña de búsqueda de solicitudes de predicción ingresadas, la pestaña de instrucciones de uso de la herramienta, la pestaña de información del proyecto y la pestaña de bibliografía. Una segunda vista crucial del prototipo de la herramienta pertenece a la búsqueda de la solicitud de predicción registrada, tal como se observa en la Figura 8. Esta pantalla mostrará los datos ingresados, el estado de la solicitud de

predicción y los resultados de esta. Incluye también una sección donde se podrá visualizar la estructura predicha.

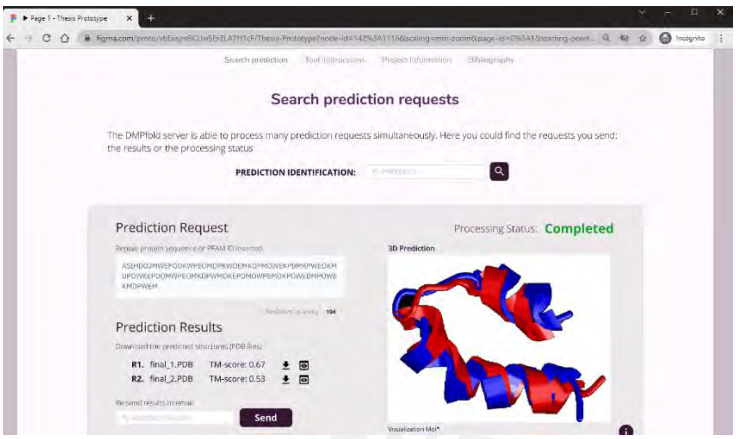

Figura 8. Pantalla de la sección de búsqueda de solicitudes de predicción del prototipo de alta fidelidad de la interfaz de la herramienta propuesta. (Elaboración propia).

El reporte completo contiene más detalle de las pantallas diseñadas como parte del prototipo de alta fidelidad de la interfaz de la herramienta propuesta. Además, presenta el diagrama de flujo de las actividades de la herramienta desarrollado con la notación BPMN con la herramienta Diagramas.net. Por último, contiene el acta de validación elaborado por un experto. Todo ello se podrá encontrar en el Anexo H. Con ello, el documento verifica el cumplimiento del resultado esperado definido en el Capítulo 1 del presente proyecto.

5.2.3 Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas

El último resultado alcanzado del objetivo específico cubierto en el presente capítulo del proyecto corresponde al desarrollo de la interfaz de la herramienta propuesta. Este se verifica por medio de un reporte descriptivo de la interfaz desarrollada y un video de su navegación.

Las herramientas, métodos y procedimientos utilizados para su elaboración fueron diversos. En primer lugar, se requirió de un repositorio en GitHub que aloje el código fuente de la interfaz. Este se actualizaba al final de cada avance significativo en el desarrollo con el fin de almacenar una copia de respaldo del proyecto. En segundo lugar, se incluyeron algunas librerías que facilitaron la transformación del prototipo hacia un proyecto web real. Algunas de ellas fueron: Bootstrap, para la integración de elementos GUI preconfigurados; Fonts, para el uso de la colección de fuentes seleccionadas; e Icons fontawesome para poder incluir iconos a las pantallas de forma fácil e intuitiva.

En ese sentido, los lenguajes de programación aplicados en esta etapa del proyecto fueron JavaScript, CSS y HTML, este último denominado como un lenguaje de etiquetas. En adición, se incluyó el framework Vue que fue una pieza clave tanto para el desarrollo eficiente y ordenado de la interfaz como para poder desarrollar la interacción entre las vistas de la interfaz web. Cabe mencionar que las pantallas de la interfaz de la herramienta propuesta se han desarrollado en base a lo especificado en el prototipo de alta fidelidad. Así, la vista principal de esta desplegada en un entorno local se observa en la Figura 9. En este punto del proyecto de fin de carrera, aún no se cuenta con la integración con el algoritmo adaptado en el objetivo específico anterior. Es por ello que para la grabación del video de la navegación de la interfaz se han configurado situaciones de ejemplo que surgirán en base a la generación de números aleatorios. La videograbación se encuentra disponible en la nube y se podrá acceder a ella a través del enlace especificado a continuación:

Grabación de la interacción de la interfaz desarrollada de la herramienta propuesta

Esta grabación de la interacción de la interfaz desarrollada se complementa con un reporte descriptivo de cada una de sus funcionalidades. Este reporte se puede encontrar en el Anexo K del presente documento. Con la presentación de la grabación y el reporte mencionado se da como completado este resultado esperado.

5.3 Discusión

El presente capítulo contiene la descripción detallada del proceso seguido para lograr el cumplimiento del segundo objetivo específico de este proyecto de tesis. Este objetivo tiene como finalidad el diseño y la implementación de una interfaz para la herramienta propuesta. Esta interfaz se integrará al algoritmo adaptado en el capítulo anterior, con lo cual se alcanzaría el objetivo general del proyecto de forma exitosa.

La superación de esta segunda etapa depende de la consecución de tres resultados esperados. El primero de ellos corresponde a la definición de un catálogo de requisitos funcionales y no funcionales para el desarrollo de la herramienta objetivo. Para conseguirlo se tomó en cuenta el contexto en el que se desenvuelven otros recursos bioinformáticos afines a la que se propone. El reconocimiento de este factor, así como de distintas necesidades de los usuarios de estas herramientas se obtuvo a partir de la revisión sistemática llevada a cabo en capítulos anteriores. El segundo resultado alcanzado es el diseño de un prototipo de la interfaz de la herramienta propuesta. El hecho de que el mockup sea de alta definición requirió de la definición de algunos

estándares básicos de entornos gráficos como la paleta de colores y las fuentes a utilizar. Esto en adición a la especificación de los pasos que un usuario debe seguir para interactuar con la herramienta. Cabe mencionar que también se utilizaron las respuestas de la revisión sistemática relacionadas a la aplicación de lineamientos de usabilidad en recursos bioinformáticos. Además de poner en práctica algunos recomendados como las heurísticas de Nielsen para poder brindarle una mejor experiencia al usuario. Se espera que la aplicación de estos criterios dé buenos resultados en la evaluación de usabilidad que se realizará en el siguiente capítulo.

El tercer resultado alcanzado abarca la transformación de las vistas diseñadas como parte del prototipo de la interfaz hacia un entorno real. Para la implementación de esta sección se utilizaron todas las herramientas disponibles, y de las cuales se tenía conocimiento, para el desarrollo web.

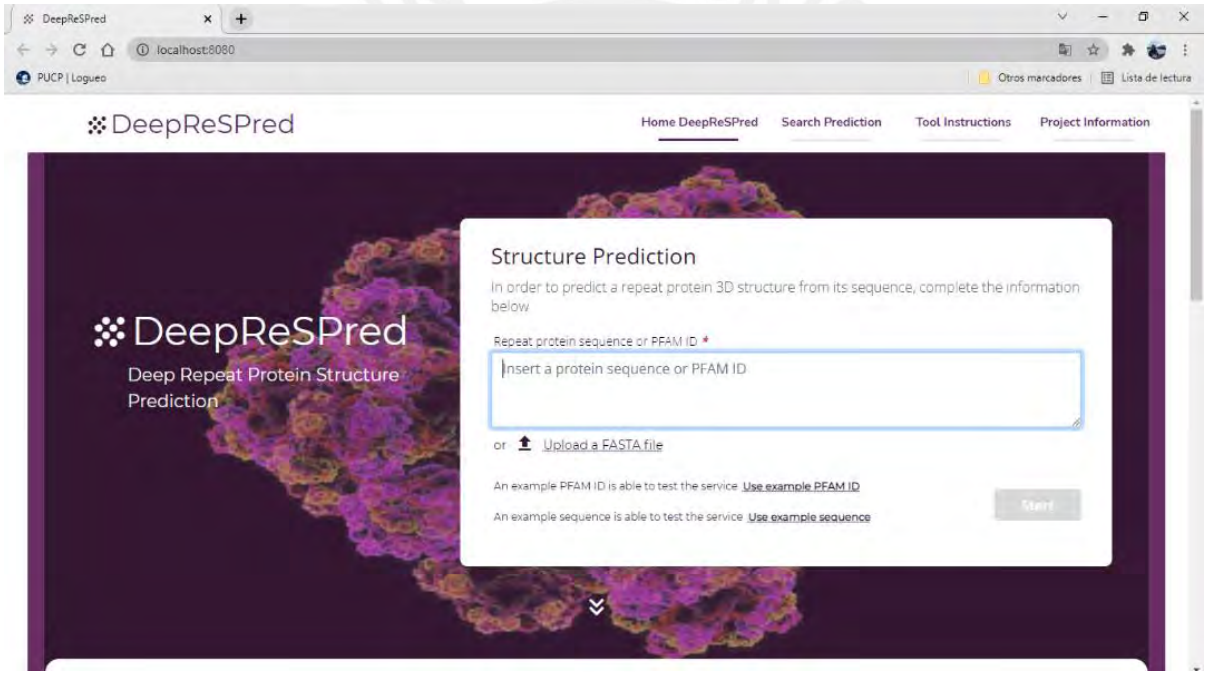

Figura 9. *Pantalla principal de la interfaz desarrollada de la herramienta propuesta. (Elaboración propia).*

Con el cumplimiento de los resultados esperados de este objetivo se intenta fomentar la inclusión de los criterios de usabilidad en el proceso de desarrollo de aplicaciones de uso científico, en particular, a las bioinformáticas. Así, se dará inicio a un sinceramiento respecto a la usabilidad de los recursos, dado que, según la revisión sistemática, muchos de ellos se autodenominan como amigables para el usuario cuando no siempre es correcto afirmarlo.

Capítulo 6. Integración del algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

6.1 Introducción

En este capítulo se presentarán los resultados alcanzados pertenecientes al tercer objetivo específico de este proyecto de fin de carrera[26]. Dicho objetivo busca la integración de los resultados obtenidos en los dos objetivos específicos previos. Es decir, busca implementar de forma integra la herramienta bioinformática propuesta inicialmente en el objetivo general de este proyecto. Esta herramienta tiene como finalidad explotar la información existente de las estructuras primarias de las proteínas repetidas para poder predecir sus estructuras terciarias. Es por ello que se utilizará el algoritmo de predicción adaptado del primer objetivo específico y se integrará a la interfaz web desarrollada como un resultado esperado del objetivo específico.

6.2 Resultados alcanzados

El cumplimiento del tercer objetivo específico del presente proyecto de tesis dependerá del logro de los tres resultados esperados definidos a detalle en el apartado 1.2 Objetivos. En esta sección, se presenta un resumen de cada uno de ellos y del proceso que se realizó para poder obtenerlos.

6.2.1 Interfaz que permite utilizar el algoritmo adaptado

Previamente al inicio de este capítulo, el algoritmo adaptado es capaz de llevar a cabo un flujo de actividades para poder predecir las estructuras terciarias de las proteínas repetidas a partir de la información de su secuencia de aminoácidos. Asimismo, se cuenta con una interfaz web desarrollada que permite realizar un flujo de predicción de proteínas a través del ingreso de solicitudes. No obstante, esta interfaz aún no puede realizar predicciones puesto que aún no tiene integrado un algoritmo que los realice.

Los desarrollos de cada uno de estos artefactos se han realizado de forma independiente, por lo cual se requiere que se hagan las modificaciones necesarias para

[26] Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

que ambos trabajen de la mano. Ese desarrollo es lo que se realizará para poder lograr el resultado esperado en esta sección.

Así, como parte un proceso de integración que sigue lineamientos ágiles, se ha determinado desarrollar la documentación mínima requerida que sirva de soporte. Con ello, se ha determinado la necesidad de elaborar dos diagramas elementales para esta etapa: el diagrama de arquitectura y el diagrama de relacional de bases de datos.

La Figura 10 muestra el diagrama de la arquitectura a desarrollarse. Como se puede se puede observar, la herramienta finalmente será desplegada en un entorno de la nube, por lo tanto, la arquitectura planteada contiene los diversos recursos de Amazon Web Services a utilizarse. Cabe mencionar que los elementos utilizados corresponden a la capa gratuita de Amazon, tal y como se describió en el documento del plan de proyecto ubicado en el Anexo C.

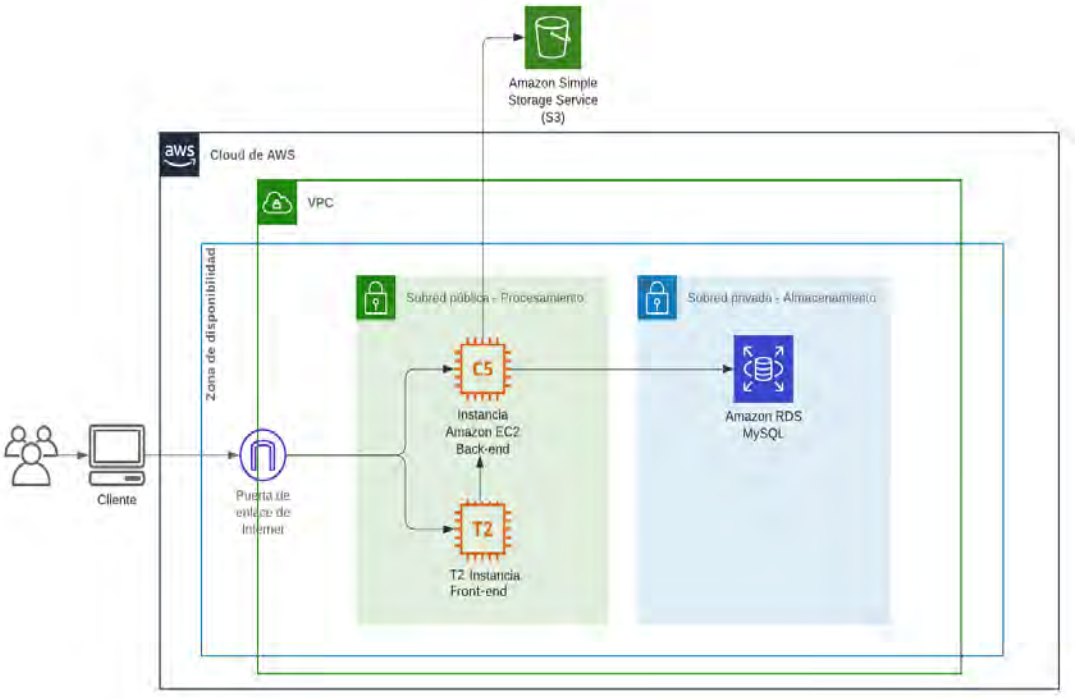

Figura 10. *Diagrama de arquitectura en la nube Amazon Web Services de DeepReSPred. (Elaboración propia).*

El proyecto fue distribuido en dos instancias: Front-end, el cual alojará a la interfaz de la herramienta planteada, y Back-end, el cual alojará tanto a las api's con las que interactúa el front-end como el algoritmo adaptado en la primera parte de este proyecto.

De acuerdo a lo anterior, el front-end será desplegado en una instancia EC2 tipo T2, ya que es un recurso computacional que no recibe mucha carga. Por otro lado, el back-end

se desplegará en una instancia EC2 de tipo C5 ya que es un tipo de recurso optimizado para soportar cargas de trabajo de computación intensiva[27]. En el mismo sentido, se está utilizando el servicio de Amazon S3 con un bucket para alojar los archivos iniciales, intermedios y finales del procesamiento de una predicción de proteínas. La instancia EC2 de back-end será la única que tendrá acceso al bucket. Finalmente, se utilizará una instancia RDS para alojar la base de datos de tipo MySQL correspondiente a los requisitos planteados en el Catálogo de requisitos no funcionales del Anexo G. Esta base de datos será alimentada y consultada por el back-end y guardará coherencia con los datos alojados en el bucket de la plataforma.

Respecto a la base de datos, se ha determinado seguir una distribución de las entidades relacionadas como la que se muestra en la Figura 11. Se puede observar que el diagrama relacional plasma una estructura básica para soportar tanto la información de las solicitudes de predicción como de los archivos que intervienen en el proceso, además de almacenar la información de los accesos a través de los cuales se han enviado las solicitudes de predicción. Esto último teniendo en cuenta el caso de que los administradores de la herramienta consideren habilitar un límite de ingresos de solicitudes por punto de acceso.

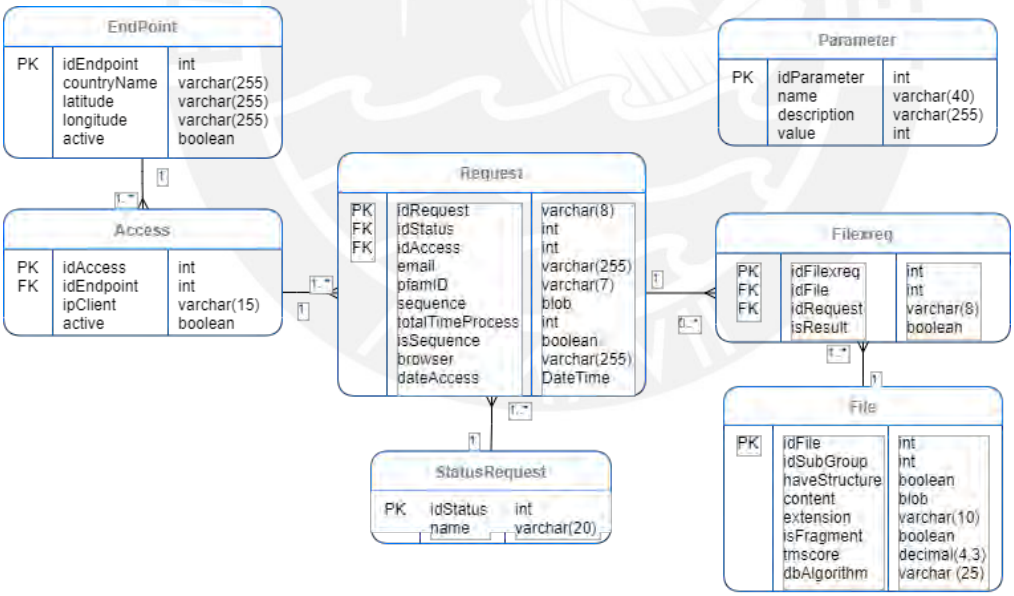

Figura 11. *Diagrama relacional de base de datos DeepReSPred. (Elaboración propia).*

[27] La utilización de los recursos optimizados de Amazon Web Services no restringuen la implementación del ya que como se explica en el apartado de consideraciones adicional del anexox se ha procurado establecer configuraciones aptas para un recurso computacional promedio

Se menciona que la base de datos tendrá coherencia con el contenido del bucket debido a que el back-end consultará a la base de datos por el nombre de los archivos pertenecientes a cada solicitud de predicción para luego buscarlos y obtenerlos desde bucket. De forma adicional, en el caso del ingreso de un código PFAM como dato de entrada a una solicitud se ha planteado un enfoque de reducción de procesamiento. Este enfoque permitirá que el back-end revise el estado de las solicitudes previamente registradas. En caso de que alguna ya haya sido completada exitosamente y que contenga como dato de entrada el mismo código PFAM que la nueva solicitud entonces el back-end automáticamente le asignará los resultados obtenidos en esa predicción.

Cabe mencionar que el diagrama relacional de la base de datos se adapta correctamente a la capa de modelos que se utilizará para realizar las api's de back-end, por lo tanto, es admitible su reutilización como un diagrama de clases. Las api's fueron desarrolladas en Python con el framework Flask.

Una vez establecidas tanto la arquitectura como la capa de modelos se dio pase a la programación de lo que se denomina como un Daemon. Este corresponde a un archivo ejecutable que va verificando de forma iterativa el ingreso de nuevas solicitudes de predicción, así como el estado de la predicción actual. Este es el que ejecuta el algoritmo de predicción y envía los datos generados hacia el bucket.

Gran parte de esta etapa de integración ha permitido cumplir los requisitos no funcionales y funcionales planteados en el Catálogo de requisitos del Anexo G como parte de un resultado alcanzado en el segundo objetivo específico del proyecto. No obstante, muchos otros ya fueron contemplados en los objetivos previos.

Al finalizar la integración propuesta se obtiene una herramienta integrada que permite la predicción de estructuras terciarias de proteínas repetidas, la cual se ha denominado como DeepReSPred. Esta herramienta devuelve los resultados que han sido generados por el algoritmo adaptado en el primer objetivo específico de este proyecto. En ese sentido, el reporte de funcionamiento de la herramienta implementada verificará de forma objetiva que se ha logrado lo mencionado. Este reporte se encuentra en el Anexo L de este documento y constituye el primer medio de verificación del resultado alcanzado. Asimismo, el segundo resultado esperado corresponde a la propia herramienta implementada, la cual ya ha sido exitosamente desplegada en la nube utilizando el servidor NginX.

En este punto cabe mencionar que existieron ciertas limitaciones en tanto a los créditos con los que se cuenta dentro de la plataforma Amazon Web Services, por lo cual se mantienen apagadas las instancias. No obstante, ya habiendo acabado la implementación requerida se considerará como validado el resultado esperado de esta sección.

6.2.2 Pruebas funcionales de la herramienta de predicción

A modo de verificación del cumplimiento de las funcionalidades propuestas para la herramienta, se realizaron pruebas funcionales a la misma. Estas pruebas estuvieron relacionadas a los requisitos funcionales capturados como uno de los resultados alcanzados en el segundo objetivo específico[28] de este proyecto. En adición, se incorporaron pruebas que puedan verificar el correcto funcionamiento de la herramienta en base al objetivo principal del proyecto: la predicción de estructuras terciarias de proteínas repetidas.

Este resultado cuenta con tres indicadores objetivamente verificables: el documento de especificación y resultados de pruebas deberá contener información detallada de su ejecución y sus resultados, se deberá cumplir el 100% de las pruebas funcionales y se deberá recibir la aprobación de los medios de verificación a través del juicio experto de un investigador en el área de bioinformática.

El documento de especificación y resultados de pruebas corresponde al primer medio de verificación del resultado esperado y se encuentra como parte del Anexo M. Para poder elaborarlo se utilizaron diversas herramientas tales como las bases de datos de proteínas en caso de la obtención de las secuencias con las cuales se realizaron las pruebas.

El segundo medio de verificación corresponde a un video de la herramienta en ejecución. Dicha grabación la herramienta de predicción denominada como DeepReSPred se encuentra disponible en la nube y se podrá acceder a ella a través del enlace especificado a continuación:

Grabación de la herramienta de predicción de estructuras terciarias de proteínas repetidas DeepReSPred en ejecución

[28] El documento correspondiente al Catálogo de requisitos funcionales y no funcionales de la herramienta propuesta se encuentra en el Anexo G.

6.2.3 Evaluación de usabilidad de la herramienta planteada

El último resultado alcanzado definido dentro del tercer objetivo específico del presente proyecto de fin de carrera es la evaluación de usabilidad de la herramienta implementada. Este resultado cuenta con un medio de verificación: un reporte de la evaluación de usabilidad para el cual se definieron dos indicadores objetivamente verificables. El primer indicador consiste en que el reporte contenga el detalle del procedimiento realizado además de que los resultados obtenidos sean satisfactorios para la herramienta. El segundo indicador corresponde a la aceptación de la prueba realizada por parte de un experto en usabilidad.

En este punto cabe mencionar que la intención de esta evaluación es obtener retrospectiva del desarrollo ya que, la revisión sistemática realizada en el Capítulo 3. Estado del Arte dio a relucir el contexto de la usabilidad en el desarrollo de las herramientas bioinformáticas como la que propone este proyecto. Esa investigación previa concluye en que la usabilidad no parece ser un factor muy considerado dentro de las herramientas de este tipo (Paixão-Cortes et al., 2018).

Para poder llevar a cabo la evaluación de usabilidad se ha elegido utilizar a la herramienta propuesta en el proyecto denominado como Usabilidad en servicios web bioinformáticos (Bezerra Brandao Corrales et al., 2020). Esta herramienta fue seleccionada en tanto es la última herramienta reconocida dedicada a la evaluación de usabilidad en herramientas web, en específico, en herramientas web bioinformáticas. En definitiva, ese recurso de evaluación se adapta en su totalidad a la herramienta implementada en el presente proyecto de tesis.

La evaluación de usabilidad consistió principalmente en la realización de un formulario anónimo con diez preguntas acerca de la percepción del usuario luego de utilizar la herramienta por primera vez. Se tomaron en cuenta tres perfiles de usuarios a los que se envió la encuesta.

Se obtuvieron ocho respuestas y a partir de ello se calculó el resultado de la evaluación de usabilidad en base a los lineamientos de la herramienta seleccionada. Obteniendo un total de 90.625 puntos de un rango de 0 a 100. Este puntaje pertenece al primer grupo SUS. Con ello se concluye que, el servicio web de DeepReSpred califica para los usuarios evaluados como un servicio web ideal y promotor, con un alto grado de admisibilidad y, en general, con una calificación A+.

El detalle de la evaluación realizada se encuentra en el Anexo N. Ese mismo anexo contiene el acta de aceptación de la prueba realizada por parte de un experto en usabilidad.

6.3 Discusión

Este sexto capítulo del presente proyecto de tesis detalla la descripción y la validación del cumplimiento de los resultados esperados del tercer objetivo específico. En este objetivo se abarca la integración del algoritmo de predicción de estructuras terciarias adaptado a las proteínas repetidas y la interfaz de la herramienta propuesta.

El primer resultado esperado corresponde a la implementación de los recursos necesarios para poder conectar el algoritmo con la interfaz. Para ello, se realizó un proceso de integración en el cual se reconoció la necesidad de elaborar dos diagramas que facilitarían el desarrollo: el diagrama de arquitectura y el diagrama relacional de bases de datos. Estos últimos se apoyan claramente en los demás diagramas presentados como parte de los objetivos específicos previos. Se realizó la creación de un script que encole las solicitudes de predicción registradas, ejecute el algoritmo y almacene los resultados. Asimismo, se llevó a cabo la configuración de la base de datos y la creación de diversas api's con herramientas como Flask, Python y MsqAlchemy. Se utilizaron, adicionalmente, diversos servicios de Amazon Web Services para desplegar los desarrollos de back-end como front-end, mantener una base de datos MySQL y almacenar los archivos de cada predicción.

El segundo resultado alcanzado corresponde a la definición de pruebas funcionales a la herramienta desarrollada, para la cual se tuvo en consideración el flujo de actividades que sigue el algoritmo, los datos de entrada admitidos y los requisitos funcionales del catálogo de requisitos elaborado en el objetivo específico número dos. Con la ejecución de las pruebas se verificó que la herramienta desarrollada cumple correctamente el flujo determinado para realizar la predicción de estructuras terciarias de proteínas repetidas.

El tercer resultado alcanzado contituyó la evaluación de usabilidad de la herramienta propuesta. Para poder realizarla se seleccionó una investigación enfocada en la evaluación de usabilidad de herramientas web como la que corresponde a este proyecto de tesis. Con lo cual se obtuvo la retroalimentación por parte de ocho usuarios acerca de su percepción al usar la herramienta por primera vez.

Llegado a este punto se verifica el cumplimiento de este último objetivo específico, lo cual da pie al cierre del desarrollo de una herramienta amigable al usuario que es capaz

de predecir estructuras terciarias de proteínas repetidas a partir de estructuras primarias. Esta herramienta ha sido denominada como el predictor profundo de estructuras de proteínas repetidas, DeepReSPred, por su traducción en inglés.

Capítulo 7.Conclusiones y trabajos futuros

7.1 Conclusiones

En base al objetivo general del presente proyecto de fin de carrera[29] se estableció un plan de acción para poder lograrlo de forma exitosa. Esto constituía la segregación del objetivo general en tres objetivos específicos, cada uno con sus propios resultados esperados. El cumplimiento total de estos resultados da origen a una herramienta amigable para el usuario que integra tanto una interfaz gráfica en un entorno web como un algoritmo de predicción de estructuras terciarias adaptado para responder a las particularidades de las proteínas repetidas.

El primer objetivo específico corresponde a la adaptación de un algoritmo de predicción de estructuras terciarias de proteínas en general para aplicarlo en proteínas repetidas. Para poder lograrlo se requirió, en primera instancia, la identificación de algoritmos *end-to-end*, es decir, algoritmos que generen las conformaciones tridimensionales de las proteínas a partir de una secuencia de aminoácidos. Cabe resaltar que este criterio reduce el área de búsqueda dado que existen pocas soluciones que cuentan con una arquitectura verdaderamente acorde (Laine et al., 2021). A partir de esta y otras consideraciones adicionales, se seleccionaron dos algoritmos DMPfold (Greener et al., 2019). y trRosetta (Yang et al., 2020). Siendo la primera de ellas la seleccionada para continuar los siguientes pasos en base a tres criterios: la eficacia del algoritmo luego de seguir las instrucciones para su instalación y uso, el empleo de librerías actualizadas y el recurso computacional requerido.

Llegado a este punto, se propusieron diversas modificaciones, entendiéndose también como adiciones, al algoritmo. Tras un proceso iterativo constituido por la aplicación de las modificaciones y la ejecución de pruebas de su funcionamiento, se llegó a la definición de cuatro: dos de ellas para el preprocesamiento, una para la generación de resultados y una más para la evaluación de la predicción. Las dos primeras trabajan en

[29] El objetivo general del proyecto es desarrollar una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria que cumpla con lineamientos de usabilidad.

conjunto para el preprocesamiento de datos de los fragmentos de una familia de proteínas repetidas obtenidos a partir de los servicios de PFAM (Mistry et al., 2021). Esto permite que, a partir del ingreso de un identificador de un dominio PFAM de proteínas repetidas, se puedan obtener los fragmentos presentes en distintas secuencias de proteínas. Con ello, se determinan dos caminos posibles: la predicción de cada uno de los fragmentos obtenidos que no cuenten con una estructura registrada en el Banco de Datos de Proteínas (PDB); y la predicción de las secuencias a las que pertenecen estos mismos fragmentos.

La tercera modificación interviene en la generación del resultado y se enfoca más en la secuencia de proteínas ingresada como dato de entrada en la predicción. En este caso, se procederá a predecir la estructura de la secuencia ingresada. No obstante, la estructura terciaria deberá refinarse con otra herramienta para poder identificar los fragmentos pertenecientes a las diversas familias de proteínas repetidas. Así, será importante entregar al usuario una alineación de estas últimas estructuras generadas.

La cuarta modificación corresponde a la evaluación de la predicción en sí. Para ello se plantea alinear la estructura predicha con cualquier estructura almacenada en PDB perteneciente a alguna proteína de la misma familia.

Estas cuatro modificaciones finales permitirán cumplir con el objetivo del proyecto cuyo enfoque principal se centra en las proteínas repetidas.

Como segunda instancia se cuenta con el objetivo número dos. Este abarca el diseño y la implementación de la interfaz de la herramienta propuesta. Para esto, es importante resaltar que la revisión sistemática realizada en el Capítulo 3. Estado del Arte dio a relucir que la aplicación de criterios de usabilidad, en el desarrollo de servicios bioinformáticos afines al propuesto, no es un factor al que se le ha dado prioridad (Machado et al., 2018). A partir de ello se tomaron en cuenta algunos lineamientos básicos de usabilidad para poder diseñar las vistas del prototipo de la interfaz de la herramienta, y definir su interacción con el usuario. Habiendo definido lo mencionado, se prosiguió a la implementación de la interfaz propuesta. Esto se realizó en forma paralela al logro del primer objetivo específico: la adaptación del algoritmo de predicción.

Como última instancia del proyecto se tiene la integración del algoritmo adaptado y la interfaz desarrollada. Asimismo, se incluyó un procedimiento de pruebas funcionales de la herramienta y una evaluación de usabilidad para obtener retrospectiva del desarrollo respecto a la percepción de los usuarios en su primera interacción con la plataforma

web. Para esto último, se utilizó una herramienta que define su enfoque desde su título: Usabilidad en servicios web bioinformáticcos (Bezerra Brandao Corrales et al., 2020). Los perfiles que se tuvieron en consideración para la evaluación de usabilidad incluyeron a alumnos, docentes e investigadores que pueden o no haber tenido interacción previa con otro recurso bioinformático ya que al medir la usabilidad se espera que la herramienta sea intuitiva y que brinde las facilidades para ser usado por cualquier usuario.

En este punto es importante mencionar que se tuvo muy en consideración los problemas causa y efecto descritos en la definición de la problemática general del Capítulo 1 Generalidades. Esto último en cuanto a que es posible afirmar que la plataforma web desarrollada representa una propuesta que afronte la carencia de una herramienta para la predicción de estructuras terciarias de proteinas repetidas a partir de su secuencia de aminoácidos, que requiere de recursos informáticos promedios y cumple con lineamientos de usabilidad en el diseño de sus interfaces. Esto último ayudará a que el tiempo que sea utilizado en la interacción con la herramienta sea bien aprovechada y no se incurra en gastos innecesarios de energía por parte del usuario.

Asimismo, dado que constituye una propuesta para la predicción de estructuras de proteínas repetidas, permitirá que su uso reduzca en cierto porcentaje la brecha entre la cantidad de secuencias y la cantidad de estructuras conocidas. Esto último respecto a las proteínas repetidas.

7.2 Trabajos futuros

A lo largo del desarrollo del presente proyecto de tesis y la herramienta que propone han surgido nuevas alternativas de solución a la problemática de la predicción de estructuras terciarias de proteínas. El contexto bioinformático avanza en torno al surgimiento de nuevas tecnologías, por lo cual es necesario ir actualizando las herramientas que se están usando. En ese sentido, cabe mencionar que la herramienta propuesta, en particular, el algoritmo seleccionado, está abierto a modificaciones que aporten a su eficiencia y eficacia. Asimismo, se propone utilizar la información de las proteínas de las bases de datos Trembl y SwissProt con la herramienta y realizar el análisis de resultados. Adicionalmente, esto se puede replicar haciendo uso de toda la base de datos Pfam.

Dado que este trabajo es parte de los proyectos de la Dra. Hirsh, en los próximos meses se procederá a desplegar la herramienta en la página web de Bioinformatica.org[30] y a realizar la respectiva publicación.

Finalmente, cabe destacar que el presente proyecto está enfocado en las proteínas repetidas, dado que presentan particularidades que las definen como tal. Es por ello que se incentiva a que se realicen más proyectos con enfoques diferentes y en relación a otros tipos de proteínas, o incluso a otras necesidades bioinformáticas.

Referencias

Abeysinghe, E., Brylinski, M., Christie, M., Marru, S., & Pierce, M. (2019). LSU computational system biology gateway for education. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3332186.3333259

Adhikari, B., & Cheng, J. (2018). CONFOLD2: Improved contact-driven ab initio protein structure modeling. *BMC Bioinformatics*, *19*(1). https://doi.org/10.1186/s12859-018-2032-6

Association for Computing Machinery. (2021). *About ACM DL*. https://dl-acm-org.ezproxybib.pucp.edu.pe/about

Association for Computing Machinery [ACM], Association for Information Systems [AIS], & Computer Society [IEEE-CS]. (2005). *Computing Curricula 2005*. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2005-march06final.pdf

Bevan, N., Kirakowski, J., & Maissel, J. (1991). *What is Usability?*

Bezerra Brandao Corrales, M. A., Hirsh Martinez, L., & Pow Sang Portillo, J. A. (2020). *Usabilidad en servicios web bioinformáticos*.

Billett, H. (1990). Hemoglobin and Hematocrit. In H. Walker, W. Hall, & J. Hurst (Eds.), *Clinical Methods: The History, Physical, and Laboratory Examinations* (3era edición). Butterworths. https://www.ncbi.nlm.nih.gov/books/NBK259/pdf/Bookshelf_NBK259.pdf

[30] Para acceder a la página web de bioinformatica.org se deberá ingresar al siguiente enlace: https://bioinformatica.org/

Bolchini, D., Finkelstein, A., Perrone, V., & Nagl, S. (2009). Better bioinformatics through usability analysis. *BIOINFORMATICS ORIGINAL PAPER*, *25*(3), 406–412. https://doi.org/10.1093/bioinformatics/btn633

Brunette, T. J., Parmeggiani, F., Huang, P. S., Bhabha, G., Ekiert, D. C., Tsutakawa, S. E., Hura, G. L., Tainer, J. A., & Baker, D. (2015). Exploring the repeat protein universe through computational protein design. *Nature*, *528*(7583), 580–584. https://doi.org/10.1038/nature16162

Brunger, A. T. (2007a). Version 1.2 of the crystallography and nmr system. *Nature Protocols*, *2*(11), 2728–2733. https://doi.org/10.1038/nprot.2007.406

Brunger, A. T. (2007b). Version 1.2 of the crystallography and nmr system. *Nature Protocols*, *2*(11), 2728–2733. https://doi.org/10.1038/nprot.2007.406

Brunger, A. T., Adams, P. D., Marius Clore, G., Delano, W. L., Gros, P., Grosse-kunstleve, R. W., Jiang, J., Kuszewski, fJOHN, Nilges, M., Pannu, N. S., Read, R. J., Rice, L. M., Simonson, T., & Warren, G. L. (1998). Crystallography & NMR System: A New Software Suite for Macromolecular Structure Determination. In *Acta Cryst* (Vol. 54).

Burley, S. K., Bhikadiya, C., Bi, C., Bittrich, S., Chen, L., Crichlow, G. v., Christie, C. H., Dalenberg, K., di Costanzo, L., Duarte, J. M., Dutta, S., Feng, Z., Ganesan, S., Goodsell, D. S., Ghosh, S., Green, R. K., Guranovic, V., Guzenko, D., Hudson, B. P., … Zhuravleva, M. (2021). RCSB Protein Data Bank: Powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic Acids Research*, *49*(1), D437–D451. https://doi.org/10.1093/nar/gkaa1038

Daros, W. R. (2002). ¿Qué es un marco teórico? In *Enfoques: Vol. XIV* (Issue 1). Universidad Adventista del Plata. https://www.redalyc.org/articulo.oa?id=25914108

DeepMind. (2021). *AlphaFold: a solution to a 50-year-old grand challenge in biology*. https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology

Deng, H., Jia, Y., & Zhang, Y. (2018). Protein structure prediction. *International Journal of Modern Physics B*, *32*(18). https://doi.org/10.1142/S021797921840009X

di Domenico, T., Potenza, E., Walsh, I., Gonzalo Parra, R., Giollo, M., Minervini, G., Piovesan, D., Ihsan, A., Ferrari, C., Kajava, A. v., & Tosatto, S. C. E. (2014). RepeatsDB: A database of tandem repeat protein structures. *Nucleic Acids Research*, *42*(D1), D352–D357. https://doi.org/10.1093/nar/gkt1175

Elsevier. (2021). *About Scopus*. https://www-elsevier-com.ezproxybib.pucp.edu.pe/solutions/scopus?dgcid=RN_AGCM_Sourced_300005030

Gao, M., Zhou, H., & Skolnick, J. (2019). DESTINI: A deep-learning approach to contact-driven protein structure prediction. *Scientific Reports*, *9*(1). https://doi.org/10.1038/s41598-019-40314-1

Greener, J. G., Kandathil, S. M., & Jones, D. T. (2019). *Deep learning extends de novo protein modelling coverage of genomes using iteratively predicted structural constraints*. https://doi.org/10.1038/s41467-019-11994-0

Guo, W., Shea, J. E., & Berry, R. S. (2006). The physics of the interactions governing folding and association of proteins. In *Annals of the New York Academy of Sciences* (Vol. 1066, pp. 34–53). https://doi.org/10.1196/annals.1363.025

Hirsh, L., Piovesan, D., Paladin, L., & Tosatto, S. C. E. (2016). Identification of repetitive units in protein structures with ReUPred. *Amino Acids*, *48*(6), 1391–1400. https://doi.org/10.1007/s00726-016-2187-2

Hou, J., Wu, T., Guo, Z., Quadir, F., & Cheng, J. (2020). The MULTICOM Protein Structure Prediction Server Empowered by Deep Learning and Contact Distance Prediction. *Methods in Molecular Biology*, *2165*, 13–26. https://doi.org/10.1007/978-1-0716-0708-4_2

Huang, Y., Li, H., & Xiao, Y. (2018). 3dRPC: A web server for 3D RNA-protein structure prediction. *Bioinformatics*, *34*(7), 1238–1240. https://doi.org/10.1093/bioinformatics/btx742

Jin, S., Contessoto, V. G., Chen, M., Schafer, N. P., Lu, W., Chen, X., Bueno, C., Hajitaheri, A., Sirovetz, B. J., Davtyan, A., Papoian, G. A., Tsai, M.-Y., & Wolynes, P. G. (2020). AWSEM-Suite: A protein structure prediction server based on template-guided, coevolutionary-enhanced optimized folding landscapes. *Nucleic Acids Research*, *48*(W1), W25–W30. https://doi.org/10.1093/NAR/GKAA356

Jordan, B. (2021). A giant step towards elucidating the function of proteins. *Medecine/Sciences*, *37*(2), 197–200. https://doi.org/10.1051/medsci/2020281

Kajava, A. V. (2012). Tandem repeats in proteins: From sequence to structure. *Journal of Structural Biology*, *179*(3), 279–288. https://doi.org/10.1016/j.jsb.2011.08.009

Kajava, A. V., & Steven, A. C. (2006). β-Rolls, β-Helices, and Other β-Solenoid Proteins. In *Advances in Protein Chemistry* (Vol. 73). https://doi.org/10.1016/S0065-3233(06)73003-0

Kordic, B., Popovic, M., Popovic, M., Goldstein, M., Amitay, M., & Dayan, D. (2019). A Protein Structure Prediction Program Architecture Based on a Software Transactional Memory. *Proceedings of the 6th Conference on the Engineering of Computer Based Systems*. https://doi.org/10.1145/3352700.3352701

Kuhlman, B., & Bradley, P. (2019a). Advances in protein structure prediction and design. *Nature Reviews Molecular Cell Biology*, *20*(11), 681–697. https://doi.org/10.1038/s41580-019-0163-x

Kuhlman, B., & Bradley, P. (2019b). Advances in protein structure prediction and design. *Nature Reviews Molecular Cell Biology*, *20*(11), 681–697. https://doi.org/10.1038/s41580-019-0163-x

Laine, E., Eismann, S., Elofsson, A., & Grudinin, S. (2021). *Protein sequence-to-structure learning: Is this the end(-to-end revolution)?* https://arxiv.org/abs/2105.07407

Lambert, K. A., & Osborne, M. (2018). *Fundamentals of Python : first programs* (Cengage Learning, Ed.; 2nd ed.).

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lopes, G. R., de Souza, P. S. L., & Delbem, A. C. B. (2019). A Systematic Mapping on High-Performance Computing for Protein Structure Prediction. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 11333 LNCS*. https://doi.org/10.1007/978-3-030-15996-2_6

Machado, V. S., Tanus, M. S. S., Paixão-Cortes, W. R., de Souza, O. N., Campos, M. B., & Silveira, M. S. (2018). wCReF – a web server for the cref protein structure

predictor. *Advances in Intelligent Systems and Computing*, *558*, 831–838. https://doi.org/10.1007/978-3-319-54978-1_103

Makigaki, S., & Ishida, T. (2020). Sequence alignment using machine learning for accurate template-based protein structure prediction. *Bioinformatics*, *36*(1), 104–111. https://doi.org/10.1093/bioinformatics/btz483

Marcotte, E. M., Pellegrini, M., Yeates, T. O., & Eisenberg, D. (1999). A census of protein repeats. *Journal of Molecular Biology*, *293*(1), 151–160. https://doi.org/10.1006/jmbi.1999.3136

Marie Sklodowska-Curie Actions [MSCA], & Research and Innovation Staff Exchange [RISE]. (2018). *Grant Agreement-823886-REFRACT*. European Commission.

Mistry, J., Chuguransky, S., Williams, L., Qureshi, M., Salazar, G. A., Sonnhammer, E. L. L., Tosatto, S. C. E., Paladin, L., Raj, S., Richardson, L. J., Finn, R. D., & Bateman, A. (2021). Pfam: The protein families database in 2021. *Nucleic Acids Research*, *49*(D1), D412–D419. https://doi.org/10.1093/NAR/GKAA913

Nelson, D. L., & Cox, M. M. (2017). *Lehninger Principles of Biochemistry*. https://scholar.google.com/scholar?hl=es&as_sdt=0%2C5&q=Lehninger+Principles+of+Biochemistry+2017&btnG=

Olivares, C. A. (2016). *Revisión sistemática sobre la aplicación de ontologías de dominio en el análisis de sentimiento. [Tesis de Maestría, Pontificia Universidad Católica del Perú]*. http://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/7514

Oller, J. (2003). *Elementos teórico-prácticos útiles para comprender el uso de los motores de búsqueda en Internet*. ACIMED. http://scielo.sld.cu/scielo.php?pid=S1024-94352003000600007&script=sci_arttext&tlng=pt

Paixão-Cortes, V. S. M., Tanus, M. S. S., Paixão-Cortes, W. R., de Souza, O. N., Campos, M. B., & Silveira, M. S. (2018). Usability as the key factor to the design of a web server for the CReF protein structure predictor: The wCReF. *Information (Switzerland)*, *9*(1). https://doi.org/10.3390/info9010020

Paladin, L., Bevilacqua, M., Errigo, S., Piovesan, D., Mičetić, I., Necci, M., Monzon, A. M., Fabre, M. L., Lopez, J. L., Nilsson, J. F., Rios, J., Menna, P. L., Cabrera, M., Buitron, M. G., Kulik, M. G., Fernandez-Alberti, S., Fornasari, M. S., Parisi, G., Lagares, A., … Tosatto, S. C. E. (2021). RepeatsDB in 2021: Improved data and

extended classification for protein tandem repeat structures. *Nucleic Acids Research*, *49*(D1), D452–D457. https://doi.org/10.1093/nar/gkaa1097

Paladin, L., Hirsh, L., Piovesan, D., Andrade-Navarro, M. A., Kajava, A. v, & Tosatto, S. C. E. (2017). RepeatsDB 2.0: improved annotation, classification, search and visualization of repeat protein structures. *Nucleic Acids Research*, *45*. https://doi.org/10.1093/nar/gkw1136

Parmeggiani, F., & Huang, P. S. (2017). Designing repeat proteins: a modular approach to protein design. *Current Opinion in Structural Biology*, *45*, 116–123. https://doi.org/10.1016/j.sbi.2017.02.001

Pérez, J. I. (2020). *Alpha Fold 2: un logro impresionante que marca un antes y un después en el estudio de las proteínas*. The Conversation. https://theconversation.com/alpha-fold-2-un-logro-impresionante-que-marca-un-antes-y-un-despues-en-el-estudio-de-las-proteinas-151702

Protein Structure Prediction Center. (2020). *CRITICAL ASSESSMENT OF TECHNIQUES FOR PROTEIN STRUCTURE PREDICTION ABSTRACT BOOK*. https://www.predictioncenter.org/casp14/doc/CASP14_Abstracts.pdf

PyMOL | pymol.org. (n.d.). Retrieved November 11, 2021, from https://pymol.org/2/

Sander, M. E., Ablin, P., Blondel, M., & Peyré, G. (2021). *Momentum Residual Neural Networks*.

Savage, N. (2015). Proteomics: High-protein research. *Nature*, *527*(7576), S6–S7. https://doi.org/10.1038/527S6a

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W. R., Bridgland, A., Penedones, H., Petersen, S., Simonyan, K., Crossan, S., Kohli, P., Jones, D. T., Silver, D., Kavukcuoglu, K., & Hassabis, D. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, *577*(7792), 706–710. https://doi.org/10.1038/s41586-019-1923-7

Septiana, L., Suzuki, H., Ishikawa, M., Obi, T., Kobayashi, N., Ohyama, N., Wihardjo, E., & Andiani, D. (2019). Classification of Elastic and Collagen Fibers in H&E Stained Hyperspectral Images. *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 7031–7035. https://doi.org/10.1109/EMBC.2019.8856371

SESAR Joint Undertaking. (2021). *SESAR and Artificial Intelligence*. https://www.sesarju.eu/node/3356

SJÖSTRAND, T. (1949). The Total Quantity of Hemoglobin in Man and its Relation to Age, Sex, Bodyweight and Height. *Acta Physiologica Scandinavica*, *18*(4), 324–336. https://doi.org/10.1111/j.1748-1716.1949.tb00623.x

Steinegger, M., Meier, M., Mirdita, M., Vöhringer, H., Haunsberger, S. J., & Söding, J. (n.d.). *HH-suite3 for fast remote homology detection and deep protein annotation*. https://doi.org/10.1186/s12859-019-3019-7

TM-align: A protein structure alignment algorithm using TM-score rotation matrix. (n.d.). Retrieved November 11, 2021, from https://zhanggroup.org/TM-align/

UniProt Consortium. (2021a). *About UniProt*. https://www.uniprot.org/help/about

UniProt Consortium. (2021b). *Current Release Statistics < Uniprot < EMBL-EBI*. https://www.ebi.ac.uk/uniprot/TrEMBLstats

Vaquero, J. R. (1997). *Recuperación de la información en Internet: motores y otros agentes de búsqueda | Scire: representación y organización del conocimiento*. https://www.ibersid.eu/ojs/index.php/scire/article/view/1078

Xiong, Jin. (2006). *Essential bioinformatics*. Cambridge University Press.

Yang, J., Anishchenko, I., Park, H., Peng, Z., Ovchinnikov, S., & Baker, D. (2020). *Improved protein structure prediction using predicted interresidue orientations*. *117*(3), 1496–1503. https://doi.org/10.1073/pnas.1914677117/-/DCSupplemental

Zhao, Z., & Gong, X. (2019). Protein-protein interaction interface residue pair prediction based on deep learning architecture. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *16*(5), 1753–1759. https://doi.org/10.1109/TCBB.2017.2706682

Zheng, W., Zhang, C., Wuyun, Q., Pearce, R., Li, Y., & Zhang, Y. (2019). LOMETS2: improved meta-threading server for fold-recognition and structure-based function annotation for distant-homology proteins. *Nucleic Acids Research*, *47*(W1), W429–W436. https://doi.org/10.1093/nar/gkz384

Anexos

En esta sección se detallan los anexos relacionados al presente documento.

Anexo A: Ficha de registro de idea de tesis y asesor

En este anexo se plantea la definición inicial del tema de tesis, la presentación de la asesora del proyecto y el cronograma de actividades respecto a los entregables del curso. Así mismo, se define el área al que pertenece el presente trabajo y la descripción de la problemática en la que se enfoca.

Título del tema de tesis

El título del proyecto de tesis a trabajar es el siguiente: “Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria”.

Asesor

La asesora principal del proyecto de tesis será la Dra. Layla Hirsh Martínez, quien cuenta con un doctorado en Biociencia y Biotecnología. Ella es investigadora registrada en el Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica (CONCYTEC), es egresada y docente de la Pontificia Universidad Católica del Perú (PUCP) en la sección de Ingeniería Informática y colaboradora del Instituto de Bioinformática Europeo (EMBL-EBI). Con aprobación de la asesora, se ha determinado tener reuniones virtuales semanales a llevarse a cabo los días martes a las 7 p.m. para organizar, plantear y, en algunos casos, desarrollar las actividades relacionadas a los objetivos del proyecto de tesis. Asimismo, se acuerda mantener comunicación constante conformando un grupo de trabajo por mensajería instantánea para realizar consultas puntuales o para absolver dudas sobre la metodología o el presente proyecto. En el mismo sentido, las versiones preliminares de los entregables se presentarán dos días antes de la fecha de entrega programada por el calendario del curso, para que la asesora pueda brindar las observaciones correspondientes, razón por la cual éstas serán entregadas para su revisión y/o corrección a más tardar un día después.

El cronograma de entregables y actividades, de acuerdo con la organización del curso Tesis 1 y la planificación con la asesora del proyecto, se detalla en la Tabla 22

Entregable	Plan de Trabajo	Fecha de reunión con asesora	Elaboración del entregable	Fecha de envío a la asesora	Corrección de observaciones de la asesora	Fecha de publicación en el repositorio	Revisión de observaciones con asesora	Corrección de observaciones del docente
EP1.1	Revisión y acuerdo del plan de trabajo. Definición de tema.	30/03/2021	30/03/2021	02/04/2021	04/04/2021	05/04/2021	06/04/2021	06/04/2021
EP1.2	Preguntas de revisión, motores de búsqueda, documentos existentes y formulario de extracción.	06/04/2021	06/04/2021	09/04/2021	11/04/2021	12/04/2021	13/04/2021	13/04/2021
EP1.3	Ejecución de la revisión. Revisión de herramientas y formulario de extracción.	13/04/2021	13/04/2021	16/04/2021	18/04/2021	19/04/2021	20/04/2021	20/04/2021
EP1.4	Revisión de correcciones, respuestas a preguntas de revisión y conclusiones.	20/04/2021	20/04/2021	23/04/2021	25/04/2021	26/04/2021	27/04/2021	27/04/2021
EP1.5	Definición del marco	27/04/2021	27/04/2021	30/04/2021	02/05/2021	03/05/2021	04/05/2021	04/05/2021

	conceptual y términos a incluir.							
E1	Revisión de correcciones, definición de problemática y marco (conceptual y teórico).	04/05/2021	04/05/2021	05/05/2021	06/05/2021	**07/05/2021***	11/05/2021	11/05/2021
E2.1	Planteamiento del árbol de objetivos.	11/05/2021	11/05/2021	14/05/2021	16/05/2021	17/05/2021	18/05/2021	18/05/2021
E2	Revisión de correcciones. Definición de objetivos. Determinación de resultados esperados y medios de verificación.	18/05/2021 25/05/2021 01/06/2021	18/05/2021 25/05/2021 01/06/2021	02/06/2021	03/06/2021	**04/05/2021***	08/06/2021	08/06/2021
E3	Revisión de correcciones. Definición de resultados esperados, herramientas, métodos y procedimientos.	08/06/2021 15/06/2021	08/06/2021	14/06/2021	15/06/2021	**16/06/2021***	22/06/2021	22/06/2021

E4	Revisión de correcciones. Ajuste de anexos. Evaluación del documento de tesis completo y verificación del cumplimiento de todos los aspectos requeridos.	22/06/2021	22/06/2021	25/06/2021	27/06/2021	28/06/2021	-	-

Tabla 22. *Cronograma de entregables y actividades para Tesis 1. Anexo A.*

Nota 1: La elaboración del entregable como la corrección de observaciones del docente por parte de la alumna inicia una vez que haya terminado la reunión con la asesora del proyecto.

Nota 2: El asterisco en las fechas de la quinta columna de la Tabla 22 hace referencia a que la encargada de subir el documento al repositorio será la asesora.

Área

El proyecto corresponde a dos áreas de la Informática[31]: **Ciencias de la Computación e Ingeniería de Software**. Esto debido a que estas dos áreas se complementan sobremanera para llevar a cabo los objetivos de la tesis; sin embargo, se tendrá un enfoque principal en la primera mencionada.

Respecto a las Ciencias de la Computación, se emplean algoritmos aplicables a diversos contextos, en este caso a la bioinformática, en particular a la proteómica. La bioinformática es un campo de investigación interdisciplinario centrado en el análisis cuantitativo de información relacionada a las macromoléculas biológicas como el ADN, ARN y las proteínas, con ayuda indispensable de la computación (Xiong, 2006). En ese sentido, el apoyo en los fundamentos de la bioinformática para la definición del problema y el contexto de aplicación es crucial.

Por otro lado, respecto a la Ingeniería de Software, el proyecto propone el diseño y desarrollo de una herramienta que satisfaga los requerimientos planteados, una vez se haya incorporado el algoritmo mencionado en el párrafo anterior. Por lo cual se tendrá como base al ciclo de vida de un software, así, se aplicarán metodologías para la evaluación de la necesidad y la implementación del mismo, teniendo en cuenta el aseguramiento de la usabilidad para lograr sus objetivos.

Descripción

Cuando escuchamos hablar sobre las proteínas tendemos a relacionarlas a los músculos, lo cual es razonable puesto que son tejidos que están conformados en su mayoría por ellas (Pérez, 2020). Sin embargo, es importante saber que en realidad las proteínas son el constituyente principal de las células, por ende, son el núcleo de la mayoría de los procesos biológicos de nuestro organismo (Senior et al., 2020).

[31] Según la Asociación de Maquinaria de Computación (ACM, por sus siglas en inglés), la Informática se divide en cinco disciplinas: Ingeniería de Computadoras, Ciencias de la Computación, Sistemas de Información, Tecnologías de Información e Ingeniería de Software (ACM et al., 2005).

Las proteínas están constituidas por una secuencia de varios cientos o miles de aminoácidos, y la organización de estos elementos determina aspectos importantes como su estructura tridimensional y su función (Xiong, 2006). Actualmente, se utilizan diversas técnicas para conocer esas estructuras, tales como la cristalografía de rayos X y la criomicroscopía electrónica, las cuales por el mismo hecho de ser experimentales son costosas y, por ello, la obtención de resultados se vuelve un proceso prolongado (Pérez, 2020). Por otro lado, tenemos a las predicciones basadas en algoritmos; sin embargo, estas requieren de gran capacidad computacional tanto para desarrollarlos como para aplicarlos (Pérez, 2020).

Con los métodos explicados anteriormente se ha logrado obtener información necesaria de más de ciento setenta mil estructuras de proteínas, de las más de doscientas millones de proteínas existentes en la naturaleza (Pérez, 2020). Además, las bases de datos siguen creciendo exponencialmente, sin embargo, aún se requiere de una herramienta integral que pueda explotar la data conocida de las secuencias de las proteínas (estructura primaria), es decir, sus cadenas de aminoácidos, para lograr predecir sus estructuras tridimensionales; esto ya que “los intentos de los últimos 50 años no han tenido éxito debido a la complejidad del problema, aunque ha habido algunos avances” (Jordan, 2021). La situación no mejora si acotamos esta situación a un contexto reducido y nos enfocamos en una familia de proteínas como, por ejemplo, las proteínas repetidas, que tienen una degeneración de secuencia muy alta (Deng et al., 2018; Hirsh et al., 2016; Parmeggiani & Huang, 2017).

A partir de ello, se identifica el problema principal como la carencia de herramientas eficientes para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria. El objetivo central abarcaría el desarrollo de una herramienta eficiente y eficaz capaz de predecir estructuras tridimensionales de proteínas repetidas a partir de su estructura primaria.

Como parte de los resultados esperados se aspira a identificar un algoritmo eficiente que pueda transformar la data disponible de las secuencias de proteínas repetidas en información útil para la predicción de su estructura, así como desarrollar una interfaz usable que lo albergue y asegure el cumplimiento de los objetivos, la cual será validada por juicio experto. Para determinar la usabilidad de la herramienta se aplicará una evaluación de usabilidad implementada para un contexto bioinformático.

Bibliografía

Association for Computing Machinery [ACM], Association for Information Systems [AIS], & Computer Society [IEEE-CS]. (2005). *Computing Curricula 2005*. http://shop.ieee.org/store/

Jordan, B. (2021). A giant step towards elucidating the function of proteins. *Medecine/Sciences*, *37*(2), 197–200. https://doi.org/10.1051/medsci/2020281

Pérez, J. I. (2020). *Alpha Fold 2: un logro impresionante que marca un antes y un después en el estudio de las proteínas*. The Conversation. https://theconversation.com/alpha-fold-2-un-logro-impresionante-que-marca-un-antes-y-un-despues-en-el-estudio-de-las-proteinas-151702

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W. R., Bridgland, A., Penedones, H., Petersen, S., Simonyan, K., Crossan, S., Kohli, P., Jones, D. T., Silver, D., Kavukcuoglu, K., & Hassabis, D. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, *577*(7792), 706–710. https://doi.org/10.1038/s41586-019-1923-7

Xiong, Jin. (2006). *Essential bioinformatics*. Cambridge University Press.

Anexo B: Formulario de extracción de datos

Este anexo contiene el formulario final de extracción empleado en el Capítulo 3. Estado del Arte, sección 3.5 Formulario de extracción de datos. Dicho documento es una hoja de cálculo en la nube y está nombrada como se observa a continuación:

20140104_SolangePalomino_LaylaHirsh_FE.xlsm

Para poder acceder al archivo es necesario ingresar al siguiente enlace:

Formulario de extracción de datos

Anexo C: Plan de Proyecto

En esta sección se presenta la planificación para la ejecución del proyecto de tesis planteado. Se encuentra dividida en diez apartados que describen a detalle los pasos con los cuales se llevará a cabo el desarrollo de la investigación.

- **Justificación**

Las proteínas cumplen un rol biológico vital dentro de las funciones orgánicas de los seres vivos (Lopes et al., 2019). Son macromoléculas complicadas y ampliamente extendidas en la naturaleza que están compuestas por una secuencia de aminoácidos, o estructura primaria, que se pliega en una estructura tridimensional única para cada una de ellas, denominada como estructura terciaria (Deng et al., 2018).

Es de suma importancia conocer la información tridimensional de las bioestructuras, en particular de las proteínas, puesto que permite entender cuál es su función dentro de los procesos biológicos en los que participan (Makigaki & Ishida, 2020). Es importante también porque se conoce su relación con distintas enfermedades humanas, tanto en el proceso de diagnóstico como el tratamiento (Kajava & Steven, 2006; Marcotte et al., 1999; Marie Sklodowska-Curie Actions [MSCA] & Research and Innovation Staff Exchange [RISE], 2018); además porque da pie al descubrimiento y desarrollo de nueva medicina (Burley et al., 2021). Esto último en base al diseño e ingeniería de nuevas proteínas (Parmeggiani & Huang, 2017).

Mucha de la información conseguida hasta el momento sobre las estructuras terciarias de las proteínas ha sido obtenida a partir de métodos experimentales u otros que hasta el día de hoy significan una gran inversión económica y temporal (Lopes et al., 2019). Asimismo, a lo largo de los años, se ha tratado de implementar distintas soluciones computacionales que ayuden a determinar la posición de los átomos de las proteínas en un espacio tridimensional a partir de su estructura primaria, lo cual fue denominado como la problemática de la predicción de estructuras terciarias (Lopes et al., 2019). Sin embargo, este es un desafío dentro de la bioinformática estructural, que aún queda por resolver (Deng et al., 2018; Gao et al., 2019; Lopes et al., 2019; Machado et al., 2018).

Lo mencionado es mucho más evidente en el caso de las proteínas repetidas, una familia de proteínas reconocidas por la presencia de unidades de repetición en su estructura, así como su alto grado de degeneración estructural (Hirsh et al., 2016; Paladin et al., 2021). Actualmente, en la base de datos de unidades de repetición en proteínas repetidas, RepeatsDB, se cuenta con información estructural detallada de

alrededor de ocho mil proteínas repetidas (Paladin et al., 2021), cuando el repositorio mundial de proteínas, UniProt, contiene más de doscientas catorce millones de secuencias de proteínas[32] (UniProt Consortium, 2021b). Se sabe que al menos el 14% de todas las proteínas existentes son proteínas repetidas (Marcotte et al., 1999).

Nos encontramos en un contexto en el cual la obtención de datos de las estructuras primarias supera exponencialmente al proceso por el cual se reconoce a las estructuras terciarias que les corresponden (Makigaki & Ishida, 2020). Es por ello que el presente proyecto busca desarrollar una herramienta informática integral y eficiente que aproveche los datos de las secuencias de aminoácidos de proteínas repetidas alojadas en las diversas bases de datos para predecir sus consiguientes estructuras tridimensionales.

Por último, diversos reportes de investigación empírica y teórica, así como la revisión sistemática realizada en el Capítulo 3. Estado del Arte, han evidenciado la necesidad de la inclusión de la usabilidad en los entornos bioinformáticos (Paixão-Cortes et al., 2018). Por ello, el presente proyecto también busca promover su aplicación en el desarrollo de herramientas afines, evaluando su usabilidad para lograr desarrollar un recurso informático con una interfaz amigable al usuario que permita la predicción de estructuras terciarias de proteínas repetidas a partir de sus estructuras primarias.

- **Viabilidad**

El desarrollo del proyecto requiere de conocimiento en torno a la bioinformática, en específico a las proteínas y sus estructuras. No obstante, quien va a asesorar la investigación tiene amplia experiencia y conocimiento en la temática que se está abarcando, además de encontrarse en el entorno de expertos que enfocan sus esfuerzos en la investigación de proteínas repetidas a nivel mundial.

Respecto a la obtención de los datos de las proteínas, es importante destacar que, al ser parte de la investigación bioinformática, toda la información necesaria se encuentra en repositorios y bases de datos disponibles a través de la web para todas las personas interesadas.

[32] En el repositorio mundial de proteínas, UniProt, se encuentran albergadas más de doscientas catorce millones de secuencias de proteínas (UniProt Consortium, 2021b). En relación a la información estructural 3D de proteínas, aproximadamente ciento ochenta mil de ellas (~1.2%), están alojadas en el Banco de Datos de Proteínas (Burley et al., 2021).

Por otro lado, respecto a los recursos del proyecto, cabe mencionar que se utilizarán librerías, IDE's y herramientas, en general, que sean de acceso libre o educativo, con lo cual no se involucrarán costos adicionales a los presupuestados.

Asimismo, el desarrollo del proyecto está siendo planificado a detalle y de forma minuciosa para que las tareas se realicen dentro del plazo establecido. Con ello, el tiempo será un factor controlado para poder cumplir los objetivos.

En el mismo sentido, dado que el proyecto está enfocado en el desarrollo de una herramienta web, se establecerá una arquitectura que pueda ser desplegada en entornos de Amazon Web Service (AWS). La universidad otorga créditos educativos para poder usar esas instancias.

Por último, el proyecto plantea evaluar la usabilidad de la herramienta desarrollada para asegurar que sus interfaces y las acciones que requiera para procesar los datos sean amigables al usuario. Es por ello que se llevará a cabo la aplicación de una metodología que se adapte a la línea de investigación de la herramienta. Con ello cabe mencionar que el contexto de pandemia no influye en su realización puesto que la evaluación de usabilidad se realizará de forma interna, es decir, no requerirá de evaluadores externos.

Luego de considerar los seis aspectos mencionados, se puede confirmar la viabilidad del proyecto.

- **Alcance**

El alcance del presente proyecto de ingeniería está centrado en dos áreas de la Informática[33]: Ciencias de la computación e Ingeniería de Software. Ambos enfocados en una temática del campo de investigación de la bioinformática. Según los objetivos del proyecto, se pretende desarrollar una herramienta que permita predecir estructuras terciarias de proteínas repetidas a partir de sus estructuras primarias.

La predicción de estructuras se llevará a cabo a través del uso de un algoritmo que aplique inteligencia artificial. Este algoritmo no será un nuevo algoritmo, sino que será resultado de la integración de modificaciones a un algoritmo que ya haya sido implementado previamente, adaptaciones que serán evaluadas para poder cumplir los objetivos respecto a una familia específica de proteínas: las proteínas repetidas. Cabe

[33] Según la Asociación de Maquinaria de Computación (ACM, por sus siglas en inglés), la Informática se divide en cinco disciplinas: Ingeniería de Computadoras, Ciencias de la Computación, Sistemas de Información, Tecnologías de Información e Ingeniería de Software (ACM et al., 2005).

resaltar este punto puesto que las proteínas son moléculas ampliamente extensas en la naturaleza y cada familia cuenta con características distintas, siendo las proteínas repetidas una de las más particulares (Marcotte et al., 1999; MSCA & RISE, 2018), pero de las cuales no se cuenta con mucha información respecto a sus estructuras terciarias (Paladin et al., 2021).

Se pretende también evaluar la usabilidad de la herramienta desarrollada, sin embargo, la presente investigación no propone una metodología de evaluación. En cambio, se aplicará una metodología recientemente propuesta. El marco de evaluación a considerar fue desarrollado con un enfoque bioinformático, por lo cual su aplicación resulta conveniente. En este punto, cabe mencionar que la evaluación mencionada será llevada a cabo solo en la última etapa del desarrollo y de forma interna dentro del proyecto, lo cual permitirá obtener información retrospectiva acerca del uso de la herramienta.

- **Limitaciones**

El proyecto de tesis está siendo construido en medio de un contexto de pandemia mundial en el que no se permite presencialidad, es por ello que las reuniones con la asesora y todo tipo de apoyo que se reciba para su desarrollo, incluyendo la evaluación de la herramienta por parte de expertos en bioinformática, se realizará de forma remota. Por otro lado, el tiempo destinado para el mismo será de 4 meses, los cuales corresponden a la duración de un semestre académico, teniendo en cuenta las actividades del curso en el que se está desarrollando el proyecto.

- **Identificación de los riesgos del proyecto**

En esta sección se presenta una lista de riesgos identificados con relación al proyecto. Para cada uno de ellos se detalla una breve descripción, sus síntomas, su probabilidad de ocurrencia, su impacto y su severidad. Asimismo, se especifican los planes de mitigación y de contingencia de la gestión de esos riesgos.

Para poder establecer sus características cuantitativas se ha tomado en cuenta ciertos rangos de valores por cada criterio, los cuales se encuentran en la Tabla 23. Teniendo en consideración que la severidad (S) de un riesgo se obtiene a partir del producto entre su probabilidad (P) y su impacto (I).

Leyenda de valores por criterio cuantitativo					
Probabilidad (P)					
Nivel	Muy baja	Baja	Moderada	Alta	Muy alta
Valor	0.10	0.30	0.50	0.70	0.90
Impacto (I)					
Nivel	Muy baja	Baja	Moderada	Alta	Muy alta
Valor	0.05	0.10	0.20	0.40	0.80
Severidad (S)					
Nivel	Baja		Media		Alta
Valor	]-∞; 0.06[		[0.06; 0.14]		]0.14; ∞[

Tabla 23. *Leyenda de valores por criterio cuantitativo: probabilidad, impacto y severidad.*

La lista de riesgos del proyecto se presenta en la Tabla 24, como se muestra a continuación:

Riesgos del proyecto						
Riesgo	**Síntomas**	**P**	**I**	**S**	**Mitigación**	**Contingencia**
Pérdida total o parcial de avances del proyecto debido a fallos en la computadora personal	• La computadora se reinicia o se apaga de forma repentina • El sonido del procesamiento de la computadora se incrementa	0.50	0.40	0.20	• Mantenimiento periódico de la computadora • Creación diaria de una copia de respaldo del proyecto en repositorio en la nube	Compra y uso de una computadora alternativa.
Carencia de conocimiento en temas relacionados a la bioinformática o	• Dificultad en comprender la dinámica de los algoritmos en torno a la	0.70	0.40	0.28	• Realizar una preparación autodidacta, anticipada y frecuente acerca de	Solicitar apoyo a expertos de bioinformática. Se cuenta con la posibilidad

uso de librerías relacionadas a esta área de investigación	predicción de estructuras de proteínas • Dificultad en entendimiento de terminología bioinformática				los temas afines al proyecto en torno a la bioinformática • Recibir capacitaciones sobre bioinformática previo y en paralelo al desarrollo del proyecto	de mantener contacto con la comunidad de expertos por medio de la asesora
Los expertos que aprueban los resultados esperados por cada objetivo no cuentan con disponibilidad para realizarlas	• No se obtiene una respuesta por parte de los expertos luego de una semana de haber enviado la solicitud de revisión	0.30	0.40	0.12	• Planificar con anticipación fechas estimadas de entrega de resultados para su aprobación • Acordar un plazo de respuesta de las revisiones • Establecer medios de comunicación convencionales y alternativos • Preparar una lista de especialistas alternos para que realicen la revisión	Solicitar apoyo a especialistas alternos para poder revisar los resultados de cada objetivo
Las bases de datos de donde se obtienen la información de las estructuras de las proteínas se encuentran inactivas o son dados de baja	• Anuncios en la página web de las bases de datos de un mantenimiento programado • Procesamiento lento o errores frecuentes en la navegación de las bases de datos	0.30	0.40	0.12	• Obtener información de una muestra significativa de estructuras de proteínas de utilidad • Preparar una lista de bases de datos alternativas • Preparar una lista de investigaciones afines al proyecto que ofrecen sus conjuntos de datos de prueba	Solicitar acceso a bases de datos alternativas para capturar la información requerida de las estructuras de proteínas
Los usuarios que realizarán la evaluación de la usabilidad de la	• No se obtiene respuesta por parte de los evaluadores	0.70	0.20	0.14	• Planificar con anticipación la fecha estimada para la evaluación	Contacto con los usuarios alternativos para poder

herramienta no cuentan con disponibilidad para llevarlos a cabo	luego de una semana de haber enviado la solicitud • Se reciben constantes solicitudes de prórroga de la evaluación				de usabilidad • Acordar un plazo de entrega de la evaluación • Establecer medios de comunicación alternativos • Preparar una lista de usuarios alternativos para que realicen la evaluación	llevar a cabo la evaluación de usabilidad
Las actividades programadas que requieren de conectividad de internet se interrumpen o postergan por problemas de latencia	• La conectividad de internet se interrumpe frecuentemente	0.70	0.80	0.56	• Cambio de clave de la red de internet semanalmente para descongestionar la red doméstica por dispositivos conectados • Adquisición de un plan de datos ilimitados	Uso del plan de datos ilimitados para continuar con la actividad remota que se estaba llevando a cabo

Tabla 24. *Riesgo del proyecto identificados.*

- **Estructura de descomposición del trabajo (EDT)**

A continuación, se presenta la estructura de descomposición del trabajo para la planificación del proyecto y su desarrollo a través de una representación esquemática en la Figura 12.

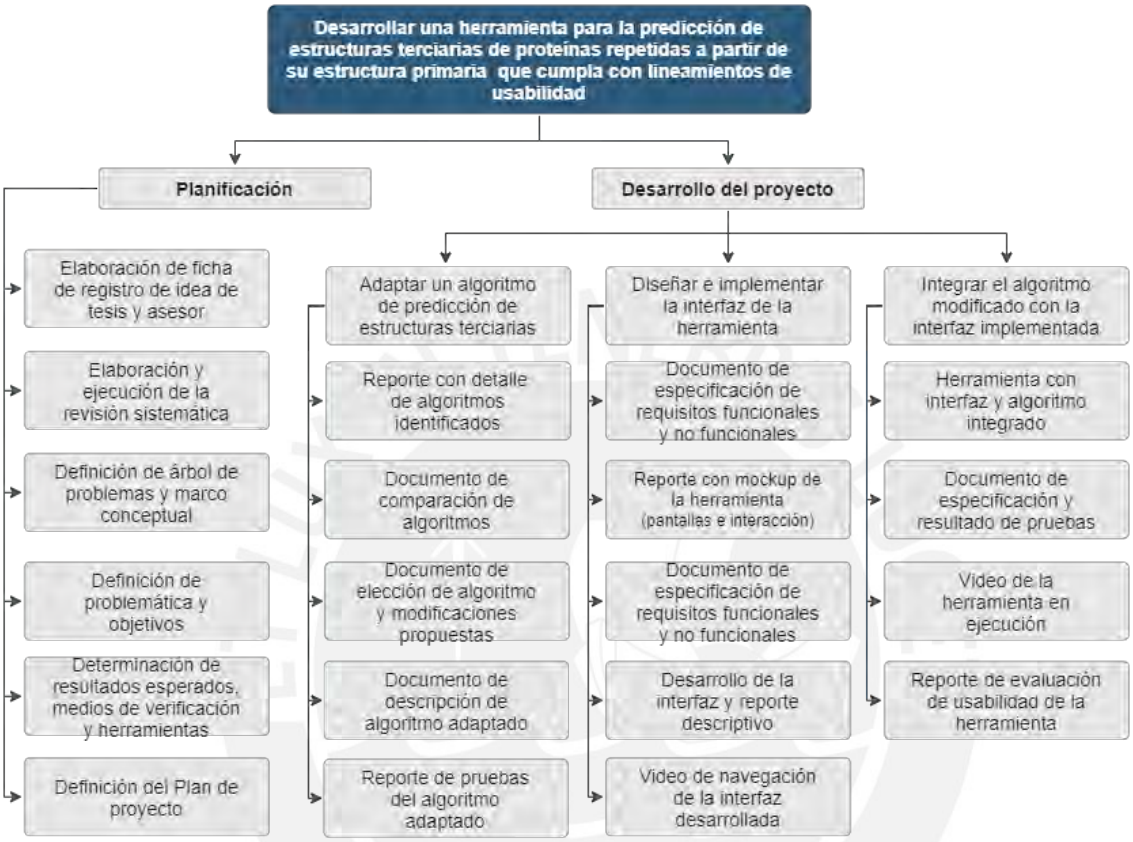

Figura 12. *Estructura de descomposición del trabajo del proyecto. (Elaboración propia).*

- **Lista de tareas**

En la presente sección se encuentra detallada la lista de tareas que se realizarán en el proyecto. La información se encuentra organizada en la Tabla 25, según la etapa del proyecto (planificación y desarrollo), el objetivo específico y resultado esperado en el que se enfoca. Se describe, además, la duración estimada, el esfuerzo asociado y el costo estimado por cada una de las tareas identificadas. La descripción pormenorizada de los esfuerzos asociados y los costos estimados se encuentran en la sección de Costeo del proyecto.

Lista de tareas del proyecto			
Planificación del proyecto			
Tarea	**Duración estimada (días)**	**Esfuerzo asociado (horas-persona)**	**Costo estimado (S/.)**
Elaboración del entregable parcial 1.1 (Tesista)	2	6	300
Elaboración del entregable parcial 1.2 (Tesista)	2	8	400
Elaboración del entregable parcial 1.3 (Tesista)	4	16	800
Elaboración del entregable parcial 1.4 (Tesista)	3	12	600
Elaboración del entregable parcial 1.5 (Tesista)	4	8	400
Elaboración del entregable 1 (Tesista)	2	6	300
Elaboración del entregable parcial 2.1 (Tesista)	4	8	400
Elaboración del entregable 2 (Tesista)	2	10	500
Elaboración del entregable 3 (Tesista)	3	6	300
Elaboración del entregable 4 (Tesista)	4	8	400
Reuniones semanales con la asesora (Tesista y asesora)	13	Tesista : 26 Asesora: 26	5,200
Reuniones con profesores del curso (Tesista)	13	26	13,000
Elaboración de presentación	2	6	300
Presentación de proyecto (Tesista)	1	0.5	25

Desarrollo del proyecto			
O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas			
Tarea	**Duración estimada (días)**	**Esfuerzo asociado (horas-persona)**	**Costo estimado (S/.)**
R1. Lista de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general			
Identificación de algoritmos de interés y obtención de información detallada de los mismos	2	10	500
Obtención de información adicional sobre la tecnología que utilizan los algoritmos	1	5	250
Redacción del reporte que contiene el detalle de los algoritmos identificados	1	6	300
R2. Algoritmo seleccionado y planteamiento de las modificaciones necesarias al mismo			
Verificación del funcionamiento de los algoritmos identificados	5	40	2,000
Revisión de las fuentes de los algoritmos para entender la lógica de trabajo de cada uno de ellos	3	6	300
Redacción del documento que contenga la comparación de los algoritmos	1	4	200
Selección de un algoritmo y análisis de posibles modificaciones	4	6	300
Redacción del documento que justifique la elección y describa las modificaciones propuestas al algoritmo seleccionado	1	4	200
R3. Implementación del algoritmo adaptado			
Aplicación de las modificaciones planteadas al algoritmo seleccionado	2	12	600
Recolección de los datos de entrada y datos de salida esperados para realizar el benchmarking	1	6	300
Preparación del algoritmo adaptado	9	50	2,500

Evaluación del performance del algoritmo adaptado (datos obtenidos vs datos esperados)	3	15	750
Redacción del documento de descripción funcional del algoritmo adaptado incluyendo código fuente	2	4	200
Redacción del reporte con los resultados de las pruebas del algoritmo adaptado	2	4	200
Solicitud y obtención de aprobación experta del documento que contiene la información de la evaluación realizada	3	6	3,600
Redacción del entregable de tesis	2	6	300
O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria			
Tarea	**Duración estimada (días)**	**Esfuerzo asociado (horas-persona)**	**Costo estimado (S/.)**
R1. Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta			
Definición de los requisitos funcionales de la herramienta	1	4	200
Definición de los requisitos no funcionales de la herramienta	1	4	200
Verificación de compatibilidad entre herramientas y refinamiento de los requisitos no funcionales de la herramienta	1	3	150
Redacción del documento con la especificación de los requisitos funcionales y no funcionales de la herramienta	1	4	200
Solicitud y obtención de aprobación experta del documento con la especificación de requisitos funcionales y no funcionales de la herramienta	3	6	3,600

R2. Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria			
Definición y diseño del diagrama de flujo de la herramienta	1	5	250
Diseño de un prototipo de baja fidelidad de la interfaz de la herramienta (wireframes)	1	3	150
Diseño un prototipo de media fidelidad de la interfaz de la herramienta (mockups)	3	20	1,000
Adición de interacción y refinamiento del prototipo de la interfaz de la herramienta (mockup de alta fidelidad)	1	4	200
Redacción del reporte con el mockup de la interfaz de la herramienta propuesta incluyendo pantallas y navegación	2	4	200
Solicitud y obtención de aprobación experta del reporte con el prototipo de la interfaz de la herramienta propuesta incluyendo pantallas y navegación	4	6	3,600
R3. Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas			
Preparación del ambiente de desarrollo de la interfaz visual (entorno, librerías, frameworks, IDE…)	1	4	200
Construcción de los objetos gráficos de la interfaz (pantallas del mockup)	5	30	1,500
Programación de la interacción entre los objetos gráficos de la interfaz	7	32	1,600
Verificación de la inclusión de los requisitos funcionales y no funcionales planteados en el R1 y el cumplimiento del flujo completo del R2 en la interfaz	3	6	300
Redacción del reporte descriptivo de la interfaz desarrollada	1	3	150
Producción del video de la interfaz en ejecución	2	6	300
Redacción del entregable de tesis	2	6	300

O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria			
Tarea	**Duración estimada (días)**	**Esfuerzo asociado (horas-persona)**	**Costo estimado (S/.)**
R1. Interfaz que permite utilizar el algoritmo adaptado			
Integración del algoritmo adaptado y la interfaz desarrollada	5	25	1,250
Validación de la compatibilidad entre el algoritmo adaptado y la interfaz desarrollada	1	4	200
Verificación de la inclusión de los requisitos funcionales y no funcionales, el cumplimiento del flujo completo y el funcionamiento del algoritmo adaptado en la herramienta	1	4	200
Redacción del reporte de funcionamiento de la herramienta implementada	2	4	200
R2. Pruebas funcionales de la herramienta de predicción con el algoritmo integrado			
Especificar las pruebas a realizar en la herramienta	2	4	200
Obtener los datos de entrada y los datos de salida esperados de la herramienta	1	4	200
Programación de pruebas especificadas	2	8	400
Ejecución de las pruebas especificadas en la herramienta desarrollada	3	15	750
Redacción del documento de especificación y resultado de pruebas	2	4	200
Producción del video de la herramienta en ejecución	1	6	300
Solicitud y obtención de aprobación experta del documento de especificación y resultado de pruebas	3	6	3,600

R3. Evaluación de usabilidad de la herramienta implementada			
Definición de la metodología de evaluación de usabilidad a aplicar	1	4	200
Aplicación de la metodología de evaluación de usabilidad seleccionada	2	6	300
Reajuste de la herramienta por observaciones de la evaluación de usabilidad	3	15	750
Redacción del reporte de la evaluación de usabilidad de la herramienta implementada	1	4	200
Solicitud y obtención de aprobación experta del reporte de evaluación de usabilidad realizada	3	6	3,600
Redacción del entregable de tesis	2	4	200
General			
Tarea	**Duración estimada (días)**	**Esfuerzo asociado (horas-persona)**	**Costo estimado (S/.)**
Reuniones semanales con la asesora (Tesista y asesora)	13	Tesista : 26 Asesora: 26	5,200
Reuniones o exposiciones semanales con los profesores del curso (Tesista)	12	24	1,200
Elaboración de presentación	2	6	300
Presentación de proyecto (Tesista)	1	0.5	25

Tabla 25. *Lista de tareas del proyecto por etapa del proyecto, objetivo específico y resultado esperado al que pertenecen.*

- **Cronograma del proyecto**

En esta sección se presenta el cronograma del proyecto de tesis, tanto para la planificación como para el desarrollo del mismo. En la Tabla 26, se detalla la fecha de inicio y la fecha de fin por cada tarea especificada.

Cronograma del proyecto		
Planificación del proyecto		
Tarea	**Inicio**	**Fin**
Elaboración del entregable parcial 1.1	30/03/2021	31/03/2021
Elaboración del entregable parcial 1.2	06/04/2021	07/04/2021
Elaboración del entregable parcial 1.3	13/04/2021	16/04/2021
Elaboración del entregable parcial 1.4	20/04/2021	22/04/2021
Elaboración del entregable parcial 1.5	27/04/2021	30/04/2021
Elaboración del entregable 1	04/05/2021	05/05/2021
Elaboración del entregable parcial 2.1	11/05/2021	14/05/2021
Elaboración del entregable 2	18/05/2021	19/05/2021
Elaboración del entregable 3	08/06/2021	10/06/2021
Elaboración del entregable 4	22/06/2021	25/06/2021
Reuniones semanales con la asesora	30/03/2021 06/04/2021 13/04/2021 20/04/2021 27/04/2021 04/05/2021 11/05/2021 18/05/2021 25/05/2021 01/06/2021 08/06/2021 15/06/2021 22/06/2021	30/03/2021 06/04/2021 13/04/2021 20/04/2021 27/04/2021 04/05/2021 11/05/2021 18/05/2021 25/05/2021 01/06/2021 08/06/2021 15/06/2021 22/06/2021
Reuniones con profesores del curso	29/03/2021 05/04/2021 12/04/2021 19/04/2021 26/04/2021	29/03/2021 05/04/2021 12/04/2021 19/04/2021 26/04/2021

	03/05/2021 10/05/2021 17/05/2021 24/05/2021 31/05/2021 07/06/2021 14/06/2021 21/06/2021	03/05/2021 10/05/2021 17/05/2021 24/05/2021 31/05/2021 07/06/2021 14/06/2021 21/06/2021
Elaboración de presentación	28/06/2021	29/06/2021
Presentación de proyecto	05/07/2021	05/07/2021
Desarrollo del proyecto		
O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas	23/08/2021	16/09/2021
R1. Lista de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general	23/08/2021	25/08/2021
Identificación de algoritmos de interés y obtención de información detallada de los mismos	23/08/2021	24/08/2021
Obtención adicional sobre la tecnología que utilizan los algoritmos	25/08/2021	25/08/2021
Redacción del reporte que contiene el detalle de los algoritmos identificados	25/08/2021	25/08/2021
R2. Algoritmo seleccionado y planteamiento de las modificaciones necesarias al mismo	25/08/2021	01/09/2021
Verificación del funcionamiento de los algoritmos identificados con datos de prueba	25/08/2021	29/08/2021
Revisión de las fuentes de los algoritmos para entender la lógica de trabajo de cada uno de ellos	26/08/2021	28/08/2021
Redacción del documento que contenga la comparación de los algoritmos	29/08/2021	29/08/2021
Selección de un algoritmo y análisis de posibles modificaciones (60%)	29/08/2021	31/08/2021
Selección de un algoritmo y análisis de posibles modificaciones (40%)	31/08/2021	01/09/2021
Redacción del documento que justifique la elección y describa las modificaciones propuestas al algoritmo seleccionado	01/09/2021	01/09/2021

R3. Implementación del algoritmo adaptado	01/09/2021	16/09/2021
Aplicación de las modificaciones planteadas al algoritmo seleccionado	01/09/2021	02/09/2021
Recolección de los datos de entrada y datos de salida esperados para realizar el benchmarking	02/09/2021	02/09/2021
Preparación del algoritmo adaptado	03/09/2021	11/09/2021
Evaluación del performance del algoritmo adaptado (datos obtenidos vs datos esperados)	11/09/2021	13/09/2021
Redacción del documento de descripción funcional del algoritmo adaptado incluyendo código fuente	10/09/2021	11/09/2021
Redacción del reporte con los resultados de las pruebas del algoritmo adaptado	12/09/2021	13/09/2021
Solicitud y obtención de aprobación experta del documento que contiene la información de la evaluación realizada	14/09/2021	16/09/2021
Redacción del entregable de tesis	15/09/2021	16/09/2021
O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	17/09/2021	10/10/2021
R1. Lista de requisitos funcionales y no funcionales para el desarrollo de la herramienta propuesta	17/09/2021	21/09/2021
Definición de los requisitos funcionales de la herramienta	17/09/2021	17/09/2021
Definición de los requisitos no funcionales de la herramienta	17/09/2021	17/09/2021
Verificación de compatibilidad entre herramientas y refinamiento de los requisitos no funcionales de la herramienta	18/09/2021	18/09/2021
Redacción del documento con la especificación de los requisitos funcionales y no funcionales de la herramienta	18/09/2021	18/09/2021
Solicitud y obtención de aprobación experta del documento con la especificación de requisitos funcionales y no funcionales de la herramienta	19/09/2021	21/09/2021
R2. Prototipo de la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	21/09/2021	28/09/2021
Definición y diseño del diagrama de flujo de la herramienta	21/09/2021	21/09/2021

Diseño de un prototipo de baja fidelidad de la herramienta (wireframes)	21/09/2021	21/09/2021
Diseño de un prototipo de media fidelidad de la herramienta (mockups)	22/09/2021	24/09/2021
Adición de interacción y refinamiento del prototipo de la interfaz de la herramienta (mockup de alta fidelidad)	24/09/2021	24/09/2021
Redacción del reporte con el mockup de la herramienta propuesta incluyendo pantallas y navegación	24/09/2021	25/09/2021
Solicitud y obtención de aprobación experta del reporte con el prototipo de la interfaz de la herramienta propuesta incluyendo pantallas y navegación	25/09/2021	28/09/2021
R3. Interfaz de la herramienta propuesta para realizar la predicción de la estructura terciaria de proteínas repetidas	26/09/2021	10/10/2021
Preparación del ambiente de desarrollo de la interfaz visual (entorno, librerías, frameworks, IDE…)	26/09/2021	26/09/2021
Construcción de los objetos gráficos de la interfaz (pantallas del mockup)	26/09/2021	30/09/2021
Programación de la interacción entre los objetos gráficos de la interfaz	30/09/2021	06/10/2021
Verificación de la inclusión de los requisitos funcionales y no funcionales planteados en el R1 y el cumplimiento del flujo completo del R2 en la interfaz	06/10/2021	08/10/2021
Redacción del reporte descriptivo de la interfaz desarrollada	08/10/2021	08/10/2021
Producción del video de la interfaz en ejecución	08/10/2021	09/10/2021
Redacción del entregable de tesis	09/10/2021	10/10/2021
O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria	11/10/2021	31/10/2021
R1. Interfaz que permite utilizar el algoritmo adaptado	11/10/2021	17/10/2021
Integración del algoritmo adaptado y la interfaz desarrollada	11/10/2021	15/10/2021
Validación de la compatibilidad entre el algoritmo adaptado y la interfaz desarrollada	15/10/2021	15/10/2021
Verificación de la inclusión de los requisitos funcionales y no funcionales, el cumplimiento del flujo completo y el funcionamiento del algoritmo adaptado en la herramienta	15/10/2021	15/10/2021

Redacción del reporte de funcionamiento de la herramienta implementada	16/10/2021	17/10/2021
R2. Pruebas funcionales de la herramienta de predicción con el algoritmo integrado	18/10/2021	26/10/2021
Especificación de las pruebas a realizar en la herramienta	18/10/2021	19/10/2021
Obtención de los datos de entrada y los datos de salida esperados de la herramienta	19/10/2021	19/10/2021
Programación de pruebas especificadas	20/10/2021	21/10/2021
Ejecución de las pruebas especificadas en la herramienta desarrollada	21/10/2021	23/10/2021
Redacción del documento de especificación y resultado de pruebas	18/10/2021	24/10/2021
Producción del video de la herramienta en ejecución	24/10/2021	24/10/2021
Solicitud y obtención de aprobación experta del documento de especificación y resultado de pruebas	24/10/2021	26/10/2021
R3. Evaluación de usabilidad de la herramienta implementada	25/10/2021	31/10/2021
Definición de la metodología de evaluación de usabilidad a aplicar	25/10/2021	25/10/2021
Aplicación de la metodología de evaluación de usabilidad seleccionada	26/10/2021	27/10/2021
Reajuste de la herramienta por observaciones de la evaluación	27/10/2021	29/10/2021
Redacción del reporte de la evaluación de usabilidad de la herramienta implementada	29/10/2021	29/10/2021
Solicitud y obtención de aprobación experta del reporte de evaluación de usabilidad realizada	29/10/2021	31/10/2021
Redacción del entregable de tesis	30/10/2021	31/10/2021
General		
Tarea	**Inicio**	**Fin**
Reuniones semanales con la asesora del proyecto	24/08/2021 31/08/2021 07/09/2021 14/09/2021 21/09/2021 28/09/2021	24/08/2021 31/08/2021 07/09/2021 14/09/2021 21/09/2021 28/09/2021

	05/10/2021 12/10/2021 26/10/2021 02/11/2021 16/11/2021 30/11/2021 03/12/2021	05/10/2021 12/10/2021 26/10/2021 02/11/2021 16/11/2021 30/11/2021 03/12/2021
Reuniones o exposiciones semanales del curso	23/08/2021 30/08/2021 06/09/2021 13/09/2021 20/09/2021 27/09/2021 04/10/2021 11/10/2021 25/10/2021 01/11/2021 15/11/2021 29/11/2021	23/08/2021 30/08/2021 06/09/2021 13/09/2021 20/09/2021 27/09/2021 04/10/2021 11/10/2021 25/10/2021 01/11/2021 15/11/2021 29/11/2021
Elaboración de presentación	01/12/2021	01/12/2021
Presentación de proyecto	13/12/2021	13/12/2021

Tabla 26. *Cronograma del proyecto por etapa del proyecto, objetivo específico y resultado esperado*

- **Lista de recursos**

En esta sección se presentan todos los recursos que serán necesarios para el proyecto de tesis.

- **Personas involucradas y necesidades de capacitación**

A continuación, se describe a las personas involucradas en todo el proceso de desarrollo del presente proyecto de fin de carrera.

- Asesora de Tesis: Dra. Layla Hirsh Martinez. La asesora principal del proyecto de tesis cuenta con un doctorado en Biociencia y Biotecnología. Ella es investigadora registrada en el Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica (CONCYTEC), es egresada y docente de la Pontificia Universidad Católica del Perú (PUCP) en la sección de Ingeniería Informática y colaboradora del Instituto de Bioinformática Europeo (EMBL-EBI).

- ❖ Expertos en proteínas repetidas: Profesionales con experiencia en el campo de la bioinformática y proteínas repetidas, quienes se encargarán de evaluar los resultados esperados del proyecto.
- ❖ Tesista: La estudiante que desarrolla el proyecto de tesis. Se requerirá de capacitación constante sobre proteínas repetidas.

- o **Materiales requeridos para el proyecto**

En este apartado se describen los materiales que serán necesarios para llevar a cabo el proyecto de fin de carrera.

- ❖ Conexión a internet: Se requiere de conexión estable a una red de internet con la cual se pueda acceder a todos los recursos remotos para desarrollar el proyecto.
- ❖ Plan de datos: Será un material requerido de forma contingente para poder continuar con el proyecto ante cualquier fallo en la conexión a internet anteriormente mencionada.

- o **Estándares utilizados en el proyecto**

El presente proyecto utilizará el marco de trabajo para el desarrollo ágil denominado Scrum. Se tendrán en cuenta algunas de sus buenas prácticas como la planificación eficaz de los entregables, el incremento de valor al producto de forma iterativa y la comunicación constante entre los miembros involucrados en el proyecto.

- o **Equipamiento requerido**

Para el desarrollo de la tesis se requiere de una computadora que cuente con acceso a internet.

- o **Herramientas requeridas**

A continuación, se describen las herramientas que serán necesarias para llevar a cabo el presente proyecto de fin de carrera:

- ❖ Algoritmos de predicción de proteínas: Con apoyo de la revisión sistemática realizada se obtendrán algoritmos para posteriormente adaptarlos al problema

- GitHub: Se manejará un repositorio que aloje a los algoritmos en evaluación.
- Herramientas para la programación: Se requieren de herramientas para el desarrollo de la programación tales como Jupyter Notebook y Visual Studio Code.
- Lenguajes de programación: Se utilizarán diversos lenguajes de programación para poder implementar la herramienta objetivo del proyecto. Algunos lenguajes de programación a utilizar serán Python, CSS, JavaScript y HTML como lenguaje de hipertexto para la programación web.
- Librerías de programación: Se requieren de algunas librerías de código abierto como Pytorch, Numpy y Bootstrap.
- Bases de datos: Se recurrirá a diversas bases de datos web como Protein Data Bank y RepeatsDB para obtener diversas estructuras de proteínas.
- Figma: Es considerada una de las mejores herramientas de prototipado web. Se utilizará para desarrollar uno de los objetivos del proyecto.
- LucidChart: Es una aplicación para construir el diagrama de flujo de la interfaz.
- Zoom: Las interacciones entre los involucrados del proyecto se realizarán a través de reuniones online.
- Noun Project: Hace referencia a un proyecto web para la obtención de iconos a utilizar en la herramienta objetivo.
- Frameworks: Se utilizarán diversos frameworks como son Vue.js y Flask tanto para poder construir la interfaz de la herramienta como para poder integrarla con el algoritmo adaptado.
- Xbox Game Bar: Es una herramienta de escritorio que se utilizará para videograbar la navegación de la interfaz desarrollada.
- Nginx: Un servidor web donde se desplegará la herramienta desarrollada.
- Amazon Web Services: Se utilizarán algunos servicios de AWS para poder instanciar y potenciar la herramienta en la nube.
- Herramienta de evaluación de usabilidad: Se considera conveniente usar una herramienta innovadora y reciente que se adapte a la herramienta desarrollada y a su contexto.

- **Costeo del Proyecto**

En este apartado se detalla el costo estimado de cada recurso involucrado en la planificación y desarrollo del presente proyecto de fin de carrera. En la Tabla 27 se especifica la descripción, la unidad de medida, la cantidad, el valor unitario y los montos calculados por cada ítem utilizado.

Costeo del proyecto						
Ítem	**Descripción**	**Unidad**	**Cantidad**	**Valor Unitario (S/.)**	**Monto Parcial (S/.)**	**Monto Total (S/.)**
0	**Costo total del proyecto**	---	---	---	---	65,916
1	**Participantes del proyecto**	---	---	---	---	58,800
1.1	Asesora de tesis	Horas	52	150	7,800	
1.2	Expertos en bioinformática	Horas	30	600	18,000	
1.5	Tesista	Horas	660	50[34]	33,000	
2	**Materiales e insumos**	---	---	---	---	560
2.1	Internet	Mes	8	40	320	
2.2	Plan de datos	Mes	8	30	240	
3	**Bienes y equipos**	---	---	---	---	3,500
3.1	Laptop ASUS X515EA CI7	Equipo	1	3500	3,500	
4	**Licencias de Software[35]**	---	---	---	---	3,056
4.1	Instancia EC2 AWS On-demand	Mes	8	381.60[36]	3,056	

Tabla 27. *Costeo del proyecto.*

[34] Cálculo en base a la retribución por hora de un instructor en la Pontificia Universidad Católica del Perú.

[35] La mayoría de las herramientas que se utilizarán son de libre acceso.

[36] Basado en la calculadora de precios de la web de Amazon Web Services. El tipo de cambio para cuando se ha realizado el cálculo del costo de la instancia es 3.89.

Anexo D: Reporte de algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general

Este anexo contiene la información recopilada de los algoritmos que han sido identificados con relación a la explotación de los datos existentes de secuencias de aminoácidos de proteínas en general para predecir sus estructuras tridimensionales. Su contenido abarca la delimitación de la búsqueda, la presentación de los algoritmos identificados y la descripción de su funcionamiento en general, así como la información sobre sus datos de entrada y de salida, modo de procesamiento, recurso computacional y tiempo requerido. Se debe tener en cuenta el alcance del presente proyecto descrito con más detalle en el Anexo C.

1. Introducción

Un aspecto clave del presente proyecto de fin de carrera corresponde a la adaptación de un algoritmo dedicado a la predicción de estructuras terciarias de proteínas en general para aplicarlo a proteínas repetidas. Para lograrlo es necesario identificar, en primer lugar, una serie de algoritmos que puedan explotar los datos de las estructuras primarias de estas proteínas para poder transformarlos en información de sus estructuras tridimensionales. En este documento se presentarán los detalles de los algoritmos que hayan sido identificados. Con esa información se procederá a seleccionar uno de ellos para poder plantear una serie de modificaciones que nos permitan cumplir los objetivos del proyecto. Esto último mencionado corresponde a un siguiente resultado esperado del mismo objetivo abordado en este anexo. Se puede revisar más detalle de este en el Capítulo 4 Adaptación de algoritmos de predicción de estructuras terciarias de proteínas en general.

2. Delimitación de búsqueda de algoritmos de predicción

Mientras más complejo sea un problema y se cuente con gran cantidad de datos sobre este, las técnicas de inteligencia artificial, en particular, las de aprendizaje profundo, se hacen cada vez más reconocidas (Zhao & Gong, 2019). Su inclusión dentro de las propuestas de solución de diversas problemáticas bioinformáticas como la predicción de estructuras de proteínas ha marcado un hito importante (Greener et al., 2019). Esto

dado que, en los diversos experimentos de CASP[37], han demostrado una mejora significativa tanto en desempeño como en eficacia respecto a los métodos usualmente aplicados (Greener et al., 2019; Zheng et al., 2019).

Todos los métodos de predicción que las aplican se pueden categorizar en tres tipos, en base a los datos de entrada y de salida de su arquitectura: “end-to-X learning”, “X-to-end learning” y “end-to-end learning” (Laine et al., 2021). El término “end” hace referencia a los datos de entrada y salida idóneos, en este caso, a la secuencia de aminoácidos y a las coordenadas 3D de la estructura de la proteína, respectivamente (Laine et al., 2021). El término “X” corresponde a cualquier dato de entrada o salida intermedio que los métodos requieran u obtengan a partir de su aplicación, como pueden ser los alineamientos multisecuencia o los mapas de contacto o distancia, entre otros (Laine et al., 2021).

Hoy en día se cuenta con una vasta cantidad de datos de estructuras primarias: más de doscientas catorce millones de secuencias de proteínas en general albergadas en UniProt (UniProt Consortium, 2021b), un número que crece exponencialmente. De ellas, aproximadamente ocho mil son proteínas repetidas con información estructural detallada (Paladin et al., 2021). Con ello, se confirma la posibilidad de aplicación de técnicas de aprendizaje profundo al contexto de proteínas repetidas, en específico. A partir de lo mencionado, cabe precisar que este proyecto busca aprovechar los datos de las estructuras primarias de las proteínas para predecir sus estructuras terciarias, por lo cual la búsqueda se delimitó y se centró en los algoritmos de aprendizaje profundo con una arquitectura “end-to-end”.

3. Presentación de algoritmos identificados

Para poder capturar los algoritmos que serán presentados a continuación se ha tomado en consideración los mismos motores de búsqueda utilizados en la revisión sistemática del Capítulo 3. Estado del Arte, además de partir de muchas de las investigaciones obtenidas en ese capítulo para revisar nuevas propuestas que hayan surgido luego de llevar a cabo dicha revisión. Como resultado de esa búsqueda de información, se obtuvieron dos algoritmos dedicados a la predicción de estructuras terciarias de proteínas a partir del ingreso de las secuencias de aminoácidos. A continuación, se

[37] Evaluación Crítica de las Técnicas para la Predicción de Estructuras de Proteínas (CASP, por sus siglas en inglés).

presentará una descripción de los mismos, así como de sus características más resaltantes.

3.1 DMPfold

Este algoritmo es una continuación del método de predicción de contacto conocido como DeepMetaPSICOV (Greener et al., 2019). No obstante, se evidencia un cambio de perspectiva. Antes el método se centraba en la generación de la matriz de contacto entre los residuos de una proteína. Ahora se enfoca en la predicción de límites de distancia interatómica, ángulos de torsión y enlaces de hidrógeno de la cadena principal a través de redes neuronales profundas (Greener et al., 2019).

Con toda la información generada construye modelos de estructuras terciarias de proteínas mediante un proceso iterativo (Greener et al., 2019). Tanto el proceso de predicción de contacto como el modelado de la estructura usualmente se consideran como procedimientos aislados, pero DMPfold crea un solo proceso combinando ambas etapas (Greener et al., 2019).

Se considera que la predicción de distancias entre los residuos brinda más información para el modelado de estructuras que los mapas de contactos, los cuales generalmente son binarios (Greener et al., 2019). En ese mismo sentido, esta es una técnica robusta que utiliza la información de la covariación para obtener un único modelo tridimensional de la proteína ingresada aprovechando los recursos computacionales de forma eficiente.

Solo se requiere de los datos disponibles de la secuencia de la proteína para que en cuestión de pocas horas y con la capacidad de una computadora estándar de un núcleo se ejecute el algoritmo y se obtenga una predicción muy precisa de su estructura terciaria (Greener et al., 2019). En el caso de una proteína de 200 aminoácidos, fue necesario de aproximadamente 3 horas de ejecución, teniendo en cuenta que 1 hora y media fue destinada para la generación del modelo a partir de las distancias de residuos predichas (Greener et al., 2019).

Cabe mencionar que el código fuente de DMPfold, la documentación completa y el modelo entrenado de la red neuronal se encuentran disponibles en la web para el libre uso de los interesados. La información se detalla en el siguiente enlace: https://github.com/psipred/DMPfold.

3.1.1 Métodos aplicados

DMPfold utiliza redes neuronales profundas previamente entrenadas para poder predecir distribuciones de probabilidad de distancia entre residuos, los enlaces de hidrógeno de la cadena principal y los ángulos de torsión, a partir de los datos de entrada ingresados al algoritmo (Greener et al., 2019). La información obtenida se envía hacia CNS (Brunger, 2007a), una herramienta con la cual se modelarán las estructuras terciarias (Greener et al., 2019). En la Figura 13 se observa el diagrama general del proceso descrito.

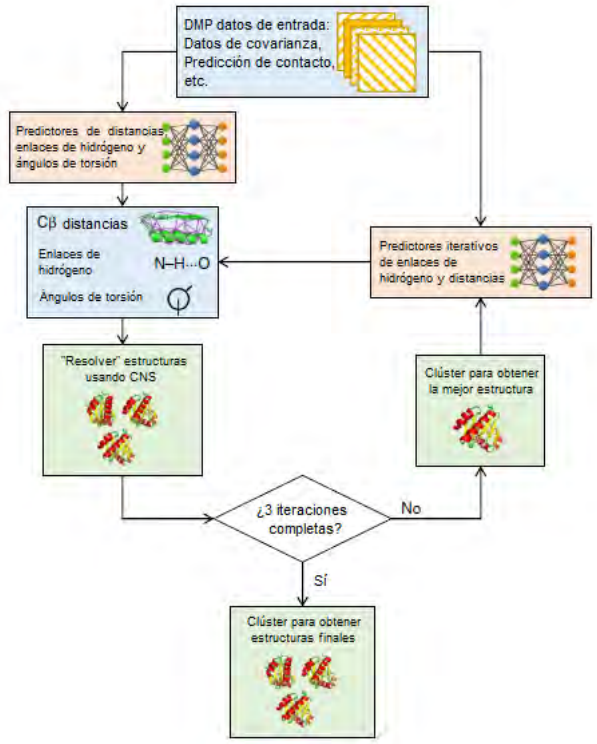

Figura 13. *Diagrama general del proceso DMPfold. Adaptado de* (Greener et al., 2019).

Para las diversas predicciones se utilizaron las arquitecturas detalladas en la Figura 14. En el caso de la predicción de los enlaces de hidrógeno se utiliza una arquitectura que devuelve un mapa de contactos, donde las filas representan los contactos de los donantes y las columnas a los aceptores (Greener et al., 2019). Para la predicción de distancias se utiliza una arquitectura con una capa softmax que devuelve un mapa de distancias en forma de 20 canales de probabilidades agrupados (Greener et al., 2019). Cada canal corresponde a un rango de distancias. Esto conllevaría al incremento del poder de representación de la red neuronal, para lo cual se reemplazó las dos capas

convolucionales de las capas de bloque residual por una única capa softmax convolucional, con cuatro unidades maxout por capa (Greener et al., 2019).

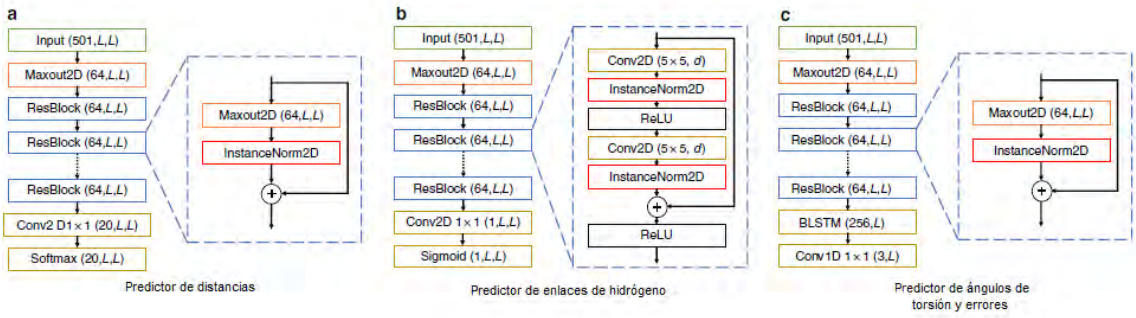

Figura 14. Arquitectura de los predictores de DMPfold. a). Predictor de distancias, b). Predictor de enlaces de hidrógeno, y, c). Predictor de ángulos de torsión y errores. Adaptado de (Greener et al., 2019).

3.2 trRosetta

Esta técnica ha ido progresando a medida que han ido surgiendo novedades en torno a la aplicación de diversas arquitecturas de aprendizaje profundo dentro de la predicción de estructuras. Para esta versión se ha considerado el hecho de que la información de distancias es mucho más conveniente y ventajosa para el modelado de estructuras que solo la predicción de los contactos entre residuos de proteínas (Yang et al., 2020). No obstante, se incluye un factor adicional: la predicción de orientaciones interresiduales (Yang et al., 2020). Este aspecto resultaría ser clave y positivamente influyente en la mejora de la predicción de la estructura de las proteínas en general.

Esta técnica denominada como “transform-restrained Rosetta” o trRosetta es un algoritmo que predice la estructura tridimensional de las proteínas de forma rápida y con un alto índice de precisión, ya que utiliza redes neuronales profundas (Yang et al., 2020). Recibe como dato de entrada la secuencia de aminoácidos de una proteína y retorna como dato de salida un total de cinco de los mejores modelos de estructuras terciarias de la proteína (Yang et al., 2020). Dado que en el artículo del algoritmo no se precisa información del tiempo de ejecución, se validó a través del servicio web que para el caso de una proteína de 200 aminoácidos, fueron necesarias 2 horas de ejecución aproximadamente, teniendo en cuenta que la espera de atención dentro de la cola de ejecuciones tuvo una duración de casi 30 minutos.

Cabe destacar que trRosetta ha demostrado predecir estructuras muy acertadas para el caso de las proteínas diseñadas, aun cuando este tipo de proteínas tienen una ausencia

de señales de coevolución (Yang et al., 2020). A partir de esto, se deduce que el modelo es capaz de aprender sobre las propiedades fundamentales de las estructuras de las proteínas y el efecto de las mutaciones, así como de la relación entre la secuencia y la estructura tridimensional (Yang et al., 2020).

Por último, es importante mencionar que los alineamientos múltiples de secuencias para las proteínas pertenecientes a los conjuntos de datos, los archivos fuente par la predicción de geometrías interresiduales y el protocolo para el modelado y la generación de estructuras restringida por transformación se encuentran disponibles para el público en general a través del siguiente enlace: https://yanglab.nankai.edu.cn/trRosetta/ y https://github.com/gjoni/trRosetta.

3.2.1 Métodos aplicados

En adición a la información tradicional como los mapas de contacto y los mapas de distancias que pueden ser predichos por trRosetta, se ha incluido la capacidad de predecir seis coordenadas de orientación espacial para cada par de residuos (i,j) de las proteínas (Yang et al., 2020). Estas coordenadas (d, ω, θij, φij, θji, φji) serán provechosas para ubicar en el espacio tridimensional a un aminoácido respecto al otro y son capturadas en base a las redes neuronales residuales profundas (Yang et al., 2020). En la Figura 15 sección A, se puede apreciar un esquema del funcionamiento de las coordenadas mencionadas. En la sección B de la misma figura, se observa la arquitectura de las redes neuronales que las predicen.

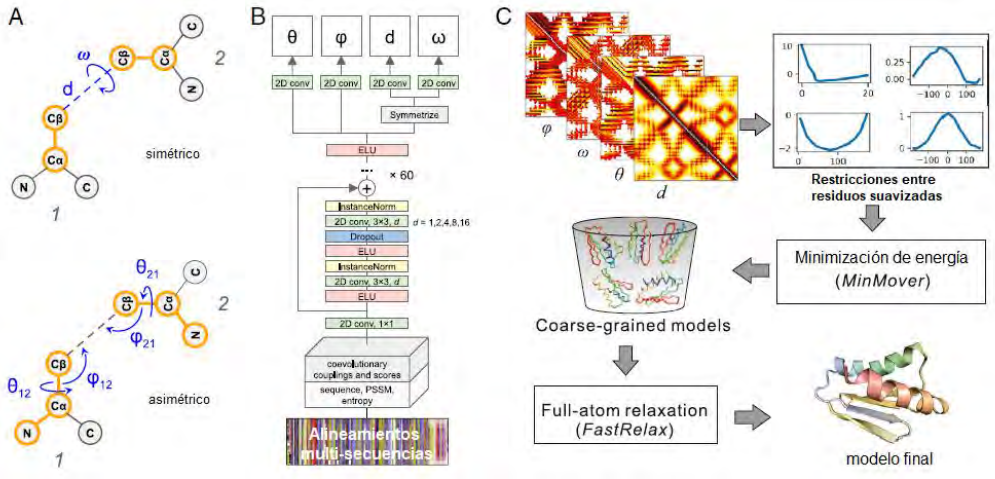

Figura 15. Diagrama general de las predicciones realizadas por trRosetta. A). Representación de una transformación de los residuos por las coordenadas predichas, B). Arquitectura de la red neuronal profunda de trRosetta para la predicción de distancias interresiduales, y C). Esquema de la aplicación de las restricciones del protocolo de Rosetta para la generación de los modelos tridimensionales. Adaptado de (Yang et al., 2020).

Una vez que se cuenta con la información estructural como las distancias, los contactos y las coordenadas por cada par de residuos, se utiliza el protocolo de construcción de modelos estructurales 3D de Rosetta para generar la estructura terciaria de la proteína ingresada (Yang et al., 2020), tal como se observa en el esquema de la sección C, de la Figura 15.

Anexo E: Reporte comparativo de los algoritmos identificados capaces de explotar datos existentes de estructuras primarias para predecir estructuras terciarias de proteínas en general

En este anexo se encuentra el documento del reporte de comparación de los algoritmos identificados con relación a la explotación de los datos existentes de secuencias de aminoácidos de proteínas en general para predecir sus estructuras terciarias. Su contenido abarca una breve presentación de los algoritmos identificados, el proceso de verificación del funcionamiento de los algoritmos identificados y, finalmente, la comparación de los algoritmos respecto a sus características generales.

1. Introducción

De momento se cuenta con dos algoritmos identificados, DMPfold y trRosetta, seleccionados por su capacidad de explotación de datos de secuencias de aminoácidos de las proteínas en general. Por consiguiente, es necesario verificar su funcionamiento y realizar una comparación de sus características generales para luego elegir solo un algoritmo y proseguir con su adaptación en base a las proteínas repetidas. El presente documento cubrirá ambos aspectos.

2. Algoritmos identificados

A partir de la delimitación de algoritmos y la búsqueda realizada que puede analizarse con más detalle en el Anexo C, se lograron capturar do algoritmos que cumplen con los requisitos propuestos: DMPfold y trRosetta. A continuación, se presentará una breve descripción de cada uno de ellos.

2.1 DMPfold

DMPfold es la versión actualizada del método de predicción de contactos entre residuos de proteínas denominado como DeepMetaPSICOV (Greener et al., 2019). Este algoritmo basado en redes neuronales profundas predice límites de distancias interatómicas, enlaces de hidrógeno de la cadena principal y los ángulos de torsión, a partir de uno o más datos de entrada. De estos últimos, el más relevante es la secuencia de aminoácidos de la proteína en cuestión. Una vez se cuenta con estas características predichas se utiliza la herramienta CNS (Brunger, 2007a) para poder modelar la estructura tridimensional.

2.2 trRosetta

La técnica denominada como “transform-restrained Rosetta” o trRosetta es un algoritmo que predice la estructura tridimensional de las proteínas con un alto índice de precisión gracias a la aplicación de redes neuronales profundas (Yang et al., 2020). Requiere del ingreso de la estructura primaria o la secuencia de residuos de la proteína para poder predecir una serie de atributos cruciales como el mapa de distancias, el mapa de contactos, además de un grupo de seis coordenadas de orientación espacial. Estas propiedades predichas a partir de la data inicial funcionan como pilares para el consiguiente modelado de la estructura terciaria de la proteína. Para lograrlo se utiliza el protocolo de la misma herramienta para poder construir el modelo tridimensional en base a sus coordenadas en el espacio.

3. Verificación del funcionamiento

Los algoritmos identificados deben cumplir con una validación de su funcionamiento. Para llevarlo a cabo se requiere de la implementación de los mismos en un entorno común, esto para reducir las variables que puedan influenciar en los resultados de las ejecuciones respecto a eficiencia y eficacia.

El entorno de pruebas consta de las siguientes características:

- Sistema Operativo : Linux x86_64 (64-bit)
- Kernel : Linux 5.4.0-80-generic
- Distribución : Ubuntu 18.04.5 LTS
- Velocidad de Procesador : 1200 MHz
- Memoria Total : 125 GB
- Swap Total : 2 GB
- Versión de Python : Python 3.9.5

A continuación, se explicará el proceso de verificación del funcionamiento de cada uno de los métodos identificados.

3.1 DMPfold

En este apartado se detallarán los pasos que se llevaron a cabo para poder preparar el ambiente de desarrollo, así como para instalar y evaluar el funcionamiento del algoritmo DMPfold. Para poder realizarlo se ha recabado toda la información disponible a partir de los sitios web de cada una de las herramientas utilizadas por el algoritmo, del artículo publicado de DMPfold y del repositorio oficial de GitHub del proyecto, el cual se podrá

encontrar ingresando al siguiente enlace: https://github.com/psipred/DMPfold/blob/master/README.md. Asimismo, se han recopilado datos adicionales para completar la instalación de forma satisfactoria en base a procedimientos de instalación de otros algoritmos no considerados en este reporte.

DMPfold cuenta con un servidor web desde donde se puede consumir el servicio y generar predicciones; no obstante, el presente documento se enfocará en la validación de su funcionamiento desde un entorno local, para poder evaluar tanto su procesamiento como los requisitos necesarios para poder realizarlo.

Para poder mantener la configuración de nuestro entorno local sin perjudicar cualquier otro tipo de proyecto, es recomendable crear un ambiente conda donde instalaremos las librerías necesarias de forma independiente[38]. Esto se realiza mediante el comando: `conda -y -n <nombreEntorno> python=3.7`. Seguidamente se habilita el entorno mediante `conda activate <nombreEntorno>`. A partir de este punto se puede continuar con la instalación de algunas librerías como Pytorch 0.4 o posterior, Numpy y Scipy. Esto se puede realizar a través del comando `pip3 install <nombreLibreria>` o `conda install pytorch torchvision torchaudio cudatoolkit=10.2 -c pytorch`, en el caso de Pytorch.

Como parte de una primera sección de implementación, se deberán instalar una serie de herramientas de forma obligatoria. Para cada una de estas herramientas se deberán seguir procedimientos independientes, los cuales se explicarán a continuación.

3.1.1 HHblits 3.3.0

HHblits es una herramienta iterativa para la búsqueda de secuencias de proteínas (Steinegger et al., 2019), el cual puede ser obtenido desde el siguiente repositorio `wget https://github.com/soedinglab/hh-suite/releases/download/v3.3.0/hhsuite-3.3.0-SSE2-Linux.tar.gz; tar xvfz hhsuite-3.3.0-SSE2-Linux.tar.gz`. Se necesita que su carpeta bin y scripts sean añadidos a la variable de entorno PATH, ya sea desde los archivos .bashrc.

[38] La librería conda debe estar instalada de forma previa a la implementación de ambos algoritmos.

Requiere de la instalación de diversas bases de datos como Uniclust y Pfam. Se deberá seguir los siguientes comandos para poder descargarlos en segundo plano[39]:

```
wget
https://wwwuser.gwdg.de/~compbiol/uniclust/2020_06/UniRef30_2020_06_hhs
uite.tar.gz -b -o output.txt
wget
https://wwwuser.gwdg.de/~compbiol/data/hhsuite/databases/hhsuite_dbs/pf
amA_31.0.tgz -b -o outputPfam.txt
```

Toda la información necesaria fue obtenida desde el siguiente enlace: https://github.com/soedinglab/hh-suite

3.1.2 FreeContact

Se recomienda instalar la librería a partir del siguiente comando:

```
sudo apt get install freecontact
```

3.1.3 CCMPred

Para poder instalar CCMPred, se debe tener instalada previamente la librería cmake. Para evitar errores se recomienda que la versión sea superior a 3.10. Por ejemplo, la versión 3.16 se puede descargar desde el repositorio de cmake. Posteriormente, se deberán ejecutar los siguientes comandos de forma consecutiva:

```
sudo apt install cmake
git clone --recursive https://github.com/soedinglab/CCMpred.git
cd CCMpred
cmake -DWITH_CUDA=off -DWITH_OMP=off .
make
sudo cp ./bin/ccmpred /bin/ccmpred
```

3.1.4 CNSsolve

El sistema de cristalografía y NMR (CNS, por sus siglas en inglés) es una herramienta para la búsqueda y determinación a detalle de estructuras macromoleculares. Para poder descargarlo se deberá contar con credenciales que se obtendrán en la página web oficial luego de registrar un formulario y

[39] Es recomendable realizar la descarga de las bases de datos con el comando “-b” (segundo plano) ya que los ficheros de gran tamaño podrían tomar un tiempo considerable en descargarse.

validar que el dominio pertenece a una organización sin fines de lucro. La dirección web corresponde a http://cns-online.org/v1.3/.

Una vez que se hayan descargado los ficheros se deberán descomprimir, para luego modificar el archivo cns_solve_env definiendo la ruta del directorio de CNSsolve (directorio actual) dentro de la variable CNS_SOLVE.
En la misma carpeta se encontrará el archivo .cns_solve_env_sh, el cual deberá renombrarse a cns_solve_env.sh y realizar la misma modificación descrita en el párrafo anterior.
Como un paso adicional, se deberá instalar la librería flex siguiendo el siguiente comando:

```
sudo apt-get update
sudo apt-get install flex
```

Seguidamente se debe ingresar al entorno Csh con `csh` en el terminal y ejecutar el comando: `source cns_solve_env && make install.` En caso no contar con un entorno con Csh o Tcsh Unix Shell, se deberá instalar con el comando `sudo apt-get install csh`.

Posterior a eso, se deberán modificar algunos de los archivos creados a partir del comando anterior:

- En el archivo cns_solve_1.3/source/rtf.inc, se deberá modificar el valor de la variable MXRTP a 4000.
- En el archivo cns_solve_1.3/source/machvar.f, se deberá añadir `WRITE (6,'(I6,E10.3,E10.3)') I, ONEP, ONEM` debajo de la línea 67 que dice `IF (ONE .EQ. ONEP .OR. ONE .EQ. ONEM) THEN`.
- En el archivo cns_solve_1.3/modules/nmr/readdata, se deberá modificar el valor de la variable nrestraints a 50000 y la variable nassign a 3000. Esto en caso se requiera ajustarse a arquitecturas más grandes.
- En el archivo cns_solve_1.3/intel-x86_64bit-linux/source/machvar.inc, se deberá modificar el valor de la variable MXFPEPS2 a 8192.
- En el archivo cns_solve_1.3/Intel-x86_64bit-linux/source/Makefile, se deberá retirar el flag –ffast-math

Una vez realizado lo anterior, se deberá volver a ingresar al entorno csh en la carpeta source y ejecutar: `csh` y luego `make cns_solve`

Luego, continuando en el entorno csh, se deberá redirigir a la carpeta principal cns_solve_1.3 y ejecutar los siguientes comandos:

```
source cns_solve_env
make clean
make no-fastm
exit
```

Asimismo, se deberá dar los permisos correctos y ejecutar el archivo .sh con el comando:

```
chmod +x cns_solve_env.sh
./cns_solve_env.sh
```

Llegado a este punto aún no se podrá comprobar que el recurso se ha instalado correctamente. Sin embargo, se realizará en los pasos posteriores.

3.1.5 Modeller

La librería Modeller se instalará ejecutando el comando descrito a continuación:

```
conda install modeller -c salilab
```

Luego de ejecutar el comando se descargarán algunas dependencias y se obtendrá un mensaje que alude al ingreso de una licencia. Sin embargo, no es necesario configurarlo ya que solo se usará el script de Python que se descarga con el comando anterior.

3.1.6 CD-hit

La librería CD-hit se utilizará para poder predecir el TM-score de los modelos generados. Es opcional.

```
git clone https://github.com/weizhongli/cdhit.git && make
```

3.1.7 Legacy BLAST

El software legacy Blast deberá ser instalado como una librería, para ello, se deberá tener instalada miniconda. Posteriormente se podrá actualizar a Blast+.

```
conda install -c bioconda blast-legacy
```

Llegado a este punto, todas las herramientas anteriores ya se encuentran instaladas dentro del ambiente local de desarrollo. Ahora bien, es necesario clonar el repositorio del algoritmo y configurar algunos archivos con el objetivo de establecer los directorios donde están almacenados los ficheros instalados previamente.

Los archivos que deberán ser modificados son los siguientes: seq2maps.csh, aln2maps.csh, bin/runpsipredandsolvwithdb, run_dmpfold.sh y predict_tmscore.sh. En cada uno de ellos, en caso se encuentren, se deberán modificar los valores de las siguientes variables:

- DMPFOLDDIR: Definir la ruta del directorio de DMPfold

```
dmpfolddir=~/R1.2_verificacion_func/dmpfold/DMPfold
```

- CNS_SOLVE: Definir la ruta del directorio de CNSsolve

```
setenv CNS_SOLVE '~R1.2_verificacion_func/dmpfold/CNS/cns_solve_1.3'
```

- CDHITCMD: Definir el comando cd-hit incluyendo el directorio

```
cdhitcmd=~/R1.2_verificacion_func/dmpfold/cdhit/cd-hit
```

- HHLib: Definir la ruta de la librería

```
setenv HHLIB ~/miniconda3/pkgs/hhsuite-3.3.0-py37pl5262h21043fe_2
```

- HHBin: Definir la ruta del directorio bin de la librería

```
setenv HHBIN ~/miniconda3/pkgs/hhsuite-3.3.0-py37pl5262h21043fe_2/bin
```

- HHDB: Definir la ruta del directorio donde se encuentra descargado la base de datos Uniclust30

```
setenv HHDB /data/spalomino/databases
```

- CCMPred: Definir la ruta del directorio bin de la librería

```
set ccmpreddir = ~/R1.2_verificacion_func/dmpfold/CCMpred/bin
```

- FreeContactCMD: Definir el comando de ejecución de la librería incluyendo el directorio

```
set freecontactcmd = /usr/bin/freecontact
```

- NCBIDIR: Definir la ruta del directorio bin de la librería Blast Legacy

```
set ncbidir = ~/miniconda3/pkgs/blast-legacy-2.2.26-h9ee0642_3/bin
```

En el caso del archivo run_dmpfold.sh, se deberá agregar el comando `RANDOM=$$` justo antes del comando `seed=$RANDOM`; además, se deberá modificar el comando `((counter++))` que se encuentra aproximadamente en la línea 191 del script, cambiándolo por `counter=$((counter+1))`.

Seguidamente, se deberá dirigir al directorio cnsfiles y ejecutar el script que contiene. El comando para realizarlo es el siguiente: `cd /DMPfold/cnsfiles && ./installscripts.sh`. Este script agregará algunos ajustes a la configuración previa del recurso CNS, por lo cual, llegado a este punto, CNS ya se encontrará disponible. Para eso se deberá comprobar su correcta instalación ejecutando `cns` en el terminal.

En el momento en el que las herramientas requeridas, así como los archivos de configuración que las invocan, ya están configuradas, se deberán ejecutar para generar dos archivos (.21c y .map), los cuales servirán como input para el algoritmo principal. En ese sentido, se requerirá ejecutar uno de los dos comandos siguientes, dependiendo del dato de entrada con el que se cuenta:

`csh seq2maps.csh example/PF10963.fasta` (En caso se cuente con la secuencia)

`csh aln2maps.csh example/PF10963.aln` (En caso se cuente con un alineamiento)

Con este último paso, el algoritmo es considerado como instalado.

Para poder verificar su funcionamiento exitoso se deberá ejecutar el siguiente comando: `sh run_dmpfold.sh example/PF10963.fasta PF10963.21c PF10963.map ./PF10963`, que requiere de cuatro parámetros. El primero representa al archivo en formato FASTA que contiene la secuencia de la proteína a predecir, el segundo y el tercero son los archivos generados en el paso anteriormente explicado, y el último es el directorio donde se almacenarán los archivos que fueron resultado del procesamiento de la predicción de estructura terciaria de la proteína ingresada.

El resultado de la ejecución del algoritmo DMPfold puede ser observado en la Figura 16. Se obtiene un mensaje de error respecto a la librería Modeller. No obstante, como se mencionó en las instrucciones de instalación era lo esperado dado que no se cuenta con una licencia. Aun así el programa puede ejecutarse correctamente debido a que solo se hace uso de uno de sus scripts.

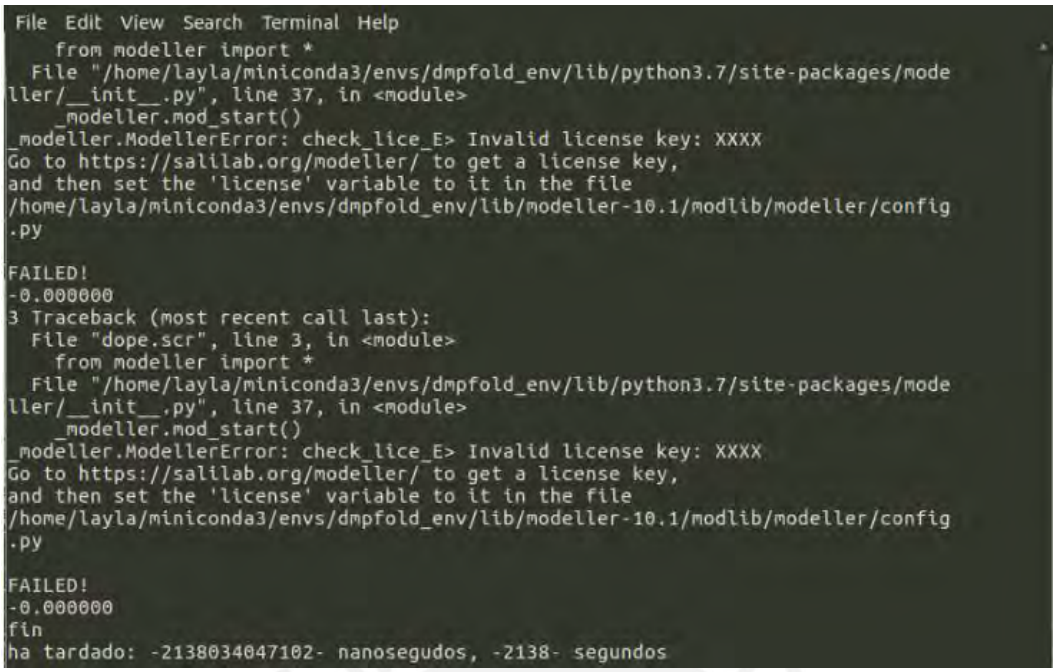

Figura 16. *Verificación del funcionamiento del algoritmo DMPfold para la predicción de estructuras terciarias de proteínas. (Elaboración propia).*

De acuerdo con lo experimentado, se verifica el correcto funcionamiento dado que la ejecución del algoritmo dio como resultado una serie de archivos en formato PDB, tal como se observa en la Figura 17. Así, el algoritmo predice estructuras terciarias de forma exitosa a partir de la secuencia de proteínas ingresada en formato FASTA.

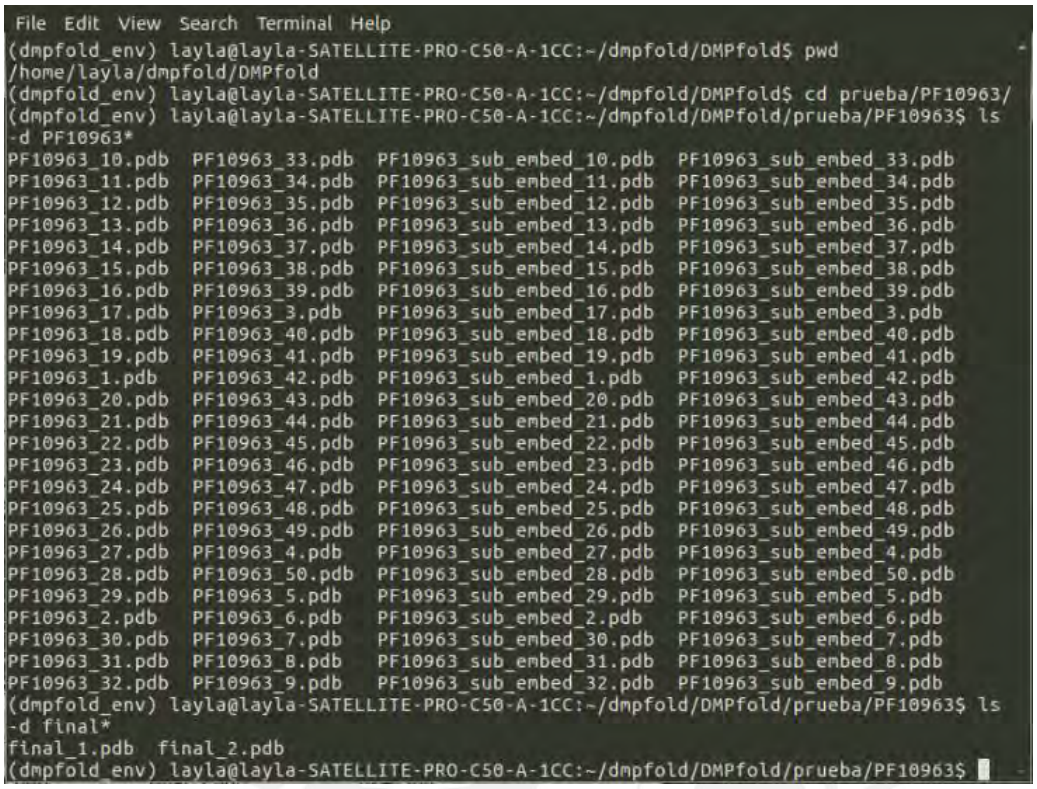

Figura 17. *Resultados de la ejecución de algoritmo DMPfold. Se generan archivos en formato PDB. Los archivos con prefijo FINAL obtuvieron una mejor predicción. (Elaboración propia).*

3.2 trRosetta

En esta sección se describe la secuencia de pasos a seguir para poder instalar y utilizar la herramienta trRosetta[40]. Se ha obtenido toda la información de distintas fuentes de información como los artículos publicados por los investigadores de la herramienta, los repositorios GitHub de los códigos fuente y las páginas web oficiales de cada una de las librerías requeridas para completar la instalación.

Cabe mencionar que el repositorio del código fuente de la primera versión solo aloja el código fuente de la predicción de las distancias y las coordenadas de orientación espacial. No obstante, adicionalmente la segunda versión incluye la obtención del alineamiento de secuencias múltiples, el modelado y el refinamiento de la estructura terciaria, obteniendo, así, un modelo predicho en formato PDB.

[40] La información recopilada sobre trRosetta abarca tanto la versión 1 como la versión 2 del método. Esta puede ser obtenida a través de los siguientes enlaces: https://github.com/gjoni/trRosetta y https://github.com/RosettaCommons/trRosetta2

Algunos requerimientos previos que se deben tener en cuenta con relación a las librerías necesarias son: Python 3.6, Tensorflow 1.14 o Tensorflow-gpu 1.14, Numpy 1.19, PyTorch 1.4, BioPython 1.78, Scipy 1.5, Psipred 4.01, hhsuite, entre otros.

En primer lugar, se deberá realizar la instalación de PyRosetta, una herramienta para el modelado molecular de las proteínas, que funciona como soporte al algoritmo en general. La instalación se realizará desde la consola de comandos bash siguiendo la documentación del siguiente enlace: https://www.pyrosetta.org/downloads/windows-10. Sin embargo, se deberán solicitar las credenciales pertinentes (usuario y contraseña) a través del canal de obtención de licencias de la plataforma CoMotion[41]. Es importante recordar que trRosetta se encuentra disponible con la versión 3.6 de Python, por tanto, la versión de pyRosetta deberá coincidir.

En segundo lugar, haciendo uso del repositorio GitHub a disposición, se deberá realizar la clonación de las fuentes. Luego, se procederá a instalar las librerías requeridas. Para ello, se recomienda crear un entorno de conda utilizando uno de los archivos .yml del repositorio. Este archivo contiene la lista de todas las librerías a instalar y su versión correspondiente.

En tercer lugar, se deberán descargar archivos de tamaño considerable: los pesos de la red neuronal, la base de datos UniClust que contiene secuencias y alineamientos de proteínas y una base de datos de plantillas de estructuras. Cada uno de los archivos pesa alrededor de 1G, 46G y 8G, respectivamente.

Adicionalmente, tras un proceso de solución de errores, se llegó a necesitar la modificación del archivo /trRosetta2/tape/get_embeddings.py. Se agregó la siguiente importación, además de realizar el cambio descrito a continuación:

```
from tape import ProteinBertModel, TAPETokenizer, ProteinBertConfig
```

Antes:

```
### Create BERT model
model = ProteinBertModel.from_pretrained('bert-base')
  tokenizer = TAPETokenizer(vocab='iupac')
  model.eval()
```

Después:

```
### Create BERT model
config = ProteinBertConfig()
model = ProteinBertModel(config)
  tokenizer = TAPETokenizer(vocab='iupac')
  model.eval()
```

[41] La plataforma solo generará credenciales para fines académicos, a través del siguiente enlace: https://els2.comotion.uw.edu/product/pyrosetta

De la misma manera, en el archivo /trTosetta2/run_pipeline.sh se corrigió la línea de invocación al servicio HHSearch en la línea 48 de la siguiente manera:

```
$HH -i $WDIR/t000_.msa0.ss2.a3m -o $WDIR/t000_.hhr -v 0 -
premerge 0 > $WDIR/log/hhsearch.stdout 2> $WDIR/log/hhsearch.stderr
```

Se agregó el flag 'premerge 0' para poder ignorar los archivos _a3m faltantes y esto porque la base de datos PDB100 no contiene una base de datos _a3m[42].
Por último, se tendrá que ejecutar el script `./install_dependencies.sh` para poder instalar otras dependencias como csblast.

Una vez instalado lo descrito en los pasos anteriores, el entorno se encuentra listo para probar el algoritmo. Se deberá dirigir a la carpeta principal de trRosetta2 para poder ejecutar el siguiente comando: `./run_pipeline.sh example/T1078.fa example/T1078`. El primer parámetro corresponde a la secuencia de la proteína en formato FASTA y el segundo, al directorio donde se almacenará a la estructura terciaria predicha en formato PDB, así como los distintos archivos intermedios generados en el procesamiento.

En la Figura 18, se puede observar la ejecución del algoritmo trRosetta luego de completar la instalación. Se ejecutó el comando descrito en el párrafo anterior, por lo cual se esperaba la predicción de la estructura terciaria de la proteína T1078 a partir de los datos ingresados. La imagen muestra una ejecución exitosa terminando en un mensaje de "Final models saved in: example/T1078/model". No obstante, al revisar la carpeta de salida /T1078/model, no se encontró el modelo generado con el mejor score, aunque se encontraron otros intermedios en otros directorios. Se supone que el algoritmo elige el mejor de todos los generados en el proceso.

[42] Previo a la modificación se obtenía un fallo de segmentación. La solución se encontró en el siguiente *issue* del repositorio Github: https://github.com/RosettaCommons/trRosetta2/issues/7

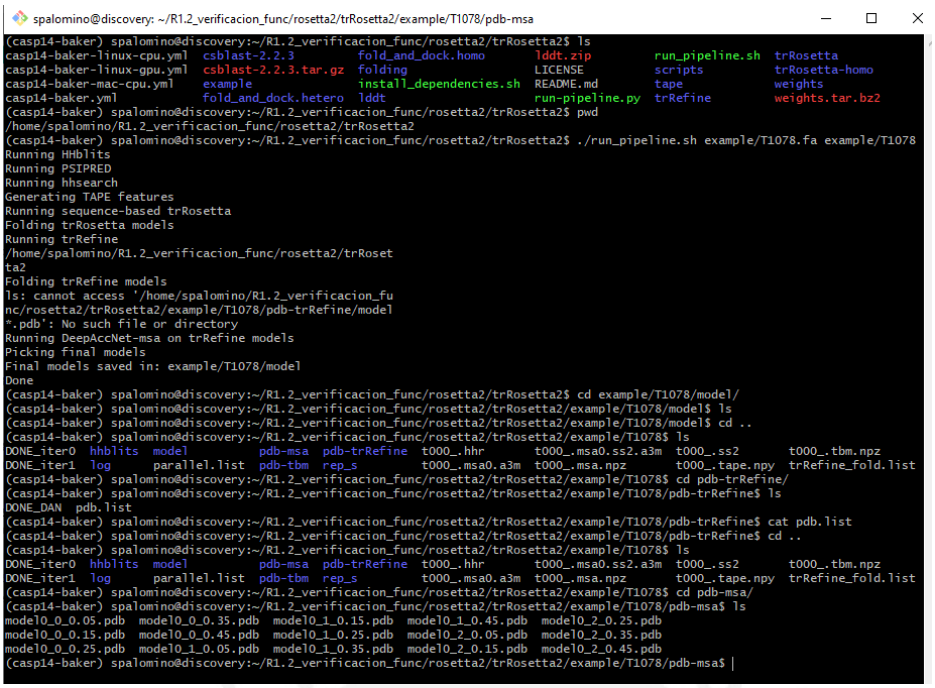

Figura 18. *Verificación del funcionamiento del algoritmo trRosetta para la predicción de estructuras terciarias de proteínas. (Elaboración propia).*

4. Comparación de características de los algoritmos identificados

En esta sección del reporte se presenta un cuadro comparativo con las características más relevantes de los dos algoritmos identificados. Se busca que mediante una visualización resumida y concreta se puedan definir criterios que permitan la toma de decisión respecto a la selección de uno de ellos. El algoritmo elegido será el que continue en el presente proyecto de fin de carrera y se propondrán una serie de modificaciones para poder adaptarlo a las particularidades de las proteínas repetidas.

El cuadro comparativo de los algoritmos en base a distintos criterios de análisis se observa en la Tabla 28.

Característica	DMPfold	trRosetta
Técnica de inteligencia artificial	Redes neuronales profundas convolucionales	Redes neuronales profundas residuales
Factor diferenciador	Predice los enlaces de hidrógeno de la cadena principal, los ángulos de torsión y los límites de distancias interatómicas. Utiliza mapas de contacto y datos de covarianza como	Predice las coordenadas de orientación espacial, los mapas de distancias y los mapas de contacto

	datos de entrada.	
Técnica de modelado	CNS	PyRosetta
Datos de entrada	Secuencia de aminoácidos	Secuencia de aminoácidos
Datos de salida	Coordenadas 3D de estructura (archivo PDB)	Coordenadas 3D de estructura (archivo PDB)
Lenguaje de programación	C y Python	Python y Bash
Versión de Python	3.7	3.6
Librería de Aprendizaje de Máquina	PyTorch 0.4 o posterior (Evaluado con PyTorch 1.9)	Tensorflow 1.14 PyTorch 1.4
Complejidad de instalación	Alta	Alta
Recurso computacional requerido	Requiere de la capacidad de una computadora estándar de un núcleo y de memoria suficiente para la descarga de algunas bases de datos	Requiere de servidores con GPU y de memoria suficiente para la descarga de distintas bases de datos
Recursos de libre acceso	El código fuente, la documentación completa y el modelo entrenado de la red neuronal. Los recursos adicionales que se utilizan también son de libre acceso	Los alineamientos múltiples, los archivos fuente para la predicción de geometrías interresiduales y el protocolo para el modelado de estructuras
Resultado de verificación de predicción	Éxito (Genera 2 estructuras en formato PDB)	Sin éxito (No genera estructuras)
Errores obtenidos	Error no significativo por carencia de la licencia de Modeller	Error en el refinamiento de estructuras. Arrastre de error para la selección de uno final
Tiempo promedio de ejecución	~36 minutos	~35 minutos
Secuencia de proteína de prueba[43]	PF10963 Escherichia coli UPI0006A5DD14	T1078 Trichoderma virens

[43] La secuencia de prueba usada para verificar el funcionamiento de cada algoritmo se obtiene de su propio repositorio.

Cantidad de residuos	82 residuos	138 residuos

Tabla 28. Cuadro comparativo entre algoritmos seleccionados: DMPfold y trRosetta.

5. Discusión

Como parte del análisis de los algoritmos seleccionados, se vio la necesidad de corroborar el correcto funcionamiento de cada uno de ellos. Ese sería el punto de partida del proceso comparativo de los algoritmos dado que se requería reducir la posibilidad de fallo en caso existan incompatibilidades respecto a las versiones de las librerías o los recursos utilizados. En ese sentido, es muy probable que luego de aplicar de forma exitosa algún recurso, se hayan reportado incidencias que podrían haber impulsado el lanzamiento de una nueva versión que las corrija, y que esa versión no se adapte de forma adecuada al procedimiento del cual era parte en los algoritmos.

Se destaca mucho este aspecto dado que el proceso de instalación de ambos algoritmos se tornó complicado debido a la variedad de librerías requeridas y la diversificación de las listas de instrucciones a seguir para instalarlas de forma independiente, las cuales, además, muchas veces fueron tediosas de comprender.

Respecto a la ejecución de los algoritmos cabe mencionar que se dio prioridad al uso de los datos de prueba que ofrecen los propios desarrolladores y que se pueden encontrar en sus repositorios. Ambas secuencias fueron de distintas proteínas y de distintos tamaños: 82 y 138 residuos para DMPfold y trRosetta, respectivamente. Sin embargo, en el cuadro comparativo se observa que el tiempo de ejecución es aproximadamente igual: 36 y 35 minutos respectivamente. Esta situación no necesariamente representa eficiencia, sino que corresponde a un escenario de ineficacia dado que el segundo algoritmo no finalizó su ejecución de forma exitosa. Tal como se explicó en los apartados anteriores, los directorios generados no contenían las estructuras tridimensionales finales que se esperaban, con lo cual se explicaría la reducción de tiempo de ejecución aún cuando la cantidad de aminoácidos en la secuencia es cerca al doble.

Debido a esa situación no se pudo continuar con una segunda etapa de comparación, es decir, a la predicción de una misma estructura de proteína usando la misma secuencia en ambos algoritmos. Sin duda, este aspecto influyó de forma significativa en la selección de uno de los dos algoritmos presentados.

Anexo F: Documento descriptivo de la justificación de la elección del algoritmo de predicción de estructuras terciarias de proteínas en general y sus modificaciones propuestas para adaptarse a las proteínas repetidas

Este anexo contiene el documento que justifica la elección de uno de los dos algoritmos identificados capaces de realizar predicciones de estructuras terciarias de proteínas en general. Asimismo, presenta una serie de modificaciones propuestas a incorporarse en el algoritmo para poder adaptarlo a las proteínas repetidas. Su contenido abarca la descripción de los criterios de selección, la elección del algoritmo y la propuesta de algunas modificaciones que aprovechen las particularidades de las proteínas repetidas.

1. Introducción

Los algoritmos identificados en las primeras instancias del presente proyecto de fin de carrera fueron dos: DMPfold (Greener et al., 2019) y trRosetta (Yang et al., 2020). Estos fueron seleccionados por su capacidad de explotación de datos de estructuras primarias de proteínas en general para predecir sus estructuras terciarias.

El objetivo es adaptar un algoritmo para poder obtener mejores predicciones de estructuras tridimensionales de proteínas repetidas. Para poder lograrlo, como segunda instancia, es necesario centrar el análisis en solo uno de ellos. Por consiguiente, en este documento se presentarán los criterios que se tuvieron en cuenta para elegir el algoritmo, así como el resultado de la selección. Además, se propondrán algunas modificaciones al algoritmo seleccionado en base a las particularidades de las proteínas repetidas.

2. Criterios de selección

Para poder definir los criterios de selección del algoritmo que se utilizará en la herramienta de predicción de estructuras terciarias de proteínas repetidas, se tomará en cuenta, y en gran medida, los dos anexos previos relacionados al objetivo específico al que le corresponde la adaptación del algoritmo. El Anexo D contiene la presentación y la descripción teórica de cada uno de los algoritmos, mientras que el Anexo E contiene la comparación de sus técnicas y funcionamiento basado en sus procesos de instalación y sus ejecuciones en sí.

Si nos enfocamos en la técnica de inteligencia artificial aplicada a cada algoritmo, según los artículos publicados por cada equipo autor, tanto DMPfold como trRosetta utilizan Redes Neuronales Profundas. Sin embargo, el primero aplica las redes neuronales de

tipo convolucionales, las cuales están diseñadas para el procesamiento de datos que vienen en forma de arreglos multidimensionales y su arquitectura usualmente está conformada por capas tras capas de abstracción (Lecun et al., 2015). En cambio, trRosetta utiliza redes neuronales residuales, una técnica reconocida por las representaciones extremadamente profundas y su gran extensión respecto a los conjuntos de datos reales (Sander et al., 2021).

Ambos tipos fueron inicialmente estudiados y aplicados al procesamiento de imágenes, sin embargo, cada vez van tomando más relevancia en otros dominios (Lecun et al., 2015). En ese sentido, su aplicación no se centra solo en una problemática, por lo cual su rendimiento dependerá de sus niveles de representación, el tiempo de entrenamiento, su tasa de aprendizaje, entre otros (Lecun et al., 2015). Así, la eficiencia de esas técnicas deberá ser analizada tras la ejecución de los algoritmos que los implementan.

Respecto al factor diferenciador, cabe mencionar que ambos presentan alternativas novedosas y prometedoras respecto al aprovechamiento de la información de la secuencia de la proteína en cuestión. Por su lado, trRosetta utiliza pyRosetta para modelar el plegamiento de la proteína en base a la predicción de diversas coordenadas de orientación espacial y distancias (Yang et al., 2020). Por otro lado, DMPfold realizará el modelado de la estructura terciaria predicha utilizando la librería CNS, además de utilizar datos de entrada intermedios como los mapas de contacto y los datos de covarianza, generados por librerías, o como los ángulos de torsión, las distancias y los enlaces de hidrógeno, predichos por scripts internos (Greener et al., 2019).

En ambos casos se rescata el hecho de que el único dato que inicia la predicción y es imprescindible para realizarlo (a pesar de requerir scripts intermedios) sea la secuencia de aminoácidos en formato FASTA. Así como que el dato de salida siempre será un archivo en formato PDB que contendrá la estructura terciaria predicha.

No obstante, lo que verdaderamente discrimina a un algoritmo del otro son tres factores importantes. En primer lugar, la eficacia del algoritmo luego de seguir las instrucciones para su instalación y uso. En segundo lugar, el empleo de librerías actualizadas en su implementación. En tercer lugar, el recurso computacional requerido.

2.1 Eficacia del algoritmo

Se considera a este primer criterio como el más influyente en la toma de decisión dado que es crucial contar con el correcto funcionamiento de ambos algoritmos para

dar paso a una posible elección. Ya sea para realizar una comparación de rendimiento o calificación funcional, donde se evaluarían los resultados obtenidos, es fundamental que los algoritmos entreguen los datos de salida de forma exitosa.

Según los artículos publicados por cada equipo desarrollador de los algoritmos, los datos de salida de ambos serán uno o más archivos en formato PDB que contienen las coordenadas de los átomos de las estructuras terciarias de las proteínas. No obstante, tal como se detalla en el Anexo F, el algoritmo trRosetta no generó ningún archivo luego de su ejecución. Las evidencias de la ejecución del algoritmo como del directorio vacío luego de haber terminado se encuentran en el mismo anexo.

Por otro lado, el algoritmo DMPfold generó dos archivos PDB, los cuales se ingresaron a PyMol, una herramienta de visualización molecular. Las estructuras tridimensionales predichas se generaron con los nombres 'final_1' y 'final_2', representados en PyMol con el color verde y turquesa, respectivamente, tal como se muestra en la Figura 19.

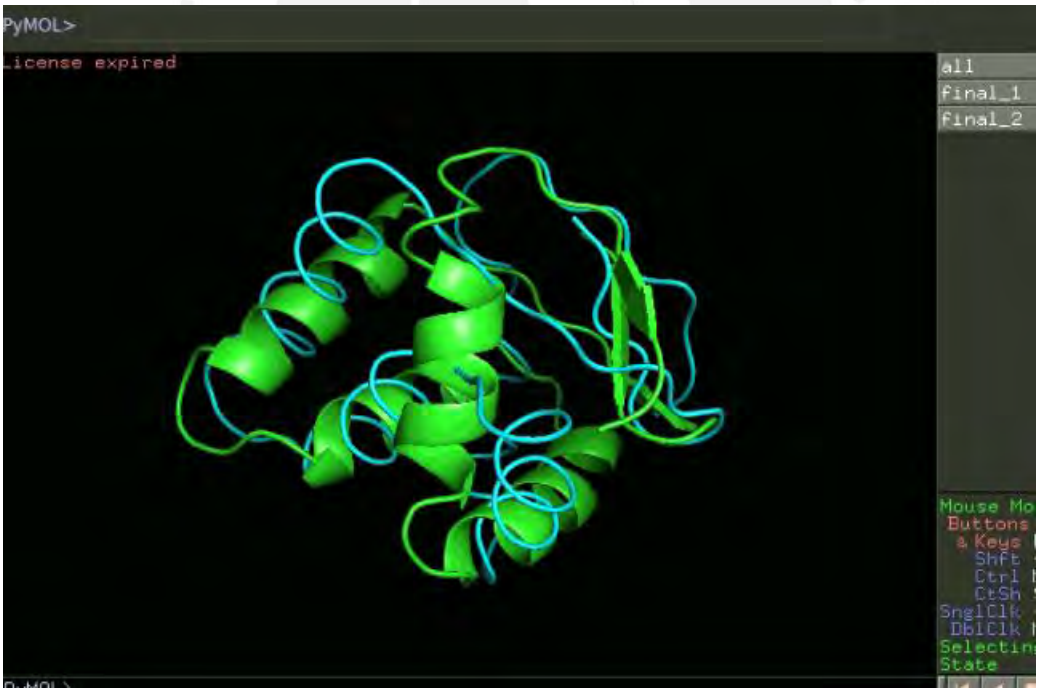

Figura 19. *Visualización de predicción de estructura terciaria realizado por el algoritmo DMPfold. La estructura de color verde corresponde a la predicción final_1 (mayor puntaje TM-score) y la estructura de color turquesa corresponde a la predicción final_2. Secuencia: PF10963. (Elaboración propia en PyMol).*

Dado que las verificaciones de la ejecución de los algoritmos se realizaron con los datos de prueba obtenidos del mismo repositorio, se procedió a ejecutarlos con los datos de entrada inversos. El archivo de entrada de prueba de trRosetta era el T1078.fa, mientras que el de DMPfold era el PF10963.fasta, es por ello que, en una segunda ejecución, se evaluó al algoritmo trRosetta con el archivo PF10963.fasta y a DMPfold con el archivo T1078.fa.

Los resultados fueron iguales, el algoritmo trRosetta no generó ningún archivo de salida, el cual según las instrucciones deberá almacenarse en la carpeta '/model', tal como se observa en la Figura 20.

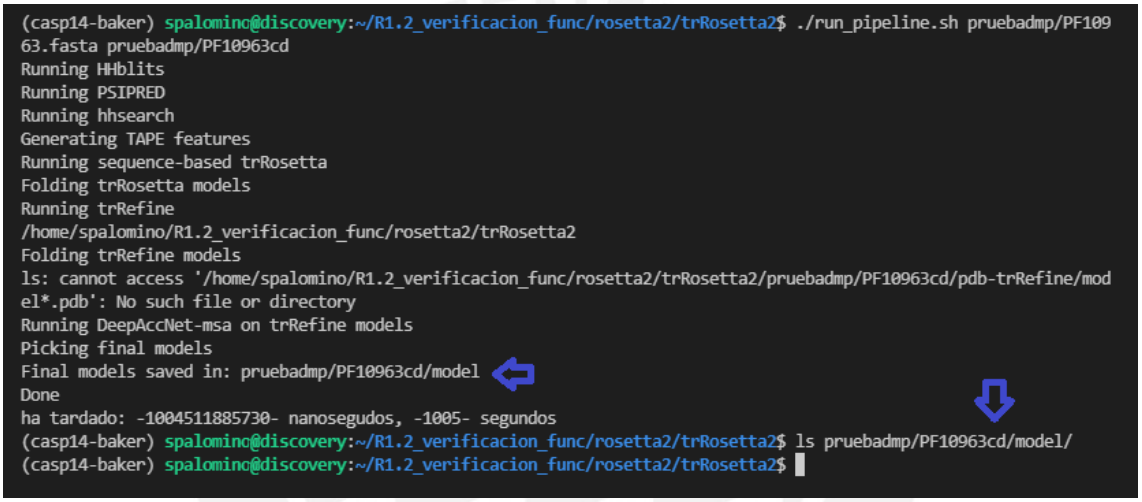

Figura 20. *Resultado de ejecución del algoritmo trRosetta con dato de entrada PF1063 (repositorio de DMPfold). (Elaboración propia).*

Sin embargo, el algoritmo DMPfold volvió a generar dos archivos PDB con los nombres 'final_1' y 'final_2', los cuales se pueden visualizar en la Figura 21 de color verde y turquesa, respectivamente. La herramienta utilizada para elaborar la visualización es PyMol.

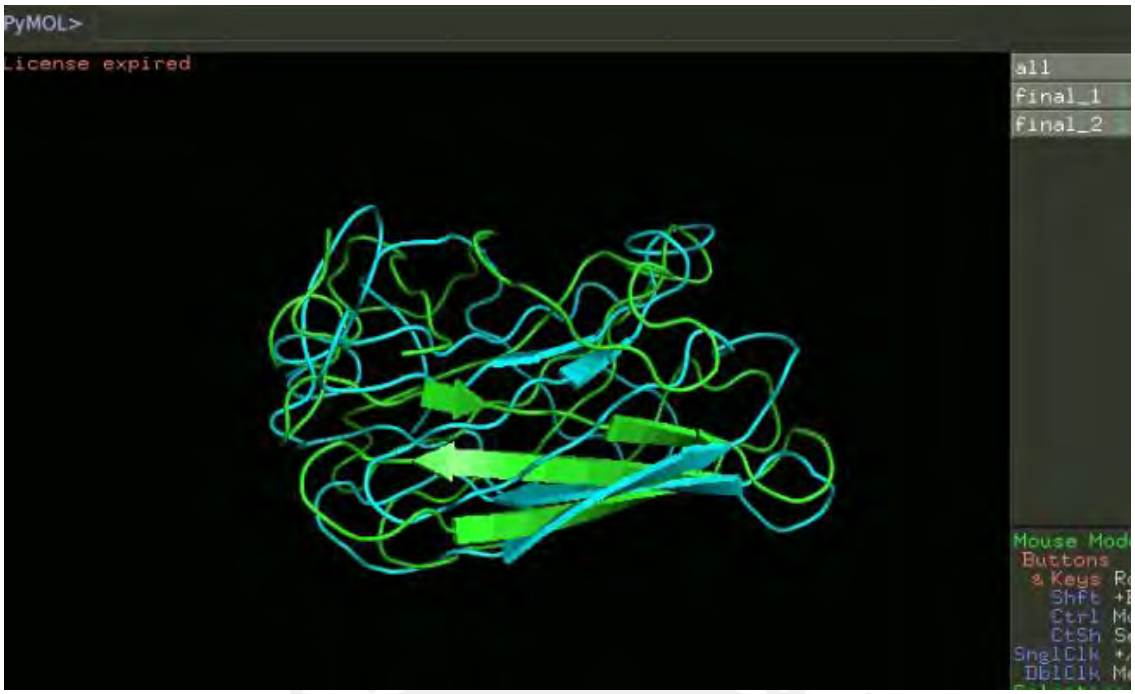

Figura 21. *Visualización de predicción de estructura terciaria realizado por el algoritmo DMPfold. La estructura de color verde corresponde a la predicción final_1 (mayor puntaje TM-score) y la estructura de color turquesa corresponde a la predicción final_2. Secuencia: T1078. (Elaboración propia en PyMol).*

2.2 Librerías en la implementación

Respecto a las librerías utilizadas en la implementación de los algoritmos se destaca la facilidad de reconocimiento y de instalación de los que fueron utilizados en trRosetta, dado que el propio repositorio cuenta con tres archivos en formato .yml. Estos archivos contienen la información de las librerías requeridas y para instalarlos solo se necesita de invocar el archivo en la creación del entorno conda. Solo se debe ejecutar un archivo pero la elección depende de la arquitectura a utilizar, es decir, si el sistema operativo es Mac o Linux, y, en caso de este último, si se cuenta con GPU o no.

En el caso del algoritmo DMPfold, no se cuenta con estos archivos que facilitan la instalación de librerías. Sin embargo, no se considera necesario, ya que las librerías que se requerían de instalación desde el entorno de desarrollo eran pocas. Sin embargo, este algoritmo requería de la instalación previa de otros recursos, los cuales fueron descritos detalladamente en las instrucciones del propio repositorio.

En ambos casos, se considera que el proceso la instalación tuvo una complejidad alta. En ese sentido, cabe mencionar que, si bien trRosetta brindó pequeñas facilidades de instalación, las librerías que utiliza son de versiones inferiores a las actuales. Por el

contrario, el algoritmo DMPfold menciona que se ha verificado su funcionamiento en versiones posteriores al recomendado. Esto fue validado también dado que la ejecución de este algoritmo se realizó con la versión de PyTorch 1.9, cuando la versión recomendada es PyTorch 0.4.

2.3 Recurso computacional requerido

El algoritmo DMPfold es apto para ejecutarse en ambientes con capacidades promedio tales como una computadora estándar que cuente con un núcleo de procesamiento y 8GB de memoria (Greener et al., 2019). Asimismo, se menciona que el uso de GPU's no es de carácter obligatorio ni restrictivo al usar PyTorch. La velocidad de ejecución no aumentará dado que la carga de la red neuronal no es una actividad que consuma demasiado recursos de procesamiento.

En el caso de trRosetta, el algoritmo requerirá de servidores con GPU para poder realizar la predicción con un mejor rendimiento (Yang et al., 2020).

Por último, para ambos algoritmos se requiere contar con espacio de memoria suficiente para alojar las diversas bases de datos que utilizan. Por ejemplo, la base de datos UniClust30 que ambos procesos de instalación recomiendan descargar pesará aproximadamente 46G en un archivo comprimido y alrededor de 150G una vez se haya descomprimido.

3. Algoritmo seleccionado para la adaptación

Acorde a los criterios descritos en el apartado anterior, el algoritmo que será seleccionado para proseguir con los objetivos del presente proyecto de fin de carrera será DMPfold (Greener et al., 2019).

4. Modificaciones propuestas para adaptar el algoritmo seleccionado hacia las proteínas repetidas

Recapitulando lo descrito en el acápite del Marco Conceptual de este proyecto de tesis, las proteínas repetidas son grupos de familias de proteínas que cuentan con ciertas particularidades. El propósito de esta sección de la investigación es aprovechar esas características para poder adaptar el algoritmo seleccionado, de tal manera que se obtenga la mayor ventaja del proceso de predicción de estructuras terciarias a partir de

las secuencias de aminoácidos. Esto en base a las necesidades del contexto de las familias de proteínas repetidas en particular.

En general, todas las modificaciones formuladas están enfocadas en torno al aprovechamiento de la particularidad más reconocible de las proteínas repetidas: los patrones de repetición dentro de una misma familia (Kajava, 2012). Es por ello que el hecho de que, de forma original, el algoritmo DMPfold permita trabajar con secuencias de aminoácidos obtenidas a partir de la base de datos PFAM, reconocida por poseer colecciones de familias de proteínas (Mistry et al., 2021), es una ventaja para las proteínas en las que se enfoca este proyecto.

El algoritmo inicial requiere de tres archivos como dato de entrada: el archivo de la secuencia en formato fasta obtenido de PFAM, un archivo de mapa de contactos y un archivo que contiene datos de covarianza (Greener et al., 2019). Estos dos últimos se obtienen tras la ejecución de un script que utiliza como input al archivo de PFAM.

El archivo PFAM se obtiene a través de un servicio de la propia base de datos al cual puede accederse mediante el siguiente link: https://pfam.xfam.org/family/alignment/download/format?acc={{pfamCode}}&alnType=full&format=fasta&order=t&case=l&gaps=default&download=1. Se deberá modificar la variable *{{pfamCode}}* con el código de la familia requerida.

Este archivo contiene fragmentos de proteínas también denominados dominios. Esos fragmentos pertenecen a diversas proteínas y se consolidan como parte de una familia ya que comparten ciertas similitudes.

Tal como se observa en la Figura 22, una secuencia (o estructura primaria) de una proteína puede estar conformada por uno o más dominios. Teniendo en cuenta que cada dominio puede pertenecer a la misma familia o a una diferente. La proteína W2T741_NECAM es uno de los cuatro mil sesenta y tres ejemplos de secuencias que cumplen con esa arquitectura.

Figura 22. *Arquitectura de dominio constituida por las familias ArgoN, ArgoL1, PAZ, ArgoL2, ArgoMid y Piwi. W2T741_NECAM es una proteína que comparte ese dominio. Obtenido de (Banco de Datos de Proteínas en Europa, 2021).*

Al tratarse de proteínas repetidas esos fragmentos corresponden a las unidades de repetición de una familia particular de proteínas. Cada una de esas unidades de repetición pueden, o no, tener una estructura tridimensional previamente relacionada. Esas estructuras terciarias estarán alojadas en el Banco de Datos de Proteínas (PDB, por sus siglas en inglés).

En ese sentido, se propone dividir a las secuencias del archivo en dos: el primer grupo de fragmentos que estén relacionados a una o más estructuras terciarias en PDB, mientras que el segundo, alojará a las que no tengan ninguna estructura tridimensional ligada. Esta distinción respecto a las estructuras terciarias se realizará a través del uso de otros servicios, los cuales se encuentran disponibles gracias a PDB y a Swiss-Prot.

A partir de esta primera división, se descartarán los fragmentos pertenecientes al primer grupo, ya que no forma parte del objetivo del proyecto, y se seguirán dos caminos con los del segundo grupo. Estos dos caminos corresponden a la primera y segunda modificación propuestas para adaptar el algoritmo a las proteínas repetidas.

El primero implica el uso de cada uno de los fragmentos de forma independiente. Estos se ingresarán directamente al proceso de generación de archivos intermedios .21c y .map mencionados al inicio de este acápite.

La segunda modificación corresponde al uso de los límites de posición de los fragmentos de las proteínas en la secuencia completa. Se debe tener en cuenta que cada uno de los dominios o fragmentos contenidos en el archivo PFAM cuenta con su identificador Uniprot y su posición inicial y final dentro de la misma.

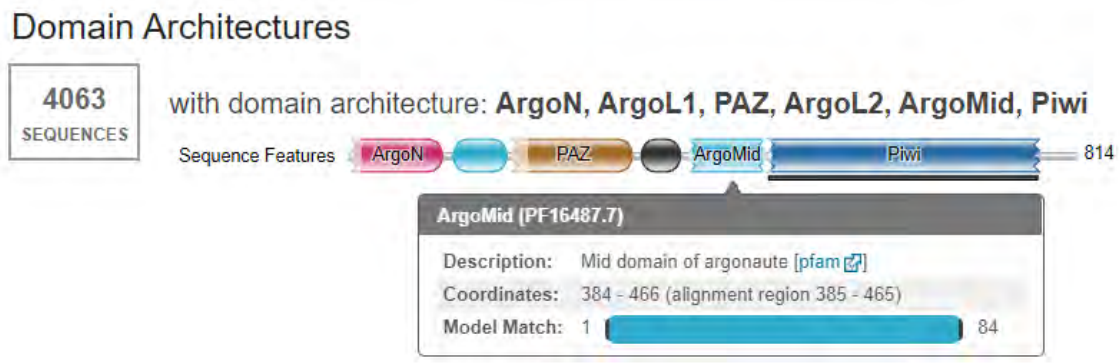

Figura 23. *Detalle de la arquitectura de dominio constituida por las familias ArgoN, ArgoL1, PAZ, ArgoL2, ArgoMid y Piwi. W2T741_NECAM es una proteína que comparte ese dominio. Se observa la posición inicial y final (384-466) del dominio ArgoMid dentro de la secuencia completa de la proteína. Obtenido de (Banco de Datos de Proteínas en Europa, 2021).*

En ese sentido, tal como se muestra en la Figura 23, el fragmento perteneciente a la familia ArgoMid con código PFAM PF16478, se encuentra en la secuencia de la proteína W2T741_NECAM a partir del aminoácido en la posición 384 hasta la posición 466.

Estas posiciones son las que se tomarán como referencia para la segunda modificación que describe el segundo camino a seguir con los fragmentos sin estructura.

En este punto, se deberán obtener las secuencias completas de las proteínas en las que se encuentran los fragmentos que llegaron a esta fase. Esto se logrará mediante el uso de servicios de Uniprot y el identificador encontrado en el archivo PFAM. Una vez que se cuente con la secuencia de obtendrá una subsecuencia en base a la mínima posición inicial de todos los fragmentos que las constituyan. De forma análoga, el límite superior de la subsecuencia será definido por la máxima posición final de todos los fragmentos. Con ello, se logra una reducción del tamaño de la secuencia, teniendo en cuenta que el algoritmo adaptado no se deberá enfocar en la predicción de toda la secuencia puesto que no corresponde al alcance, sino que tiene principal interés en el modelamiento de la estructura terciaria de los fragmentos pertenecientes a las familias de proteínas repetidas. Esto último lo conseguimos con apoyo crucial del archivo PFAM perteneciente a este grupo en particular de familias de proteínas, las proteínas repetidas.

Estas subsecuencias se almacenarán en archivos con formato fasta y serán los datos de entrada del script de generación de archivos intermedios con extensión .21c y ,map.

Llegado a este punto, se cuenta con los tres archivos requeridos para utilizar el algoritmo central de predicción con el cual se obtiene un archivo PDB. Este archivo contiene el detalle de las coordenadas de cada uno de los aminoácidos de la proteína.

Cabe mencionar que las dos modificaciones descritas hasta ahora corresponden al ingreso de un código PFAM como dato de entrada del algoritmo adaptado. No obstante, en caso se ingrese una secuencia de proteína de forma directa de deberá utilizar el script de archivos intermedios y, posteriormente, la predicción de la estructura general.

La tercera actividad que se suma al algoritmo, entendiéndose como una modificación, se ajusta mucho más al segundo tipo de dato de entrada definido: la secuencia de proteínas. Para la inclusión de la propuesta ya se debe haber generado la estructura terciaria; sin embargo, se debe tener en cuenta que la predicción en esta sección no está enfocada a las proteínas repetidas. Esto es por lo cual se utilizará una herramienta adicional denominado RepeatsDBLite. Este nuevo servicio permite la identificación de

los fragmentos de la secuencia pertenecientes a las familias de proteínas repetidas y la generación de sus estructuras tridimensionales independientes. Todo ello, en base a la estructura predicha ingresada como dato. Adicionalmente, se obtiene una alineación de estas estructuras, donde se deberá observar la afinidad estructural de los fragmentos pertenecientes a una misma familia.

Por último, la cuarta modificación se ubica más en la etapa de evaluación de la estructura terciaria generada a partir de la predicción. Esta se fundamenta en la afirmación de que las unidades de repetición que pertenecen a una misma familia comparten características similares. Lo cual se reconoce en mayor profundidad cuando se comparan las estructuras de las mismas. Así, para poder evaluar la predicción de estructura de un fragmento de proteína, esta se alineará con la estructura alojada en PDB de algún fragmento perteneciente a la misma familia. El hecho de que se encuentre en PDB significa que la estructura ha sido generada previamente y validada por la comunidad bioinformática a través de distintos métodos tanto computacionales como manuales.

Anexo G: Catálogo de requisitos funcionales y no funcionales de la herramienta propuesta

En este anexo se encuentra el listado de los requisitos funcionales y no funcionales de la herramienta propuesta. Su contenido abarca el catálogo de requisitos, además del acta de aceptación de lo descrito elaborado por un experto en bioinformática.

1. Introducción

Siguiendo las buenas prácticas del marco de trabajo seleccionado para desarrollar la herramienta propuesta en el presente proyecto de tesis, es fundamental contar con la definición de un catálogo de requisitos especificados en conjunto con el cliente. Para este contexto en particular se tomó en cuenta la revisión sistemática realizada en el Capítulo 3. Estado del Arte. Las respuestas a las preguntas de la revisión sirvieron de apoyo para entender el contexto en el que se desarrolla el proyecto y para comprender el funcionamiento actual de las herramientas afines a la propuesta. En el mismo sentido, facilitaron el reconocimiento de los requisitos primordiales para los usuarios que requieren de la predicción de estructuras de proteínas. Estos requisitos identificados posteriormente serán revisados y aprobados por parte de un experto en bioinformática.

2. Requisitos funcionales y no funcionales de la herramienta

Este apartado contiene la lista de requisitos funcionales y no funcionales de la herramienta propuesta. A continuación, en la Tabla 29, se presentan las exigencias identificadas, así como su identificación, el tipo de requisito (Funcional o No Funcional) y el tipo de exigencia (Deseable o Exigible).

Listado de requisitos			
ID	**Descripción**	**Tipo**	**Exigencia**
R1	El sistema deberá generar modelos de apoyo como la predicción de contacto entre residuos de proteínas o los resultados de las alineaciones de secuencias múltiples (MSA, por sus siglas en inglés)	Funcional	Deseable
R2	El sistema deberá permitir al usuario cambiar el idioma del servicio	Funcional	Deseable
R3	El sistema deberá permitir el ingreso de secuencias de proteínas sin límite de residuos	Funcional	Deseable

R4	El sistema deberá permitir al administrador el ingreso a un módulo privado de configuración a través de un usuario y contraseña	Funcional	Deseable
R5	El sistema deberá predecir la estructura terciaria de una proteína repetida a partir de su estructura primaria	Funcional	Exigible
R6	El sistema deberá predecir la estructura terciaria de proteínas repetidas a partir de un identificador de la familia de proteínas	Funcional	Deseable
R7	El sistema deberá verificar que la secuencia de proteína ingresada como dato de entrada solo esté conformado por los veinte aminoácidos fundamentales	Funcional	Exigible
R8	El sistema deberá permitir el registro de solicitudes de predicción sin un inicio de sesión	Funcional	Exigible
R9	El sistema deberá permitir solo el ingreso de la secuencia de aminoácidos en formato de texto plano, en formato Fasta o en formato Stockholm	Funcional	Exigible
R10	El sistema solicitará de forma opcional el registro de un identificador de la tarea y un correo previo al registro de la solicitud de predicción	Funcional	Exigible
R11	El sistema deberá almacenar la información de cada solicitud de predicción ingresada (ID de solicitud, fecha de solicitud, correo electrónico, tipo de dato de entrada)	Funcional	Exigible
R12	El sistema deberá enviar la estructura tridimensional predicha al usuario a través de un correo electrónico	Funcional	Exigible
R13	El sistema deberá enviar los resultados intermedios obtenidos en la predicción a través de un correo electrónico	Funcional	Deseable
R14	El sistema deberá mostrar la estructura terciaria predicha a través de links dentro de la interfaz	Funcional	Exigible
R15	El sistema deberá permitir al usuario consultar solicitudes que hayan sido ingresadas previamente (solicitudes en general)	Funcional	Exigible
R16	El sistema deberá mostrar al usuario un resumen del proceso de predicción (los datos ingresados, el estado del proceso y la predicción obtenida).	Funcional	Exigible
R17	El sistema deberá permitir al usuario la visualización de la estructura terciaria predicha en la propia interfaz	Funcional	Exigible
R18	El sistema deberá permitir exportar los resultados de la predicción en PDB	Funcional	Exigible

R19	El sistema deberá contar con una sección de tutoriales, documentación y ayuda online para guiar al usuario en torno a su uso	Funcional	Exigible
R20	El sistema deberá notificar al usuario todo tipo de error en la misma ventana en la que ocurrió y solicitar su confirmación para proseguir	Funcional	Exigible
R21	El sistema deberá contar con una sección de información adicional y links externos sobre la problemática y el proyecto	Funcional	Exigible
R22	El sistema deberá permitir al administrador la gestión de parámetros generales del procesamiento	Funcional	Deseable
R23	El sistema deberá permitir al usuario diferenciar los campos obligatorios y opcionales de los formularios	Funcional	Exigible
R24	El sistema deberá permitir a un mismo usuario ingresar varias solicitudes de predicción	Funcional	Exigible
R25	El sistema deberá permitir la carga de la estructura primaria de una proteína repetida a través de un archivo	Funcional	Exigible
R26	El sistema deberá poner a disposición del usuario datos de entrada de ejemplo para utilizar la plataforma	Funcional	Exigible
R27	El sistema deberá permitir al administrador gestionar el límite de solicitudes de predicción por usuario	Funcional	Deseable
R28	El sistema deberá ser compatible con navegadores como Chrome, Edge y Explorer.	No funcional	Deseable
R29	El sistema deberá entregar resultados del procesamiento en un lapso no mayor a 24 horas	No funcional	Deseable
R30	La interfaz del sistema deberá estar desarrollado con HTML, CSS y JavaScript	No funcional	Exigible
R31	La base de datos del sistema deberá ser MySQL	No funcional	Exigible
R32	El sistema deberá poder recibir múltiples solicitudes de predicción	No funcional	Exigible
R33	El sistema deberá refrescar la información a visualizar sobre el estado de la predicción de forma automática	No funcional	Deseable
R34	El sistema deberá contar con un diseño intuitivo y minimalista	No funcional	Exigible
R35	El sistema deberá estar disponible las 24 horas de los 7 días de la semana. (En caso ocurra algún error del sistema,	No funcional	Exigible

	se permitirá la no disponibilidad de este hasta que se corrija el desperfecto)		
R36	El sistema deberá constituir el desarrollo de una plataforma web	No funcional	Exigible
R37	El sistema deberá cumplir con estándares de usabilidad (niveles aceptables)	No funcional	Exigible
R38	El sistema deberá solicitar información solo antes del inicio del procesamiento	No funcional	Exigible
R39	La interfaz del sistema deberá contar con elementos consistentes respecto a sus colores y formas	No funcional	Exigible
R40	El sistema deberá estar disponible en el idioma inglés	No funcional	Exigible
R41	El sistema deberá generar archivos logs de las ejecuciones del usuario y de los errores	No funcional	Deseable

Tabla 29. *Listado de requisitos funcionales y no funcionales de la herramienta propuesta.*

3. Apéndice

En esta sección se presenta el acta de validación del presente documento de requisitos funcionales y no funcionales de la herramienta propuesta.

3.1 Validación del documento por medio de juicio experto

La validación del documento del catálogo de requisitos funcionales y no funcionales de la herramienta propuesta ha sido realizada por una experta en bioinformática. Esta verificación implica la revisión completa del informe y la entrega de observaciones en caso se requieran. A continuación, se observa el acta de validación recibida por la experta.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Acta de validación de documento

Título de tesis: Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

Tesista: Solange Estrella Palomino Chahua

Nombre del documento: Catálogo de requisitos funcionales y no funcionales de la herramienta propuesta

Descripción del documento: Documento que contiene la especificación de los requisitos funcionales y no funcionales de la herramienta propuesta

Mediante la presente acta, yo, la Dra. Layla Hirsh Martinez, dejo constancia de que, en mi calidad de experta en bioinformática, he revisado el documento descrito en los puntos anteriores perteneciente al proyecto de tesis en mención. En ese sentido, en la siguiente sección se especifica el veredicto y, en caso hubiesen, las observaciones correspondientes al documento.

Veredicto:

(X) Aprobado ( ) Requiere observación

Observaciones:

Lima, 02 de octubre de 2021

Firma

Anexo H: Reporte del mockup de la interfaz de la herramienta propuesta

Este anexo contiene el detalle del prototipo de alta fidelidad de la interfaz de la herramienta propuesta. Se muestran las pantallas diseñadas para una plataforma web, además de una descripción de la interacción entre ellas. Asimismo, muestra y describe el diagrama de flujo de la herramienta. Finalmente, se adjunta la aceptación por parte de juicio experto de lo presentado.

1. Introducción

Previo al desarrollo de la interfaz de la plataforma propuesta como parte del resultado esperado del segundo objetivo del presente proyecto se ha capturado la necesidad de diseñar un prototipo de alta fidelidad. El desarrollo de este mockup constituye la definición de los elementos que se mostrarán en las pantallas, las secciones que se incluirán en la herramienta, los estilos de las fuentes y la paleta de colores a utilizar. Además, implica la definición de la interacción entre las pantallas y el usuario, con lo cual se detallará el diagrama de flujo de las actividades necesarias para poder llevar a cabo el registro de una solicitud de predicción de estructuras terciarias de proteínas repetidas.

2. Definición de estándares de diseño de interfaz

El presente documento está relacionado al resultado alcanzado perteneciente al segundo objetivo específico del presente proyecto. Este corresponde al prototipo de la interfaz de la herramienta de predicción. Es por ello que, en este apartado, se incluyen las pantallas que lo conforman. No obstante, es importante definir, en primer lugar, ciertos estándares que se seguirán a lo largo del proceso de diseño.

En este punto cabe mencionar que, para elaborar los estándares de diseño, se ha tomado en cuenta la respuesta a la tercera pregunta de la revisión sistemática desarrollada en el Capítulo 3. Estado del Arte. En esa sección se describe el contexto en el que se han desarrollo diversos servicios relacionados al presente proyecto. Se detallan además ciertos criterios de usabilidad relevantes para el desarrollo de una plataforma web.

A partir del reconocimiento de la necesidad de la definición de ciertos aspectos de usabilidad, se ha establecido que la paleta de colores estará constituida por la colección de ocho colores. En la primera fila de la Figura 24 se muestran los cinco colores principales escogidos para el diseño de la interfaz, mientras que los colores de la

segunda fila serán utilizados de forma secundaria, ya sea en iconos, mensajes particulares o el fondo de las secciones.

Figura 24. Paleta de colores de la interfaz de la herramienta propuesta. (Elaboración propia).

Posteriormente, se prosiguió a definir las fuentes que se utilizarán en el diseño de las interfaces. Se eligieron tres fuentes, incluyendo sus variaciones, las cuales se muestran en la Figura 25: Open Sans, Nunito Sans y Montserrat. Esta última será usada básicamente para la especificación del nombre de la herramienta.

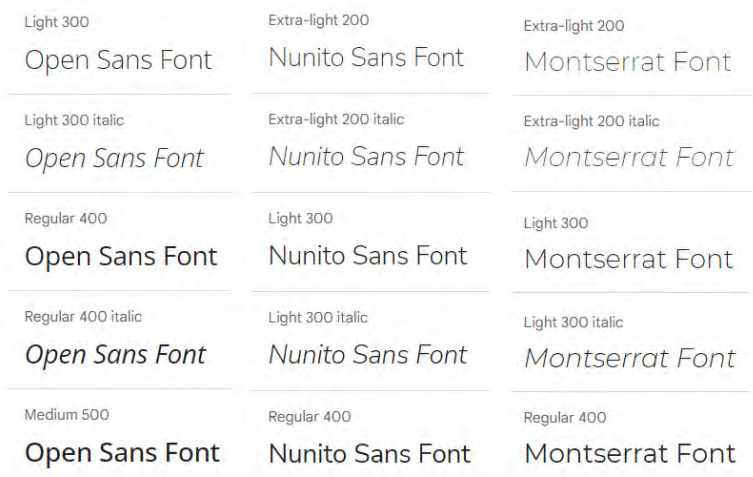

Figura 25. Colección de fuentes seleccionadas para el diseño de la interfaz de la herramienta propuesta. (Elaboración propia en Google Fonts).

3. Diseño de las pantallas del prototipo de la interfaz de la herramienta propuesta

La definición de las pantallas de la interfaz del prototipo de la herramienta propuesta se ha basado en la especificación de requisitos funcionales y no funcionales descritos en el catálogo del Anexo G. En ese sentido, se ha establecido la inclusión de cinco vistas fundamentales y exigibles en la plataforma web.

La primera de las cinco vistas fundamentales es la pantalla principal de la interfaz. Esta será la primera interacción entre el usuario y la herramienta por lo cual contendrá un formulario que permitirá el ingreso directo de los datos de entrada para el registro de una solicitud de predicción. Esta vista se puede observar en la Figura 26.

La predicción de las estructuras terciarias de las proteínas no se realiza de forma instantánea sino que forma parte de un proceso que requiere de un tiempo de espera para la obtención de resultados. Es por ello que la herramienta trabajará en base al registro de solicitudes que deberán ser ingresadas para, posteriormente, consultar por ellas y visualizar sus resultados.

Acorde al catálogo de requisitos definidos, los datos de entrada necesarios para poder registrar una solicitud de predicción son los siguientes: en primer lugar, el identificador PFAM de una familia de proteínas repetidas o la secuencia de aminoácidos de una proteína repetida, esta última ya sea directamente desde el campo de entrada de texto o desde un archivo. En este caso, la herramienta solo permitirá el ingreso de archivos con formato Fasta, Stockholm o un archivo de texto simple con extensión .fasta, .sto o .txt, respectivamente.

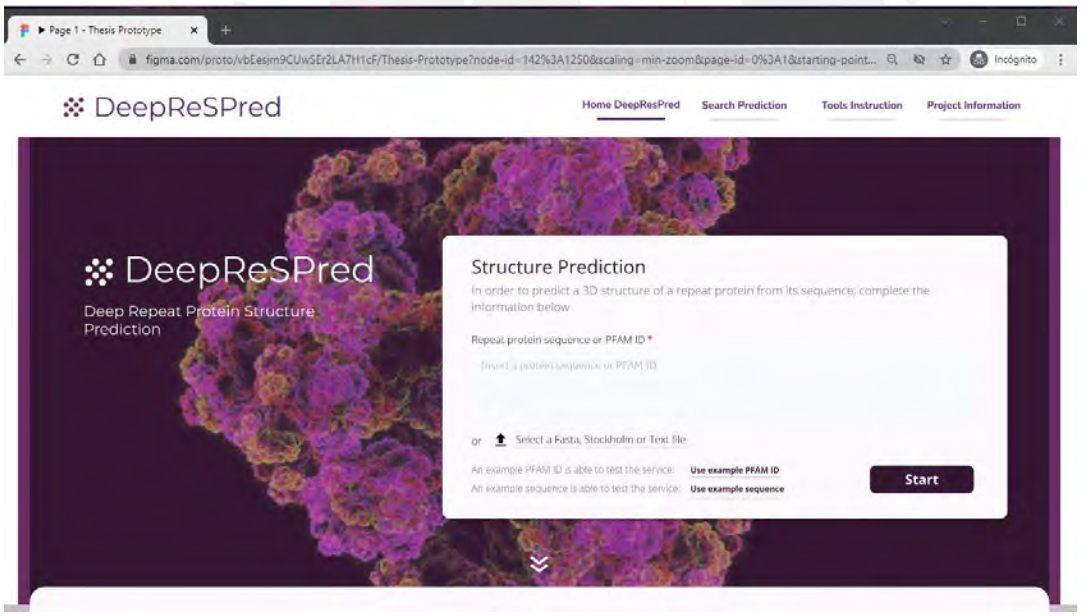

Figura 26. *Pantalla principal del prototipo de la interfaz de la herramienta propuesta. (Elaboración propia).*

De forma adicional, el formulario de esta pantalla pondrá a disposición del usuario dos enlaces ubicados en su zona inferior. Estos podrán ser seleccionados para ingresar datos de entrada de ejemplo correspondientes a los dos tipos de datos permitidos para realizar la solicitud de predicción, es decir, un identificador PFAM o la secuencia de una

proteína. De esta manera, con pocos *clicks*, el usuario podrá hacer uso del servicio de predicción denominado como DeepReSPred, Deep Repeat protein Structure Predictor.

En este punto, el usuario deberá presionar el botón 'Start' con el cual se mostrará una pantalla modal como la que se observa en la Figura 27. Esta pantalla funciona como una confirmación por parte del usuario del proceso que está por iniciar. En ese sentido, muestra un resumen de los datos ingresados, además de solicitar el ingreso de datos adicionales como el identificador de la solicitud y un correo. La herramienta generará un identificador único para la nueva solicitud; sin embargo, se dejará la posibilidad modificarla según se requiera, por lo tanto este campo es considerado como opcional. El correo ingresado, también de forma opcional, servirá para enviar una confirmación del registro de la solicitud de predicción especificando su identificador. Su funcionalidad se describirá posteriormente.

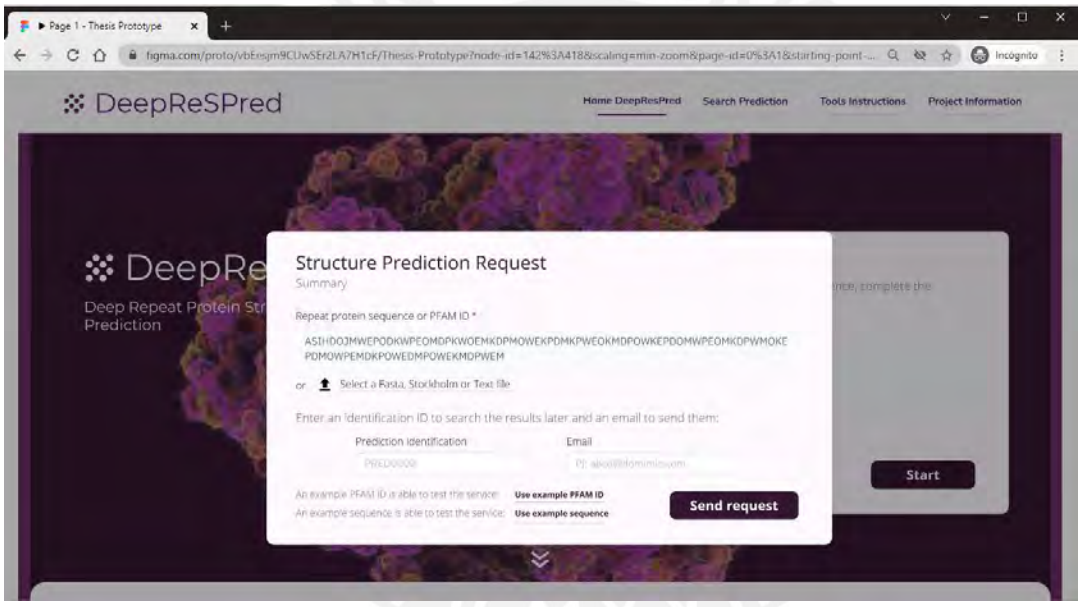

Figura 27. *Pantalla de prototipo de la interfaz de la herramienta propuesta. Resumen de la solicitud de predicción. (Elaboración propia).*

A medida que el usuario ingrese los datos necesarios, la interfaz irá validándolos. En caso no sean admitidos, se mostrará un mensaje descriptivo que ayude al usuario a identificar el error. Por ejemplo, se verificará la estructura del correo, en caso se ingrese. Asimismo, se validará que los caracteres ingresados en el campo de entrada de texto obligatorio correspondan a la simbología de los aminoácidos existentes. Esto en caso de que se ingrese una secuencia de proteínas y no un identificador PFAM. Se puede revisar la simbología de los aminoácidos permitidos en el apartado de la descripción del término *estructura primaria de proteína* del Capítulo 2. Marco Conceptual.

En la Figura 28, se muestra el mensaje de validación obtenido a partir del ingreso de caracteres inválidos en una secuencia de proteína. El mensaje traducido indica lo siguiente: “Entrada de secuencia no válida. Verifique los símbolos correctos de los residuos”.

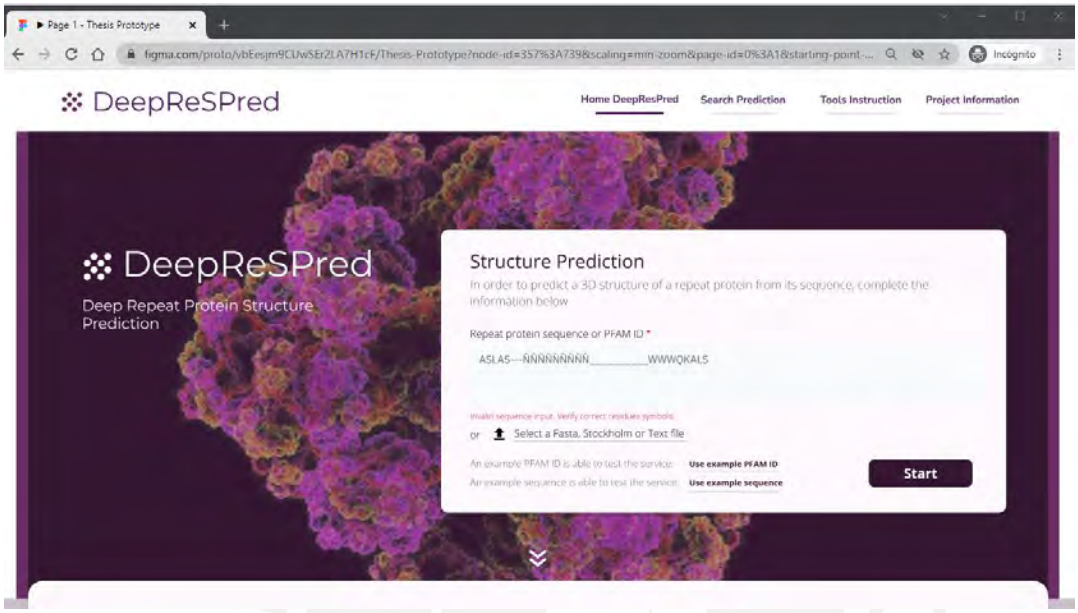

Figura 28. *Pantalla de prototipo de la interfaz de la herramienta propuesta. Ejemplo de mensaje de validación del campo de entrada de texto de una secuencia. (Elaboración propia).*

Una vez que los datos sean correctos, se habilitará la opción de envío de la solicitud de predicción. Al presionarlo se espera que se muestre un mensaje de confirmación o un mensaje de error en base a una primera respuesta del algoritmo. En caso no exista ningún error, el modal que contiene el mensaje permitirá el acceso a la sección de búsqueda de la solicitud ingresada. Esta sección corresponde a una de las cuatro pestañas ubicadas en la zona inferior de la pantalla principal: la pestaña de búsqueda de solicitud de predicción, la pestaña de las instrucciones de uso de la herramienta, la pestaña de la información del proyecto y la pestaña de la bibliografía. De forma adicional, se podrá acceder a estas secciones a través de la barra de navegación de la zona superior de la interfaz.

Tal como se muestra en la Figura 29, la sección de búsqueda de una solicitud de predicción permitirá el ingreso del identificador único generado en su registro. Inicialmente, se visualizarán los campos que serán completados una vez que se obtengan los datos de la solicitud registrada y su procesamiento.

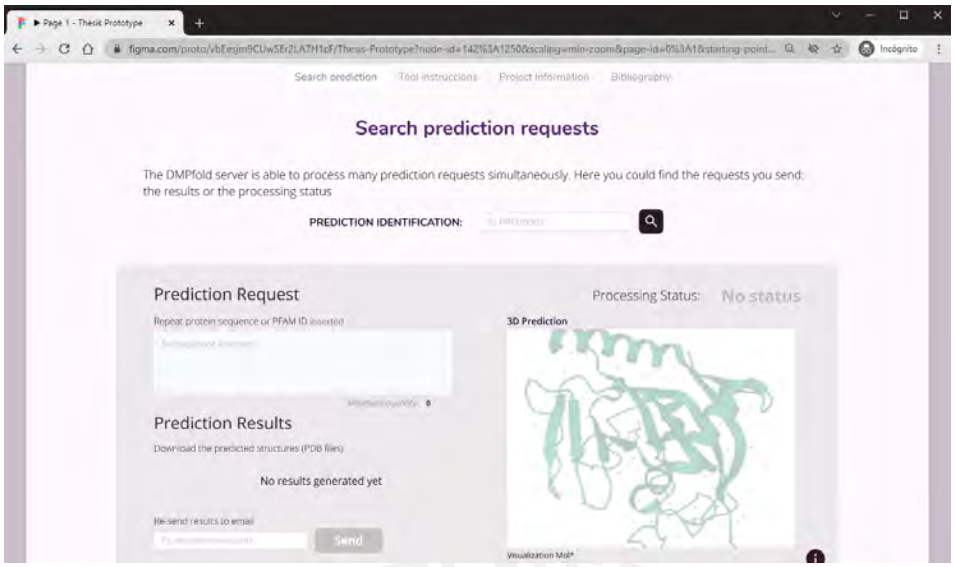

Figura 29. Pantalla de prototipo de la interfaz de la herramienta propuesta. Vista inicial de la sección de búsqueda de solicitud de predicción. (Elaboración propia).

Luego de ingresar un identificador de una solicitud de predicción registrada, se podrán esperar dos estados: en proceso o completado. De forma excepcional, si no se encuentra el identificador, se mostrará una ventana modal con un mensaje descriptivo. La Figura 30 detalla el caso exitoso en el que el algoritmo ha terminado el procesamiento de la predicción. Así, se podrá visualizar la secuencia de la proteína ingresada, el recuento de sus aminoácidos, los archivos PDB generados y una sección que permitirá su visualización. Por último, el usuario tendrá la posibilidad de ingresar un correo para volver a enviar los resultados obtenidos.

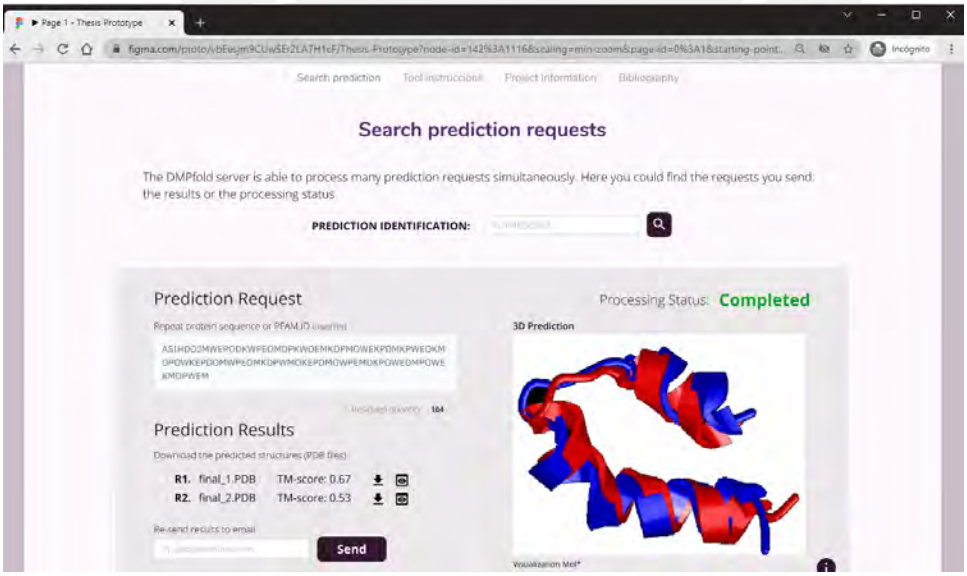

Figura 30. Pantalla de prototipo de la interfaz de la herramienta propuesta. Vista de proceso completado en la sección de búsqueda de solicitud de predicción. (Elaboración propia).

4. Diagrama de flujo de actividades de la herramienta propuesta

El presente acápite detalla la interacción esperada entre la interfaz de la herramienta propuesta y los usuarios. En ese sentido, las actividades que se deberán seguir para ingresar una solicitud de predicción de la estructura terciaria de una proteína repetida se especifican en el diagrama de flujo de la herramienta propuesta.

Tal como se observa en la Figura 31, el proceso inicia con el ingreso del usuario a la página web de la herramienta. Ante ello, el sistema reacciona desplegando su pantalla principal que contiene el formulario de datos de entrada y los enlaces de datos de prueba. En este punto, lo siguiente depende de la decisión del usuario respecto a usarlos o ingresar nuevos datos: una secuencia de proteínas o un identificador PFAM desde el campo de entrada de texto o desde un archivo. Los formatos permitidos de los archivos serán Fasta, Stockholm y texto plano.

Al presionar la opción de inicio de predicción, la herramienta generará un identificador de la nueva solicitud, la cual podrá ser modificada por el usuario. Este campo, y el que permite el ingreso de un correo electrónico del usuario, se visualizará en una ventana modal. Adicionalmente, se detallará el dato de entrada obligatorio requerido en la predicción.

La plataforma validará los datos ingresados y, en caso alguno no corresponda a lo requerido, mostrará un mensaje acorde en cada campo. Esto exigirá que los usuarios los revisen y vuelvan a ingresar otros.

Una vez que los datos sean correctos se inician dos flujos paralelos. El primero corresponde al envío de un correo de confirmación del registro de la predicción, en caso se haya ingresado un correo electrónico, y a lo visible para el usuario desde la interfaz de la herramienta. Esto implica el redireccionamiento hacia la pestaña de búsqueda de solicitudes de predicción, además de la actualización del estado del procesamiento. Por otro lado, el segundo flujo corresponde al proceso que se ejecute en un entorno back-end.

Este segundo camino está conformado por las actividades que realiza un servidor con comunicación directa con la base de datos. Así, a partir de la selección del botón de ‘Enviar solicitud’ por parte del usuario, este registrará los datos ingresados en una tabla específica. Seguidamente, se verificará la existencia de alguna solicitud de predicción registrada previamente, el cual contenga los mismos datos ingresados en la nueva solicitud. En caso negativo, se encolará y se esperará a la finalización de los procesos

en actual ejecución, para luego proseguir con la nueva predicción. Esta sección se especifica en el diagrama como un subproceso que corresponde al algoritmo adaptado en el primer objetivo específico del presente proyecto de fin de carrera.

En caso se encuentre una solicitud de predicción análoga a la ingresada, la herramienta asignará en la base de datos los resultados generados de ese procesamiento.

Finalmente, si el proceso de predicción se ejecuta con errores entonces se le notificará al usuario con un mensaje acorde. Por otro lado, si la ejecución se realiza satisfactoriamente, se enviará un correo electrónico con los archivos generados en la predicción. Esto en caso el usuario haya registrado un correo.

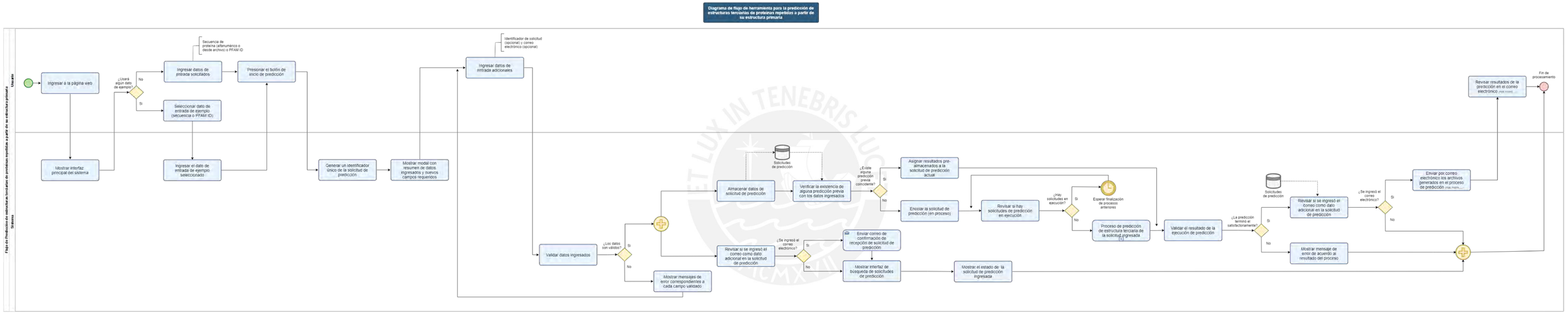

Figura 31. *Diagrama de flujo de actividades de la herramienta propuesta – Notación BPMN. (Elaboración propia).*

5. Apéndice

En esta sección se presenta un enlace de acceso al prototipo diseñado de la interfaz de la herramienta. Asimismo, se incluye el acta de validación del presente documento de presentación del prototipo de alta definición de la interfaz de la herramienta propuesta aprobado por parte de un experto en bioinformática.

5.1 Prototipo interactivo de la interfaz de la herramienta propuesta

El prototipo de la interfaz de la herramienta propuesta ha sido desarrollado con la herramienta de diseño Figma. Mediante el siguiente enlace se podrá acceder al mockup de alta definición e interactuar con los elementos de las pantallas.

https://www.figma.com/proto/vbEesjm9CUwSEr2LA7H1cF/Thesis-Prototype?node-id=142%3A1250&scaling=min-zoom&page-id=0%3A1&starting-point-node-id=142%3A1250

Cabe mencionar que la visualización del prototipo es de acceso libre, por lo cual no se requiere de una cuenta de usuario en la herramienta utilizada.

5.2 Validación del documento por medio de juicio experto

La validación del documento del prototipo de alta fidelidad de la interfaz de la herramienta propuesta ha sido realizada por una experta en bioinformática. Esto brinda un punto adicional a la evaluación del mockup propuesto dado que plataforma será utilizada por usuarios pertenecientes al campo de investigación al que también está relacionada la experta seleccionada. Esta verificación implica la revisión completa del informe, la navegación del prototipo de la interfaz de la plataforma propuesta y la entrega de observaciones en caso se requieran. A continuación, se observa el acta de validación recibida por la experta.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Acta de validación de documento

Título de tesis: Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

Tesista: Solange Estrella Palomino Chahua

Nombre del documento: Reporte del mockup de la interfaz de la herramienta propuesta

Descripción del documento: Documento que contiene el prototipo de alta fidelidad de la interfaz de la herramienta propuesta incluyendo las pantallas y su navegación. Asimismo, cuenta con el diagrama de flujo que describe los pasos a seguir para llevar a cabo la predicción de la estructura terciaria de la proteína repetida a partir de su secuencia de aminoácidos.

Mediante la presente acta, yo, la Dra. Layla Hirsh Martinez, dejo constancia de que, en mi calidad de experta en bioinformática, he revisado el documento descrito en los puntos anteriores perteneciente al proyecto de tesis en mención. En ese sentido, en la siguiente sección se especifica el veredicto y, en caso hubiesen, las observaciones correspondientes al documento.

Veredicto:

(X) Aprobado ( ) Requiere observación

Observaciones:

Lima, 12 de octubre de 2021

Firma

Anexo I: Documento descriptivo del algoritmo adaptado para la predicción de estructuras terciarias de proteínas repetidas

El presente anexo contiene el documento que describe el algoritmo adaptado para la predicción de estructuras terciarias de proteínas repetidas. Este reporte es uno de los dos medios de verificación del tercer resultado esperado del primer objetivo específico[44] de este proyecto de tesis. Su contenido abarca el diagrama y la descripción funcional del algoritmo de predicción que incluye las modificaciones propuestas previamente. En adición, se adjunta como un apéndice al documento, el acceso al código fuente del desarrollo.

1. Introducción

A modo de recapitular los avances del presente proyecto, cabe mencionar que en el segundo resultado del primer objetivo específico se alcanzó lo esperado. Se propusieron distintas modificaciones al algoritmo que fue seleccionado previamente.

El presente documento detalla el proceso de aplicación de esas modificaciones propuestas al algoritmo seleccionado con el fin de cubrir los resultados esperados del mismo objetivo específico. Las modificaciones darán pie a la obtención de un algoritmo adaptado según los requerimientos de las personas interesadas en la predicción de estructuras terciarias de proteínas, en específico de proteínas repetidas, a partir de su estructura primaria.

En general, el algoritmo adaptado es el resultado de la compilación del algoritmo original con un preprocesamiento de los datos ingresados y un procesamiento posterior a la predicción realizada. Para lograr una funcionalidad completa y relevante para los usuarios interesados en las proteínas repetidas, se ha utilizado una serie de servicios gratuitamente disponibles en la web. Muchos de ellos pertenecen a diversos consorcios como Uniprot, RSCB Protein Data Bank, Swiss-Prot y PFAM. Los enlaces a los mismos se incluirán en la descripción funcional del algoritmo.

2. Aplicación de modificaciones propuestas y descripción funcional del algoritmo adaptado

La primera de las cuatro modificaciones propuestas en base a las necesidades en torno a las proteínas repetidas corresponde a un enfoque en particular. Este requiere de un

[44] O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.

identificador PFAM como dato de entrada al algoritmo adaptado. Se utilizará un servicio de PFAM para poder obtener los distintos fragmentos pertenecientes a las diversas familias de proteínas repetidas.

El servicio utilizado para generar el archivo de fragmentos de secuencias de una familia de proteínas es el siguiente:

https://pfam.xfam.org/family/alignment/download/format?acc={{pfamCode}}&alnType=full&format=fasta&order=t&case=l&gaps=default&download=1

Para poder usarlo se deberá reemplazar la variable *{{pfamCode}}* por un identificador de una familia en específico, como por ejemplo PF03377 perteneciente a la familia de proteínas "TAL effector repeat". La Figura 32 muestra un ejemplo del contenido en formato Fasta del archivo obtenido por el servicio mencionado. Cada par de líneas corresponde a la información de un fragmento de secuencia de proteína. En el contexto de las proteínas repetidas, cada fragmento de la familia corresponde a una unidad de repetición. La primera línea está formada por el identificador uniprot de la secuencia donde la unidad de repetición fue identificada, seguido de la posición inicial y la posición final que ocupa el fragmento dentro de la secuencia. La segunda línea es la secuencia de aminoácidos de la unidad de repetición.

```
>Q5H0J1_XANOR/371-404
..IGGNQALETVQRL......LP......VLCQ.aHGLTPDQVVAIASH.......
>BAT1_PARRH/161-193
..NGGAQALYSVLDV......EP......TLGK..RGFSRADIVKIAGN.......
>Q5H0J1_XANOR/881-914
..DGGKQALETVQRL......LP......VLCQ.dHGLTPDQVVAIASH.......
>Q5GUW4_XANOR/758-791
..DGGKQALETVQRL......LP......VLCQ.dHGLTPDQVVAIASN.......
>Q5H0I8_XANOR/733-766
..DGGKQALETVQRL......LP......VLCQ.dHGLTPDQVVAIASN.......
>Q5GUW4_XANOR/826-859
..IGGKQALETVQRL......LP......VLCQ.dHGLTPDQVVAIASH.......
>BAT1_PARRH/491-523
..RGGAQALQAVLAL......EL......TLRE..RGFSQPDIVKIAGN.......
>Q5H0I8_XANOR/1004-1037
..IGGKQALETVQRL......LP......VLCQ.dHGLTPDQVVAIASN.......
```

Figura 32. *Archivo en formato fasta obtenido a partir del consumo del servicio de PFAM. Familia de proteínas repetidas TAL-effector PF03377. (Elaboración propia).*

Una vez que contamos con los fragmentos de la familia de proteínas repetidas en cuestión, se ha planteado discriminar a los fragmentos que hayan sido identificados en secuencias que ya cuentan con una estructura terciaria predicha debido a que no corresponde al objetivo del presente proyecto. La información estructural de las secuencias se encuentra alojada en el Banco de Datos de Proteínas. Esta base de datos también pone a disposición de los interesados, una serie de recursos para consultar la

información que almacena. En este punto se utilizará la siguiente API para poder capturar la información estructural de cada una de las secuencias obtenidas de la familia de proteínas que se está analizando:

https://www.ebi.ac.uk/pdbe/search/pdb/select?q=uniprot_accession:{{datoAccession}}&wt=json

Este recurso requiere que se le envíe el acceso de uniprot en reemplazo de la variable *{{datoAccession}}*. No obstante, la información que se obtiene del archivo PFAM en cada encabezado de una secuencia corresponde al identificador de uniprot, los cuales son valores distintos que identificarán a una secuencia. Se requiere de un intermediario que permita relacionar un valor con el otro para poder utilizar el recurso de PDB.

La base de datos Swiss-Prot del consorcio Uniprot es un repositorio de información biológica de secuencias de proteínas de donde se puede obtener la relación entre el identificador Uniprot y el acceso de uniprot. Por ello, se decidió utilizar sus datos como una especie de traductor de identificadores. A modo de incrementar la eficiencia de esta actividad, se realizó un filtrado de la información de esa base de datos, para eliminar todos los datos adicionales no concernientes al objetivo del presente proyecto, y se generó un nuevo fichero con dos tipos de datos: los identificadores uniprot y sus correspondientes accesos uniprot. En la Figura 33, se observa una pequeña muestra del contenido del fichero generado, donde la columna izquierda corresponde al acceso uniprot y la columna derecha, al identificador uniprot. En otras palabras, se muestran en dos columnas lo que se necesita vs lo que se tiene.

```
Q6GZX4 001R_FRG3G
Q6GZX3 002L_FRG3G
Q197F8 002R_IIV3
Q197F7 003L_IIV3
Q6GZX2 003R_FRG3G
Q6GZX1 004R_FRG3G
Q197F5 005L_IIV3
Q6GZX0 005R_FRG3G
Q91G88 006L_IIV6
Q6GZW9 006R_FRG3G
Q6GZW8 007R_FRG3G
Q197F3 007R_IIV3
```

Figura 33. *Ejemplo del fichero generado en base a los identificadores uniprot (columna derecha) y sus correspondientes accesos uniprot (columna izquierda) obtenidos de SwissProt. (Elaboración propia).*

Con la ayuda de ese fichero se puede traducir el identificador uniprot obtenido del archivo PFAM de la familia de proteínas hacia el acceso uniprot, lo cual se requiere para

poder utilizar el API de PDB para conocer la información estructural de las secuencias en cuestión.

En el caso del identificador uniprot “BAT1_PARRH”, se obtuvo el acceso uniprot “E5AV36”, el cual es ingresado al API de PDB y se obtiene información de las familias de proteínas que contiene, la cantidad de estructuras relacionadas y demás. Cuando el parámetro “numFound” contiene un valor distinto a cero, la secuencia está relacionada a una estructura terciaria, por lo tanto, se deberá descartar del proceso del algoritmo adaptado. Esta reducción de la cantidad de las predicciones ayudará a que el algoritmo ofrezca resultados en un menor tiempo y a que los resultados se enfoquen enteramente en el objetivo del proyecto.

Por cada uno de los fragmentos que no se encuentra en una secuencia relacionada a una estructura de proteína se generará un archivo en formato fasta. La cadena de aminoácidos se registrará luego de la depuración de sus datos. Esto significa que los caracteres de punto “.”, guión “-“, o cualquier otro que no corresponda a la simbología de los veinte aminoácidos naturales, serán retirados de la secuencia. Asimismo, todos los caracteres serán convertidos a mayúsculas a modo de estandarización. Estos archivos serán nombrados según el identificador uniprot de la cabecera de la unidad de repetición que contiene, añadido a un prefijo “is” correspondiente a lo que se denominará en este proyecto como una subsecuencia independiente o “independent subsequence”.

Dado que una misma secuencia puede contener diversos fragmentos de una misma familia, se buscará agruparlos para poder predecir la estructura de la secuencia más pequeña que incluya a todas las unidades de repetición. Esto es especialmente interesante ya que representa un segundo enfoque donde el algoritmo de predicción solo se tendría que ejecutarse una vez, en vez de predecir la estructura de cada unidad para posteriormente interpolarla en la estructura completa. En todo caso, ambos caminos serán incluidos en el algoritmo adaptado.

Así, continuando con la segunda perspectiva, se buscará el menor mínimo de las posiciones de los fragmentos dentro de cada secuencia. Del mismo modo, se buscará el mayor máximo de las posiciones de los fragmentos. Seguidamente se deberá obtener la secuencia completa de aminoácidos de la secuencia en cuestión. Esto se realizará en base a otro recurso web, esta vez perteneciente a Uniprot. Para este servicio el ingreso del acceso uniprot o el identificador uniprot es indiferente.

Para poder acceder a la información se ingresará el valor uniprot, ya sea el identificador o el acceso, reemplazando la variable *{{uniprotAccession/identifier}}* en el siguiente enlace:

https://www.uniprot.org/uniprot/{{uniprotAccession/identifier}}.fasta

De esa manera, se obtiene la secuencia completa, la cual será recortada en base a la posición mínima y máxima de las unidades de repetición identificadas. Cada recorte se alojará en un archivo con formato fasta, y su nombre contendrá el prefijo “nr” acorde a lo que en este proyecto se denomina como una nueva secuencia representativa, “new representative sequence”. Este recorte es importante, además de lo mencionado anteriormente, puesto que en las pruebas del algoritmo original se percibió que el tiempo de ejecución de la predicción es directamente proporcional a la longitud de la secuencia en procesamiento. En ese sentido, se identificó que una longitud de una secuencia superior a los seiscientos aminoácidos demorará más de cuatro días en completar su predicción. Esto último, incluso teniendo en cuenta las capacidades del primer entorno en el que fueron evaluados los algoritmos identificados, las cuales superaban a las características promedio. Con ello, se determinó considerar ese número como una restricción adicional al filtrado de datos. Es decir, aquella subsecuencia representativa que supere esa longitud será separada del proceso de predicción adaptado a las proteínas repetidas.

Llegado a este punto se cuenta con varios archivos fasta, pertenecientes a dos enfoques planteados para la adaptación del algoritmo. Estos archivos serán ingresados a un script de generación de mapas, el cual a partir del fasta genera mapas de contacto en un archivo con extensión .map y diversos datos de covarianzas en un archivo con extensión .21c. Como se deduce, de cada archivo fasta se generarán los dos archivos mencionados.

El algoritmo originalmente seleccionado utiliza un grupo de tres archivos de diversos tipos para iniciar una predicción. Estos archivos son, precisamente, los generados en los pasos anteriores. Un grupo para predicción estará conformado por un archivo fasta, un archivo 21c y un archivo map, teniendo en cuenta que estos dos últimos deben haber sido generados en base al fasta en cuestión.

Llegado a este punto se procederá a iniciar la predicción de límites de distancia interatómica, ángulos de torsión y enlaces de hidrógeno de la cadena principal a través

de redes neuronales profundas (Greener et al., 2019). Esta sección corresponde a lo obtenido de parte del algoritmo seleccionado inicialmente.

Luego de la predicción de los datos mencionados, continuará la resolución de la estructura a través del uso del Sistema de Cristalografía y NMR, CNS, por sus siglas en inglés (Brunger, 2007b; Brunger et al., 1998). Cabe mencionar que para contar con acceso al recurso se deberá ingresar un formulario que valida el uso académico de la herramienta.

Por cada iteración se generará una cantidad configurables de estructuras que van mejorando su precisión en cada iteración. Para poder realizarlo se incurre a la clusterización de las estructuras predichas, con lo cual se obtiene a la mejor. Para el presente proyecto, se han seguido las configuraciones por defecto del algoritmo inicial, las cuales comprenden a tres iteraciones y un total de cincuenta estructuras predichas por cada una de ellas. Una vez que se acabaron las iteraciones se realizará una última clusterización para poder obtener las estructuras finales con mejor precisión. La cantidad de estructuras predichas por cada secuencia es indistinta, con lo cual una predicción en base a una secuencia podría generar tres estructuras como una sola. No obstante, un punto a recalcar es que todas las estructuras predichas para cada secuencia tienen un número al final del nombre del archivo que las contiene. Este número inicia en uno y termina en la cantidad de estructuras predichas, siendo la estructura con valor uno la que tiene mayor probabilidad de ser la mejor.

Llegado a este punto también cabe precisar que hasta este momento toda ejecución del algoritmo adaptado se ha realizado a través del terminal de comandos de Linux. No obstante, se fue añadiendo la posibilidad de ingresar datos de entrada distintos al código PFAM de una familia de proteína repetida. Algunos de los que fueron agregados son: una secuencia directa, las secuencias pertenecientes a una familia, pero en formato Stockholm o las secuencias pertenecientes a una familia de proteínas, pero directamente con el archivo. Todas esas modificaciones parten de un inicio un tanto distinto dentro del flujo de procesamiento, pero se encuentren en un punto del mismo.

En la Figura 34, se muestra un ejemplo de los archivos que son procesados a lo largo del flujo del algoritmo adaptado y la evolución de sus tipos de datos, desde los archivos fasta, pasando por los archivos intermedios y terminando con los archivos PDB.

Figura 34. *Diferenciación de los archivos involucrados en el proceso de la predicción de estructuras terciarias de proteínas repetidas. El algoritmo de predicción inicia con archivos de secuencias como los de la primera columna. La segunda columna corresponde a archivos intermedios generados. La tercera columna contiene al archivo final de la predicción en formato PDB, el cual contiene el detalle de las coordenadas espaciales de cada átomo de cada aminoácido. (Elaboración propia).*

La última sección del flujo corresponde a una evaluación del performance de la predicción, dependiendo del caso en el que se encuentre. Si fuese una secuencia completa la que se ha predicho, se utilizará la herramienta RepeatDBLite para poder identificar las unidades de repetición o proteínas repetidas dentro de la secuencia. Esto último con el fin de no divagar en torno al objetivo y el enfoque de la herramienta: las proteínas repetidas. Por otro lado, en caso se trate de un fragmento, se incurrirá a la obtención de una estructura predicha de alguna secuencia que contenga unidades de repetición de la misma familia que la del fragmento que se ha predicho.

El código de la estructura se obtiene luego de la ejecución de uno de los procesos descritos previamente. Se utilizará el siguiente enlace para obtener las estructuras requeridas, donde se requerirá reemplazar la variable *{{ID_PDB}}* por el código PDB de la estructura requerida. Solo se admiten códigos de cuatro caracteres alfanuméricos:

https://files.rcsb.org/download/{{ID_PBD}}.pdb

Con esta estructura auxiliar se procederá a ejecutar la librería TM-align para poder evaluar el alineamiento estructural de la estructura predicha y la auxiliar.

Llegado a este punto, se considera que el 100% de las funcionalidades del código fuente han sido cubiertos y explicados dentro del presente documento.

3. Diagrama del algoritmo adaptado

En esta sección se presentará al diagrama de flujo de actividades del algoritmo adaptado. Este diagrama fue elaborado con la herramienta Diagrams.net y en base a la notación BPMN. Esta visualización contempla en su totalidad la descripción realizada en el apartado anterior de este documento.

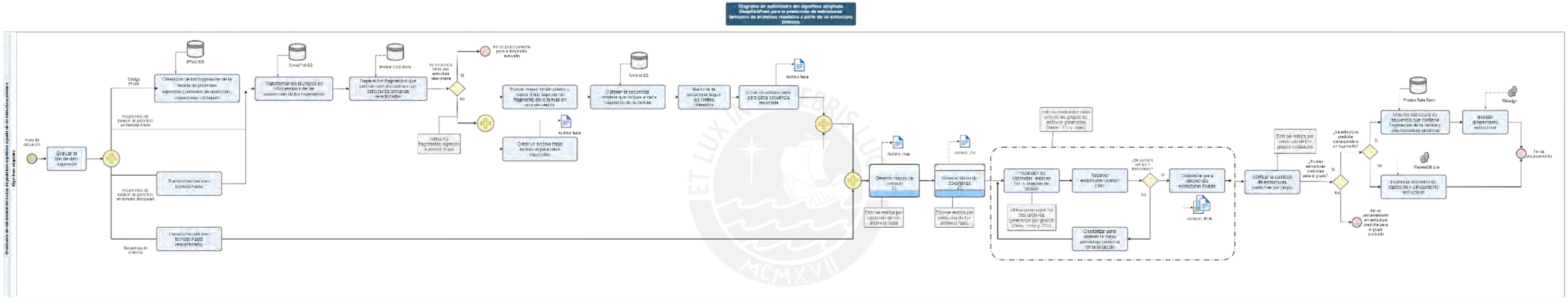

Figura 35. Diagrama de flujo de actividades del algoritmo adaptado DeepReSPred. (Elaboración propia).

4. Revisión de los scripts

Este apartado presentará la organización de los archivos ejecutables responsables de cada actividad dentro del flujo del algoritmo adaptado. Se debe tener en cuenta que algunos de ellos se obtuvieron del propio repositorio del algoritmo seleccionado. No obstante, se hicieron algunas pequeñas correcciones en tanto en la primera evaluación luego de la instalación no funcionaba ninguno de los dos algoritmos.

Cabe mencionar también que se incurrieron la programación de estos scripts en distintos lenguajes de programación de acuerdo a la afinidad de los otros archivos con los que se iba a interactuar o, incluso, el nivel de conocimiento técnico de los lenguajes.

De acuerdo a las actividades descritas tanto de forma textual como visual, a través del diagrama, a realizarse para poder llevar a cabo la predicción de estructuras terciarias: se crearon dos scripts y se modificó solo uno ya existente. Los archivos creados fueron los que están denominados en el repositorio como run_repeat_prediction.sh y MappingFasta.py, el primero de ellos escrito en lenguaje Shell y el segundo en Python. Mientras que solo se realizó la modificación de un archivo del algoritmo original denominado como run_dmpfold.sh. Las modificaciones descritas a continuación fueron realizadas en pro de la dinámica del flujo de actividades totales.

Respecto al archivo modificado, se resalta el hecho de haber ingresado líneas de comandos para poder calcular el tiempo de ejecución de la predicción de cada una de las secuencias ingresadas. Este indicador a gran escala podría ayudar a definir ciertos patrones en el flujo de la información o la estimación del tiempo de predicción. Asimismo, se incluyeron líneas para la creación de ficheros ante errores de ejecución. Estos archivos se generarán en un directorio específico para poder realizar el seguimiento de todas las incidencias ocurridas en el procesamiento. En tercer lugar, se agregaron líneas de eliminación de información que no cumplan con un formato específico. Esto ayuda en gran medida debido a que muchas de las familias cuentan con miles de secuencias. Dado que en la predicción se crearán archivos intermedios por cada uno de ellos, teniendo en cuenta que no solo se crean los archivos fasta, map y 21c, el almacenamiento podría llegar a coparse, mas si se van eliminando los archivos innecesarios en paralelo a la ejecución, ese riesgo se reducirá. Por último, se incluyeron algunas correcciones pequeñas de algoritmo original, sin las cuales el algoritmo no funcionará correctamente.

Debido a las modificaciones incluidas el fichero run_dmpfold.sh perteneciente al algoritmo original, en cada instalación del algoritmo adaptado se procuró simplemente reemplazar el fichero con el mismo nombre del directorio del algoritmo original por el archivo con el mismo nombre ubicado en el repositorio del algoritmo adaptado.

Respecto a los archivos creados, el primero de ellos denominado como run_repeat_prediction.sh es el que se encarga de realizar las invocaciones a todas las herramientas necesarias para cumplir cierta sección del proceso de predicción.

En ese mismo sentido, se encarga de diferenciar los caminos que seguirán cada tipo de dato ingresado a la solicitud de predicción.

Por otro lado, el archivo MappingFasta.py es el encargado del consumo de todos los recursos descritos en el primer apartado de este documento, tales como el consumo de api's o de otras herramientas web. A partir de eso, el script se encargará de crear los archivos fasta por cada tipo de secuencia "nr" o "is".

5. Apéndice

En esta sección se presenta el acceso al código fuente del algoritmo adaptado y la descripción general del contenido del repositorio que lo contiene.

5.1 Código fuente del algoritmo adaptado

La gestión del código fuente se ha llevado a cabo mediante la herramienta GitHub. Así, para poder acceder a este recurso se deberá ingresar a un repositorio. Este repositorio nombrado como "DeepReSPred-back" tiene habilitado el acceso público, por lo cual para poder obtener las fuentes del algoritmo adaptado bastará con el ingreso al siguiente enlace:

Repositorio back-end de la herramienta DeepReSPred

Este repositorio cuenta con una carpeta principal que contiene dos carpetas secundarias. En el directorio "programsAuxiliar" se encuentran algunas herramientas necesarias para la implementación del algoritmo adaptado, teniendo en cuenta que para

la realizar la instalación de este algoritmo será necesario instalar primero el algoritmo original siguiendo los pasos de instalación descritos en el Anexo E.

Cabe mencionar que el uso de las herramientas contenidas en el repositorio del algoritmo adaptado deberá tener una finalidad enteramente académica.

El directorio “back_project” contiene otro directorio denominado como “deepReSPred”, en el cual se encuentran primordialmente los archivos creados MappingFasta.py, SP_collection.txt, run_repeat_prediction.sh y el archivo modificado del algoritmo original, run_dmpfold.sh.

Cabe recordar que el archivo SP_collection es el fichero que contiene la relación entre los identificadores Uniprot y los accesos Uniprot necesarios para poder utilizar el api del Banco de Datos de proteínas.

Anexo J: Reporte de pruebas del algoritmo adaptado

En este anexo se presentará el reporte de pruebas del algoritmo adaptado para la predicción de estructuras terciarias de proteínas repetidas a partir de sus estructuras primarias. Este reporte es el segundo medio de verificación del tercer resultado alcanzado del primer objetivo específico[45] de este proyecto de fin de carrera. Contiene la descripción de las pruebas realizadas, la explicación a detalle del procedimiento de captura de datos para la evaluación, así como el procedimiento de ejecución de pruebas. En adición, dado que este documento corresponde a la última sección referente al primer objetivo, se adjunta como un apéndice al documento, el acta de validación del reporte de pruebas, y, por tanto, del algoritmo adaptado en base al contexto de las proteínas repetidas.

1. Introducción

Las modificaciones que se han incluido al algoritmo seleccionado en resultados previos al presente dieron pie a la obtención de un algoritmo adaptado según los requerimientos de las personas interesadas en la predicción de estructuras terciarias de proteínas, en específico de proteínas repetidas.

Una vez que se cuenta con el algoritmo adaptado es indispensable evaluar su nivel de confiabilidad. Es por ello que en el presente documento se especifica, en primer lugar, la descripción en general de las pruebas a realizar, seguido de la explicación del procedimiento para la obtención de datos requeridos de proteínas repetidas, y por último se encuentra la descripción del proceso de ejecución de esas pruebas. Por último, se adjunta el acta de validación del presente documento a través de juicio experto.

2. Descripción de las pruebas a realizar

El criterio más importante que seguirán las evaluaciones propuestas en esta sección del documento corresponderá al alineamiento estructural entre las estructuras predichas y alguna estructura relacionada.

Este punto tiene en consideración las particularidades de los grupos de familias de proteínas repetidas en las que se ha hecho hincapié a lo largo de todo el proyecto. Es decir, a la presencia de unidades de repetición en su estructura, las cuales pertenecen a una familia de proteínas por su similaridad en características; así como el alto grado de conservación en su estructura (Deng et al., 2018; Hirsh et al., 2016; Parmeggiani &

[45] O1. Adaptar un algoritmo capaz de explotar los datos existentes de estructuras primarias de proteínas en general para predecir estructuras terciarias de proteínas repetidas.

Huang, 2017). De esa manera, se podrá reconocer que el resultado obtenido en la comparación entre una estructura predicha por el algoritmo adaptado y una estructura previamente almacenada en el Banco de Datos de Proteínas representa un indicador lo suficientemente significativo como para evaluar el performance del algoritmo adaptado. Se ha determinado que la evaluación va a tomar un enfoque enteramente cuantitativo, por la cual se podrá decidir definitivamente si los resultados obtenidos por la predicción de estructuras son lo suficientemente buenos como para considerarse significativas para las personas interesadas.

Se utilizarán dos herramientas para poder llevar a cabo la evaluación. El primero de ellos corresponde a la herramienta de visualización PyMol (PyMOL, 2021) a la cual se le entregará el archivo en formato PDB de la estructura terciaria de la proteína predicha y el archivo PDB de alguna estructura relacionada. Se entiende que la relación entre dos estructuras se determina en base a las secuencias relacionadas a esas estructuras. Cada secuencia de proteína puede contener fragmentos o, en el caso de las proteínas repetidas, unidades de repetición pertenecientes individualmente a una familia de proteínas en específico. Aunque, también podría darse el caso donde una secuencia completa solo posea unidades de repetición y que todas pertenezcan a la misma familia. El análisis se realiza con las estructuras dado que es bien conocido que las estructuras de las proteínas repetidas tienen un factor de degeneración menor que el de su secuencia (Deng et al., 2018; Hirsh et al., 2016; Parmeggiani & Huang, 2017). Esto significa que, si un aminoácido varía debido a condiciones externas, es muy probable que otro aminoácido afín a este también varie, a modo de reacción al primer cambio. Así, la estructura tridimensional de la proteína se mantendrá casi intacta a pesar de las modificaciones. Esto no sucederá si se evalúa a la secuencia, donde, como se menciona en el caso de ejemplo, sí variaron dos aminoácidos.

Para poder utilizar la herramienta PyMol, en primer lugar, es necesario contar con un correo con un dominio de educación y, en segundo lugar, se requiere completar una solicitud de uso donde se afirma que la herramienta será utilizada para estudios sin ánimos de lucros. Cabe mencionar que, si bien se utilizará la herramienta dentro de la evaluación, su uso no es indispensable, con lo cual se le otorgará una función de verificación de la evaluación que se explicará a continuación.

La segunda herramienta que se utilizará para poder evaluar el performance del algoritmo adaptado corresponde a TM-align (Zhang & Skolnick, 2005). Esta es una herramienta que permite identificar los niveles de alineamiento entre dos proteínas con estructura

tridimensional. Retorna valores entre 0 y 1, donde 1 corresponde a un alineamiento perfecto, el cual podría obtenerse al alinear dos estructuras iguales. Asimismo, es considerado que un valor superior a 0.5 significa un alto grado de similaridad estructural, donde de forma general se observa un plegado bastante semejante.

El uso de esta herramienta es la estrategia principal de este proceso de pruebas al algoritmo adaptado. En ese sentido, se deberán seleccionar una serie de datos representativos para poder realizar esta evaluación. El procedimiento para capturar los datos a ingresar en la evaluación será descrito en la sección a continuación.

3. Procedimiento para la obtención de datos requeridos de proteínas repetidas

En este apartado se detallará el paso a paso a seguir para poder obtener los datos que serán explotados por el algoritmo adaptado: secuencias de proteínas repetidas, familias de proteínas repetidas y diversas estructuras terciarias de secuencias que contienen proteínas repetidas. Ha sido posible capturar toda esa información gracias a la política de datos abiertos de distintas bases de datos tales como RepeatsDB, Uniprot, Pfam y RCSB Protein Data Bank.

Como primera instancia, se deberá obtener la información general de un grupo de familias de proteínas repetidas en específico. En este caso, se centrarán los esfuerzos en las proteínas repetidas de clase 3 y topología 3. Así, se utilizará una herramienta de RepeatsDB, el cual permite realizar consultas de acuerdo a diversos parámetros. Para obtener resultados del recurso solo es necesario construir una cadena de búsqueda como la siguiente e ingresarlo en un navegador web:

https://repeatsdb.bio.unipd.it/api/search?query=class:3%7Cclass_topology:3

La dirección anterior contiene las variables de clase y topología, y genera un archivo json con todas las secuencias de RepeatsDB que pertenecen a la clase y topología especificada. La respuesta obtenida a partir de la consulta también detalla el identificador de la estructura almacenada en Protein Data Bank y el identificador de la familia de proteínas a las que pertenece y la cual está alojada en Pfam. Se recomienda realizar el procedimiento en un entorno con memoria disponible dado que los archivos descargados pueden tener un tamaño considerable.

En el caso de la cadena de búsqueda ingresada, se obtuvo un fichero con setecientas tres secuencias de proteínas de las cuales se identificaron fragmentos pertenecientes a

trescientos cuarenta y cuatro familias de proteínas repetidas. De este grupo de familias identificadas se escogieron diez, teniendo en consideración que cada una de ellas contiene un mínimo de trescientos setenta y tres, y un máximo de ciento treinta y tres mil secuencias, o como se están denominando en este proyecto, fragmentos.

En la Tabla 30, se muestra el código de acceso PFAM, la descripción, el tipo, el número de secuencias por cada tipo de generación y la longitud promedio de las secuencias pertenecientes a las familias de proteínas repetidas seleccionadas para la evaluación del algoritmo adaptado.

Catálogo de familias de proteínas seleccionadas para pruebas					
Código PFAM	Descripción	Tipo	Número de secuencias		Longitud promedio de cada secuencia
			Semilla	Completo	
PF00023	Ankyrin repeat	Repeat	1062	23676	33.60
PF00514	Armadillo/beta-catenin-like repeat	Repeat	197	85142	40.70
PF16186	Atypical Arm repeat	Repeat	139	5093	52.50
PF18770	Armadillo tether-repeat of vescicular transport factor	Repeat	29	700	60.30
PF18773	Importin 13 repeat	Repeat	13	679	39.30
PF08569	Mo25-like	Repeat	117	2902	286.90
PF01239	Protein prenyltransferase alpha subunit repeat	Repeat	619	16557	31.70
PF00806	Pumilio-family RNA binding repeat	Repeat	1115	57203	33.10
PF08238	Sel1 repeat	Repeat	167	133325	35.30
PF03377	TAL effector repeat	Repeat	41	373	33.50

Tabla 30. *Catálogo de familias de proteínas seleccionadas para realizar las pruebas del algoritmo adaptado.*

Para proceder con la comparación de las estructuras predichas por el algoritmo adaptado se deberán recolectar una serie de estructuras que ya han sido previamente predichas con otras herramientas, y, por tanto, estén alojadas en la base de datos RSCB PDB. Para cada familia de proteínas se identificará una serie de cinco estructuras tridimensionales alojadas en el Banco de Datos de Proteínas. Estas estructuras están

relacionadas a una o más familias seleccionadas, dado que se obtienen a partir de una secuencia de proteína, la cual puede contener diversos fragmentos de proteínas repetidas, o unidades de repetición, pertenecientes a una misma o a distintas familias. En este caso, la información de las estructuras se ha obtenido del mismo archivo json. Así, en la Tabla 31, se muestra la relación entre las familias escogidas y las estructuras terciarias que han sido seleccionadas por cada una de ellas para evaluar a las estructuras que fueron predichas por el algoritmo adaptado.

Catálogo de estructuras PDB escogidas para las pruebas						
Código PFAM	Descripción	Código PDB 1	Código PDB 2	Código PDB 3	Código PDB 4	Código PDB 5
PF00023	Ankyrin repeat	2yqf	3f59	3kbt	3kbu	3ud1
PF00514	Armadillo/beta-catenin-like repeat	1xm9	2jdq	3tj3	4b18	2yns
PF16186	Atypical Arm repeat	4tnm	4uad	1bk5	1bk6	1ee4
PF18770	Armadillo tether-repeat of vescicular transport factor	2w3c	3gq2	3grl	-	-
PF18773	Importin 13 repeat	2x19	2xwu	3zjy	2x1g	3zkv
PF08569	Mo25-like	1upk	1upl	2wtk	3gni	4fzd
PF01239	Protein prenyltransferase alpha subunit repeat	4ydo	3q75	3q78	3q79	3q7a
PF00806	Pumilio-family RNA binding repeat	3q0q	3q0r	3q0s	3v71	5bz5
PF08238	Sel1 repeat	1ouv	3rjv	4bwr	5b26	-
PF03377	TAL effector repeat	2ypf	3ugm	4cj9	4gg4	4hpz

Tabla 31. *Catálogo de estructuras PDB escogidas para las pruebas del algoritmo adaptado.*

En general, se identificaron un total de cuarenta y siete estructuras de proteínas con las cuales se realizará la comparación. No obstante, aún se requiere capturar la información de esas estructuras. Esto se realizará a través del uso de la siguiente API del Banco de Datos de Proteínas:

https://files.rcsb.org/download/{{CódigoPDB}}.pdb

Se deberá modificar la variable *{{CódigoPDB}}* por el código de cada una de las estructuras escogidas.

4. Ejecución de pruebas del algoritmo adaptado

Una vez que el algoritmo se encuentra listo para realizar la predicción de las estructuras terciarias de las proteínas repetidas a partir de sus estructuras primarias, se procederá con la ejecución de las pruebas. Asimismo, en este punto, las estructuras necesarias para realizar el 'benchmarking' con las estructuras a generar también han sido identificadas y su información ya ha sido capturada.

Corresponde, entonces, ejecutar el algoritmo adaptado para cada una de las familias escogidas y seleccionar la mejor predicción de al menos tres fragmentos pertenecientes a esa familia. Cabe recordar que el algoritmo ordena las estructuras predichas asignándole un sufijo "_1" a la predicción con mejor valoración. Estas serán las estructuras que se escogerán para la evaluación. Asimismo, la elección de los fragmentos de los cuales se capturará la predicción será aleatoria.

La Figura 36 muestra el resultado del alineamiento estructural obtenido con TM-align entre la estructura 3zkv.pdb y la mejor estructura predicha para la unidad de repetición A0A091PG47_LEPDC de la familia Importin 13 Repeat PF18773. Este fragmento consta de 40 aminoácidos de longitud. El valor del resultado es 0.63 y es mayor al 0.5 esperado.

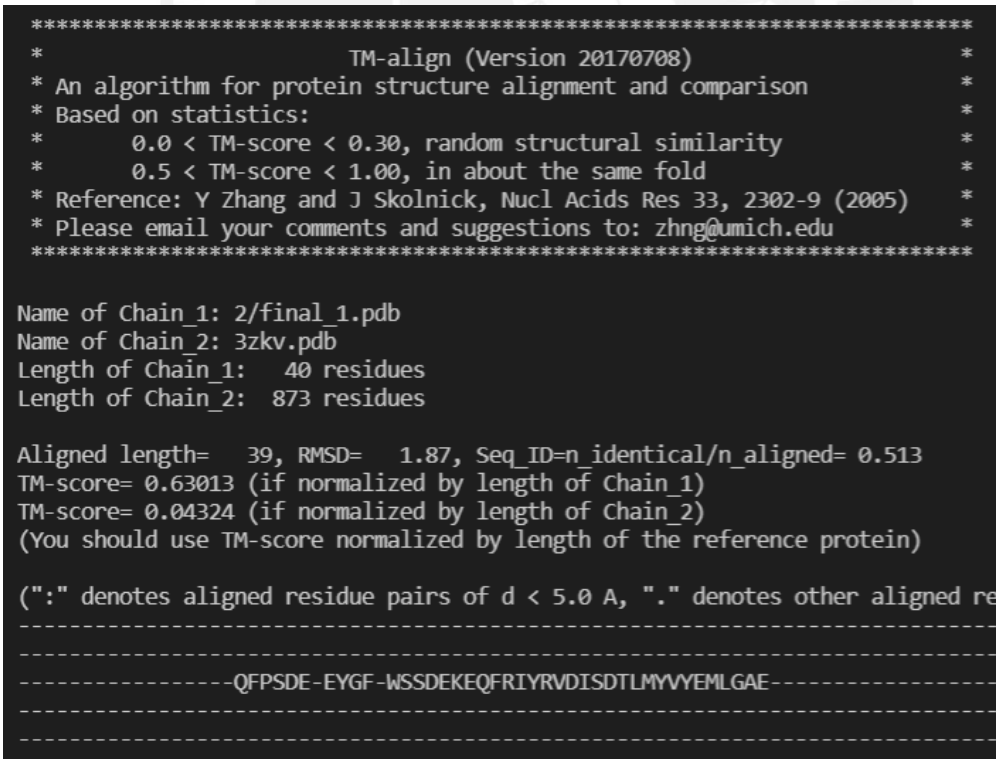

Figura 36. *Ejemplo de resultado de alineamiento estructural de TM-align entre la estructura 3zkv.pdb y la mejor estructura predicha para la unidad de repetición A0A091PG47_LEPDC de la familia PF18773. (Elaboración propia).*

El valor de la evaluación del alineamiento obtenido es mayor al 0.5, por lo que se considera que la predicción de esa unidad de repetición fue exitosa.

Como se mencionó, se utilizará la herramienta de visualización PyMol para poder analizar de forma visual el alineamiento de las dos estructuras. El resultado se puede observar en la Figura 37, donde la estructura obtenida del Banco de Datos de Proteínas con el identificador 3ZKV está en color verde y la estructura predicha por el algoritmo adaptado se encuentra de color celeste y pertenece a la familia de proteínas repetidas Importin 13 repeat, identificada con el código PFAM PF18773.

Se puede observar que la unidad de repetición predicha se alinea casi perfectamente a un fragmento de la estructura de 3ZKV. Con ello, se puede reconocer al fragmento de la secuencia de la proteína con estructura 3ZKV que pertenece a la familia Importin 13 repeat.

Para visualizar con más detalle el alineamiento de los dos fragmentos de la evaluación, se generó otra imagen con un enfoque más centrado en lo mencionado. Esta nueva visualización se encuentra en la Figura 38.

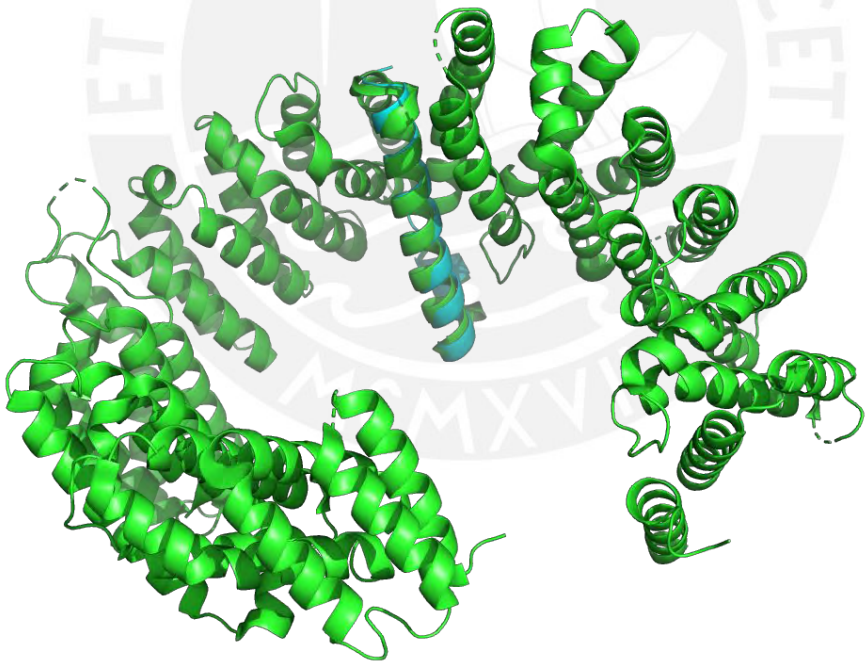

Figura 37. *Visualización del alineamiento estructura entre la estructura 3zkv.pdb en color verde y la mejor estructura predicha para la unidad de repetición A0A091PG47_LEPDC de la familia PF18773 en color celeste. (Elaboración propia en PyMol).*

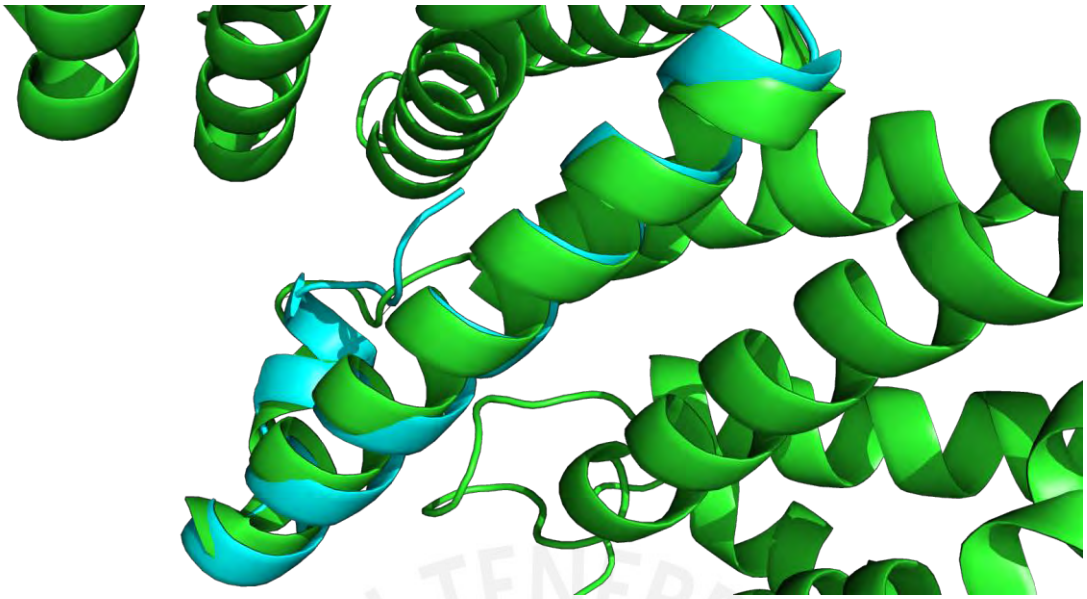

Figura 38. Visualización enfocada del alineamiento estructura entre la estructura 3zkv.pdb en color verde y la mejor estructura predicha para la unidad de repetición A0A091PG47_LEPDC de la familia PF18773 en color celeste. (Elaboración propia en PyMol).

En la Figura 38 se observa a detalle el alineamiento entre las dos estructuras, recordando que la que se encuentra en color celeste es la unidad de repetición obtenida a partir del preprocesamiento de la familia de proteínas repetidas PF18873.

Cabe mencionar que el algoritmo se ejecutó en promedio en un total de 6,9 minutos por fragmento predicho. En el caso de la familia Importin 13 repeat, esta compuesta por seiscientas setenta y nueve secuencias con un promedio de cuarenta aminoácidos en su composición.

Se ha realizado cada una de las evaluaciones propuestas, en las que cada estructura predicha se ha alineado con las cinco estructuras seleccionadas por familia. Se ha obtenido que el menor valor promedio obtenido por TM-align es 0.43, mientras que el mayor valor promedio obtenido es 0.53. Con ello, se verifica que es posible obtener predicciones de proteínas repetidas muy cercanas a lo requerido.

La evaluación del alineamiento estructual entre de los fragmentos predichos y alguna estructura PDB de la base de datos ya es una actividad considerada en el flujo de proceso que sigue el algoritmo adaptado. No obstante, ante el anális de los resultados de esta evaluación se ha decidido que la herramienta que utilice el algoritmo adaptado deberá validar que se supere el valor esperado en el alineamiento. Con ello, se asegurará que las predicciones que ofrezca sean las mejores posibles.

5. Consideraciones adicionales

Llegado a este punto cabe mencionar algunas consideraciones importantes que se han tomado en cuenta a lo largo del desarrollo este objetivo específico.

Si bien el presente documento corresponde a un reporte específico para la definición de pruebas y su ejecución, es necesario entender que las pruebas al algoritmo se llevaron a cabo de forma constante a lo largo de todo el proyecto. Gracias a ello se fueron tomando en consideración algunas restricciones del algoritmo escogido como las que serán descritas a continuación:

Tras realizar diversas ejecuciones en el algoritmo, tanto el original como el adaptado, se obtuvo que el tiempo de ejecución del algoritmo es directamente proporcional a la longitud de la secuencia ingresada. Se observó que es a partir los quinientos o seiscientos aminoácidos de longitud donde el algoritmo comienza a demorarse más de lo previsto, superando los tres días de ejecución. No se ha tenido capturado el tiempo total que se demora la predicción, pero considerando que el algoritmo tiene tres iteraciones y el mínimo de tres días consumidos corresponde solo a la primera iteración, se decidió establecerla como una restricción. Se debe tener en cuenta que, en primeras instancias, la instalación de los algoritmos se realizó en un ambiente robusto para predicciones, es decir, con recursos computacionales superiores al del promedio. Fue en ese contexto en el cual se percibió ese comportamiento.

Asimismo, cabe mencionar que la sección de generación de los archivos intermedios de mapas de contacto y datos de covarianza requiere de la instalación en el mismo entorno de ejecución de una serie de bases de datos. Al inicio del proyecto se instalaron dos bases de datos recomendados: UniRef30_2020_06_hhsuite y pfamA_31.0. Estos recursos al estar comprimidos tienen un tamaño de 45G y 1,5G, respectivamente; no obstante, al descomprimirlos ocupan un espacio alrededor a 180G y 5G, respectivamente.

Como se mencionó, se contaba con una arquitectura robusta que podía alojar toda la información; sin embargo, el objetivo del presente proyecto también contempla el hecho de presentar una herramienta que pueda ser desplegado en entornos más locales, y pueda ser soportado por arquitecturas promedio. Muchas de las arquitecturas promedios de computadoras no soportan esa cantidad de información y más cuando se incluye dentro de un procesamiento, con lo cual implicaría, también, el uso de bastante memoria RAM para realizarlo.

Esto se corroboró en tanto se realizó todo el proceso de instalación del algoritmo y sus bases de datos en un total de tres computadoras, además del que se realizó inicialmente en un servidor robusto.

El primer caso requirió del uso de un disco externo para poder alojar la base de datos UniRef, el cual es el más completo, pero también el más pesado. No obstante, al ser un recurso externo conectado a la computadora, el tiempo de procesamiento se elevó considerablemente, llegando copar los servicios del recurso y obligando a reiniciarlo en cada ejecución.

Para el segundo caso, sí se llegó a resolver la instalación completa del algoritmo y las bases de datos, incluso descomprimidas, pero no se llegó a realizar una predicción exitosa ya que, como se mencionó, el uso de la base de datos UniRef no solo implica la necesidad de gran espacio en el disco duro, sino también una gran capacidad de procesamiento en memoria. Este segundo factor es el que no permitió la ejecución de la predicción.

En una tercera instalación, se contempló la utilización de una laptop con características levemente superiores al promedio. No obstante, el sistema operativo por defecto era Windows, por lo cual se decidió instalar un entorno WSL Ubuntu para poder realizar la instalación del algoritmo. En esta situación no se pudo corroborar si el almacenamiento de la base de datos y procesamiento eran soportados, ya que el propio entorno restringió la aplicación de algunas configuraciones necesarias para el proceso de instalación del algoritmo. Se validó a través de canales de discusión de los desarrolladores de WSL, que esas restricciones se debían a complicaciones que aún no se habían resuelto en un entorno no nativo como ese.

Llegado a este punto se reconoció que UniRef era una base de datos extensa que no podía ser incluida en el procesamiento final de la herramienta, no obstante, sus resultados sí. Este punto será discutido más adelante como parte de la integración de la interfaz y el algoritmo adaptado correspondiente a este capítulo.

Por otro lado, se decidió realizar pruebas del performance del algoritmo con uso de la base de datos alternativa: PFAM.

Las pruebas realizadas en base a un alineamiento entre las estructuras predichas utilizando las bases de datos UniRef y PFAM indicaron de forma positiva que los resultados son casi idénticos. Esto se podría explicar teniendo en cuenta que los datos de entrada del algoritmo también corresponden a datos que han sido obtenido de la

base de datos PFAM, lo cual indica que toda información ingresada será encontrada también en la base de datos localmente instalada.
La ejecución del algoritmo en base a esta segunda base de datos se verificó en los dos primeros ambientes en los que se había verificado el funcionamiento del algoritmo con la primera base de datos, además del entorno robusto mencionado inicialmente. Esto resulto en la ejecución exitosa del algoritmo en ambos ambientes. Para parte final de este proceso de pruebas adicional, se decidió instalar todo el algoritmo en un entorno Cloud. Esto también obtuvo resultados positivos. Con ello, se determinó que PFAM sería la base de datos que se utilizaría en el despliegue final del algoritmo.
En conclusión, se tomaron en cuenta dos restricciones del algoritmo en general. La primera de ellas corresponde a la longitud de la secuencia ingresada al algoritmo de predicción. A raíz de esto se decide considerar una longitud máxima de quinientos cincuenta aminoácidos como la admitida para proceder a la predicción. En caso una secuencia supere esa longitud y no corresponda reducirla en tanto sea una secuencia representativa[46] entonces será separada del grupo de secuencias encoladas para predecir. Asimismo, como parte de la segunda situación descrita en este apartado, se ha determinado que la base de datos que se utilizará como parte del algoritmo adaptado final será PFAM.

6. Apéndice

En esta sección se presenta el acta de validación del presente documento de requisitos funcionales y no funcionales de la herramienta propuesta.

6.1 Validación del documento por medio de juicio experto

La validación del documento de reporte de pruebas del algoritmo adaptado ha sido realizada por una experta en bioinformática. La realización de esta verificación implica la revisión completa del informe, el análisis de los procedimientos realizados tanto para la captura de información como para la ejecución de las pruebas, el análisis de las consideraciones adicionales y la entrega de observaciones en caso se requieran. A continuación, se observa el acta de validación recibida por la experta.

[46] La definición de esta denominación se explica en el Anexo I.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Acta de validación de documento

Título de tesis: Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

Tesista: Solange Estrella Palomino Chahua

Nombre del documento: Reporte de pruebas del algoritmo adaptado

Descripción del documento: Documento que contiene la descripción de las pruebas realizadas al algoritmo adaptado y los resultados de su ejecución

Mediante la presente acta, yo, la Dra. Layla Hirsh Martinez, dejo constancia de que, en mi calidad de experta en bioinformática, he revisado el documento descrito en los puntos anteriores perteneciente al proyecto de tesis en mención. En ese sentido, en la siguiente sección se especifica el veredicto y, en caso hubiesen, las observaciones correspondientes al documento.

Veredicto:

(X) Aprobado ( ) Requiere observación

Observaciones:

Lima, 12 de octubre de 2021

Firma

Anexo K: Reporte descriptivo de la interfaz desarrollada

En este anexo se presentará el reporte descriptivo de la interfaz. Este reporte corresponde al tercer medio de verificación del último resultado alcanzado en el segundo objetivo específico[47] de este proyecto de fin de carrera. Su contenido abarca la descripción detallada de cada una de las secciones de la interfaz desarrollada.

1. Introducción

Acorde al catálogo de requisitos del Anexo G perteneciente al primer resultado esperado del objetivo en el también que se enfoca este reporte, se deberán desarrollar una interfaz web mediante la cual los usuarios interesados puedan registrar una solicitud de predicción y puedan obtener una serie de resultados. Este flujo de actividades fue descrito en el reporte de mockup de la herramienta del Anexo H y es el que se ha tomado en consideración para desarrollar las secciones que comprenden a la interfaz de la herramienta planteado. A continuación, se describirán cada una de las secciones que contituyen a la interfaz de la herramienta, indicando a detalle la funcionalidad que cubren.

Cabe mencionar que la interfaz completa, así como la interacción entre las pantallas que la conforman, fue desarrollada utilizando HTML, JavaScript, Css y Vue, este último como framework que da soporte a toda la lógica e interacción entre sus funcionalidades.

2. Pantalla principal

El prototipo de alta definición propuesto para la interfaz de la herramienta plasma a la pantalla principal tal y como se observa en la Figura 39 . Esta será la pantalla que cualquier usuario que ingrese al recurso web verá a primera mano. Incluye un formulario por el cual directamente se puede registra una solicitud de predicción y una barra de navegación en la zona superior mediante el cual se podrán acceder a las demás secciones de la plataforma.

Este formulario solo cuenta con un campo, el cual permite el ingreso de secuencias de aminoácidos de proteínas o el código PFAM de una familia de proteínas repetidas. Asímismo, permite la carga de las secuencias desde tres tipos de archivos: en formato fasta, en formato Stockholm y en formato de texto plano. En caso se ingresen estos archivos, su contenido será desplegado en el campo textual, el cual será inhabilitado

[47] O2. Diseñar e implementar la interfaz de la herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

para edición al culminarse la carga del archivo. En el mismo sentido el formulario también cuenta con dos accesos directos. Al presionar alguno de estos accesos directos la herramienta cargará datos por defecto en el campo textual. Uno de ellos cargará un código PFAM mientras que el otro cargará una secuencia de aminoácidos.

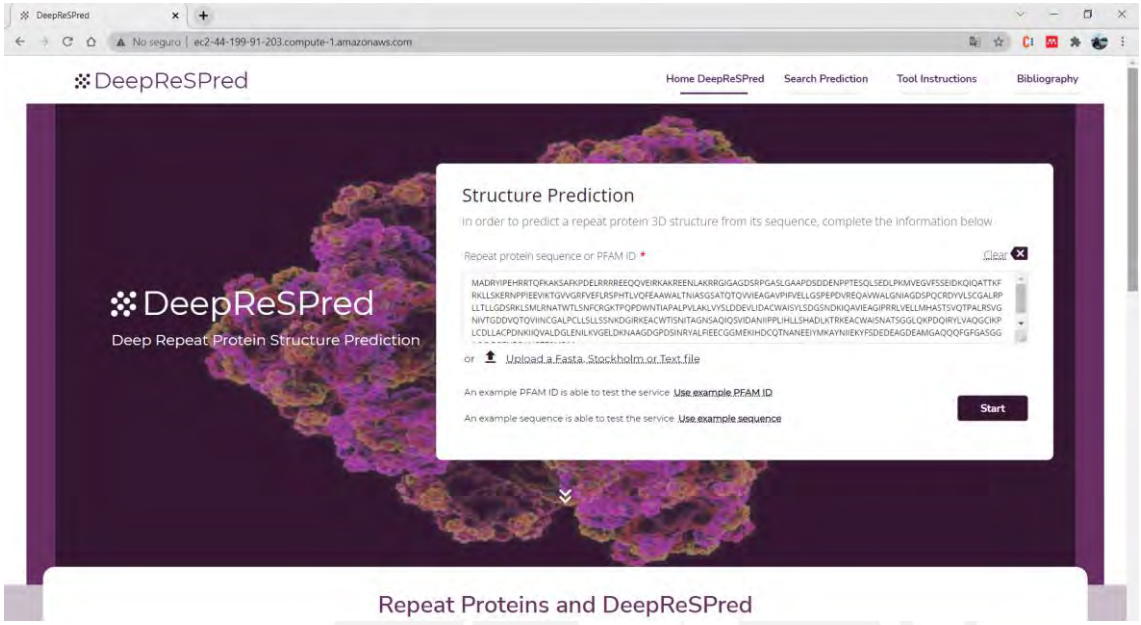

Figura 39. *Interfaz de la herramienta propuesta DeepReSPred - Pantalla principal. (Elaboración propia).*

El formulario permite la identificación de los campos obligatorios a través de un asterisco (*) de color rojo al costado de los campos de ese tipo. Asimismo, al detectar que se ingresa información en el campo textual se mostrará una opción de limpieza directa de este campo. Esta opción se encontrará en la zona superior derecha del campo textual y se ocultará en tanto este último se quede sin contenido.

El formulario también cuenta con validaciones, los cuales al detectar un dato inválido muestran un mensaje de error debajo del campo que los genera. Por ejemplo, en caso de detectar un código PFAM con más de 5 números mostrará un mensaje de error para que se verifique el dato ingresado, ya que el código PFAM siempre cumple con una estructura en el que se inicia con los caracteres “PF” y se continúa con máximo 5 números. Asimismo, en caso se trate del ingreso de una secuencia, si se detecta un caracter alfanumérico que no corresponda a la simbología de un aminoácido, la interfaz mostrará un mensaje de error debajo del campo textual.

Solo al ingresar datos, se habilitará el botón de envío de la solicitud, por el cual se mostrará un modal de resumen de la predicción a ser enviada.

Este modal cuenta con los mismos campos anteriores mas dos adicionales donde se deberán registrar un identificador de la predicción como un correo electrónico, tal y como se muestra en la Figura 40. Este último es un campo opcional.

Cabe mencionar que el identificador que el sistema generará un identificador de forma aleatoria y lo recomendará al usuario. No obstante, el usuario puede modificarlo según crea conveniente. El sistema validará que el código ingresado sea único y no corresponda a alguna solicitud de predicción previamente registrada. En caso no se cumpla, se mostrará un mensaje de error acorde a la situación y se mostrará un botón con icono de recarga junto a lado del campo. Al presionarlo se sobreescribirá el código recomendado por el sistema en el campo del identificador y se ocultará el botón.

En la zona inferior derecha de este modal se encuentra un botón denominado “Send Request” para poder enviar la solicitud.

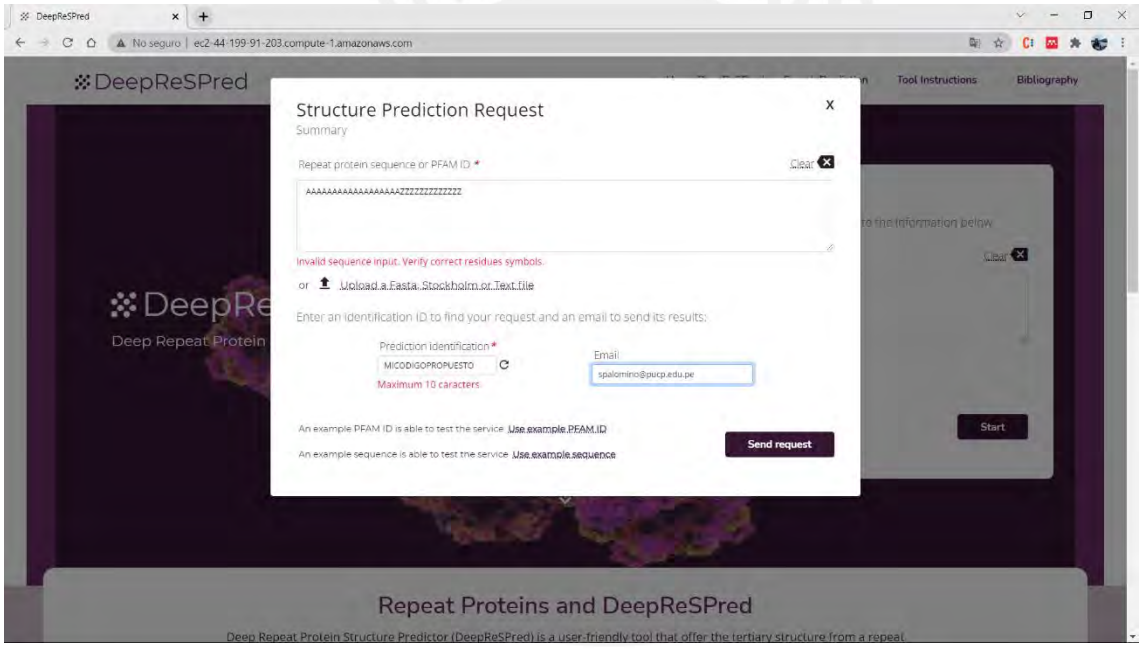

Figura 40. *Interfaz de la herramienta propuesta DeepReSPred - Modal de resumen de solicitud. (Elaboración propia).*

3. Mensajes de error o éxito

La interfaz de la herramienta permite mostrar mensajes de validación como los que fueron descritos en la sección anterior. Sin embargo, para los casos exitosos o fallidos en el registro de una predicción se deberá mostrar una notificación diferente. En este caso la plataforma mostrará un modal con una estructura como la que se observa en la Figura 41.

Estos modales contarán con título del mensaje ubicado en la zona superior izquierda, un icono con el cual el usuario podrá identificar rápidamente si el mensaje tiene connotación positiva o negativa, el mensaje principal, una zona para mensajes auxiliares y una zona de botones.

En el caso de que la solicitud de predicción se haya ingresado correctamente se mostrará el mensaje de registro exitoso conteniendo el identificador de la solicitud de predicción registrada y el correo electrónico. Este último solo se mostrará en caso se haya ingresado dentro de la solicitud. Por último, se mostrarán dos botones: el primero de ellos, ubicado en la zona de la izquierda redirigirá al usuario a la zona de consulta de la solicitud de predicción sobreescribiendo en el campo de búsqueda el identificador de la solicitud recientemente ingresada, mientras que el segundo botón solo cerrará el modal.

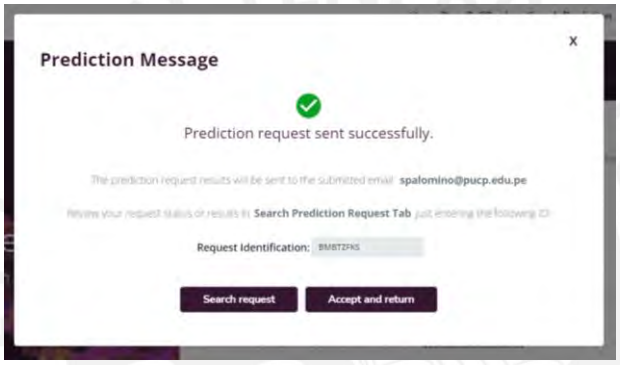

Figura 41. *Interfaz de la herramienta propuesta DeepReSPred - Modal de mensaje de éxito. (Elaboración propia).*

En caso de que la solicitud no se haya podido registrar de forma satisfactoria, se mostrará un mensaje descriptivo como el que se muestra en la Figura 42. En este modal solo se muestra el título del mensaje, el icono, el mensaje principal y un botón. Al tratarse de un mensaje de error, al presionar el botón solo se cerrará el modal.

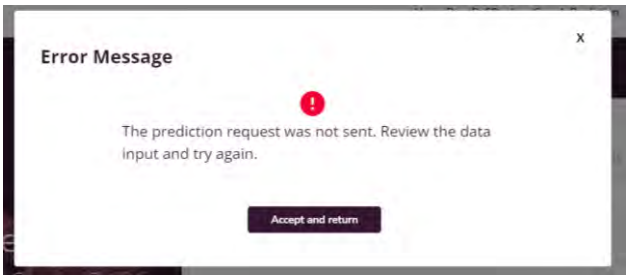

Figura 42. *Interfaz de la herramienta propuesta DeepReSPred - Modal de mensaje de error. (Elaboración propia).*

4. Sección de búsqueda de solicitudes de predicción

La plataforma permitirá la búsqueda de las solicitudes de predicción registradas. Esta actividad podrá realizarse a través de la sección de búsqueda de solicitudes que se ubica en la zona inferior de la pantalla principal. Se podrá acceder a esta sección ya sea a través de la barra de navegación ubicado en la zona inferior de la pantalla principal como al deslizar la barra de desplazamiento hacia el final de la pantalla. Tal como se observa en la Figura 43, en esta zona se mostrarán tres secciones distribuidas en tres pestañas: “Search prediction”, es la pestaña que se muestra por defecto; “Tool instructions”; y, “Bibliography”. Para acceder a alguna de ellas solo basta con seleccionarla.

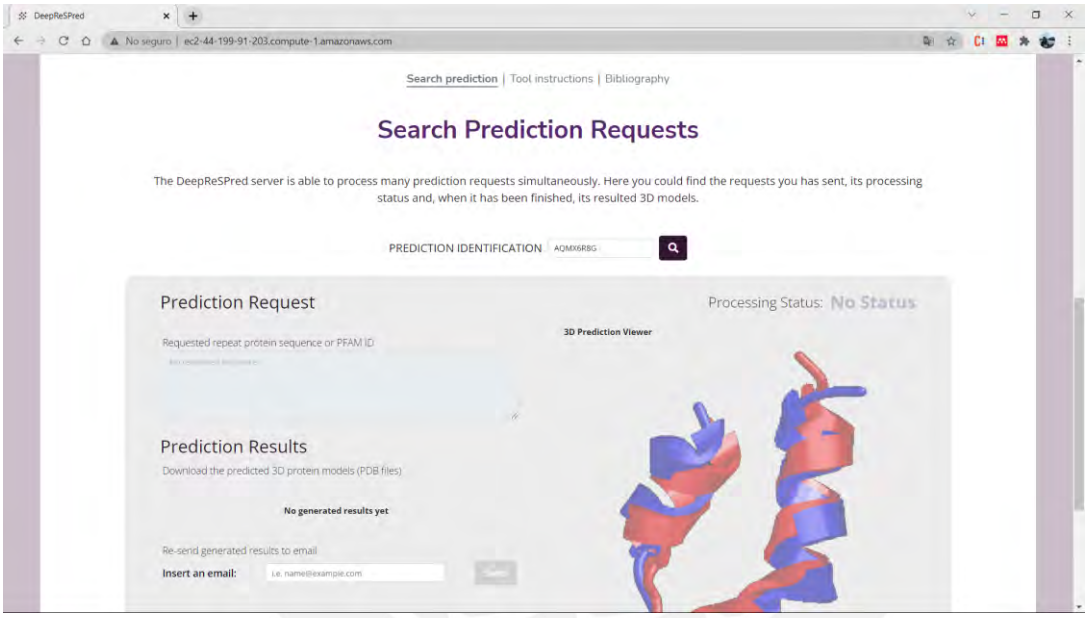

Figura 43. Interfaz de la herramienta propuesta DeepReSPred - Sección de búsqueda de solicitudes de predicción. (Elaboración propia).

En esta sección se mostrarán los resultados del procesamiento de la solicitud de predicción. Para iniciar una consulta se deberá ingresar el identificador de la solicitud registrada y presionar el botón con el icono de lupa. Seguidamente se mostrará el estado de solicitud en la zona superior derecha en el recuadro de los resultados. Los valores posibles del estado del procesamiento son los siguientes: Registrado, el estado inicial de toda solicitud que se ha registrado exitosamente; En proceso, significa que la el procesamiento de la predicción está en curso; Completo, cuando el procesamiento ha finalizado; Sin estado, cuando no se encuentra la solicitud de predicción solicitada o cuando ocurrió un error al traer la información; Error, significa que el procesamiento de la solicitud se ha realizado pero que se obtuvieron errores.

En la Figura 44, se muestra la sección de búsqueda una vez que se haya obtenido la información de la predicción consultada. Se podrá visualizar el dato de entrada registrado en la solicitud, los resultados obtenidos y su puntaje de TM-score, en caso corresponda. Por cada uno de los resultados obtenidos se contará con la opción de descarga y con la opción de visualización. A lado derecho se encuentra el panel de visualización a través del cual se podrá interactua con la estructura predicha.

Por último, se cuenta con un campo textual en el cual se podrá ingresar un correo electrónico para reenviar los resultados generados, ya sea se haya ingresado un correo electrónico al registrar la solicitud o no.

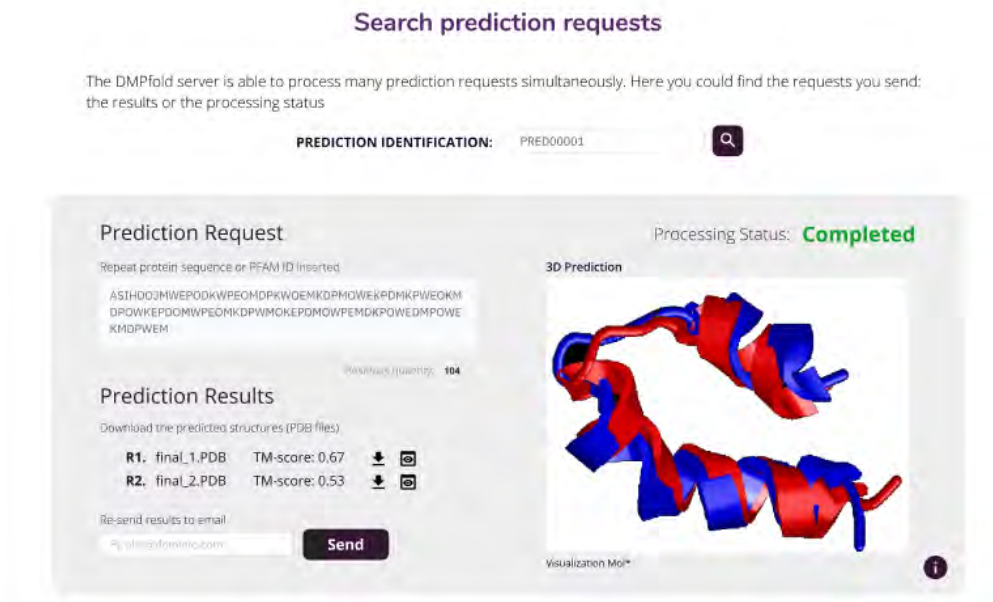

Figura 44. *Interfaz de la herramienta propuesta DeepReSPred - Sección de búsqueda de solicitudes de predicción con resultados. (Elaboración propia).*

5. Sección de instrucciones de uso

El servicio web contará con una sección de apoyo al usuario. Esta sección se denomina como “Tool instructions” y contendrá la explicación de los pasos a seguir para poder registrar una solicitud la predicción, consultarla, entender los estados de la solicitud y demás. En la Figura 45, se puede observar la estructura de esta sección. Cabe mencionar que se explica todo lo necesario en solo nueve puntos.

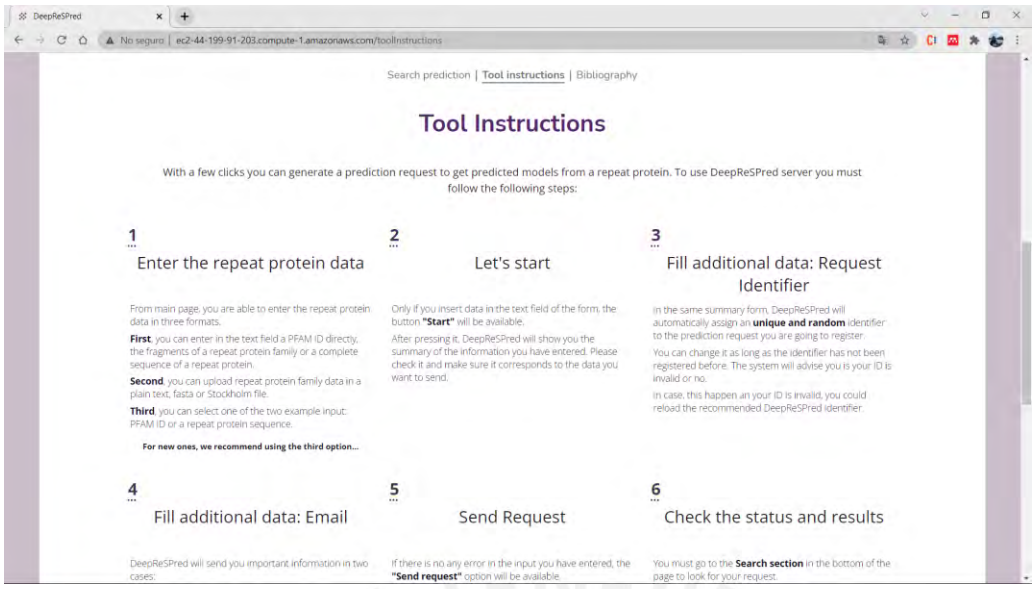

Figura 45. *Interfaz de la herramienta propuesta DeepReSPred - Sección de instrucciones de uso. (Elaboración propia).*

6. Sección de bibliografía del proyecto

La plataforma web contará con una sección de bibliografía. A través de esa sección se busca reconocer a las herramientas utilizadas para poder llevar a cabo el proyecto, en tanto muchas de ellas estuvieron a libre disposición a través de la web y otras requirieron de una licencia. No obstante, en este último caso, se registró una solicitud en sus páginas web oficiales, mediante los cuales los desarrolladores otorgaron una licencia educativa para el proyecto.

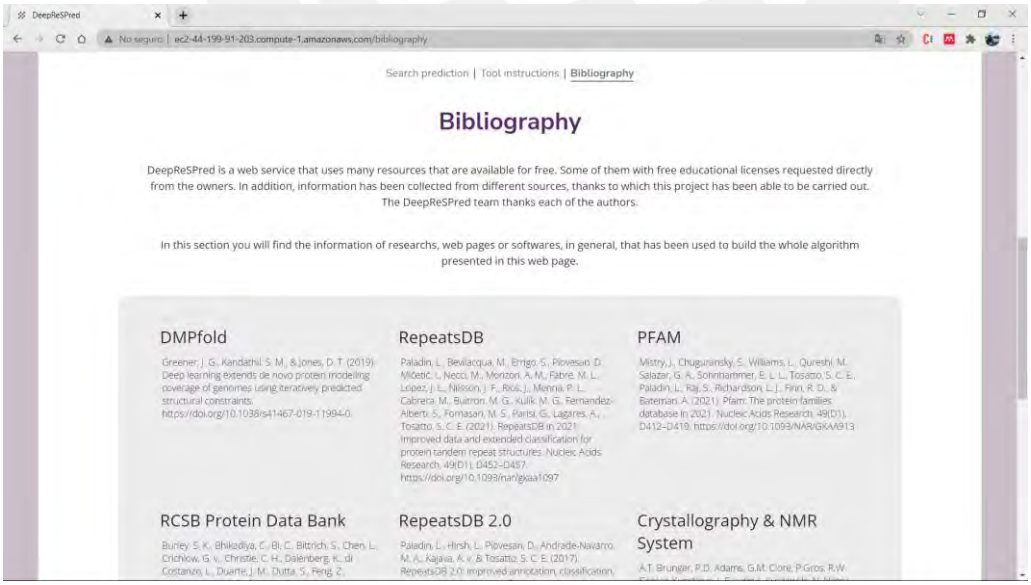

Figura 46. *Interfaz de la herramienta propuesta DeepReSPred - Sección de bibliografía. (Elaboración propia).*

7. Login de administrador

En la sección inferior de todas las pantallas del servicio web DeepReSPred se mostrará un botón para acceder a la sección administrativa de la herramienta, tal y como se muestra en la Figura 47. Al presionar este botón, ubicado en la zona inferior derecha, se redirigirá al usuario hacia la sección de login de administrador.

Figura 47. *Interfaz de la herramienta propuesta DeepReSPred - Zona de acceso a la sección de login de administrador. (Elaboración propia).*

En la sección de login se solicitará el ingreso del usuario y la contraseña del administrador. En la Figura 48, se observan los campos textuales a ingresar. Estos campos cuentan con validaciones en cuanto a la cantidad mínima y máxima de los caracteres aceptados.

El usuario administrador deberá ingresar los datos correctos para poder ingresar a la sección administrativa de la herramienta.

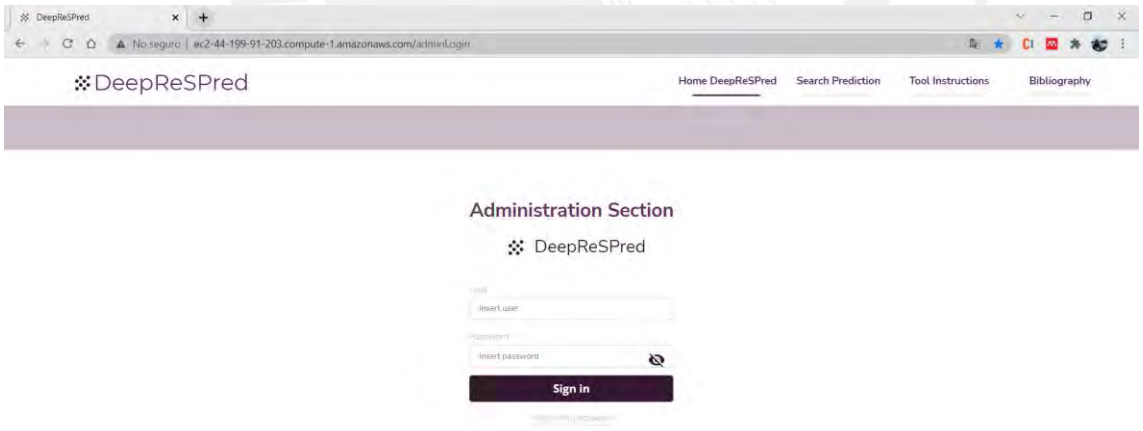

Figura 48. *Interfaz de la herramienta propuesta DeepReSPred - Sección de login de administración. (Elaboración propia).*

8. Panel administrativo

La herramienta contará con una sección administrativa al cual solo podrán acceder los usuarios que cuenten con el usuario y la contraseña correcta.

En esta sección se podrán configurar algunos parámetros generales de la herramienta, tales como la cantidad máxima de solicitudes de predicción permitida por día por punto de acceso.

Asimismo, se podrá visualizar la lista de puntos de acceso desde los cuales se ha ingresado solicitudes de predicción. En caso de que se supere la cantidad máxima configurada a través de los parámetros generales, se bloqueará al punto de acceso. Este bloqueo es temporal y se desactivará al iniciar el día siguiente.

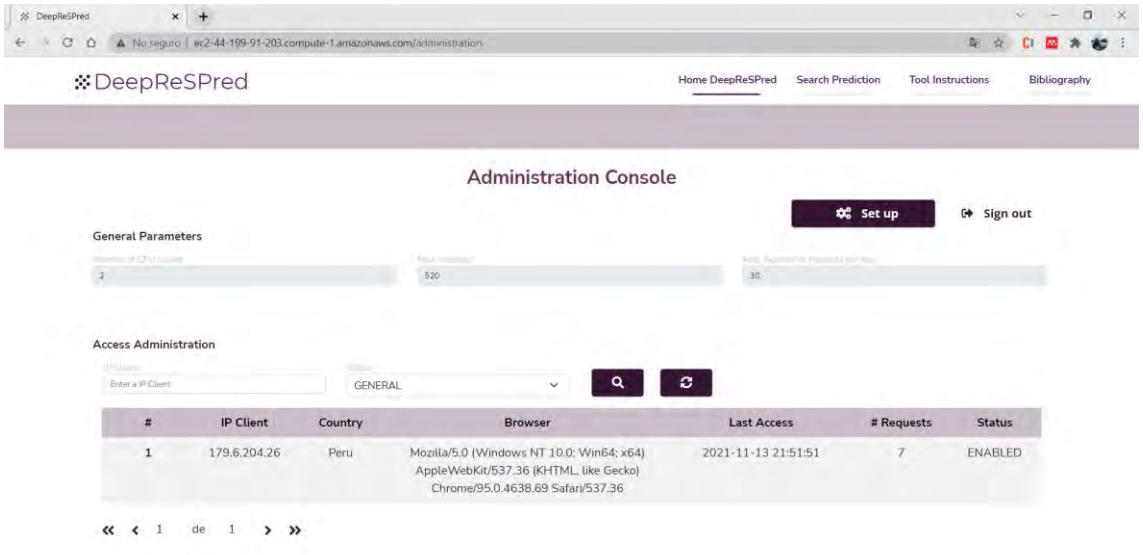

Figura 49. Interfaz de la herramienta propuesta DeepReSPred - Sección del panel administrativo. (Elaboración propia).

Tal como se puede observar en la Figura 49, el administrador de la herramienta también visualizará el menú de navegación de las secciones de DeepReSPred, así como el botón de cerrar sesión a lado del tírulo de la sección. No obstante, una vez que se salga de ese entorno dirigiéndose a cualquier otra sección a través del menú, se cerrará la sesión.

Anexo L: Reporte de funcionamiento de la herramienta implementada

El presente anexo contiene el documento que reporta la descripción del funcionamiento de la herramienta implementada. Este reporte corresponde al primer medio de verificación del resultado alcanzado número uno del tercer objetivo específico[48] de este proyecto de tesis. Su contenido abarca la descripción de la evaluación de funcionamiento realizada a la herramienta implementada, incluyendo el enlace para acceder a la grabación de la misma, la comparación entre los resultados obtenidos al utilizar el algoritmo adaptado con dos bases de datos diferentes y, finalmente, un apéndice donde se detalla la ubicación de los repositorios del código fuente de la herramienta y la explicación de su contenido.

1. Introducción

En el Capítulo 5 se planteó la necesidad de capturar una serie de requisitos funcionales y no funcionales que desde ese objetivo específico deberían ser incluidas en el diseño de la interfaz de la herramienta propuesta para posteriormente incluirlas e implementarlas. Al momento del desarrollo de esta sección del proyecto, la interfaz desarrollada ya se encuentra integrada con el algoritmo adaptado. Es por ello que se plantea la elaboración de este documento. Su primer contenido abarca la presentación de la evaluación del funcionamiento de la herramienta DeepReSPred en base al cumplimiento de los requisitos funcionales del catálogo de requisitos que finalmente se encuentra en el Anexo G.

2. Evaluación de funcionamiento de la herramienta implementada

Acorde al catálogo de requisitos del Anexo G perteneciente al primer resultado esperado del objetivo en el que también se enfoca este reporte, se deberá desarrollar una interfaz web mediante la cual los usuarios interesados puedan registrar una solicitud de predicción y puedan obtener una serie de resultados.

[48] O3. Integrar el algoritmo adaptado con la interfaz implementada para crear la herramienta que realice la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria.

Para verificar el funcionamiento de la herramienta desarrollada será necesario interactuar con ella. Esta interacción se ha grabado ya que también representa un medio de verificación del funcionamiento. Para acceder al recurso solo basta con ingresar al siguiente enlace:

Grabación de la herramienta de predicción desarrollada en ejecución

De la misma manera, en esta sección se describirá lo realizado en dicha grabación:

Mediante la plataforma web denominada como DeepReSPred se ha registrado una solicitud de predicción de la familia de proteínas PF18773. Esta familia consta de seiscientas setenta y nueve unidades de repetición. No obstante, teniendo en cuenta que se requiere visualizar el resultado en un corto tiempo, solo para esta verificación de funcionamiento se ha configurado al script que consulta a la base de datos PFAM para obtener una generación “semilla” de los fragmentos, es decir, un grupo representativo de la familia, en vez de una generación “completa”. Con ello, se redujo la cantidad de secuencias obtenidas a un total de trece.

A la predicción ingresada se le asignó automáticamente el identificador 28PYUWZS y se registró en base de datos. Esto se verificó a través de la recepción de un correo electrónico en el cual se confirmaba el registro de la solicitud de predicción y se incluía el identificador de la solicitud ingresada para poder revisar los resultados después.

A partir de las trece unidades de repetición se generaron veintidós archivos en formato fasta que se encolarían de acuerdo al flujo propuesto por el algoritmo adaptado. Este incremento en la cantidad de secuencias de las cuales se deberá predecir sus estructuras corresponde a los dos caminos planteados en el algoritmo, la creación de archivos fasta a partir de las unidades de repetición independientes y la creación de ficheros a partir de la reducción de la longitud de la secuencia completa que contiene al fragmento en cuestión.

El procesamiento de esta solicitud inició apenas se registró en base de datos debido a que la cola de procesamiento se encontraba vacía.

Al cabo de siete mil quinientos nueve segundos, lo que corresponde a un aproximado de dos horas con cinco minutos, se recibió un correo con el cual se confirmaría la finalización exitosa del proceso de predicción.

Seguidamente, se dirige una vez más a la interfaz web y se consulta por la solicitud de predicción correspondiente a la presente verificación de funcionamiento. A partir de la

consulta se obtienen los archivos PDB, los cuales contienen a las estructuras de las proteínas predichas. Se puede interactuar con las estructuras predichas a través del visualizador incluido en la interfaz, además se podrá descargar los resultados al seleccionar el botón correspondiente.

Con ello, se da por finalizado el flujo básico propuesto para realizar una predicción de estructuras de proteínas repetidas a partir de su secuencia de aminoácidos y obtener los resultados del algoritmo adaptado.

3. Comparación de resultados

Acorde a lo mencionado en la sección de consideraciones adicionales del Anexo J correspondiente al reporte de pruebas del algoritmo adaptado, las pruebas realizadas se han ejecutado en un entorno robusto capaz de soportar el procesamiento de la base de datos UniRef. No obstante, como se explicó en ese apartado, se verificó mediante el intento de la instalación de ese recurso en distintos ambientes que no sería posible la ejecución del algoritmo en entornos con recursos computacionales promedio. Ante ello surgió una alternativa que fue considerada como la solución a ese inconveniente: el uso de una base de datos alternativa como PFAM.

Aquella es una base de datos más ligera que es soportada por cualquier entorno, y permite que se ejecute el algoritmo adaptado sin que requiera de mucho recurso computacional. Es por ello que en esta sección se realizará una comparación de los resultados obtenidos entre la ejecución de una solicitud de predicción de una proteína repetida en un equipo robusto que utilice la base de datos UniRef versus su ejecución en un equipo promedio que utilice la base de datos PFAM.

Para la realización de esta comparación se utilizará la misma familia evaluada en la ejecución de la verificación de funcionamiento desarrollada en el apartado anterior: la familia de proteínas repetidas PF18773.

Dado que existe una diferencia en cuanto a la cantidad de secuencias a predecir, se buscará comparar las predicciones de las estructuras de un subconjunto de los trece fragmentos capturados como los más representativos de la familia.

Con ello, se seleccionó a cuatro fragmentos de los cuales se tuvo que buscar el resultado homólogo predicho con el algoritmo adaptado ejecutado en un equipo más robusto que trabajó con la base de datos UniRef.

A continuación, en la Figura 50, se muestra el compilado de los alineamientos realizados entre cada una de las estructuras predichas de las cuatro unidades de repetición en

evaluación. Las estructuras observables de color verde corresponden a las obtenidas a partir del uso del algoritmo adaptado con la base de deatos UniRef30_2020_06_hhsuite, mientras que las de color celeste son las obtenidas con la base de datos PfamA_31.0.

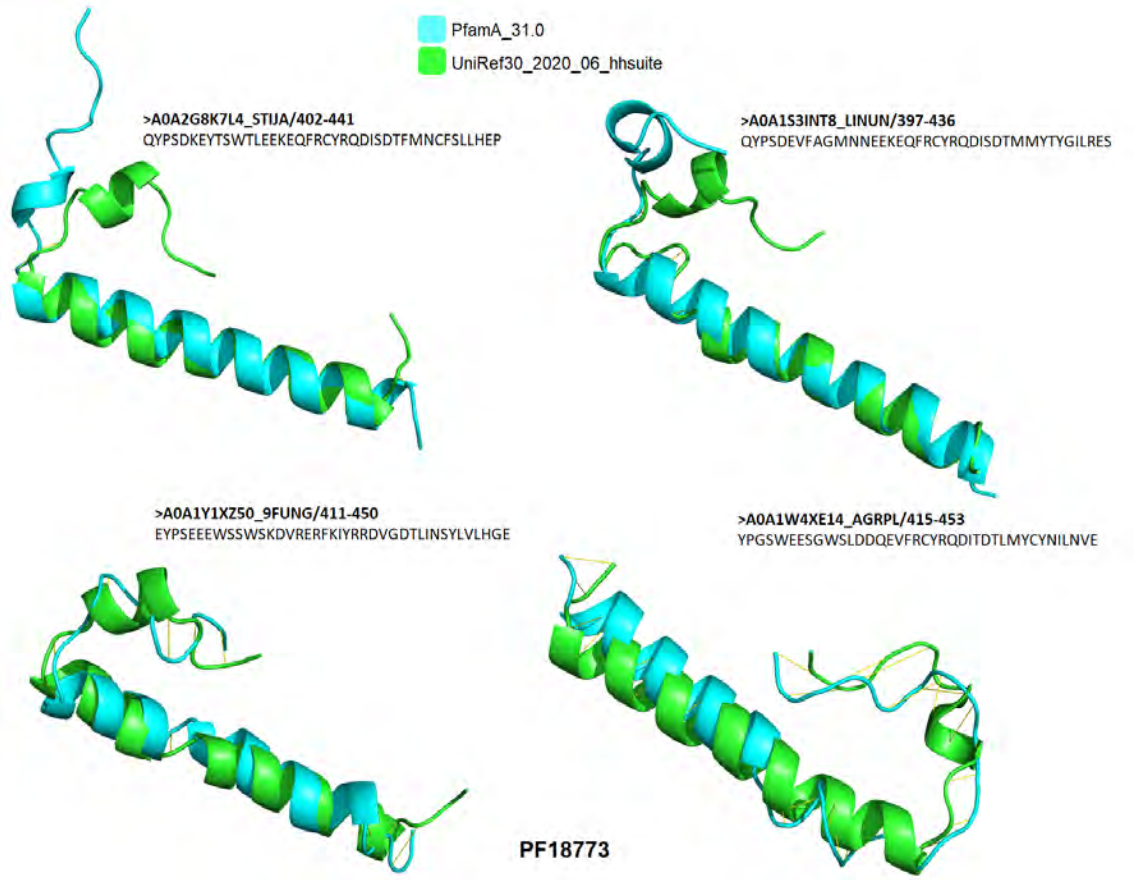

Figura 50. *Comparación por alineamiento estructural de las estructuras predichas de la familia de proteínas repetidas PF18773 con el algoritmo adaptado. Las estructuras de color celeste corresponden a las predichas con la base de datos Pfam y las de color verde, a las predichas con UniRef30. (Elaboración propia en PyMol).*

Asimismo, en la Tabla 32, se presenta el puntaje tm-score obtenido para cada uno de los alineamientos. Se observará el código de la familia PFAM a la que pertenecen, el nombre de la secuencia, la longitud de la cadena obtenida por cada base de datos utilizados, la longitud del alineamiento generado y, por último, el puntaje TM-score tanto si se normaliza por la longitud de la cadena generado con Pfam o UniRef.

Resumen cuantitativo de la comparación de estructuras por base de datos utilizado						
Código PFAM	Secuencia	Longitud de la cadena		Longitud de alineamiento	TM-score	
		Pfam	UniRef		N-Pfam	N-UniRef
PF18773	A0A2G8K7L4_STIJA	40	40	33	0.59	0.59
PF18773	A0A1S3INT8_LINUN	40	40	33	0.53	0.53
PF18773	A0A1Y1XZ50_9FUNG	40	40	34	0.39	0.39
PF18773	A0A1W4XE14_AGRPL	39	40	36	0.56	0.55

Tabla 32. *Cuadro resumen de la comparación por alineamiento estructural entre estructuras predichas con la base de datos Pfam y UniRef*

Tal como se observa también en la Tabla 32, el mayor valor obtenido en el alineamiento con TM-align es 0.59, el mínimo valor es 0.39 y el promedio de todas es 0.51. Cabe recordar que un valor superior a 0.5 indica que se trata del mismo plegado de estructura. Con ello se concluye que no existe una brecha muy grande entre los resultados obtenidos de forma independiente con cada una de las bases de datos. Lo cual funcionará como una medida probatoria de que no habrá distinción al utilizar la base de datos PFAM en ambientes que no cuenten con recursos computacionales superiores al promedio.

4. Apéndice

En esta sección se presenta el acceso al código fuente de la herramienta implementada y la descripción general del contenido del repositorio que lo almacena.

4.1 Código fuente del algoritmo adaptado

La herramienta web bioinformática propuesta en este proyecto de tesis ha gestionado sus fuentes en base a una distinción básica de entornos: back-end y front-end, y a la utilización de GitHub. Así, para poder acceder a los diversos recursos se deberá ingresar a dos repositorios. El primero de ellos contiene el código fuente relacionado al back-end de la herramienta y está nombrado como “DeepReSPred-back”. En el mismo sentido, el segundo repositorio contiene el código fuente relacionado al front-end de la herramienta

y está nombrado como “DeepReSPred-front”. Cabe mencionar que ambos repositorios tienen habilitado el acceso público, por lo cual para poder obtener las fuentes requeridas bastará con el ingreso al enlace de interés:

Repositorio GitHub back-end de la herramienta DeepReSPred

Repositorio GitHub front-end de la herramienta DeepReSPred

El repositorio back-end cuenta con una carpeta principal que contiene dos carpetas secundarias. En el directorio “programsAuxiliar” se encuentran algunas herramientas necesarias para la implementación del algoritmo adaptado, teniendo en cuenta que para la realizar la instalación de este algoritmo será necesario instalar primero el algoritmo original siguiendo los pasos de instalación descritos en el Anexo E.

Cabe mencionar que el uso de las herramientas contenidas en el repositorio del algoritmo adaptado deberá tener una finalidad enteramente académica.

El directorio “back_project” contiene, en primera instancia, a los archivos Python con los cuales se configuran los apis con los que se manejará la base de datos, la instancia S3 de Amazon Web services y con las que se atenderán las peticiones de front-end.

Asimismo, contiene dos subdirectorios. El primero denominado como “deepReSPred”, en el cual se encuentran primordialmente los archivos creados MappingFasta.py, SP_collection.txt, run_repeat_prediction.sh y el archivo modificado del algoritmo original, run_dmpfold.sh.

Cabe recordar que el archivo SP_collection es el fichero que contiene la relación entre los identificadores Uniprot y los accesos Uniprot necesarios para poder utilizar el api del Banco de Datos de proteínas.

El segundo subdirectorio corresponde a “autProcess” dentro del cual se encuentran tres ficheros Python, que se utilizaron para poder definir procesos recurrentes que encolen las solicitudes de predicción, ejecuten el algoritmo adaptado, actualicen la base de datos y almacenen los datos en el bucket de S3.

Por otro lado, el repositorio front-end cuenta con una carpeta principal que contiene todos los ficheros relacionados por la implementación de la interfaz de la herramienta. Cabe mencionar que se ha utilizado el framework Vue.js para su desarrollo, por lo cual los ficheros de cada vista de la interfaz tendrán una extensión vue.

Anexo M: Documento de especificación y resultado de pruebas funcionales

En este anexo se encuentra el documento del reporte de especificación y resultado de pruebas realizadas en la herramienta implementada. Su contenido abarca a una introducción del documento, la presentación de la matriz de trazabilidad de los casos de prueba y la especificación de las pruebas realizadas. Cada prueba funcional propuesta se desarrollará en un apartado diferente y contendrá la descripción del resultado de su ejecución. Finalmente, como parte de un apéndice, se adjunta el acta de validación del documento a través de juicio experto.

1. Introducción

Llegado a este punto del desarrollo de la herramienta propuesta cabe mencionar que, tal y como se describe en el apartado de estándares del plan de proyecto del Anexo C, se han seguido algunas de las buenas prácticas recomendadas por el marco de trabajo para el desarrollo ágil denominado Scrum. Con ello, se han realizado pruebas frecuentes a lo largo de todo el desarrollo; no obstante, al tratarse de un proyecto de software en este proyecto se tendrá en cuenta su ciclo de vida y se elaborará un plan de pruebas final. El presente documento contempla el plan de pruebas mecionado y tiene el objetivo de verificar el cumplimiento de las distintas actividades soportadas por la herramienta.

2. Trazabilidad de los casos de prueba - requisitos

En este apartado se contemplará una tabla en la cual se indica la correspondencia entre los casos de prueba definidos y los requisitos funcionales del catálogo de requisitos ubicado en el Anexo G. La Tabla 33 contiene la información mencionada.

Se reconoció un total de cuarenta y un requisitos, de los cuales veintisiete eran funcionales. Esos requisitos se han clasificado en dos tipos: cinco de ellos se denominaron como requisitos deseables y veintidós como exigibles.

Correspondencia entre casos de prueba y requisitos funcionales			
Caso de prueba	Requisitos funcionales	Cantidad de requisitos	
		Exigibles	Deseables
CP1	R5, R8, R10, R11, R24, R1, R6, R26, R12	7	2
CP2	R9, R25	2	-

CP3	R14, R15, R16, R17, R18	5	-
CP4	R19, R21	2	-
CP5	R20	1	-
CP6	R7, R23	2	-
CP7	R4, R22, R27	-	3

Tabla 33. Correspondencia entre casos de prueba y requisitos funcionales

3. Especificación y resultado de pruebas

En esta sección se procederá a describir una serie de pruebas que validen el funcionamiento esperado de la herramienta desarrollada, DeepReSPred. Las pruebas propuestas contemplarán tanto al funcionamiento del algoritmo adaptado como a las actividades que son realizables desde la interfaz de la herramienta. En el mismo sentido, las pruebas de los siguientes apartados estarán alineadas a los requisitos funcionales de la herramienta. El catálogo de requisitos se encuentra en el Anexo G.

Así, a continuación, se presentarán un total de siete acápites, los cuales contendrán la descripción de una prueba funcional a realizar, los prerrequisitos, los pasos a seguir, el resultado esperado y el resultado obtenido.

3.1 Caso de prueba CP1: Predicción exitosa de proteínas repetidas

La finalidad principal de la herramienta propuesta corresponde a la predicción de estructuras terciarias de proteínas a partir de estructuras primarias. Esto tomará en cuenta el rendimiento del algoritmo adaptado. Con ello, esta prueba busca evaluar la calidad de los resultados obtenidos a partir de una predicción exitosa.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
  - El usuario deberá encontrarse en la pantalla principal de la herramienta
- **Pasos a seguir**
  - Seleccionar la opción de dato de entrada PFAM de ejemplo y presionar “Start”
  - El sistema mostrará la ventana modal del resumen de la solicitud de predicción y generará un identificador único
  - Ingresar un correo electrónico y presionar la opción “Send request”

- **Consideraciones adicionales**

Una vez que se obtengan resultados de la predicción de proteínas se deberá verificar la la calidad de los mismos, es por ello que se ha establecido al alineamiento de las estructuras generados por la herramienta un total de cinco estructuras de proteínas alojadas en el Banco de datos de proteínas relacionadas a una cantidad de fragmentos de la familia de proteínas ingresada. A continuación, en la Tabla 34, se presentan las estructuras de proteínas escogidas para realizar dicha evaluación.

Catálogo de estructuras PDB escogidas para verificación del caso de prueba CP1						
Código PFAM	**Descripción**	**Código PDB 1**	**Código PDB 2**	**Código PDB 3**	**Código PDB 4**	**Código PDB 5**
PF18773	Importin 13 repeat	2x19	2xwu	3zjy	2x1g	3zkv

Tabla 34. *Catálogo de estructuras PDB escogidas para verificación del caso de prueba CP1*

En general, se identificaron un total de cinco estructuras de proteínas con las cuales se realizará la comparación. No obstante, para capturar la información de esas estructuras, se utilizará la siguiente API del Banco de Datos de Proteínas:

https://files.rcsb.org/download/{{CódigoPDB}}.pdb

Se deberá modificar la variable *{{CódigoPDB}}* por el código de cada una de las estructuras escogidas.

- **Resultado esperado**

  Al ingresar la solicitud de predicción el sistema ocurre lo descrito a continuación:
  - Los datos capturados en la solicitud ingresada son almacenados en la base de datos
  - Se encolará a la solicitud de predicción
  - Se iniciará el proceso de predicción
  - Se generarán modelos de apoyo como los datos de covarianza y los mapas de contacto para cada fragmento admitido
  - Se finalizará la predicción de las proteínas repetidas
  - Se deberán enviar los resultados de la predicción al usuario mediante el correo electrónico registrado

- **Resultado obtenido**

Los resultados obtenidos son los siguientes:

- La herramienta DeepReSPred muestra al usuario el código PF18773 al momento en que este selecciona la opción de inserción de datos de entrada PFAM por defecto.
- La herramienta encola la solicitud de predicción ingresada y almacena los datos en la base de datos
- Comienza el flujo del procesamiento
- Finaliza el procesamiento
- Se generaron nueve estructuras a partir de los fragmentos de proteínas o lo que se ha denominado en este proyecto como “secuencias representativas”
- La herramienta envío los resultados obtenidos al correo electrónico ingresado en la solicitud.

Dado que se cuentan con nueve estructuras predichas por el algoritmo adaptado, se realizará la evaluación de alineamiento con cada una de esas estructuras predichas versus cada una de las estructuras obtenidas desde PDB. La herramienta con la cual se ha evaluado el alineamiento es TM-align.

A continuación, en la Tabla 35, se observa el resumen del puntaje obtenido por cada combinación de alineamiento evaluada.

Resumen de TM-score por cada combinación de alineamiento evaluada					
Nombre de la secuencia del fragmento evaluado	**2x19**	**2xwu**	**3zjy**	**2x1g**	**3zkv**
A0A2G8K7L4_STIJA	0.42	0.46	0.42	0.33	**0.55**
C3ZJ96_BRAFL	0.41	0.44	0.40	0.34	**0.50**
C3ZJ96_BRAFL (nr)[49]	0.47	**0.71**	0.45	0.36	0.65

[49] Esta unidad de repetición cuenta con dos estructuras predichas, la primera de las cuales obtuvo un 0.5 como el mayor puntaje en la evaluación. La segunda estructura se difiere de la primera debido al procesamiento del que fue parte y corresponde a lo que se ha denominado en este proyecto como “secuencia representativa” por lo cual se le ha asignado el sufijo “nr” al final del nombre de la secuencia.

V4AA09_LOTGI	0.34	0.49	0.40	0.31	**0.54**
A0A1S3INT8_LINUN	0.44	0.47	0.44	0.34	**0.53**
A0A1Y1XZ50_9FUNG	0.35	0.45	0.37	0.37	**0.55**
A0A1W4XE14_AGRPL	0.43	0.41	0.43	0.34	**0.54**
D6WZG3_TRICA	0.35	0.38	0.36	0.39	**0.44**
F6T6E4_CIOIN	0.38	**0.42**	0.36	0.34	0.42

Tabla 35. *Cuadro resumen del puntaje obtenido por cada combinación de alineamiento evaluada*

Como se observa en la Tabla 35, para la mayoría de los casos se ha obtenido un puntaje superior esperado (0.5). Lo cual corresponde a una predicción exitosa con resultados totalmente aceptables.

A continuación, en la Figura 37 y Figura 38, se muestra la visualización del alineamiento de las estructuras que obtuvieron el mayor puntaje en la presente evaluación.

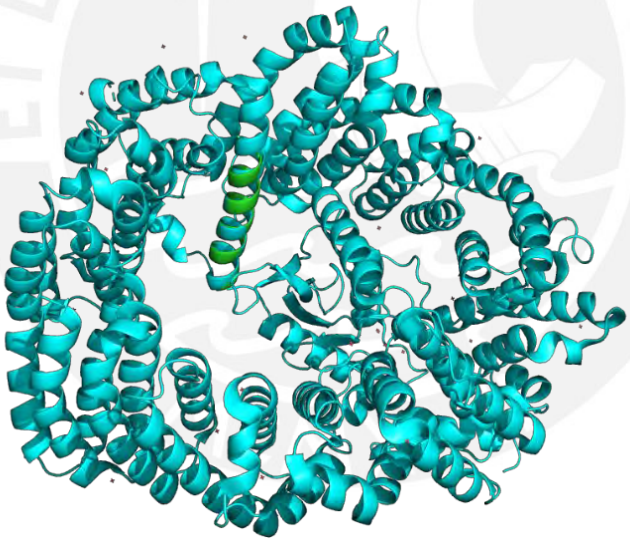

Tabla 36. *Visualización del alineamiento estructual entre el fragmento de la secuencia C3ZJ96_BRAFL (nr) predicha y la estructura 2XWU de PDB. (Elaboración propia en PyMol).*

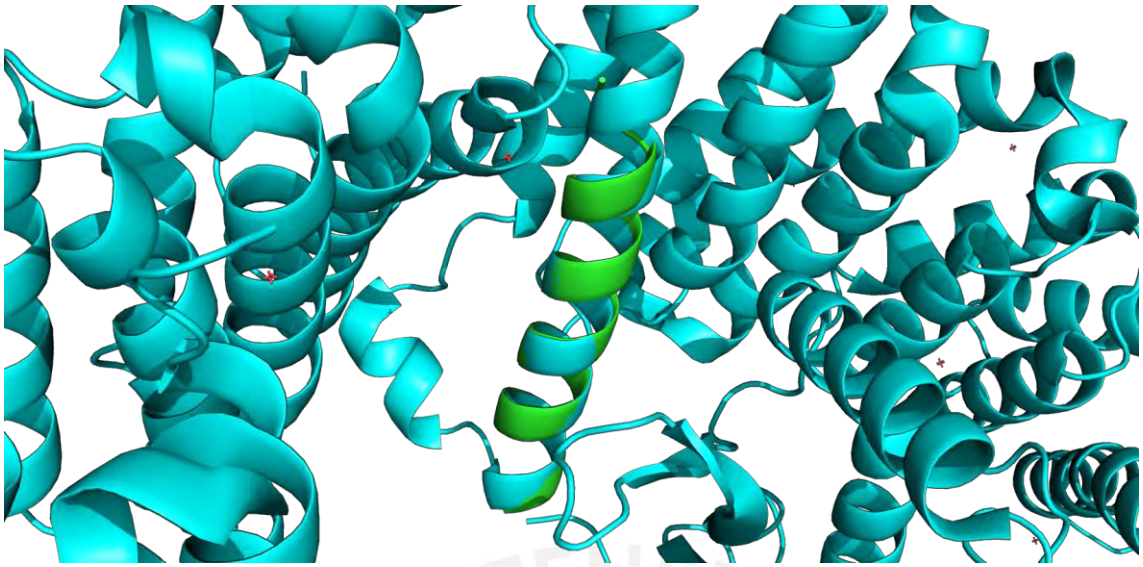

Tabla 37. Visualización enfocada del alineamiento estructual entre el fragmento de la secuencia C3ZJ96_BRAFL (nr) predicha y la estructura 2XWU de PDB. (Elaboración propia en PyMol).

3.2 Caso de prueba CP2: Datos de entrada válidos

La herramienta DeepReSPred permite ingresar diferentes tipos de datos de entrada para realizar la predicción entre los que se encuentran el código PFAM, una secuencia completa de una proteína repetida, las secuencias que conforman a una familia en formato de texto plano, en formato Fasta y en formato Stockholm. Asimismo, se permitirá el ingreso de los datos de entrada descritos a través de la carga de un archivo. Este caso de prueba busca validar la aceptación de los datos de entrada descritos.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
  - El usuario deberá encontrarse en la pantalla principal de la herramienta
  - El usuario contará con una secuencia de una proteína repetida
  - El usuario deberá contar con las secuencias de los fragmentos pertenecientes a una familia de proteínas contenidas en un archivo con extensión de texto plano.
- **Pasos a seguir**
  - El usuario deberá ingresar la secuencia de aminoácidos de la proteína repetida en el campo textual
  - El usuario limpiará el campo textual utilizando la opción “Clear”

- El usuario seleccionará el ingreso de datos por archivo y seleccionará el que contiene los fragmentos de la familia de proteínas repetidas.

- **Resultado esperado**
  - La herramienta no mostrará ningún mensaje de error debajo del campo textual con lo cual se verifica que no hubo ningún error en el ingreso de datos
- **Resultado obtenido**
  - La herramienta no mostró ningún mensaje de eror debajo del campo textual

3.3 Caso de prueba CP3: Consulta de resultados de predicción

La herramienta permite la consulta del estado y los resultados de las solicitudes de predicción ingresada. Mediante este caso de prueba se busca validar que esta funcionalidad se realice de forma exitosa.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
  - El usuario deberá encontrarse en la sección de búsqueda de la herramienta
  - El usuario ha registrado previamente una solicitud de predicción ingresando su correo electrónico
  - El usuario cuenta con el identificador de la solicitud de predicción
  - El usuario debe hacer recibido el correo de confirmación de finalización de predicción
- **Pasos a seguir**
  - El usuario deberá ingresar el identificador de la solicitud de predicción en el campo textual de la sección de búsqueda
  - Se debe presionar el botón de búsqueda. Este botón tiene un icono de lupa y está ubicado al lado derecho del campo textual
  - El usuario selecciona el botón de descarga de uno de los resultados obtenidos y guarda localmente el archivo generado
  - El usuario selecciona el botón de visualización de uno de los resultados obtenidos
- **Resultado esperado**
  - La herramienta devuelve la información del estado de la solicitud: “Completado”
  - La herramienta muestra los datos de entrada ingresados en la solicitud de predicción

- La herramienta lista los resultados obtenidos en la predicción
- La herramienta selecciona el primero resultado obtenido y lo muestra en la sección de visualización
- Cuando el usuario presiona el botón de descarga, la herramienta genera el archivo y habilita su descarga
- Cuando el usuario presiona el botón de visualización, la herramienta muestra la proteína seleccionada en la sección de visualización.

- **Resultado obtenido**
  - La herramienta devolvió la información del estado de la solicitud: "Completado"
  - La herramienta mostró los datos de entrada ingresados en la solicitud de predicción
  - La herramienta listó los resultados obtenidos en la predicción
  - En primera instancia, se visualizó la estructura de uno de los resultados obtenidos
  - Al presionar el botón de descarga, la herramienta generó el archivo y se pudo descargar el archivo de forma local
  - Al presionar el botón de visualización, la herramienta actualizó la sección de visualización y se mostró la estructura de la proteína seleccionada.

3.4 Caso de prueba CP4: Sección de tutoriales e información adicional

La herramienta cuenta con dos secciones adicionales para poder tener conocimiento del funcionamiento de la misma y sobre los recursos utilizados en su desarrollo. A través de este caso de prueba se busca evaluar que su contenido se encuentre disponible para el usuario.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
- **Pasos a seguir**
  - El usuario deberá dirigirse hacia la zona media de la página web y seleccionar la pestaña de "Tool Instructions"
  - El usuario deberá seleccionar otra pestaña, en este caso, la de "Bibliography"

- **Resultado esperado**
  - Para cada una de las interacciones la herramienta deberá mostrar la información y los elementos de la pestaña seleccionada
- **Resultado obtenido**
  - Al seleccionar la pestaña “Tool instructions” se visualizó el cambio de sección hacia la sección que contenía a las intrucciones de la herramienta
  - Al seleccionar la pestaña “Bibliography” se visualizó el cambio de sección hacia la sección que contenía a la descripción de los recursos utilizados en el desarrollo de la herramienta

3.5 Caso de prueba CP5: Notificaciones de error

Las notificaciones de error son distintos a los errores que difieren de los mensajes de error en la validación de campos. Estas notificaciones se muestran en modales en la misma ventana en donde ocurrió el error. Este caso de uso busca validar esa situación.

- **Prerequisitos**
  - El usuario se encuentra en el modal de resumen de solicitud de predicción
  - El usuario ha ingresado todos los datos correctamente
  - Los servicios de la herramienta se encuentran deshabilitados
- **Pasos a seguir**
  - El usuario presiona el botón de “Send Request”
- **Resultado esperado**
  - La herramienta muestra una notificación de error superpuesta al modal de resumen de la solicitud de predicción con un mensaje descriptivo de la situación
- **Resultado obtenido**
  - La herramienta mostró una notificación de error con el siguiente mensaje de error: “The prediction request was not sent. Review the data input and try again”.

3.6 Caso de prueba CP6: Validación de los aminoácidos naturales

La herramienta permite el ingreso directo a una secuencia perteneciente a una proteína repetida para realizar su predicción. Una secuencia válida solo contiene caracteres alfanúmericos relacionados a la simbología de los veinte aminoácidos naturales

existentes. Este caso de prueba busca verificar que la herramienta valide que la información ingresada solo contiene la simbología de los aminoácidos existentes, esto solo en el caso de que el tipo de dato de entrada de trate de una secuencia de proteína.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
  - El usuario se encuentra en la pantalla principal de la herramienta
  - El usuario cuenta con una secuencia de aminoácidos inválida (incluye caracteres como la Z)
- **Pasos a seguir**
  - El usuario ingresa la cadena en el campo textual del formulario
- **Resultado esperado**
  - La herramienta valida la información ingresada y muestra un mensaje de error debajo del campo textual
- **Resultado obtenido**
  - La herramienta mostró el siguiente mensaje de error: “Invalid sequence input. Verify correct residues symbols.”

3.7 Caso de prueba CP7: Sección administrativa

La herramienta pone a la disposición de un administrador una sección mediante el cual se pueda visualizar los accesos de los usuarios de la herramienta en cuanto al registro de solicitudes. Asimismo, mediante esta sección el administrador tiene opción a visualizar y modificar los valores de algunos parámetros generales.

- **Prerequisitos**
  - La herramienta web debe encontrarse disponible
  - El usuario tiene el rol de administrador de la herramienta
  - El usuario tiene conocimiento del nombre de usuario y de la contraseña requerida para ingresar a la sección administrativa
  - El usuario se encuentra en la sección de login administrativo
- **Pasos a seguir**
  - El usuario ingresa las credenciales de administrador en los campos correspondientes

- El sistema valida las credenciales y permite el acceso a la zona de administrador. El usuario puede observar el listado de accesos del día actual y los parámetros generales de la herramienta
- El usuario presiona el botón de “Set up”
- El sistema habilita la edición de los campos de los parámetros. El usuario modifica la cantidad máxima admitida de registros de solicitud de predicción por punto de acceso.
- El usuario presiona la opción “Set up”
- El sistema muestra un modal de confirmación. El usuario presiona el botón “Set up”

- **Resultado esperado**
  - La herramienta valida las credenciales ingresadas y permite el acceso a la sección administrativa
  - La herramienta muestra un mensaje de éxito luego de realizar la modificación de los parámetros
- **Resultado obtenido**
  - Se logró ingresar a la sección administrativa de la herramienta
  - Luego de realizar la modificación del valor del parámetro, se mostró el siguiente mensaje: “System parameters configured successfully.”

4. Apéndice

En esta sección se presenta el acta de validación del presente documento de especificación y resultado de pruebas de la herramienta propuesta.

4.1 Validación del documento por medio de juicio experto

La validación del documento de especificación y resultado de pruebas de la herramienta propuesta ha sido realizada por una experta en bioinformática. La realización de esta verificación implica la revisión completa del informe y la entrega de observaciones en caso se requieran. A continuación, se observa el acta de validación recibida por la experta.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Acta de validación de documento

Título de tesis: Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

Tesista: Solange Estrella Palomino Chahua

Nombre del documento: Documento de especificación y resultado de pruebas funcionales

Descripción del documento: Documento que contiene la descripción de las pruebas funcionales realizadas a la herramienta implementada y los resultados de su ejecución

Mediante la presente acta, yo, la Dra. Layla Hirsh Martinez, dejo constancia de que, en mi calidad de experta en bioinformática, he revisado el documento descrito en los puntos anteriores perteneciente al proyecto de tesis en mención. En ese sentido, en la siguiente sección se especifica el veredicto y, en caso hubiesen, las observaciones correspondientes al documento.

Veredicto:

(X) Aprobado ( ) Requiere observación

Observaciones:

Lima, 06 de noviembre de 2021

Firma

Anexo N: Reporte de la evaluación de usabilidad de la herramienta

Este anexo contiene el reporte de la evaluación de usabilidad de la herramienta propuesta capaz de realizar predicciones de estructuras terciarias de proteínas repetidas. Su contenido abarca la presentación de la herramienta de evaluación de usabilidad seleccionada, la explicación de la metodología de evaluación y el análisis de los resultados obtenidos. Adicionalmente, se incluye el acta de validación de este reporte a través de juicio experto como parte de un apéndice en la última sección de este anexo.

1. Introducción

La revisión sistemática realizada en el Capítulo 3. Estado del Arte del presente proyecto de fin de carrera dio a relucir ciertas definiciencias en torno a la aplicación de lineamientos de usabilidad en el proceso de desarrollo de las interfaces de las herramientas bioinformáticas. Es notable la carencia de una evaluación de usabilidad de los recursos con los que se interactúan al estar en un contexto bioinformático.

El desarrollo de la herramienta que propone este proyecto ha tenido en consideración algunos lineamientos básicos de usabilidad, teniendo como finalidad ofrecer un recurso que sea amigable al usuario y que brinde las facilidades para el entendimiento de su uso. Si bien se ha tomado en consideración al usuario para poder entender su necesidad en torno al uso de las herramientas bioinformáticas, al culminar el desarrollo es saludable realizar una evaluación de usabilidad. Esta evaluación deberá servir como retrospectiva al desarrollo de la herramienta y servir de apoyo para el desarrollo de otras.

2. Herramienta de usabilidad web

Dado que se ha reconocido la necesidad de la inclusión de la usabilidad en el desarrollo de los recursos tecnológicos, se han iniciado una serie de investigaciones en torno a esta temática. Gracias a ello es que se ha podido reconocer a uno de los estudios más recientes en relación a la usabilidad de plataformas web bioinformáticas, una descripción que se ajusta fielmente a la herramienta que propone este proyecto.

En ese sentido, se ha encontrado una investigación que propone una adaptación de los métodos de evaluación de usabilidad usualmente aplicados, si es que se aplican.

El proyecto en cuestión es denominado como Usabilidad en servicios web bioinformáticos (Bezerra Brandao Corrales et al., 2020).

3. Metodología de evaluación

La herramienta de evaluación de usabilidad propone su medición tomando enteramente la percepción de los usuarios luego de usar algún servicio web. La propuesta de ese proyecto incluye la elaboración de un formulario web que contiene diez preguntas, las cuales deberán ser respondidas por los usuarios de la herramienta en evaluación.

De acuerdo a ello, se ha elaborado un formulario web con ayuda de la herramienta Google Forms. Su contenido se distribuyó en cuatro secciones, de las cuales, la última corresponde a un agradecimiento por su participación y al registro de su correo electrónico en caso esté interesado por conocer los resultados de la evaluación de usabilidad.

En la primera sección del formulario, tal y como se muestra en la Figura 51, se realizó una presentación al proyecto de tesis al cual pertenece la evaluación de usabilidad, además del consentimiento informado para el uso de los datos recolectados.

Figura 51. *Formulario de evaluación de usabilidad en DeepReSPred - Primera sección. (Elaboración propia en Google Forms).*

En la segunda sección del formulario se captura la informacion del participante en cuanto a su ocupación y su experiencia con recursos web bioinformáticos dedicados a la prediccion de estructuras de proteínas o herramientas afines. Esto se ilustra en la Figura 52.

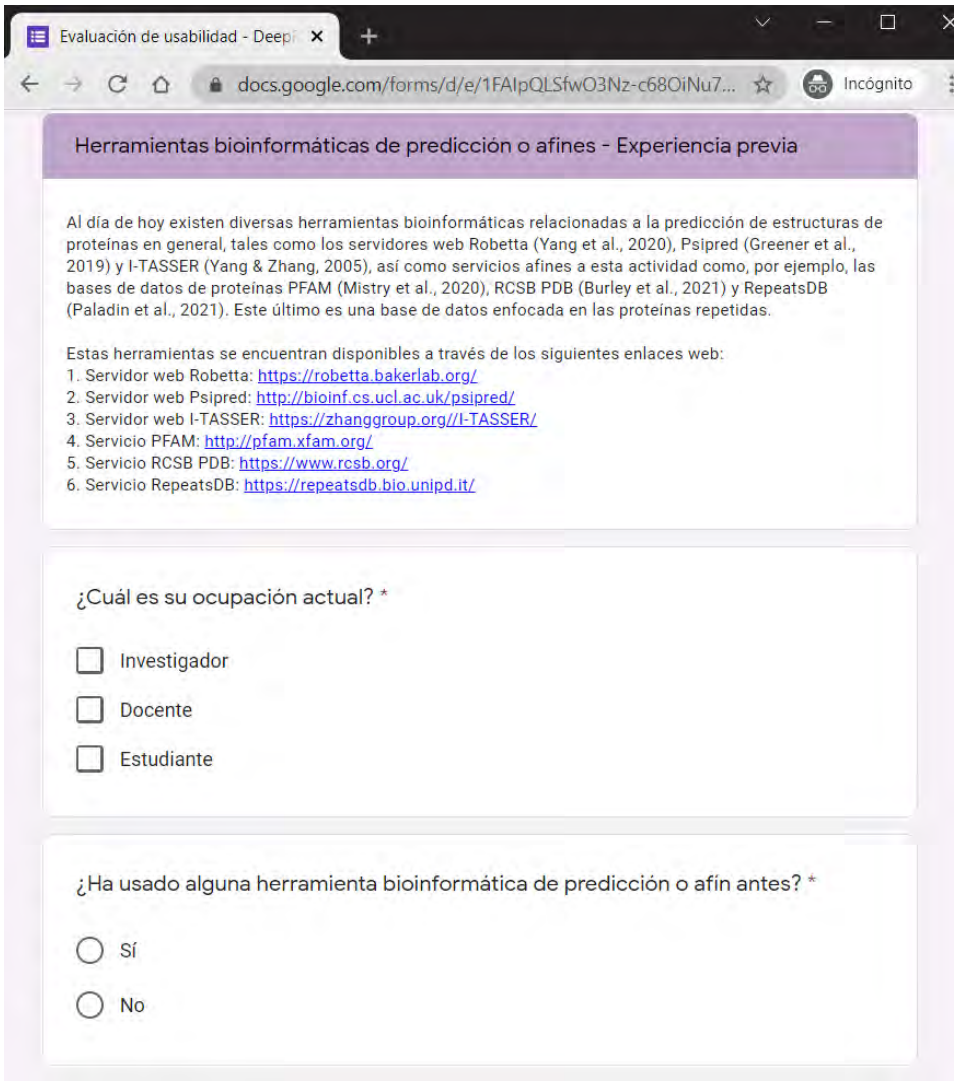

Figura 52. *Formulario de evaluación de usabilidad en DeepReSPred - Segunda sección. (Elaboración propia en Google Forms).*

En la tercera sección del formulario, tal y como se observa en la Figura 53, se presentan al usuario las diez afirmaciones del cuestionario propuesto en el proyecto de Usabilidad en servicios web bioinformáticos (Bezerra Brandao Corrales et al., 2020).

Antes de solicitar la resolución del formulario se presenta al instrumento de evaluación de usabilidad que se está utilizando y se incentiva al usuario a dirigirse al servicio web bioinformático en evaluación. Asimismo, se hace de su conocimiento un pequeño flujo que se espera que el usuario pueda llevar a cabo en la herramienta. Así, el usuario tendrá un objetivo y su percepción se evaluará en torno a ello.

Dado que la interfaz de DeepReSPred no cuenta con muchos flujos de actividades se ha solicitado al usuario evaluador que realice el registro de una solicitud de predicción de la estructura terciaria de una proteína repetida. Seguidamente, en caso el procesamiento se haya realizado exitosamente, el usuario deberá consultar por el estado de la solicitud que ha ingresado previamente y visualizar los resultados obtenidos.

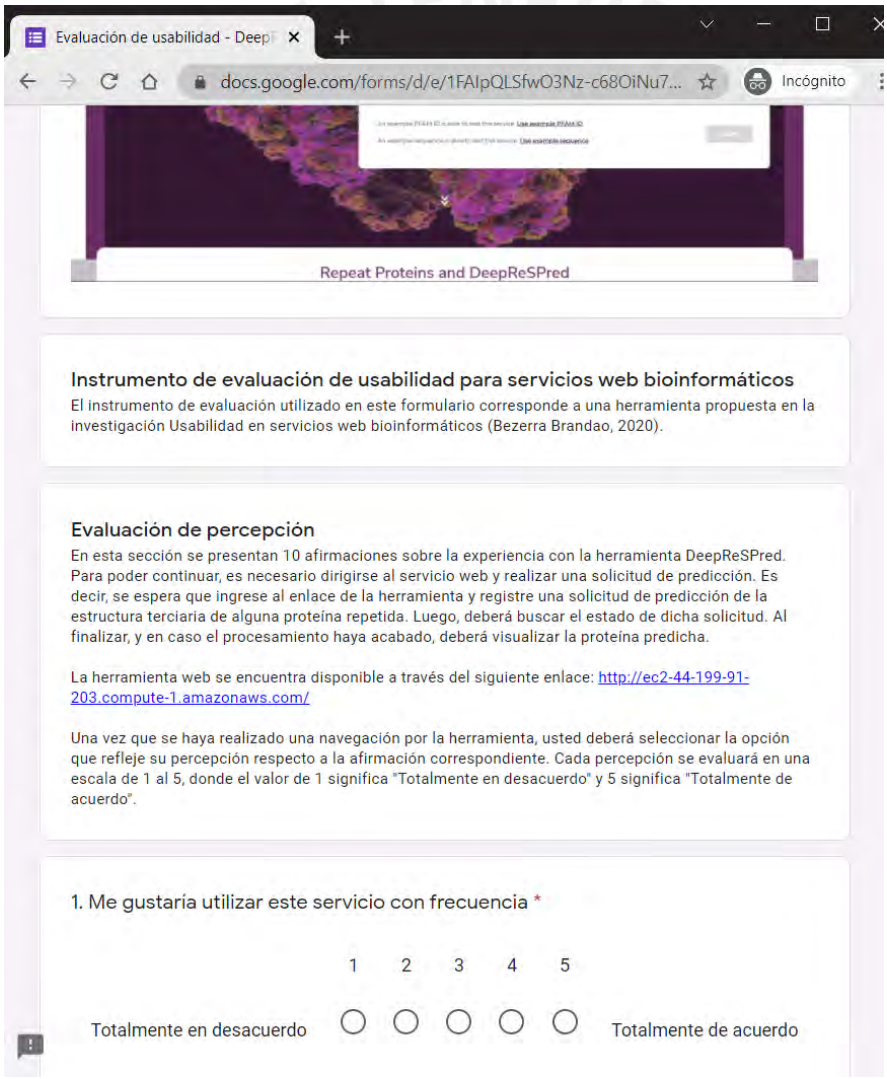

Figura 53. *Formulario de evaluación de usabilidad en DeepReSPred - Tercera sección. (Elaboración propia en Google Forms).*

De acuerdo al cuestionario original, una vez que se hayan obtenido respuestas a las afirmaciones por parte de los evaluadores estas se deberán contabilizar de acuerdo a los siguientes criterios (Bezerra Brandao Corrales et al., 2020):

- Cada afirmación tiene un puntaje del 0 al 4
- Para las afirmaciones 1, 3, 5, 7, y 9 el puntaje asignado es el valor de la posición escalar menos 1
- Para las afirmaciones 2, 4, 6, 8 y 10 el puntaje asignado es 5 menos el valor de la posición escalar
- Se deben sumar todos los puntajes y multiplicar el total obtenido por 2.5.

4. Resultados obtenidos

El formulario fue enviado a diversos usuarios potenciales de la herramienta y a personas pertenecientes al entorno académico en el que se desarrolla este proyecto. Los evaluadores cumplen al menos uno de los perfiles: estudiante, docente e investigador.

Cabe mencionar que la evaluación contempla a usuarios que han tenido relación con herramientas bioinformáticas como a algunas que no, dado que se espera que el resultado refleje una percepción de facilidad de uso y entendimiento de su contenido por parte de un usuario general.

El formulario fue respondido por ocho personas de las cuales siete se identificaron con el perfil de estudiante, una de ellas como docente y dos de ellas como investigadores. Asimismo, dentro del grupo evaluador se obtuvo que el 50% de ellas tuvieron algún tipo de interacción previa con alguna herramienta bioinformática de predicción o afín.

Para poder visualizar los resultados obtenidos de la evaluación se ha elaborado la Tabla 38. Esta presenta la cantidad de personas que han seleccionado una opción por cada afirmación, así como el puntaje promedio obtenido por cada una de ellas. Cada percepción se ha evaluado en una escala de 1 al 5, donde el valor de 1 significa "Totalmente en desacuerdo" y 5 significa "Totalmente de acuerdo".

Resumen de resultados obtenidos del cuestionario de evaluación de usabilidad						
Afirmación	**1**	**2**	**3**	**4**	**5**	**Puntaje promedio**
1. Me gustaría utilizar este servicio con frecuencia	0	0	0	1	7	3.875
2. Encuentro este servicio innecesariamente complicado	7	1	0	0	0	3.875
3. Considero que el servicio fue fácil de usar	0	0	0	1	7	3.875
4. Considero que necesitaría leer mucha documentación para ser capaz de usar el servicio	5	3	0	0	0	3.625
5. Encuentro que muchas funcionalidades en este servicio estuvieron bien integradas	0	0	0	1	7	3.875
6. Pienso que hubo muchas inconsistencias en las opciones de este servicio	4	2	0	1	1	2.875
7. Imagino que muchas personas aprenderían a usar este servicio muy rápidamente	0	0	0	1	7	3.875
8. Encuentro a este servicio muy incómodo de usar	6	2	0	0	0	3.750
9. Me sentí muy seguro al usar el servicio	0	0	1	1	6	3.625
10. Necesité aprender muchas intrucciones/opciones antes de poder comenzar con este servicio	4	2	0	2	0	3.000

Tabla 38. *Resumen de resultados obtenidos del cuestionario de evaluación de usabilidad*

El puntaje promedio ha sido calculado en base a los criterios de contabilización de los resultados presentados en el apartado anterior. La suma de cada uno de los valores corresponde a un puntaje total de 36.25 que al ser multiplicado por 2.5 se convierte en 90,625. Este puntaje corresponde al primer grupo SUS.

La Figura 54 muestra una descripción, un adjetivo, un nivel de admisibilidad y un calificativo NPS en base al percentil en el que se encuentra el resultado total de la evaluación de usabilidad realizada.

De acuerdo al grupo SUS al que pertenece la valoración del puntaje total obtenido por la evaluación de usabilidad, la herramienta propuesta en el presente proyecto de tesis

califica para los usuarios evaluados como un servicio web ideal y promotor, con un alto grado de admisibilidad y, en general, con una calificación A+.

SUS	Calificación	Rango Percentil	Adjetivo	Admisibilidad	NPS
84.1 - 100	A+	96 - 100	Ideal	Admisible	Promotor
80.8 - 84.0	A	90 - 95	Excelente	Admisible	Promotor
78.9 - 80.8	A-	85 - 89		Admisible	Promotor
77.2 - 78.8	B+	80 - 84		Admisible	Pasivo
74.1 - 77.1	B	70 - 79		Admisible	Pasivo
72.6 - 74.0	B-	65 - 69		Admisible	Pasivo
71.1 - 72.5	C+	60 - 64	Bueno	Admisible	Pasivo
65.0 - 71.0	C	41 - 59		Neutral	Pasivo
62.7 - 64.9	C-	35 - 40		Neutral	Pasivo
51.7 - 62.6	D	15 - 34	OK	Neutral	Detractor
25.1 - 51.6	F	2 - 14	Pobre	No admisible	Detractor
0 - 25	F	0 - 1.9	Atroz	No admisible	Detractor

Figura 54. *Categorías de percentiles, calificación, adjetivos, admisibilidad y NPS para la descripción de resultados. Obtenido de (Bezerra Brandao Corrales et al., 2020).*

5. Apéndice

En esta sección se presenta el acta de validación del presente reporte de evaluación de usabilidad de la herramienta propuesta.

5.1 Validación del documento por medio de juicio experto

La validación del documento del reporte de la evaluación de usabilidad de la herramienta propuesta ha sido realizada por una experta en usabilidad y bioinformática. La realización de esta verificación implica la revisión completa del informe en tanto este contenga la especificación de la metodología de evaluación llevada a cabo. Asimismo, su aprobación corresponderá al análisis de los resultados obtenidos y la entrega de observaciones en caso se requieran. A continuación, se observa el acta de validación recibida por la experta.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Acta de validación de documento

Título de tesis: Desarrollo de una herramienta para la predicción de estructuras terciarias de proteínas repetidas a partir de su estructura primaria

Tesista: Solange Estrella Palomino Chahua

Nombre del documento: Reporte de evaluación de usabilidad de la herramienta implementada

Descripción del documento: Documento que contiene la descripción de la evaluación de usabilidad de la herramienta implementada y los resultados obtenidos

Mediante la presente acta, yo, la Dra. Layla Hirsh Martinez, dejo constancia de que, en mi calidad de experta en bioinformática, he revisado el documento descrito en los puntos anteriores perteneciente al proyecto de tesis en mención. En ese sentido, en la siguiente sección se especifica el veredicto y, en caso hubiesen, las observaciones correspondientes al documento.

Veredicto:

(X) Aprobado ( ) Requiere observación

Observaciones:

Lima, 12 de noviembre de 2021

Firma