UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA

MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos

T E S I S

Que para obtener el Grado de:

MAESTRA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

Presenta

ERIKA LIZBETH MONTIEL RUIZ

Asesora interna

Dra. Lorena Díaz González

Asesora externa

Dra. Blanca Itzelt Taboada Ramírez

Revisores:

Dra. Lorena Díaz González

Dra. Blanca Itzelt Taboada Ramírez

Dr. José Alberto Hernández Aguilar

Dr. Jorge Hermosillo Valadez

Dr. Mauricio Rosales Rivera

CUERNAVACA, MORELOS NOVIEMBRE, 2020

Cuernavaca, Morelos a 27 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado de la estudiante Erika Lizbeth Montiel Ruiz, con matrícula 10023096, con el título **Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dra. Lorena Díaz González
Profesora- Investigadora
Centro de Investigación en Ciencias

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

LORENA DIAZ GONZALEZ | Fecha:2020-11-27 16:12:11 | Firmante

Bm/ACO35y6nj6VzJ0XljSdPjilvbKS0ejQ9FgK+5Qv102Uz9DuL3PH9YpQcDag+5f4QQzzRAMkJ3Vf+wvJvcFocnNlK26iPtVxWsB3Ao89bSgQbZIVq5wHdmfDlmYJxCUlusjx7XKm98Y9I87fGb/kiVFgvxVp8NO4Q7oqbJpVuGlR52wDGR3PHRxCoEt+28gmyWtB+RUiNWDhGLKe22hybGvyk9tOPvFOwJ8U71BxY55/3lAT9Z8GF+QeCTxpfd35U39x4P/JpD6AQtC40lQvxtNPBC2+eelFrdnhNlIuxSD7TwHZ1W2Zw3/0fn48DaGEifJTdN6ZkmcOUVW3Yz8w==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:

tYBJoF

https://efirma.uaem.mx/noRepudio/w6VgYfYkyAvaGHIO6oMxXQqjsAYrP36v

Cuernavaca, Morelos a 04 de diciembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAel
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Erika Lizbeth Montiel Ruiz, con matrícula 10023096, con el título **Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dra. Blanca Itzelt Taboada Ramírez
Profesor- Investigador
Instituto de Biotecnología UNAM

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

BLANCA ITZELT TABOADA RAMÍREZ | Fecha:2020-12-10 09:48:33 | Firmante

ZIqo/OWnhORD4qUzg3teR3JRwv5ChClXjRRFiHWS6OKiGy5rTB8BMqec9JrrNeoRgbLivlYF1KY8Cl70qC5O0ClZfiBY6zB7f9eNCdhc0+nI9jIoHZg/oWOShwV9uRkjCx5f8ZviNtzT0drWj7nJmBW1SfSjAV/uG6H8GcyobpJ8sFqDPB43F2uO1DZQ063FnzIEhK5QsJ2LNnki+YmyMszmXMZykFt2wXmIJLh3ktuz0M1CSsDYdDqX7y1VJmG+aMZxQS1rE59V0ZpI+VMSSG+xK10tKtQYNURxYUVFIqsA4NvbpYjeF9b32PBeBsivme5zkkjEtSeukoclYV6aOA==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:

SOvULW

https://efirma.uaem.mx/noRepudio/qztj6Vgats2FmtlwxokrYsBJdjB9GN9s

Cuernavaca, Morelos a 02 de diciembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAel
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Erika Lizbeth Montiel Ruiz, con matrícula 10023096, con el título **Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Jorge Hermosillo Valadez
Profesor- Investigador
Centro de Investigación en Ciencias

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

JORGE HERMOSILLO VALADEZ | Fecha:2020-12-02 15:04:39 | Firmante

THZY7EuWja5IdQ/LbMLBdeYnV6wVKdCPdC/IwNyv7AtEYyXaDQmlfqDQKKbF1ueuweMEgoW+LNr2DvQ7oQET8CMFlJte5FkHS7V7AwQOfxhEst0GqCBVDG1E71dBeD2XeDmS3kFlW+8e0bG+7P3scG3fRezqP+37Bgkm1im4IBjW1mM3Vs2UYUpGrEw74ekbBHYCkcmZeDBQxlK4VRSfQ+Ghi/0761NhycRi1hDIzo/KHmriN5lP/vu3zeX6FcGFXLYSAzoDV/02+BTKQRUUQkFFcGDskG892S2AtvD1gXxVZisKu/f4vzDXl5M0uEtrGyREIdliLCVZtXFZaxDvJg==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:

Dl6Xrn

https://efirma.uaem.mx/noRepudio/8BggX59trDf4QC17InIZiQmd3ng7Dz1Y

Cuernavaca, Morelos a 10 de Diciembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Erika Lizbeth Montiel Ruiz, con matrícula 10023096, con el título **Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. José Alberto Hernández
Profesor- Investigador
Facultad de Contaduría, Administración e Informática

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

JOSE ALBERTO HERNANDEZ AGUILAR | Fecha:2020-12-10 13:54:21 | Firmante

gk0SrgrBy6sLoo80FTMu2Ot826M25VwyK30+495tXKXHIS1Xuyk9iv/16O4+1GpqUggwF7ebEmkf2dbKmsdJWHa3DVGpAs/J1AsppHzoRWS5V95JVcT4i7m0tYRWB9QfcLAvSb9RH+lflit4e8teWtNOhUTbCsZyNRuE41yibR4OzJZFzcr0xAc1uTopXVYZGKjmg4N1rOhbI2oSD7vRa0Y/RyYcDHEgdHMCIleSXUSeJ6UhktW9lDAjRNg+Ti1oqv9LQPaRcpWYDqtb+ENuOZpNSIIN0BGLDvM0pYZl5ciafv814Y1BaZGCnqwEzX4fgT+uGiylVw0fC17L73wt9Q==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:

pbeRc4

https://efirma.uaem.mx/noRepudio/qGXWpTCoBfh8SsvnohGXZmnA799CRbOT

Cuernavaca, Morelos a __29_de __Noviembre__ del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Erika Lizbeth Montiel Ruiz, con matrícula 10023096, con el título **Desarrollo de una herramienta computacional para la identificación de nuevos genomas virales en estudios metagenómicos** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Mauricio Rosales Rivera
Profesor- Investigador
(adscripción temporal)
Centro de Investigación en Ciencias

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MAURICIO ROSALES RIVERA | Fecha:2020-11-29 11:59:31 | Firmante

XQeFTz+RLHr8h2JJ3mAtRYpDVsQtEKVFBAEgCzmxT6l69Mpa7p4tkuA5vUJBYrbZMp98cExK953Up3rDXtZHdUeAjx3WC31w6L0x231p8WneFDrRUy+FTn7chqqaI+fZaAKNlOrzl0xeW+hbdzYeFL4D38v2FYbcnqo70W5FDWUUFecYPeLcs/pLD50O5Bv3B5oyHtyG+hxan9G5QHHKKOyV8Mfj9ocWk9Wn3B6G61fbjf4ULN6X0ejrzrH4kH0owFndP0V2hi0M45O2a+3i6C52+9GuByHrO+yHgvRF1S1tubm9ClL3zvsTWmphW2Rozi8pGTJmPZnRmu5fBAOWQw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:

EygTqc

https://efirma.uaem.mx/noRepudio/FZ0SUqWcZ495fsWBpWf23Bs9DBjZ3i97

Agradecimientos

Agradezco ante todo a Dios por darme la vida, salud y fortaleza para cumplir mis sueños en la vida.

Gracias a mi familia: Chris, Sophie y Migue que me han apoyado y donado mucho de su tiempo para cumplir esta meta.

A mis padres Eli y Mariano que siempre han confiado en mí y me han apoyado en todo momento. Una gran bendición tenerlos como padres.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca otorgada para realización de mi maestría.

A la Universidad Autónoma del Estado de Morelos UAEM, la Facultad de Contaduría, Administración e Informática, al Centro de Investigación en Ciencias por el apoyo brindado durante mi estancia como alumna

A mis asesoras la Dra. Blanca Taboada Ramírez y la Dra. Lorena Díaz González por la confianza brindada para desarrollar este trabajo.

A mi compañera de posgrado y amiga Elizabeth Cadenas Castrejón, por compartir sus conocimientos.

Agradezco de manera muy especial el soporte computacional brindando por Jerome Veleyen, que siempre estuvo disponible y al pendiente de mis dudas, configuraciones, compilaciones y demás issues, además de compartir siempre sus conocimientos para aprovechar mejor los recursos en el Clúster Teopanzolco de la UNAM.

Se agradece los recursos computacionales otorgados, en la supercomputadora MIZTLI, por la Dirección General de Cómputo y de Tecnologías de Información y Comunicación (DGTIC) de la UNAM, mediante los proyectos LANCAD-UNAM-DGTIC-350 y UNAM-DGTIC-COVID-011. También a Juan Manuel Hurtado Ramírez y Roberto Pablo Rodríguez Bahena del Instituto de Biotecnología de la UNAM.

Resumen

Hoy en día, gracias a las tecnologías de secuenciación masiva de nueva generación, ha surgido la metagenómica la cual es una herramienta que permite obtener de una manera no sesgada millones de secuencias de ADN de los diferentes microorganismos que componen una comunidad microbiana de una muestra, sin necesidad de aislarlos o cultivarlos *(Kumar et al., 2015)*.

Existe una cantidad inmensa de estudios metagenómicos virales. Sin embargo, la efectiva identificación de los diversos virus, así como su clasificación taxonómica a nivel de especie, género y/o familia de los datos generados sigue siendo un desafío computacional.

Primero, porque se requiere de un gran número de recursos computacionales, con suficiente capacidad de memoria para el procesamiento de millones de secuencias de ADN por los algoritmos implementados en estas herramientas de análisis bioinformático. Segundo, gran parte de estas secuencias no pueden ser identificadas por los clasificadores taxonómicos y dependiendo del tipo de muestra, queda entre un 20% a un 90% de secuencias sin clasificar.

Este panorama proporciona una oportunidad única para descubrir nuevas secuencias virales utilizando modelos computacionales basados en aprendizaje profundo.

En este contexto, este trabajo presenta un modelo de predicción de aprendizaje profundo basado en una red neuronal convolucional residual, para la identificación de proteínas virales en secuencias metagenómicas que pueda ser aplicado como una herramienta de análisis bioinformático para aquellas lecturas de secuencias que quedan sin clasificar por las herramientas de clasificación taxonómico actuales, las cuales podrían representar nuevos virus, aún sin anotarse, o con baja identidad a los conocidos.

Los resultados obtenidos muestran que los métodos de aprendizaje profundo requieren una gran cantidad de ejemplos para mejorar la precisión y al estar poco anotados los virus, no pueden competir con las herramientas convencionales como Blastp o Blastx que permiten identificar secuencias virales hasta con un porcentaje de identidad del 30% con secuencias de referencia. Sin embargo, el modelo propuesto se puede utilizar como un modelo base que puede mejorarse para el propósito que fue desarrollado.

Contenido

Índice de figuras

Índice de tablas

CAPITULO 1. Introducción

Los microrganismos son los seres vivos más pequeños que sólo pueden visualizarse a través de un microscopio. Ellos están presentes en todos los ambientes que habitamos y son esenciales para la vida en el planeta. Existen los siguientes tipos de microrganismos: bacterias, arqueas, protozoos, hongos y **virus**. Estos últimos, son organismos que están formados de ARN (ácido ribonucleico) o ADN (ácido desoxirribonucleico), y se consideran parásitos obligados debido a que requieren la maquinaria celular de su huésped para poder reproducirse. Los virus también son las entidades biológicas más abundantes de la Tierra. Por ejemplo, los fagos, virus de bacterias, son un importante depósito de diversidad genética en los océanos que afectan los ciclos biogeoquímicos, y por ende la dinámica de dichos ecosistemas. Se estima que hay 10^{31} partículas virales infectando poblaciones microbianas. Sin embargo, los virus siguen siendo los grandes desconocidos del mundo. Mientras que existen unos 45,000 genomas bacterianos depositados en la base de datos del NCBI (National Center for Biotechnology Information), solo hay unos 2,000 genomas virales *(Páez-Espino et. al. 2016)*.

En los últimos años se han introducido una serie de herramientas de secuenciación de ADN de alto rendimiento basados en diferentes técnicas de detección. Estos son llamados tecnologías de Secuenciación de Nueva Generación (NGS, por sus siglas en inglés) que generan millones de secuencias cortas de ADN, denominadas lecturas (reads), en un par de días y a un costo relativamente bajo.

Los recientes avances tecnológicos en las NGS han permitido el ensamble de nuevos genomas, realizar estudios de genética, la identificación de genomas microbianos dentro de una comunidad, entre otras aplicaciones.

Estas tecnologías de secuenciación han generado un aumento masivo de los datos en bruto, sin embargo, hay una serie de nuevos retos y dificultades informáticas que deben abordarse para mejorar el estado actual de su análisis.

Por ejemplo, una de las aplicaciones de NGS es la metagenómica viral, la cual busca obtener de manera global, todas las secuencias de ADN de los diferentes virus que están presentes en una muestra dada, sin necesidad de cultivarlos o aislarlos. Se pueden resolver preguntas como: ¿Qué virus están en la muestra?, ¿Qué hacen?, es decir entender la composición/estructura de la comunidad estudiada, incluyendo: el desglose taxonómico y el relativo a la abundancia de las diversas especies identificadas, la riqueza y diversidad, así como funciones que se puedan estar llevando. Todos estos aspectos son los objetivos primordiales de un estudio de metagenómica *(Scholz et al., 2012)*.

El tiempo de análisis de los datos NGS es costoso computacionalmente y la contextualización de los resultados (geográfica, temporal, prevalencia, condiciones ambientales, condiciones funcionales, entre otros) requiere de modelos estadísticos complejos.

En el caso específico de la metagenómica viral, a pesar de obtener millones de secuencias, las virales son poco abundantes, ya que el ADN viral representa del 1 al 5% del ADN total en la muestra. También, una gran proporción de las bacterias hospederas de los fagos, no se pueden cultivar en el laboratorio, por lo tanto, no se pueden aislar y/o estudiar los virus asociados a éstas. No existen genes marcadores, como en las bacterias que permitan caracterizarlos fácilmente. En este sentido, los clasificadores metagenómicos virales arrojan una cantidad importante de datos sin clasificar que van del 20% hasta el 90% dependiendo del tipo de muestra *(Mande et al. 2012)*.

Estos resultados son en parte porque los métodos computacionales existentes están basados en alineamientos o búsquedas por similitud con genomas de referencia de bases de datos, lo cual solo permite identificar virus conocidos hasta el momento o de gran similitud con lo anotado, pero no son eficientes en la identificación de virus desconocidos. En el conjunto de datos sin clasificar, se espera que una fracción importante corresponda a virus nuevos, no anotados hasta el momento, así como otros tipos de microorganismos y de "materia negra biológica".
Lo anteriormente expuesto, motivó el desarrollo de este trabajo de investigación, el cual se basa en la implementación de una herramienta computacional que permita el análisis de las lecturas de ADN que los clasificadores taxonómicos no lograron identificar, a fin de detectar nuevas proteínas virales y contribuir hacia la identificación de nuevos genomas virales en estudios metagenómicos.

1.1. Hipótesis

El uso de modelos de aprendizaje profundo proporcionará un método de identificación viral de aquellas lecturas de secuencias de ADN, generadas por tecnologías de secuenciación masiva, que hasta el momento no se les ha identificado homólogos en bases de datos de referencia por métodos computacionales clásicos, debido a su capacidad de aprender y extraer las características más importantes.

1.2. Objetivo

Desarrollar una herramienta computacional basada en un modelo de aprendizaje profundo (red de convolución residual) para el análisis de lecturas metagenómicas virales, que quedan sin clasificar por métodos computacionales convencionales, para detectar posibles nuevos virus, mediante la clasificación de proteínas virales.

1.3. Objetivos específicos

- Obtener, depurar y quitar redundancia de las secuencias de la base de datos de GenBank de proteínas virales a fin de generar el conjunto de datos de entrenamiento y definir las clases del modelo.
- Diseñar un modelo de aprendizaje profundo para la clasificación de secuencias de fragmentos de proteínas virales.
- Evaluar el desempeño de clasificación del modelo de aprendizaje profundo.
- Implementar una herramienta para aplicar del modelo de aprendizaje profundo diseñado

1.4. Organización de la tesis

Este trabajo de tesis está organizado de la siguiente manera:

- En el capítulo dos, se expone el marco de referencia y los conceptos fundamentales del área biológica, así como los conceptos del área computacional utilizados.
- En el capítulo tres se describe la metodología desarrollada para el diseño del modelo que permite la identificación de fragmentos de proteínas virales en secuencias metagenómicas.
- El capítulo cuatro se presentan los resultados de la definición del conjunto de datos de entrenamiento y las clases obtenidas, la herramienta computacional implementada del modelo, así como la discusión y algunas conjeturas realizadas a partir de su evaluación.
- Para finalizar, en el capítulo cinco, se dan las conclusiones de este trabajo de tesis, y trabajos futuros.

CAPITULO 2. Marco teórico y antecedentes

2.1. Conceptos biológicos

En este capítulo se exponen los conceptos biológicos básicos utilizados en este trabajo de tesis, a fin de lograr una mayor comprensión del tema desarrollado para aquellos ajenos al estudio de las ciencias biológicas.

2.1.1. Ácido Desoxirribonucleico (ADN)

ADN es el nombre químico de la molécula que contiene la información genética que sirve para el desarrollo y funcionamiento de todos los organismos vivos, y también de aquellas formas no celulares como los virus. La molécula de ADN consiste en dos cadenas que se enrollan entre ellas para formar una estructura de doble hélice; está formado por componentes químicos básicos denominados nucleótidos. Los nucleótidos incluyen un grupo fosfato, un grupo de azúcar y una de cuatro tipos de bases nitrogenadas alternativas, las cuales son: Adenina (A), Tiamina o Uracilo (T), Citosina (C), y G(Guanina).

2.1.2. Gen

La información del ADN es codificada en bloques discretos, llamados genes. Un gen es la unidad física y funcional básica de la herencia y algunos de estos actúan como instrucciones para sintetizar moléculas llamadas proteínas. Un gen está formado por una larga cadena de nucleótidos (ADN), y algunas veces por partes discontinuas denominadas **exones e intrones**. Los exones son las regiones codificantes que van a proporcionar la información para la síntesis de una proteína, mientras que los intrones son regiones no codificantes, que se hallan intercaladas en el gen y tienen otras funciones. Para que la información de un gen sea "leída" por la célula, la información del ADN es copiada en moléculas de ácido ribonucleico (ARN), en un proceso llamado transcripción. Desde el punto de vista computacional, el ADN y el ARN pueden verse como una combinación de cuatro caracteres diferentes A, G, C, T (la T cambia a U -uracilo-, en el caso de ARN), los cuales se utilizan para realizar la transcripción del ADN al ARN por medio de un lenguaje determinado.

2.1.3. Genoma

El conjunto de genes y de regiones no codificantes de un organismo forma su genoma. En otras palabras, el genoma es el manual que contiene todas las instrucciones para conocer el funcionamiento y crecimiento de un organismo.

2.1.4. Proteína

Una vez que se transcribe el ADN a ARN, se realiza la síntesis de proteínas, en un proceso denominado traducción. Las proteínas son moléculas grandes y complejas que desempeñan diversas funciones básicas para la vida en un organismo, tales como catálisis, reguladoras, estructurales, motoras entre otras. Se componen de cientos o miles de unidades más pequeñas llamadas aminoácidos (20) que se unen entre sí en largas cadenas. Las diferentes combinaciones de tres nucleótidos (codones) determinan un determinado aminoácido de la proteína (Tabla 1) el cual tiene un nombre y una letra que lo representa (Tabla 2). Cada proteína tiene su propia secuencia de aminoácidos que proviene de la secuencia de nucleótidos del gen que la codifica.

Tercera letra											
Primera letra		U		C		A		G			**Segunda letra**
	U	UUU	**Phe (F)**	UCU	**Ser (S)**	UAU	**Tyr(Y)**	UGU	**Cys (C)**	U	
		UUC		UCC		UAC		UGC		C	
		UUA	**Leu (L)**	UCA		UAA	**Término**	UGA	**Término**	A	
		UUG		UCG		UAG	**Término**	UGG	**Trp (W)**	G	
	C	CUU	**Leu (L)**	CCU	**Pro (P)**	CAU	**His (H)**	CGU	**Arg (R)**	U	
		CUC		CCC		CAC		CGC		C	
		CUA		CCA		CAA	**Gln (Q)**	CGA		A	
		CUG		CCG		CAG		CGG		G	
	A	AUU	**Ile (I)**	ACU	**Thr (T)**	AAU	**Asn (N)**	AGU	**Ser (S)**	U	
		AUC		ACC		AAC		AGC		C	
		AUA		ACA		AAA	**Lys (K)**	AGA	**Arg (R)**	A	
		AUG	**Met (M)**	ACG		AAG		AGG		G	
	G	GUU	**Val (V)**	GCU	**Ala(A)**	GAU	**Asp (D)**	GGU	**Gly (G)**	U	
		GUC		GCC		GAC		GGC		C	
		GUA		GCA		GAA	**Glu (E)**	GGA		A	
		GUG		GCG		GAG		GGG		G	

Tabla 1. Codones que codifican los aminoácidos (código de tres y una letra)

Aminoácidos					
V	Val	Valina	**A**	Ala	Alanina
L	Leu	Leucina	**P**	Pro	Prolina
T	Thr	Treonina	**G**	Gly	Glicina
K	Lys	Lisina	**S**	Ser	Serina
W	Trp	Triptófano	**C**	Cys	Cisteína
H	His	Histidina	**N**	Asn	Asparagina
F	Phe	Fenilalanina	**Q**	Gln	Glutamina
I	Ile	Isoleucina	**Y**	Tyr	Tirosina
R	Arg	Arginina	**D**	Asp	Ácido aspártico
M	Met	Metionina	**E**	Glu	Ácido glutámico

Tabla 2. Nombre y código de 1 y 3 letras de los 20 aminoácidos

2.1.5. Secuenciación de ADN

La secuenciación del ADN es un conjunto de métodos y técnicas cuya finalidad es la determinación del orden de los nucleótidos (A, C, G y T) en un oligonucleótido de ADN.

2.1.6. Secuenciación masiva o secuenciación de Nueva Generación (NGS)

Es un conjunto de técnicas que permiten descifrar la información genética desde una única célula hasta comunidades muy complejas. Estas técnicas utilizan procesamiento paralelo masivo lo cual permite generar millones de secuencias cortas de entre 75-1000 bases de nucleótidos, comúnmente llamadas lecturas, en tiempos muy cortos y a un costo relativamente bajo. Algunos ejemplos de tecnologías de este tipo de secuenciación son: Illumina (Solexa), Roche 454, Ion torrent: Proton/PGM y SOLiD.
El uso de NGS ha aumentado debido a la disminución de costos (figura 1A) y por su gran utilidad en diversas áreas como biología, genética, salud, biorremediación, entre otras. Como resultado, se han generado grandes volúmenes de datos que requieren de herramientas específicas para su análisis *(Beerenwinkel et al., 2012)*. En la figura 1B se puede observar un panorama sobre cómo el desarrollo y las mejoras de las tecnologías de secuenciación de ADN, han contribuido al crecimiento exponencial de datos genómicos y su utilización en diversas áreas. Por último, en la figura 1C, se muestra una proyección del crecimiento de la base de datos GenBank, la cual es una de las bases de datos genómicas más utilizadas, de acceso abierto y forma parte de un consorcio internacional donde participan diversas instituciones de investigación: National Center for Biotechnology (NCBI), European Nucleotide Archive (ENA), y el Data Bank de Japón (DDBJ).

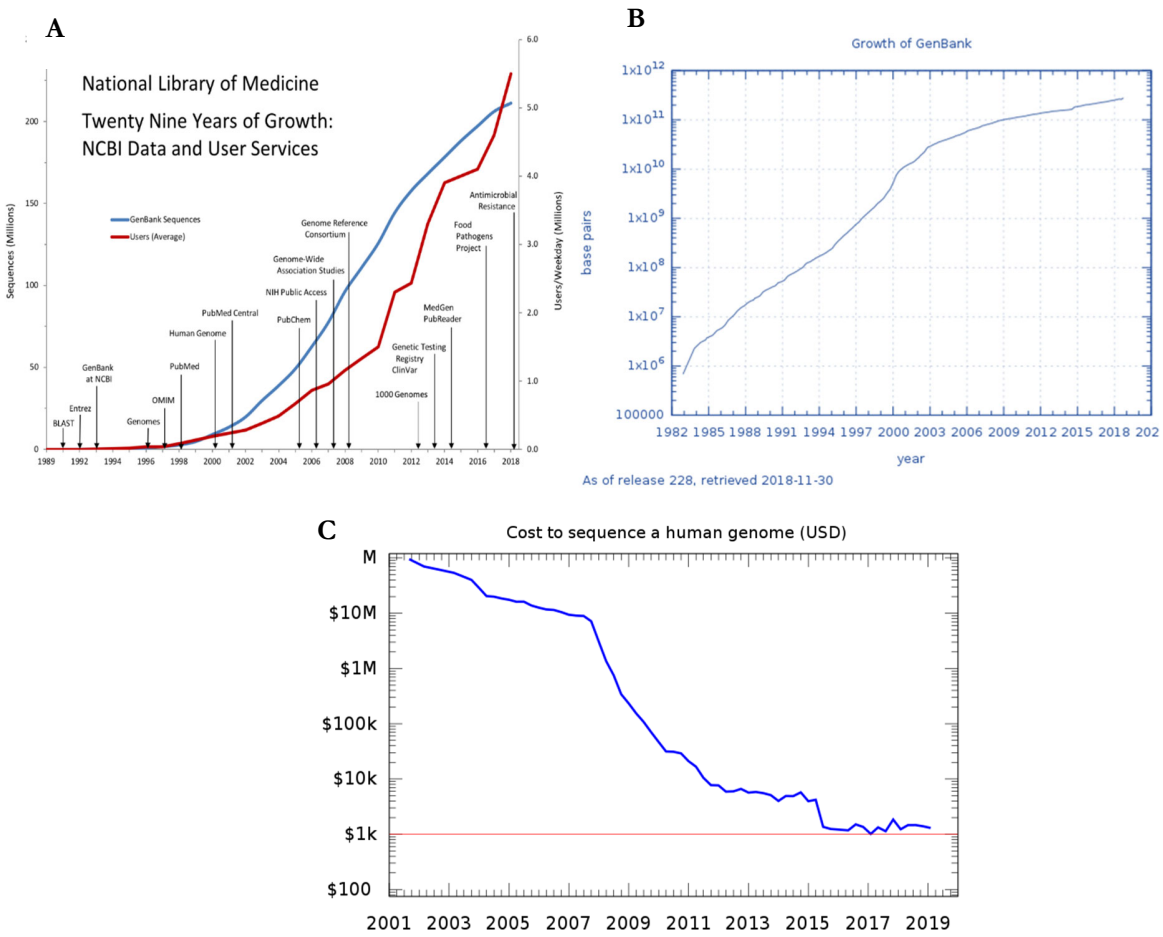

Figura 1. Crecimiento de la información genómica. ***A)*** *Gráfica de crecimiento de datos secuenciados y usuarios. Figura tomada de: https://www.nlm.nih.gov/about/2019CJ.html.* ***B)*** *Crecimiento de la base de datos GenBank. Figura tomada de: https://en.wikipedia.org/wiki/File:Growth_of_GenBank.svg,* ***C)*** *Costo por Genoma Humano. Figura tomada de: https://es.m.wikipedia.org/wiki/Archivo:Historic_cost_of_sequencing_a_human_genome.svg*

2.1.7. Ensamble de fragmentos de secuencias de ADN (lecturas).

El ensamblaje de secuencias se refiere a fusionar fragmentos de secuencias de ADN en secuencias más largas. Esto es necesario ya que la tecnología de secuenciación de ADN no puede leer genomas completos de una sola vez, sino que lee pequeños fragmentos. Sin embargo, las lecturas al ser muy pequeñas son poco precisas y pudieran pertenecer a más de un organismo por lo que se hace necesario ensamblarlos en fragmentos más grandes llamados contigs que contiene más información.

2.1.8. Metagenómica

"Metagenómica" está formada por dos palabras "meta" y "genómica". La genómica es la obtención de la secuencia de ADN y meta implica que se está haciendo al mismo tiempo para muchos organismos. Por lo tanto, la metagenómica es una de las aplicaciones de la NGS, que permite el estudio de la información genética de todos los microorganismos que se encuentran en una comunidad o muestra ambiental, sin necesidad de aislarlos y cultivarlos *(Kumar et al., 2015)*. Esta herramienta nos permite estudiar poblaciones de organismos a fin de entender cuál es el papel que desempeñan en el medio donde se encuentran. En este trabajo de tesis, la información que se analizará será el resultado de los datos generados para este tipo de estudios metagenómicos virales, centrándonos en los datos que un clasificador taxonómico no logre analizar.

La figura 2, describe el proceso general de un estudio de metagenómica, el cual inicia con una muestra ambiental, que puede ser de suelo, agua, sedimento, especímenes acuáticos, terrestres o bénticos, humanas. Una vez obtenida la muestra, se extrae el material genético *(Kumar et al. 2015)*. El ADN es fragmentado en trozos de una longitud conocida, utilizando enzimas a fin de crear librerías de ADN (copias de los fragmentos por reacción en cadena de la polimerasa, conocida como PCR por sus siglas en inglés (Polymerase Chain Reaction). Las librerías de ADN son procesadas por un secuenciador NGS, los cuales generan millones de "reads o lecturas de ADN", de una longitud aproximada de 75 a 1000 nucleótidos (bases) dependiendo la tecnología que se utilice. Estas lecturas o fragmentos obtenidos del secuenciador requieren ser ensamblados, tal si se tratará de armar un rompecabezas. Sin embargo, las lecturas al ser muy pequeñas son poco precisas por lo que se hace necesario ensamblarlos en fragmentos más grandes llamados contigs. Finalmente, después del ensamble, comienza el análisis con diversas herramientas bioinformáticas entre ellas los clasificadores taxonómicos.

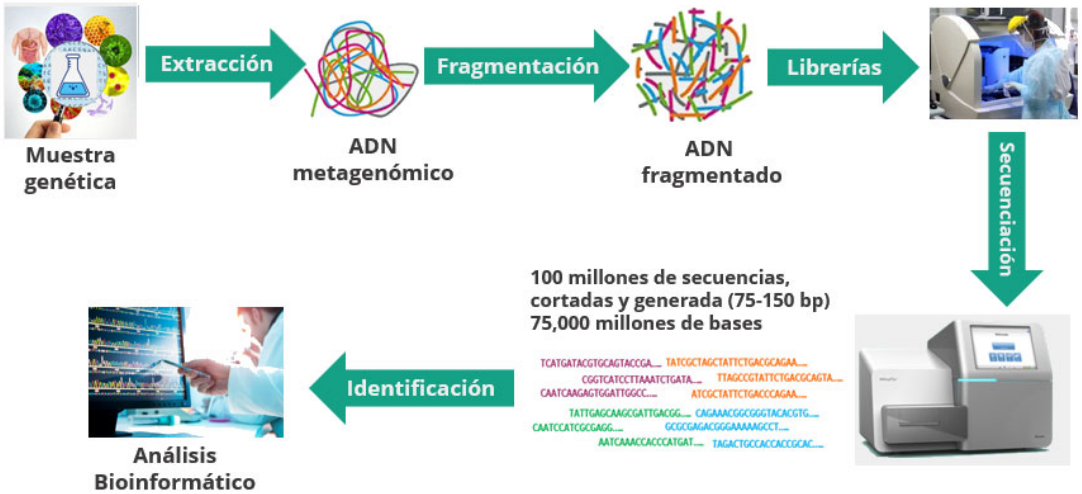

Figura 2. Esquema del proceso que se realiza en estudios metagenómicos

2.2. Conceptos computacionales

A continuación, se exponen los conceptos generales del área de aprendizaje automático y aprendizaje profundo utilizados en este trabajo de tesis a fin de lograr una mayor comprensión del objetivo de este trabajo de estudio para aquellos ajenos al estudio de las ciencias computacionales.

2.2.1. Inteligencia artificial

La inteligencia artificial, conocida por sus siglas en inglés (AI), es una rama de las ciencias computacionales, que tiene como actividad principal construir máquinas o sistemas inteligentes capaces de realizar tareas que generalmente requiere inteligencia humana. Es una ciencia interdisciplinaria con varios enfoques, unas de sus subramas es el aprendizaje automático.

2.2.2. Aprendizaje de máquina o automático (Machine Learning)

Es un método de análisis de datos que automatiza la construcción de modelos, los cuales utilizan métodos estadísticos y algoritmos a fin de identificar patrones e inferencias para resolver un problema con mínima intervención humana *(Angermueller et al., 2016)*. La mayoría de estos modelos se pueden definir a través del siguiente flujo de trabajo que consta de cuatro pasos: i) preprocesamiento, ii) definición de las características por parte del diseñador del modelo, iii) ajuste y iv) evaluación del modelo.

2.2.3. Aprendizaje profundo (Deep Learning)

El aprendizaje profundo es un tipo de algoritmo de aprendizaje automático basados en redes neuronales artificiales. Está inspirado en las redes neuronales biológicas que transforman la entrada procesando señales a través de múltiples capas de neuronas, para obtener una salida. Gracias, al desarrollo de poder de cómputo para procesar información a gran escala (big data) el aprendizaje profundo ha generado muy buenos resultados en aplicaciones de visión por computadora, reconocimiento de voz y procesamiento de lenguaje natural (Angermueller et al., 2016). En biología computacional se han desarrollados métodos de aprendizaje profundo para resolver diferentes problemas como predecir la especificidad de secuencia de la unión a proteínas, identificación de secuencias conservadas evolutivamente e identificación del potenciador y promotor *(Angermueller et al., 2016)*. Estos métodos han mostrado mejoras notables sobre aquellos basado en el aprendizaje automático y modelos basados en inferencias estadísticas. En la figura 3, se puede observar en resumen la relación que existe entre los tres conceptos descritos anteriormente.

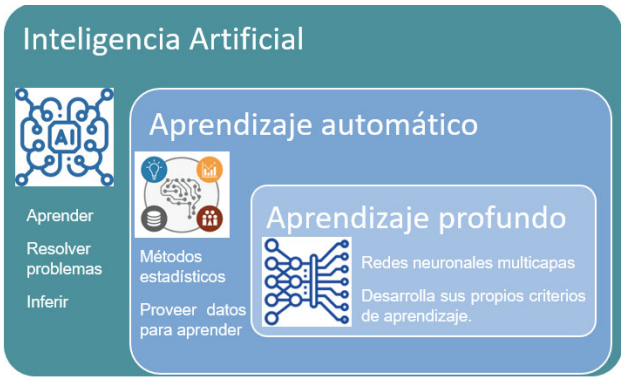

Figura 3. Inteligencia artificial y sus subramas

2.2.4. Redes neuronales artificiales

Las redes neuronales artificiales están basadas en el funcionamiento de las redes de neuronas biológicas. En una red neuronal artificial este comportamiento biológico se imita creando un sistema de interconexión en capas de neuronas artificiales (red neuronal multicapa) que colaboran para procesar datos de entrada y generar salidas o dicho en otras palabras para clasificar o hacer predicciones *(Zou et al., 2019)*. En la figura 4, se muestra cómo se imita este comportamiento de una neurona biológica (figura 4A) con una neurona artificial (figura 4B) donde el canal de entrada son las **dendritas, la sinapsis** representan los pesos, el procesador o cuerpo el **soma** y el **axón** el canal de salida.

A

B

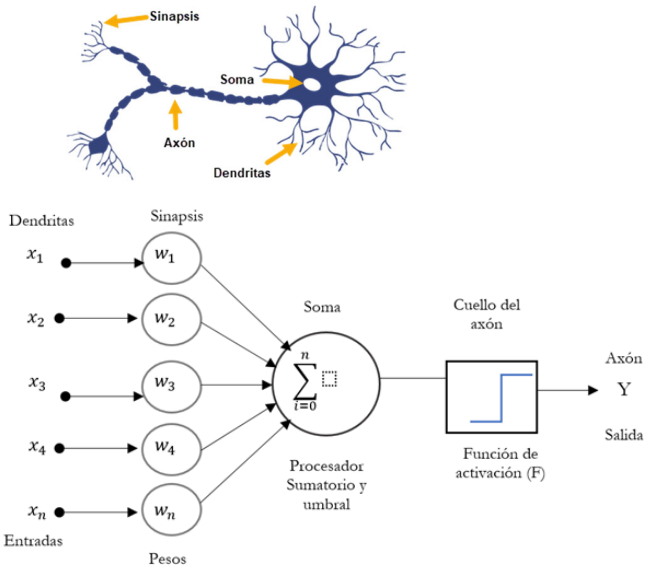

Figura 4. Esquemas de neuronas. A) Biológica. B) Artificial
Figura modificada de:(Lao, Y. et al,2017)

De la figura 4B, podemos definir los siguientes elementos básicos de una red de una sola capa.

1. **Conjunto de entradas (x_1, x_2, x_3, x_4)**. Representan las entradas, de la red neuronal.
2. **Pesos sinápticos (w_1, w_2, w_3, w_4)**. Estos se van ajustando de forma automática a medida que la red neuronal va aprendiendo.
3. **Función de agregación, Σ.** Realiza el sumatorio de todas las entradas ponderadas por sus pesos.
4. **Función de activación, F.** Se encarga de mantener el conjunto de valores de salida en un rango determinado, normalmente (0,1) o (-1,1). Existen diferentes funciones de activación que cumplen este objetivo, la más habitual es la función sigmoide.
5. **Salida, Y.** Representa el valor resultante tras pasar por la red neuronal.

2.2.5. Redes de convolución (CNN)

Las redes neuronales convolucionales son algoritmos de aprendizaje profundo que inicialmente fueron diseñados para trabajar con imágenes. Las CNNs están diseñadas para recibir los datos de entrada en forma de matrices multidimensionales. Además, contienen varias capas ocultas especializadas y con una jerarquía: esto quiere decir que cada capa permite detectar ciertas características y se van especializando hasta llegar a capas más profundas que reconocen formas complejas *(Angermueller et al., 2016).*

Los principales componentes de una red de convolución son:

a) Capa de convolución
b) Capa de "Pooling" o submuestreo
c) Capa totalmente conectada (La capa final de salida)

La figura 5, muestra los principales elementos de las redes de convolución que a continuación se detallan.

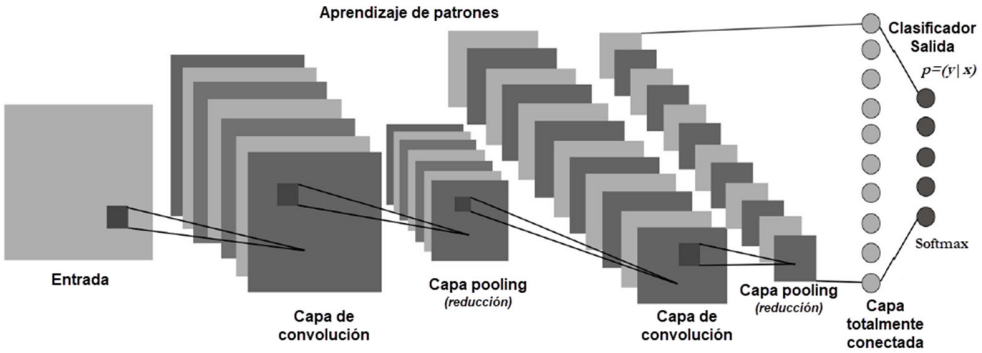

Figura 5. Esquema de los elementos principales de una CNN.
Figura modificada de: (Albelwi et Mahmood ,2017).

a) Capa de convolución

La capa de convolución se encarga de tomar porciones de la matriz de entrada y realizar operaciones del producto escalar de cada una de estas porciones contra una pequeña matriz, que se denomina Kernel, la cual es la ventana que recorre o convoluciona sobre la matriz de entrada el cual genera una nueva matriz más pequeña de salida como se puede apreciar en la figura 6.

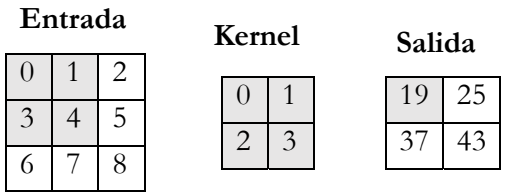

Figura 6. Ejemplo de un Kernel (tamaño 2x2) ventana que recorre una matriz de entrada

Otro elemento importante de esta capa de convolución es el filtro, el cual es un conjunto de "kernels". Para cada capa de convolución, se fija un determinado número de filtros, los cuales se conocen como mapa de detección de características o "feature mapping". Cada filtro permitirá detectar alguna característica importante de los datos de entrada. Una de las funciones de activación más utilizada en este tipo de redes es RELU (Rectified Linear Unit), la cual se define con la siguiente fórmula: $f(x) = x^+ = \max(0, x)$, donde x es la entrada a la neurona.

b) Capa de "Pooling" o de reducción

La capa de pooling generalmente se coloca después de la capa de convolución. Su propósito principal es reducir la dimensión espacial (ancho y alto) del volumen de entrada para la siguiente capa de convolución y mantener las características más importantes que detectó cada filtro. La operación realizada por esta capa se llama reducción de muestreo (submuestreo). Existen varios tipos dentro de los que destacan: max-pooling y average-pooling. En el caso del max-pooling para cada una de las regiones representadas por el filtro, se toma el máximo de esa región. Se crea una nueva matriz de salida donde cada elemento es el máximo de una región en la entrada original, como se ilustra en la figura 7. De este modo en este paso es posible reducir la matriz de salida antes de hacer una nueva convolución y así solo quedarse con las características más comunes. Por ejemplo, si tenemos una matriz de 24x24 tendríamos 576 neuronas y si aplicamos un filtro de 64, la dimensión de la matriz de salida tendría un valor de 36,864. Lo cual incrementaría bastante el procesamiento de computación de estos datos.

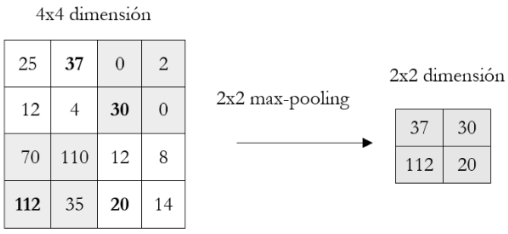

Figura 7. Ejemplo ilustrativo de la operación max-pooling

c) Capa totalmente conectada (La capa final de salida)

Esta capa corresponde a una red neuronal tradicional que conectará con la última capa a la que se le aplicó el pooling y que fue aplanado en un vector de una sola columna. Esta salida aplanada alimentará esta red neuronal, la cual tendrá la cantidad de neuronas como el número de clases a predecir o clasificar, normalmente se utiliza la técnica de clasificación Softmax. Softmax es una función que convierte un vector de K valores reales en un vector de K valores reales que suman 1. Los valores de entrada pueden ser positivos, negativos, cero o mayores que uno, pero el Softmax los transforma en valores entre 0 y 1, para que se puedan interpretar como probabilidades *(Wood Tomas, 2019)*. La definición matemática se representa con la siguiente fórmula:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Donde:

$\vec{z}$	Es el vector de entrada a la función Softmax
z_i	son los valores z_i que dio el vector de entrada a los cuales se les aplicará la función softmax para que representen las probabilidades
e^{z_i}	Es la función exponencial estándar que se aplica a cada elemento del vector de entrada
$\sum_{j=1}^{K} e^{z_j}$	Corresponde a la normalización. Asegura que todos los valores de salida de la función sumen 1 y cada uno esté en el rango (0, 1), constituyendo así una distribución de probabilidad válida.
K	Es el número de clases.

2.2.6. Redes residuales (Resnets)

En aprendizaje profundo, las redes al tener más capas detectan características más específicas y dan mejores resultados. Pero en la práctica resulta difícil optimizarlas y por ende el proceso de entrenamiento también, se puede decir que hasta cierto punto si se agregan más capas es benéfico, pero se debe tener precaución ya que agregar demasiadas puede disminuir la precisión. Las redes residuales intentan solucionar este problema agregando una conexión llamada "skip connection" o "short cut connection" *(He et al., 2016)*. A través de esta conexión se transfiere la información importante únicamente de un punto de la red hacia adelante, permitiéndole a la red aprender las funciones residuales (lo más importante) con referencia a las capas de entrada, es decir $y = F(x) + x$. Lo cual se conoce como bloque residual (figura 8).

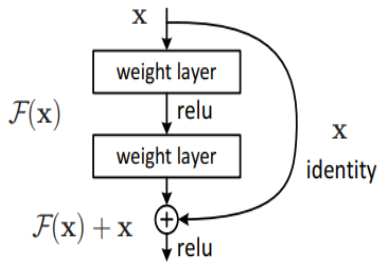

Figura 8. Esquema de un bloque residual dentro de una CNN. Imagen tomada del articulo (He et al., 2016)

2.3. Antecedentes de la investigación

Como se mencionó en la introducción de los datos metagenómicos, una gran proporción de lecturas de secuencias de ADN quedan sin clasificar ya que no se coinciden con las bases de datos de referencia. En este sentido, se han desarrollado varias aproximaciones para tratar de desarrollar metodologías computacionales que permitan identificar nuevos virus o con baja similitud con lo anotado, que quedan sin clasificar por métodos convencionales. Un primer trabajo corresponde a Páez-Espino et Al. (2016), dónde analizaron 5 Tb de datos de secuencias metagenómicas de 3,042 muestras distintas. Se descubrieron más de 125,000 nuevos genomas parciales de virus ADN, lo que supone 17 veces más del número de genes virales conocidos hasta ese momento. Esto supone más de 2.79 millones de secuencias de proteínas virales, de las que el 75% no tenía similitud con proteínas conocidas. Páez-Espino et al. (2016), propusieron un enfoque computacional para explorar el contenido viral de las muestras metagenómicas (figura 9), el cual consiste en el siguiente proceso:

(A) Preprocesamiento y filtrado de los datos adquiridos para mejorar su calidad. Donde se obtiene una definición de familias de proteínas virales que serán utilizadas para la identificación de contigs virales metagenómicos mayores de 5 kb de longitud. En la primera etapa de este método, se agruparon las proteínas de 2.300 virus de ADN bicatenario en 16,000 grupos de familias de proteínas, que se alinearon para generar perfiles de familia utilizando Modelos Ocultos de Márkov (HMM por sus siglas en inglés). Estos HMM se usaron en combinación con el análisis de composición k-mer y análisis filogenético de ADN dependiente de genes de ARN polimerasa para identificar 1,843 contigs virales metagenomicos (mVC).

(B) Esos contigs fueron validados manualmente y las proteínas de este conjunto se combinaron con proteínas virales (iVGs) para generar un conjunto final de 25,000 grupos de familias de proteínas virales.

(C) Los perfiles de HMM generados a partir de la alineación de estas familias de proteínas se utilizaron como herramienta para identificar 125,842 nuevos contigs virales metagenómicos. Los mVC finales se agruparon y se asignaron a su huésped a través de CRISPR – Cas spacer y tRNA viral coincide con microbios aislados (estos no se muestran en la figura 9).

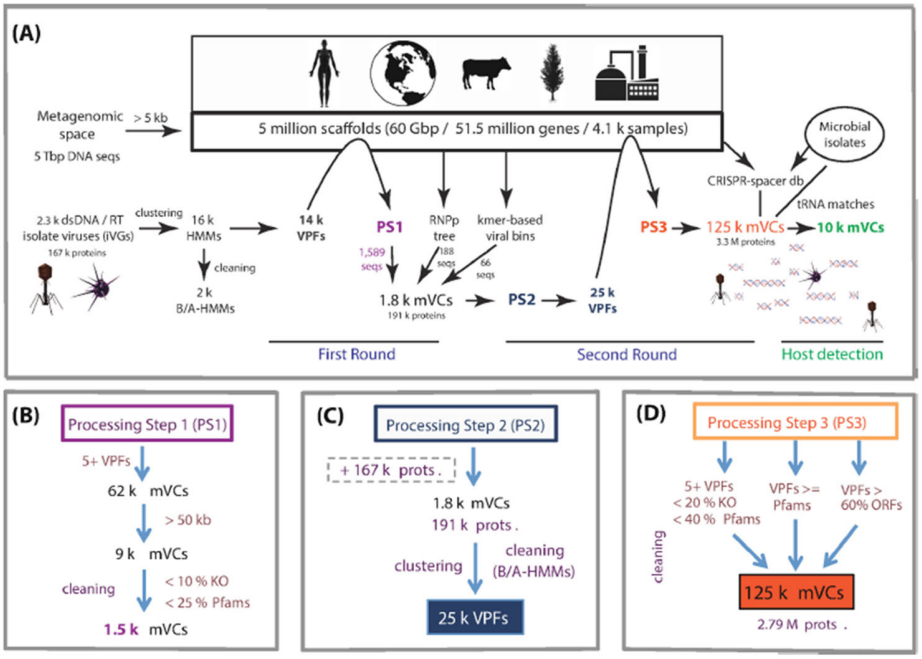

Figura 9. Proceso para la identificación de secuencias virales en datos metagenómicos propuesto por Páez-Espino et al., 2016. Imagen tomada del artículo (Páez-Espino et al., 2016)

Este estudio llevó a un notable incremento en el número de secuencias virales putativas, además de identificar conexiones virus-hospedero, lo que demostró que la diversidad de procariotas es mucho más grande de lo que se conoce. Además, proporcionó mapas globales de biogeografía viral que muestran que los virus se encuentran en hábitats similares, independientemente de su proximidad geográfica. Cabe mencionar que, en este trabajo, los contigs eran mayores a 5k, por lo que muchos fragmentos no se consideraron, ya que en metagenómica se generan muchos trozos de longitud de 75pb, los cuales podrían contener información relevante para el descubrimiento de nuevos virus.

Un segundo trabajo, enfocado a la predicción de proteínas, fue el realizado por Kulmanov et al. (2017), en el cual se abordan tres retos computacionales en la predicción de la función de las proteínas que son: aprender características para representar una proteína, predecir funciones en

un espacio de salida jerárquico con fuertes dependencias, y combinar la información de secuencias de proteínas con redes de interacción proteína-proteína. En este trabajo se propone un modelo jerárquico, basado en una red convolucional y en la ontología genética (GO) de secuencias de proteínas, formada por 3 clases: Función Molecular (MF), Proceso Biológico (BP), y la de Componente Celular (CC).

En la primera parte de este modelo se aprende una representación vectorial de una secuencia de proteínas que puede utilizarse como características para predecir las funciones de las proteínas, y en la segunda parte del modelo tiene como objetivo la codificación para las dependencias funcionales entre clases GO, y optimizar la precisión de la clasificación sobre la estructura jerárquica del GO en lugar de optimizar un modelo localmente para cada clase de la ontología.

La arquitectura del modelo se asemeja a la estructura jerárquica del GO y a las dependencias entre sus clases, para asegurar que las características discriminatorias de cada clase puedan ser aprendidas de forma jerárquica, teniendo en cuenta las relaciones simbólicas en el GO. Sin embargo, este modelo solo abarca proteínas completas lo cual en su mayoría de los casos no es lo real en datos metagenómicos. Por otra parte, necesita grandes cantidades de datos de entrenamiento para cada clase; estos datos están disponibles a través de las anotaciones del manual GO que se han creado durante muchos años, pero que no estarán fácilmente disponibles para todas las familias virales lo que dificultaría la predicción de nuevas proteínas virales.

Por otra parte, se estudió un tercer trabajo realizado por Tampuu et al. (2019) donde se desarrolló un modelo que se nombró ViraMiner, que utiliza las Redes Neuronales Convolucionales (CNN) en datos metagenómicos de diferentes muestras humanas para identificar posibles secuencias virales. La arquitectura de ViraMiner se construyó sobre el modelo de CNN de DeepVirFinder reportado por Ren et al. (2018), agregando algunos detalles para que sea más efectivo en este problema de clasificación. Para entrenar el modelo, se utilizaron 19 experimentos metagenómicos originados de muestras tipos como la piel, el suero y los condilomas. La arquitectura de ViraMiner toma como entrada las secuencias de contigs de tamaño de 300pb en forma codificada. Estas secuencias son procesadas por dos ramas convolucionales diferentes.

La rama de patrones, que devuelve que tan bien los patrones están emparejados a lo largo de la secuencia y la rama de frecuencia devuelve la frecuencia de los patrones encontrados. Las salidas de las ramas se unen y se usan para calcular la salida final. El valor del modelo está restringido al rango [0,1] y refleja la probabilidad de que la secuencia perteneciente a la clase de virus, como se muestra en la

Figura 10. El modelo logra una precisión significativamente mejorada en comparación con otros métodos existentes para la identificación de virus en muestras metagenómicas. Hasta donde se sabe, el modelo propuesto es la primera metodología que puede detectar la presencia de virus en contigs metagenómicos de varias muestras humanas. Sin embargo, este trabajo está basado en redes convolucionales para detectar contigs virales únicamente en conjuntos de datos metagenómicos humanos, en el caso de este trabajo de tesis se propone un modelo que sea capaz de detectar contigs virales para cualquier tipo de estudio metagenómico.

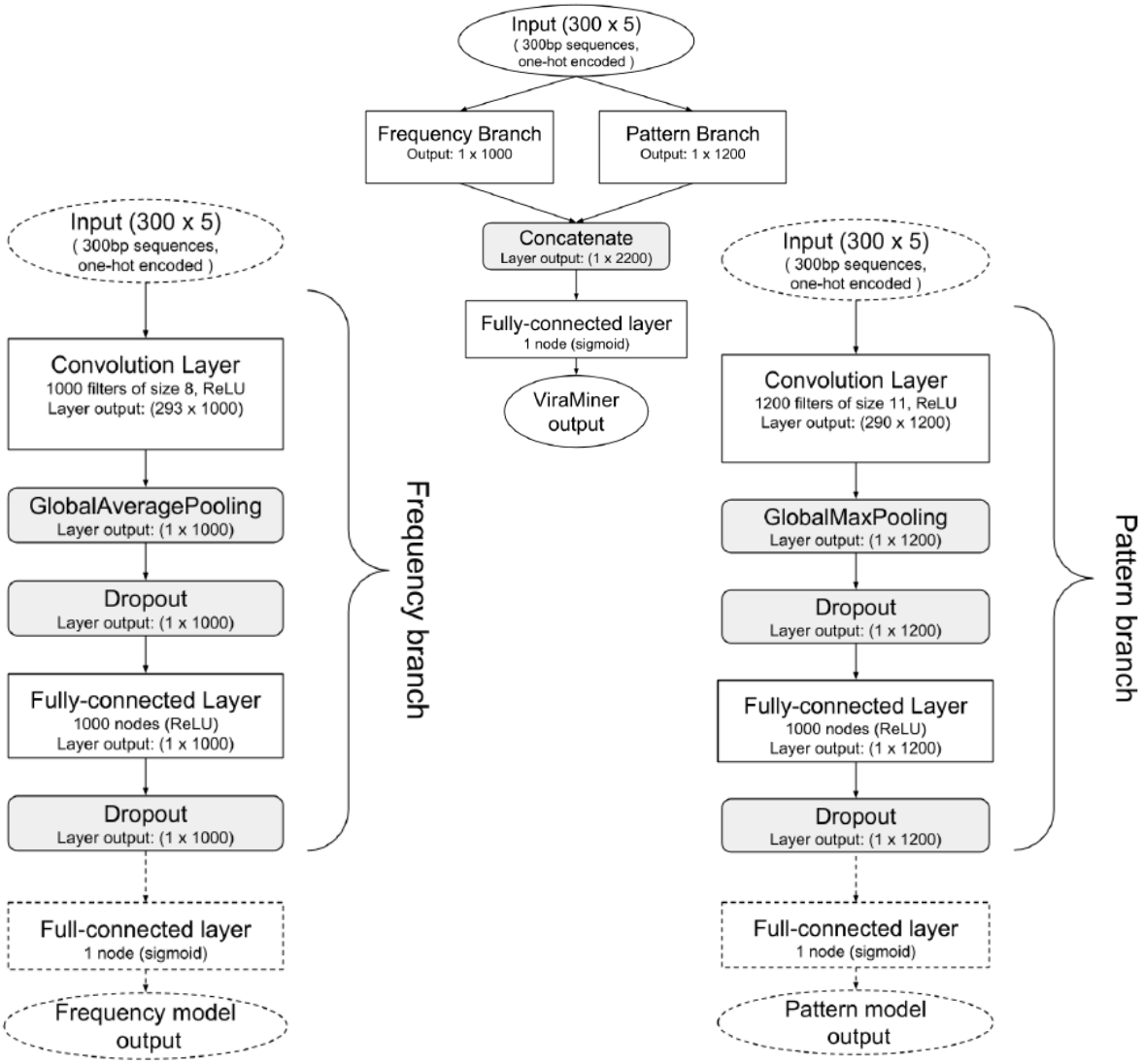

Figura 10. Arquitectura del modelo ViraMiner propuesto por Tampuu et al. 2019. Imagen tomada de (Tampu et al. 2019)

Finalmente, otro estudio que fue de gran aportación fue el realizado por Bileschi et al. (2019). Esto se debe a la dificultad que implica comprender la relación entre la secuencia de aminoácidos y la función de las proteínas lo cual es un problema en biología molecular con implicaciones científicas de gran alcance. A pesar de seis décadas de progreso, las técnicas de vanguardia no pueden anotar 1/3 de secuencias de proteínas microbianas, lo que dificulta la capacidad para explotar las secuencias recolectadas de diversos organismos. En este sentido, Bileschi et al. (2019) cuestionan si los modelos de aprendizaje profundo pueden complementar los enfoques existentes (BLASTp y Modelos Ocultos de Márkov (HMM)) y proporcionar herramientas de predicción de la función de las proteínas, usando una amplia cobertura del universo de las proteínas, permitiendo que se anoten secuencias más distantes en homología a las existentes en las bases de datos. En este trabajo, se realiza el comparativo entre el aprendizaje profundo y la tarea de anotar secuencias con dominios de proteínas alineadas de Pfam-seed (*Punta et al. ,2012*). Se incluyeron 17,929 familias y más de un 1 millón de ejemplos en el entrenamiento. Los modelos

de aprendizaje profundos en este trabajo logran una precisión extremadamente alta sin conocimiento previo de los datos de la secuencia de la proteína, ni la codificación de estás a través de matrices de sustitución, o alineación de secuencias. Esto difiere sustancialmente de enfoques como BLASTp, phmmer y HMMER que realizan anotaciones usando alineación explícita. Los resultados presentan un avance significativo respecto a trabajos anteriores para la aplicación de métodos de aprendizaje profundo en cuanto al número de familias, y el número de secuencias de entrenamiento por familia. Sin embargo, para el presente trabajo de estudio, el modelo desarrollado por Bileschi et al. (2019), difiere al aquí propuesto debido a que éste se basa en secuencias completas de proteínas, lo cual no aplica para el caso de datos metagenómicos donde se reciben fragmentos de secuencias de distintas partes de lo que podría ser una secuencia completa genómica. Pero podría ser un buen modelo base debido al manejo de una gran cantidad de datos para entrenamiento, además de que estos datos están representados como secuencias de aminoácidos (proteínas) y también realizar una clasificación hacia múltiples clases.

CAPITULO 3. METODOLOGÍA

La metodología implementada fue la siguiente, ya que los modelos de aprendizaje profundo requieren un conjunto de datos de entrenamiento, una de las primeras actividades a realizar fue: i) obtener el conjunto de datos de entrenamiento, el cual consiste en todas las secuencias de proteínas virales anotadas. Para esto, se realizó la descarga de estas secuencias de la base de datos GenBank, ii) Realizar el preprocesamiento de este conjunto de datos el cual consistió en eliminar la redundancia de la información (secuencias duplicadas) y de cierto tamaño, iii) Generación de las clases aplicando una técnica de agrupamiento de los datos anteriores, con las cuales se entrenará el modelo y permitirá definir la salida del modelo que sería la clasificación del tipo de proteína viral, iv) Definir un diseño base de la arquitectura de la CNN. Ajuste de los parámetros del modelo base en función a su desempeño, hasta encontrar el mejor modelo, y por último v) Desarrollar un script que permita al usuario preprocesamiento y post procesamiento para la utilización del modelo de clasificación de proteínas virales.

A continuación, y como parte del capítulo de metodología, se detalla cada uno de estos pasos metodológicos.

3.1. Generación del conjunto de datos

Se realizó la descarga de los archivos de todas las secuencias de proteína viral del sitio del Centro de información de Biotecnología conocido por sus siglas en inglés como NCBI (National Center for Biotechnology Information) y que se encuentra dentro de la base de datos de GenBank (https://www.ncbi.nlm.nih.gov/GenBank/).

Se depuraron las secuencias únicamente considerando aquellas con una longitud mayor a 100pb a 15,000pb. La descarga de estas secuencias se dividió en 3 grupos: FLU que contiene todas las proteínas virales de la familia Orthomyxoviridae caracterizado por incluir virus de Influenza, VIH que contiene las proteínas virales del género Lentivirus que se caracteriza por el retrovirus de Virus de Inmunodeficiencia humana (VIH), y el tercer grupo que excluye los dos anteriores al cual se le denominó: SIN. La descarga de estas secuencias en 3 grupos fue para agilizar el proceso de descarga, y por otra parte tener un mejor control de calidad de los datos al asegurar la limpieza de secuencias duplicadas, ya que los virus de Influenza y VIH son virus muy estudiados y por consecuencia son de los más anotados en esta base de datos.

3.2. Preprocesamiento del conjunto de datos para obtención de clases

Una vez obtenido el conjunto de datos de secuencias de proteínas virales, se procedió a quitar las secuencias duplicadas. Para esto, se utilizó el programa CDHIT, el cual es un programa para agrupar y comparar conjuntos grandes de secuencias de proteínas o nucleótidos *(Li, W. et al., 2001.)*. Este algoritmo heurístico funciona de la siguiente manera: i) Ordena las secuencias de largas a cortas, ii) La secuencia más larga se clasifica automáticamente como secuencia representativa del primer grupo, iii) Las secuencias restantes se comparan con las secuencias representativas de los grupos o clúster, iv) Si la similitud con cualquier representante está por encima de un umbral (porcentaje de identidad) dado, se agrupa en ese grupo. De lo contrario,

un nuevo clúster se define con esa secuencia como el representante. Obtención de clases para el modelo.

Una vez eliminadas las secuencias de proteínas redundantes, se realizó su agrupamiento jerárquico utilizando nuevamente el programa CD-HIT. El agrupamiento jerárquico a diferencia de la agrupación en un solo paso que utiliza CD-HIT, nos permite obtener resultados con mejor precisión. Esto debido a que la agrupación en un solo paso, al ser un algoritmo heurístico de complejidad n2, puede originar el siguiente problema: se pueden agrupar dos secuencias A y B muy similares en diferentes grupos. Por ejemplo: Supóngase que el umbral de agrupación sea 40%, IAB (identidad de AB) = 95%, IAC ≥ 40%, pero IBC <40%. Si C fue seleccionado por primera vez como un representante del grupo, entonces A estará en el grupo "C", pero "B" no lo hará, lo que resulta en que un AB casi idéntico se encuentre en diferentes grupos.
En este contexto se decidió utilizar el agrupamiento jerárquico con los siguientes parámetros:

a) Se realizó un CD-HIT de un solo paso del conjunto original al 80% y otro al 60% de identidad del cd-hit.
b) Del cd-hit generado al 60% se aplicó el comando *psi-cd-hit*, donde se define el nivel más bajo de identidad para agrupar que se definió al 40%.
c) Se realiza la revisión de los grupos generados al 80 y 60% de identidad con el programa *clstr_rev* para verificar si existen coincidencias en ellos. El cual genera un archivo: file80-60.clstr
d) Este último archivo, se utilizará para realizar la última revisión de los grupos con el archivo generado al 40% en el paso b, y que generará el archivo file80-60-40.clstr.

El resultado de este agrupamiento jerárquico permitió definir las clases que utilizará el modelo para realizar la clasificación de secuencias de proteínas virales

3.3. Diseño del modelo

Como un primer paso, se realizó una revisión en la literatura para identificar un modelo previamente publicado, que se pudiera utilizar como modelo base para posteriormente adecuarlo a las necesidades de este trabajo de investigación. Como se mencionó en el apartado de antecedentes, el modelo que se utilizó en este trabajo de tesis fue el propuesto por (Bileschi et al., 2019). El cual se consideró un buen modelo, ya que se utiliza para la clasificación de secuencias de aminoácidos, es multi-clase y realiza el procesamiento de una gran cantidad de datos de secuencias genómicas. El modelo de Bileschi, está basado en un modelo de red de convolución residual. Sin embargo, este modelo realiza la predicción de proteínas completas, lo cual difiere al trabajo aquí propuesto que requiere procesar datos metagenómicos, lo cual implica hacer predicción con solo fragmentos de la información de la secuencia a fin de realizar la predicción de la proteína viral. Para ello, fue necesario realizar tres actividades muy importantes para el diseño del modelo que fueron:

i) fragmentación de las proteínas virales,
ii) etiquetado de los datos y
iii) adecuación del modelo, las cuales se describen en detalle en las subsecciones siguientes.

3.3.1. Fragmentación de las secuencias de ADN

Para el diseño del modelo de red base, se realizó el preprocesamiento de las secuencias de ADN de entrada. Una actividad crucial de esta etapa fue dividir cada una de las secuencias de las proteínas virales en fragmentos más pequeños (k-mers). Esto se debe principalmente a que se está trabajando con datos metagenómicos, los cuales vienen en primera instancia como millones de lecturas cortas o fragmentos que representan cierta parte de genomas o proteínas de diversos organismos. Los k-mers son las k subsecuencias posibles de una secuencia completa, es decir el número total de éstos representa las subsecuencias de longitud k contenidas dentro de una secuencia biológica. Esto está dado por la siguiente fórmula, cuando el traslape (salto) de la secuencia es 1:

$$Totalkmers = (longitud\ secuencia - k)+1$$

La figura 11, muestra un ejemplo de la descomposición de una secuencia de longitud de 14 nucleótidos, la cual se analizará en fragmentos de longitudes de 8. Por lo tanto, el número de fragmentos de acuerdo con la formula será de 7.

Secuencia	**G**	**A**	**T**	**C**	**C**	**T**	**A**	**C**	**T**	**G**	**A**	**T**	**G**	**C**
Kmers de 8: #1	G	A	T	C	C	T	A	C						
Kmers de 8: #2		A	T	C	C	T	A	C	T					
Kmers de 8: #3			T	C	C	T	A	C	T	G				
Kmers de 8: #4				C	C	T	A	C	T	G	A			
Kmers de 8: #5					C	T	A	C	T	G	A	T		
Kmers de 8: #6						T	A	C	T	G	A	T	G	
Kmers de 8: #7							A	C	T	G	A	T	G	C

Figura 11. Ejemplo de una secuencia y k-mers obtenidos cuando Kmer =8 y salto=1

Cuando el traslape o el salto en la secuencia sea mayor a 1. La fórmula que se aplica es la siguiente:

$$Totalkmers = \frac{\big((longitud\ secuencia - k) + 1\big)}{\text{salto}}$$

Por ejemplo, cuando la secuencia es de longitud 14, el Kmer = 3 y el salto = 2, se obtendría en total 6 fragmentos. Es importante mencionar que el uso de esta técnica incrementa exponencialmente el tamaño del conjunto de datos para el entrenamiento del modelo de predicción. En este estudio, se decidió utilizar un k-mer de longitud 100 y un salto de 10. Se tomaron 100 aminoácidos ya que es la longitud mínima de una proteína viral.

3.3.2. Etiquetado de los datos

El entrenamiento de una red generalmente comienza con el etiquetado de datos (Xi, Yi), donde cada Xi, es el conjunto de características de la entrada i-ésima, y Yi es su etiqueta de salida, es decir, la clase a la cual pertenece. Para introducir estos datos de entrada a una red neuronal normalmente se realizan a través de una matriz de valores numéricos. En este trabajo, los datos

de entrada fueron los fragmentos de las secuencias de proteínas virales. Estas secuencias de proteína viral se fragmentaron en trozos de longitud de 100 aminoácidos y un salto de 10, las cuales están representadas por 20 aminoácidos con código de una sola letra. La figura 12, muestra un ejemplo de la secuencia de proteína viral representada con código de aminoácidos de una sola letra, que utilizará el modelo.

9 [Rotavirus]

GenBank: CAA64568.1

GenPept Identical Proteins Graphics

```
>CAA64568.1 9 [Rotavirus]
MYGIEYTTILTILISIILLNYILKTITNTMDYIIFRFLLLIALMSPFVRTQNYGMYLPITGSLDAVYTNS
TSGESFLTSTLCLYYPTEAKNEISDNEWENTLSQLFLTKGWPTGSVYFKDYNDITTFSMNPQLYCDYNVV
LMRYDNTSELDASELADLILNEWLCNPMDISLYYYQQNSESNKWISMGTDCTVKVCPLNTQTLGIGCKTT
DVDTFEIVTSSEKLVITDVVNGVNHKINISISTCTIRNCNKLGPRENVAIIQVGGPNALDITADPTTVPQ
VQRIMRVNWKKWWQVFYTVVDYINQIIQVMSKRSRSLDTATFYYRI
```

Figura 12. Secuencia de proteína viral

Para realizar la conversión de los fragmentos de cada secuencia para la matriz de valores numéricos de entrada a la red neuronal se utilizó la técnica: one-hot encoding. En este contexto, tenemos una matriz de $20 * k$, donde 20 representa el código del aminoácido y "k", representa la longitud del fragmento que como ya se había mencionado en la sección anterior se estableció de longitud $k = 100$. La figura 13, muestra un ejemplo ilustrativo de una secuencia longitud 9 aminoacidos y cuál es su representación en la matriz de valores utilizando la técnica de "one-hot encoding", así como para los valores de la función objetivo, valores "Y" que serían las 2079 clases.

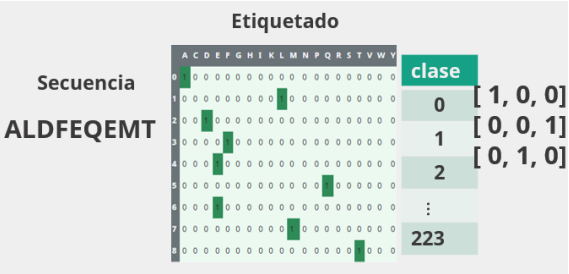

Figura 13. Ejemplo del etiquetado de una secuencia de longitud de 9 aminoácidos

3.3.3. Adecuación del modelo base

La actividad principal para adecuar el modelo base fue modificar el tamaño de la matriz de entrada, ya que como se mencionó en la sección 3.4.1, fue necesario preprocesar los datos de entrada como fragmentos de longitud de 100 aminoácidos debido a que es un modelo se utilizará para datos metagenómicos.

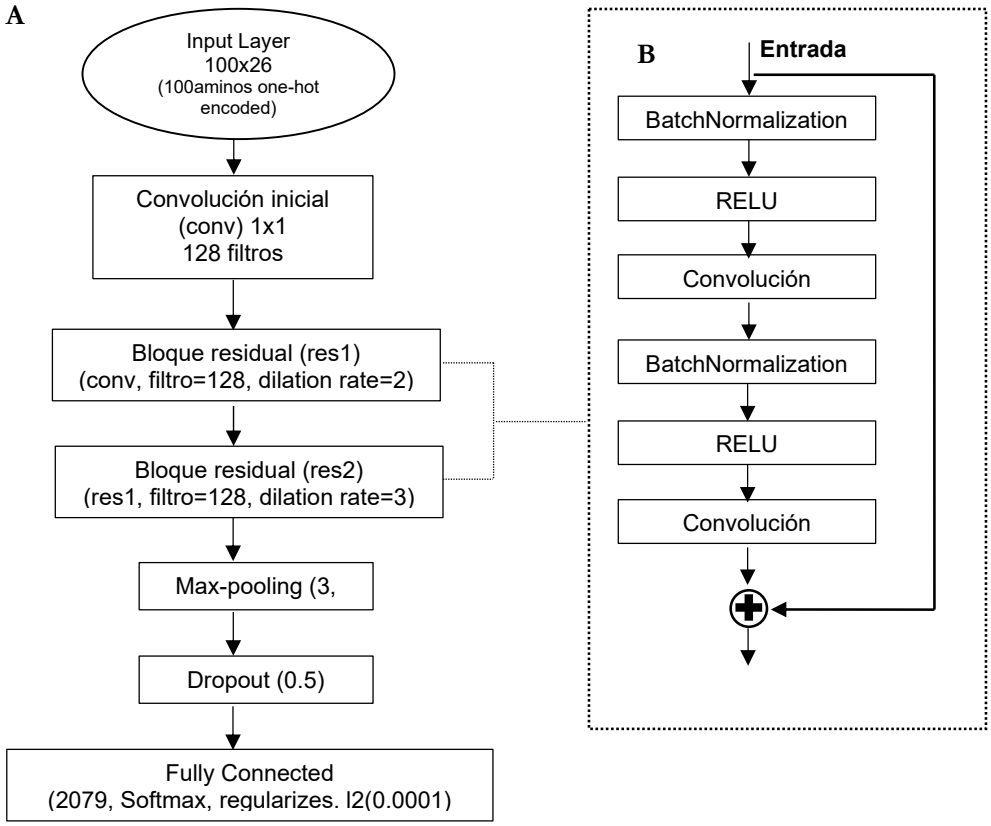

Figura 14. Esquema de la arquitectura del modelo de predicción de proteínas virales propuesto. A) Arquitectura del modelo. B) Bloque residual que se describe en el modelo general.

Como se puede apreciar en la figura 14A, en la capa de entrada se reciben las secuencias de aminoácidos con una longitud de 100 aminoácidos que son convertidas a través de la técnica "hot-encoded" a su representación numérica, por lo cual tendremos una matriz de entrada de (100,26). En este caso son 26 códigos, debido a que se consideraron los códigos de los 20 aminoácidos + 6 códigos (X, U, B, O, Z, J) que se detectaron en menor frecuencia al realizar el análisis de las frecuencias de los códigos en las secuencias del conjunto de datos del modelo como se muestra en la figura 15. Estos códigos adicionales, se debe a que al hacer la predicción ORF algunos programas no detecta con certeza que aminoácido corresponde y utilizan algún otro código para indicarlo. Se realiza una operación convolucional de kernel 1 x1 que se utilizó como técnica de proyección para hacer coincidir el número de filtros de entrada con la salida de módulos residuales en el diseño de la red residual (He, K, 2016).
Le siguen 2 bloques residuales que están basados en la arquitectura de ResNet *(He, K, 2016)*. Como se puede observar en la figura14B, los dos bloques contienen los mismos elementos, que son: "BatchNormalization" que se refiere a los pasos para arreglar la media y la varianza de las entradas a cada capa. Esto significa, que se aplica una transformación que mantenga la activación

media cercana a 0 y la desviación estándar de activación cercana a 1, lo cual mejora el entrenamiento, velocidad y desempeño de las redes neuronales.

Después, sigue la aplicación de la función de activación de RELU y una primera convolución dilatada con kernel $1x1$ con una tasa de dilatación de 2 y una segunda convolución con un kernel más grande 3x3.

Las convoluciones dilatadas permiten tener un campo más grande de recepción sin incrementar el número de parámetros del modelo y éstas utilizan un parámetro llamado tasa de dilatación (dilatation rate). Esto define un espacio entre los valores del kernel. Por ejemplo, un kernel de 3x3 con una tasa de dilatación de 2 tendrá el mismo campo de visión que un kernel de 5x5, mientras que sólo utiliza 9 parámetros, tal como se muestra en la figura 15. En resumen, las convoluciones dilatadas ofrecen un campo de visión más amplio con el mismo costo de cálculo *(Vijak, Ronan. 2019).* Finalmente, después de aplicar las operaciones de convolución, se realiza una conexión de salto (short o skip conection) muy particular de las redes residuales, que en sí lo que hace es añadir la entrada inicial y la salida de las operaciones de convolución. Continuando con la explicación del esquema general después de los 2 bloques residuales, se aplica el max-pooling para reducir el tamaño espacial de la representación. También se añade un dropout de tamaño 0.5, que es una técnica para reducir el sobreajuste del modelo, a través de "apagar" aleatoriamente ciertos nodos durante el entrenamiento. Por último, se realiza la conexión a una red neuronal totalmente conectada con 2079 nodos (uno por cada clase) y utilizando la función de clasificación "Softmax", el cual proporciona las probabilidades asignadas a cada una de las clases de proteína viral que el modelo será capaz de clasificar. Este tema será comentado con más detalle en la siguiente sección de posprocesamiento.

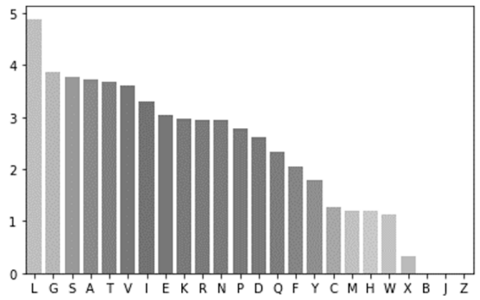

Figura 15. Frecuencia de códigos en las secuencias de aminoácidos del conjunto de datos (se muestran aquellos con más frecuencia para no saturar la gráfica)

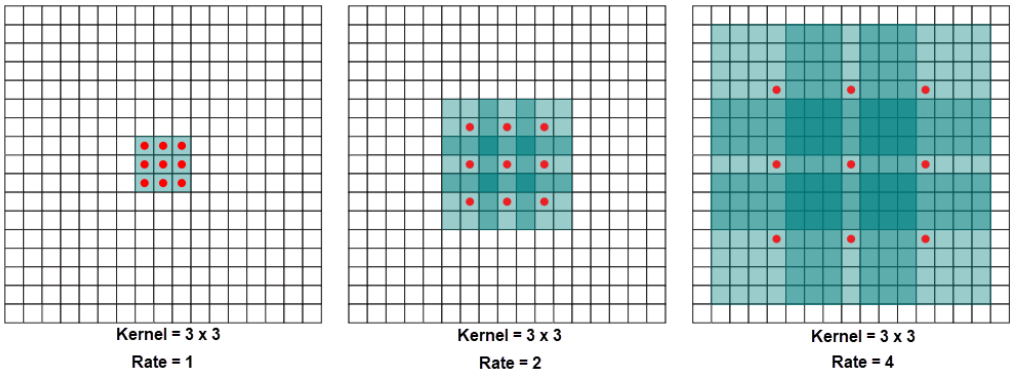

Figura 16. Ejemplo de una convolución dilatada 3 x 3 utilizando tasa de dilatación de 1, 2 y 4. Figura modificada de: (Yu, F., et Koltun, V., 2015)

3.4. Posprocesamiento

El posprocesamiento de los resultados consiste en determinar un umbral a partir de las probabilidades obtenidas a través del clasificador Softmax del modelo, con la finalidad de reportar la clase más probable o descartar que la secuencia analizada se trate de una secuencia viral. El modelo no tiene una clase negativa, por lo que se realizó un análisis de los valores obtenidos en conjuntos de secuencias virales reales y no virales. Se estima que el análisis de la distribución geométrica de los fragmentos de secuencias virales (figura 17A) presente un valor alto a la clase real, mientras que, los no virales, muestren valores de probabilidad bajas (figura 17B). A partir de este umbral de probabilidad de cada fragmento de la secuencia, se considerará la clase mayoritaria, ya sea de clase viral o no viral.

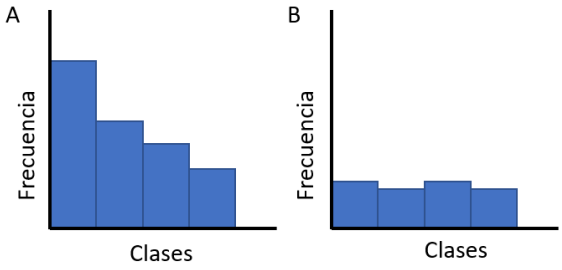

Figura 17. Distribución de probabilidades de clases. Estos valores permitirán determinar: A) Secuencias virales. B) Secuencias no virales.

3.5. Métricas usadas para la evaluación.

Para la evaluación del modelo propuesto se utilizarán las siguientes métricas de precisión, "recall" y coeficiente de correlación de Mathews (MCC).

a) **Precisión:** representa el número de elementos identificados correctamente como positivos de un total de elementos identificados como positivos y se describe con la siguiente fórmula:

$$Precision = \frac{TruePositive(TP)}{TruePositive(TP) + FalsePositive(FP)}$$

b) **Recall:** es el número de elementos identificados correctamente como positivos del total de positivos verdaderos y se representa con la siguiente fórmula:

$$Recall = \frac{TruePositive(TP)}{TruePositive(TP) + FalseNegative(FN)}$$

c) **MCC:** toma en cuenta los positivos y negativos verdaderos y falsos, y generalmente se considera como una medida equilibrada que puede usarse incluso si las clases son de tamaños muy diferentes. La fórmula es:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

CAPITULO 4. RESULTADOS Y DISCUSIÓN

En este capítulo se presentan los resultados de las etapas de generación del conjunto de datos de entrenamiento, preprocesamiento del conjunto de datos y la evaluación y ajuste de parámetros del modelo propuesto, así como una discusión de los hallazgos obtenidos durante esta evaluación.

4.1. Generación del conjunto de datos de entrenamiento

La figura 18, muestra el número de secuencias de proteínas de los diferentes organismos almacenados en la base de Datos GENBANK al 31-mayo-2019. De esta información, se seleccionó solamente la correspondiente a virus.

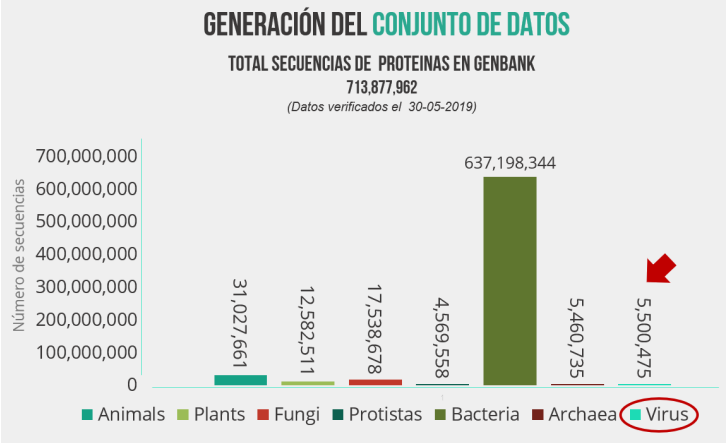

Figura 18. Gráfica de abundancia de secuencias de proteínas virales obtenidas de GenBank

El 07 de octubre de 2019 se realizó una nueva búsqueda de la información donde se obtuvieron 5,769,963 secuencias de proteína viral. Como se mencionó en la sección 3.1, a esta información se le aplicó un filtro de longitud de aminoácidos de 100 a 15,000, quedando solamente 3,015,620 secuencias (figura 19A). En la figura 19B, se muestra la cantidad de secuencias descargadas por cada grupo (FLU, VIH, SIN) que en su totalidad suman los 3,015,620, a las cuales se realizó la limpieza de secuencias duplicadas utilizando el programa de agrupamiento CD-HIT con los parámetros del 97% de identidad y una cobertura del 80% de la secuencia, del cual se obtuvieron 468,364 secuencias.

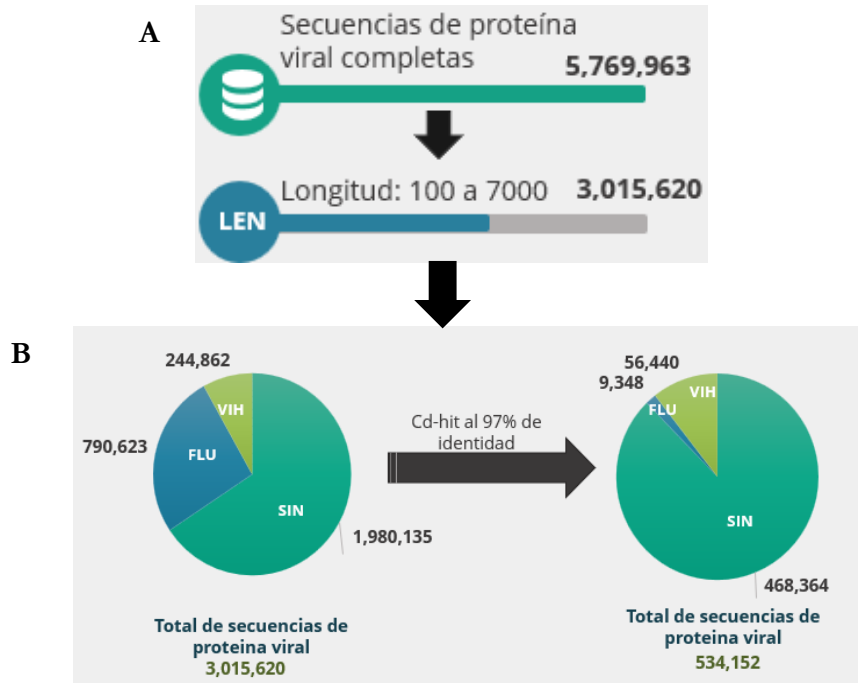

Figura 19. A) Grafico con el número de secuencias de longitud mayor a 100, obtenidas de GenBank. B) Gráfica con el número de secuencias descargadas y obtenidas al eliminar redundancia para los conjuntos de datos de FLU, VIH y SIN.

4.2. Clases obtenidas y fragmentación de secuencias

Una vez realizada la limpieza de secuencias duplicadas del conjunto de datos, se procedió a realizar el agrupamiento con el programa CD-Hit; en la figura 20 se muestra el número de clases agrupadas al 40% de identidad.

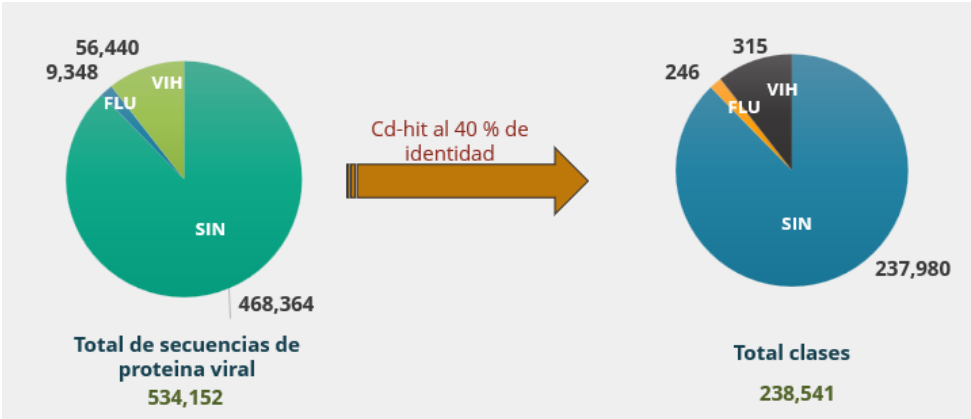

Figura 20. Número de clases, obtenidos al agrupar secuencias de virales a una identidad mayor a 40%

De las 238,541 clases obtenidas, se realizó un análisis identificándose 173,671 clases con un solo elemento o secuencia viral, lo que corresponde a casi el 70% de los grupos. Se decidió descartar los grupos que tuvieran menos de 15 elementos ya que para el entrenamiento del modelo de red

no aportan valor, debido a que los modelos de aprendizaje profundo requieren de muchos ejemplos para poder realizar una inferencia de los datos proporcionados.

La tabla 3, muestra un resumen del análisis realizado, en el primer renglón se muestra el total de grupos obtenidos al realizar el CD-Hit a un 40% de identidad entre las secuencias de cada grupo. En el segundo renglón, se indica el número de grupos con una sola secuencia. En el tercer renglón, que está señalado con el recuadro color rojo, indica un total de 2,092 grupos obtenidos, al elegir aquellos que tenían más de 15 elementos. Sin embargo, después de realizar una revisión de los resultados de los modelos implementados se reagruparon manualmente 13 grupos que estaban mal clasificados, teniendo un total de 2079 grupos finales y un total de 144,789 secuencias. Cada uno de estos 2079 grupos o clases representa un tipo de proteína viral que se usará para entrenar el modelo y realizar la clasificación de una secuencia de metagenómica.

	SIN	FLU	VIH	Total
Total de grupos después del agrupamiento con CD-Hit al 40% de identidad	237,980	246	315	238,541
Grupos con secuencia única	173,448	121	102	173,671
Grupos con más de 15 elementos	1,985	39	68	2,092
Grupos después de la reagrupación manual.	1,980	36	63	2,079

Tabla 3. Numeralía de grupos obtenidos despues del Cd-Hit al 40% de identidad

4.3. Fragmentación de secuencias

Al realizar la fragmentación de las 144,221 secuencias de proteínas virales completas (sección 3.4.1), se obtuvo un incremento del 480% del número de secuencias, obteniéndose un total 5,545,389 fragmentos correspondientes a las 2079 clases del modelo (tabla 4).

Clases	2079
Secuencias	144,221
Fragmentos	5,545,389

Tabla 4. Clases totales obtenidas con sus secuencias y fragmentos respectivos. En esta tabla se indica que las 2079 clases en total representa 144,221 secuencias y que estas a su vez representan 5,545,389 fragmentos.

4.4. Modelo usando secuencias completas (no fragmentos) de longitud de 3000 bp.

Se realizó un análisis de la distribución de la longitud de las secuencias y se determinó que podría realizar una prueba con un máximo de longitud de secuencia de 3000bp que representa el último pico de densidad de los datos, como se muestra en la figura 21. Con esta prueba se pretendió verificar si el modelo era capaz de clasificar las proteínas con secuencias completas y evitar la fragmentación de estas. El primer ejercicio consistió en entrenar el modelo solamente con el 10% del total de las clases 2092, descritos en la tabla 3, con el modelo original de Bileschi et al. (2019) y sin fragmentar las secuencias para evaluar su desempeño y posibilidad de ser usado en

este problema. Este 10% lo conformaron 255 clases y 31,373 secuencias completas. De este conjunto de datos, se tomó el 80% para el entrenamiento, 10% para validación y el 10% para pruebas.

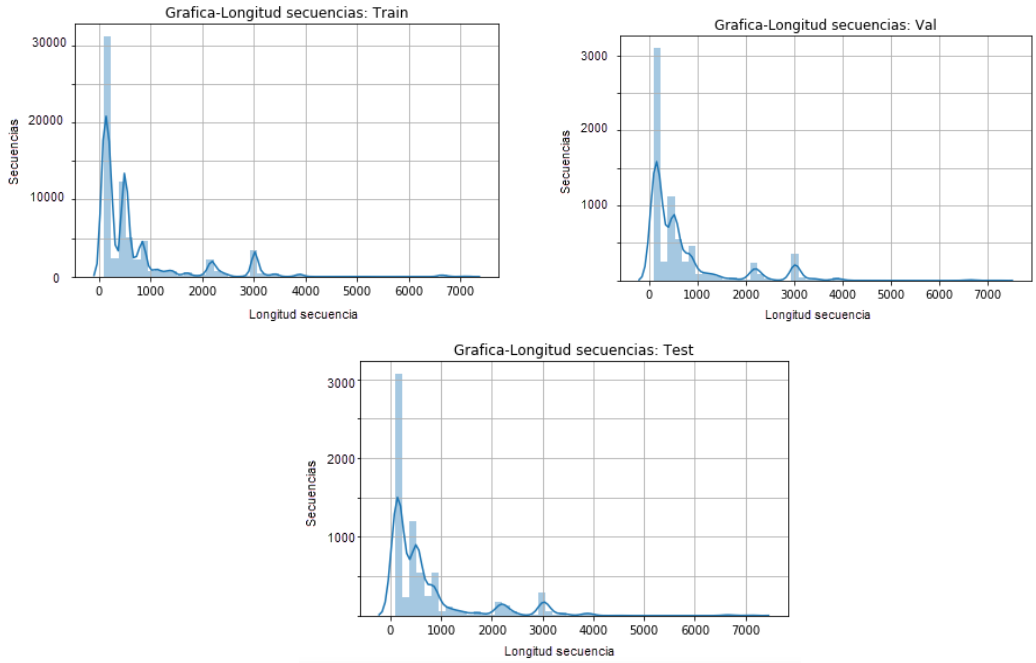

Figura 21. Distribución de la longitud de las secuencias del 10% de conjunto de datos de las 2092 clases. a) Gráfica de longitud de secuencias del conjunto de entrenamiento, b) Gráfica de longitud de secuencias del conjunto de validación y c) Gráfica de longitud de secuencias del conjunto de Prueba

La figura 22, muestra los resultados obtenidos al entrenar el modelo con el conjunto de datos anteriormente descrito. Como se puede observar en la figura 22A, el modelo alcanzó en una época temprana, antes de la 10, una alta precisión; por arriba del 93% en el conjunto de entrenamiento y cerca del 90% en el conjunto de validación. Sin embargo, en la figura 22B, se observa que la perdida del conjunto de entrenamiento inicia alrededor de 3.3 y la perdida de validación en 8.24 y termina el entrenamiento con una perdida de 0.57 y para el conjunto de validación de 1.03 en la época 50. Lo que nos indica que existe un sobreajuste que podriá deberse a que no se tiene un conjunto de datos de entrenamiento representativo, el cual proporcione suficiente información para poder generalizar sobre los datos.

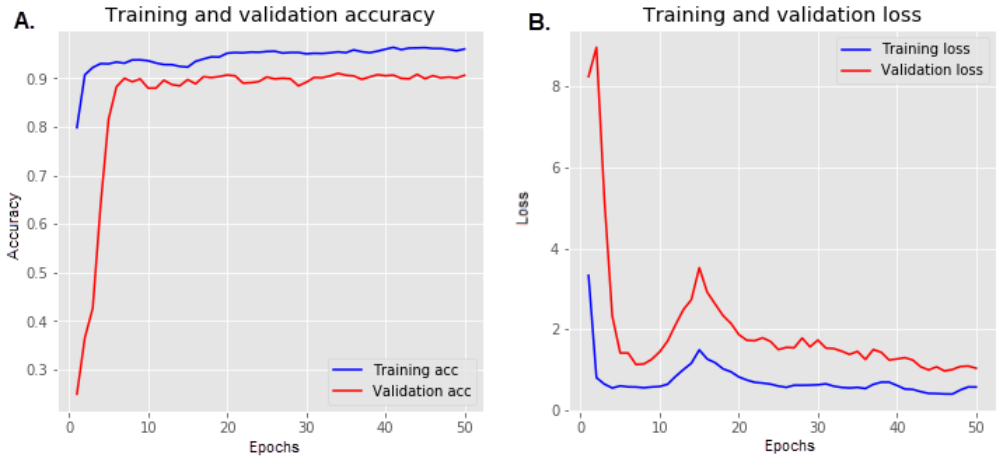

Figura 22. Resultados del modelo usando secuencias completas de longitud de 3000. A) Precisión del conjunto de datos de entrenamiento con valor de 0.93 y validación en 0.90 en la época 50. B) Perdida (loss) del conjunto de datos de entrenamiento con valor de 0.57 y validación de 1.03 en la época 50.

En base a los resultados obtenidos se decidió probar la eficiencia de este modelo con datos metagenómicos, para ello se realizaron pruebas con trozos de secuencias simulados utilizando el programa "Grinder" *(Angly et al. 2012)*. Este programa permite la generación de conjuntos de datos de lecturas simples y complejas necesarios para realizar pruebas al desarrollar software bioinformático, comparar herramientas existentes o diseñar experimentos basados en secuencias simuladas. Como resultado de este análisis, se detectó que el modelo, a pesar de haber obtenido un buen desempeño con secuencias completas, no era capaz de reconocer los fragmentos de secuencias de proteínas virales, que son comúnmente usados en estudios metagenómicos reales. En este contexto, fue necesario entrenar un nuevo modelo con las secuencias de proteínas virales fragmentadas como se especificó en la sección de 3.4.1 de esta tesis.

4.5. Modelo usando secuencias fragmentadas longitud 100, en el 10% de las clases

Para implementar este modelo se inició con la fragmentación de las secuencias. El conjunto de datos que se utilizó fue el mismo empleado en el modelo de la sección 4.4, es decir 255 clases de 2092. Una vez terminada la fragmentación, se detectó que hubo un incremento del 300% en la cantidad del número de secuencias, quedando el conjunto total con 1,020,799 fragmentos de secuencias. Como se puede apreciar en la figura 23A, se observa como la exactitud del entrenamiento se aproxima al 0.89 y la exactitud en el conjunto de validación se aproxima al 0.93. En la gráfica de la figura 23B, se observa que cambia con respecto a la gráfica del modelo anterior (sección 4.4), ahora la pérdida de validación inicia con una pérdida de 0.84 y la de entrenamiento de 1.44 lo cual indica que la de validación está por debajo del entrenamiento lo cual puede ser posiblemente a que se tenga ejemplos más fáciles en el conjunto de validación que en el de entrenamiento o bien porque se está utilizando un "dropout" muy drástico, ya que al apagar ciertas neuronas en el conjunto de entrenamiento se pierde parte de la información de cada muestra y las capas siguientes intentan construir las respuestas basándose en respuestas incompletas. Sin embargo, durante la validación todas las unidades están disponibles, de este

modo la red tiene toda su potencia de cálculo y por lo tanto podría tener un mejor rendimiento que el entrenamiento.

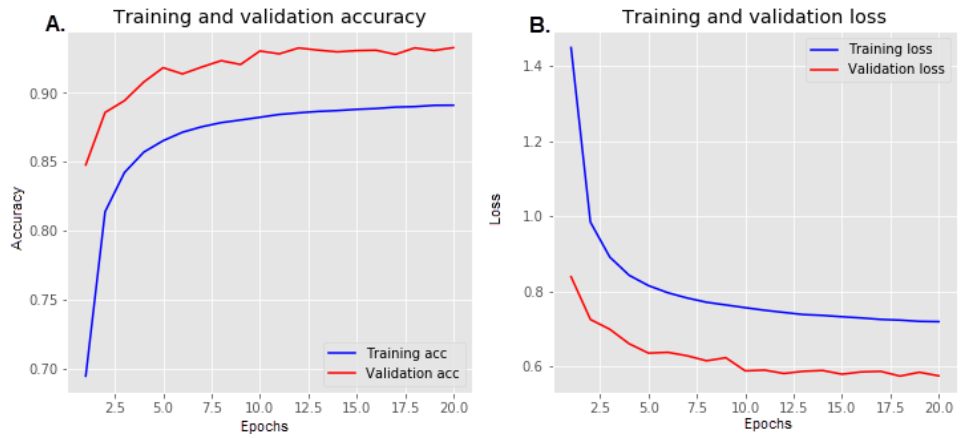

Figura 23. Resultados del modelo usando secuencias fragmentadas de longitud K=100 y salto=10 con el 10% de las clases. A) Precisión del conjunto de datos de entrenamiento con valor de 0.89 y validación en 0.93 en la época 20. B) Perdida (loss) del conjunto de datos de entrenamiento con valor de 0.71 y validación de 0.57 en la época 20.

4.6. Modelo usando las secuencias fragmentadas en el 100% de las clases (2092)

Una vez que se habían obtenido buenos resultados en un subconjunto con el 10% de las clases, se procedió a usar el modelo con la totalidad de las clases (2092) y cada una de sus secuencias en fragmentos de longitud de 100 y saltos de 10, como se explicó en la sección 3.3.1. Este conjunto se incrementó en un 450%, es decir subió de 1,020,799 a 4,741,307 fragmentos. Esto provocó que, al momento de entrenar con todo el conjunto de datos, la memoria de procesamiento del GPU alcanzara su punto máximo y no permitiera cargar los datos, por lo cual fue necesario implementar un generador de datos (data_generator). El generador de datos divide en lotes (batch) de un tamaño específico y de esta manera permite el entrenamiento del modelo en lotes. Para este caso, se determinó un tamaño igual a 2,048 secuencias de datos, que resultó en un total de 2,316 lotes para enviarlos al entrenamiento del modelo, y de este modo no sobresaturar la memoria con el procesamiento del total de fragmentos de secuencias. A partir de los resultados obtenidos en el modelo de la sección 4.5, en este modelo se aseguró que las clases estuvieran estratificadas, es decir que los subconjuntos de entrenamiento y prueba tengan las mismas proporciones de etiquetas de clase. Los resultados de este modelo se aprecian en la figura 24, la gráfica 24A muestra que los conjuntos de validación y entrenamiento, obtuvieron las precisiones de 0.92 y 0.86, respectivamente. Posiblemente este ligero decremento en la precisión del entrenamiento con respecto al modelo de la sección 4.5 puede deberse a que ahora el entrenamiento se realizó usando data_generator. En la figura 24B**,** que indica la perdida entre estos conjuntos se observa un comportamiento similar al descrito en el modelo de la sección 4.6, se observó que al final del proceso (en este caso 50 épocas), el conjunto de validación concluyó con una pérdida de 0.88 contra un valor de 1.10 del conjunto de entrenamiento.

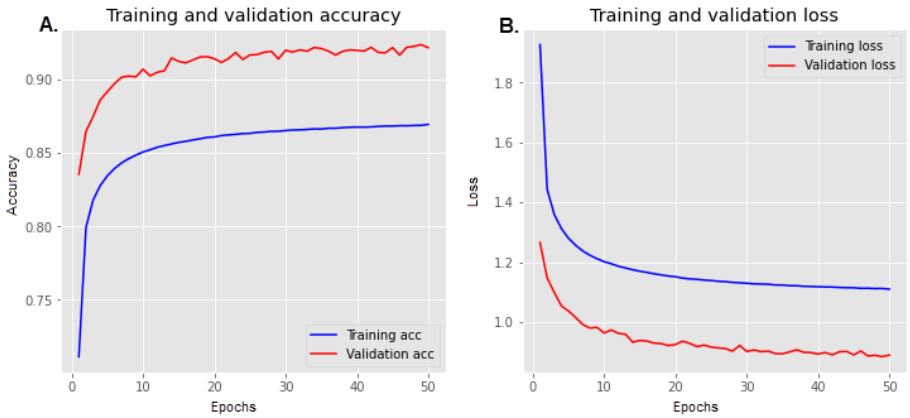

Figura 24. Resultados del modelo usando secuencias fragmentadas de longitud K=100 y salto=10 con el total de las clases (2092). A) Precisión del conjunto de datos de entrenamiento con valor de 0.86 y validación en 0.92 en la época 50. B) Perdida (loss) del conjunto de datos de entrenamiento con valor de 1.10 y validación de 0.88 en la época 50.

Por otra parte, al analizar los resultados de este modelo, se detectaron algunas clases que obtuvieron una precisión igual a cero, para las cuales, se identificaron las clases predichas y se detectó homología con las clases reales. Esto nos permitió identificar que el programa CD-HIT no había realizado correctamente el agrupamiento de ciertos grupos, probablemente por ser un algoritmo heurístico. Por lo cual fue necesario realizar una depuración manual de estas clases, que a continuación se detallan:

Del grupo de SIN que representan a otros virus que no incluyen a VIH y FLU, las clases mostradas en la columna izquierda son las clases reagrupadas por tener similitud al grupo que se describe en la columna de la derecha de la tabla 5.

Clases	**Agrupadas en:**
SIN_218885 y SIN_220225	SIN_214369
SIN_127939	SIN_103075
SIN_180250	SIN_140307
SIN_143620	SIN_63302

Tabla 5. Clases reagrupadas del conjunto SIN

En la tabla 6, se muestran los grupos de influenza (FLU) de los cuales fueron 3 clases que se unieron a la clase FLU_63 por tener homología entre ellas.

Clases	**Agrupadas en:**
FLU_70	FLU_63
FLU_73	
FLU_72	

Tabla 6. Clases reagrupadas del conjunto FLU

Por último, en la tabla 7 se muestran los grupos de virus de inmunodeficiencia (VIH). De los cuales 5 se unieron a VIH_15 y uno a VIH_200.

Clases	Agrupadas en:
VIH_52	VIH_15
VIH_63	
VIH_64	
VIH_62	
VIH_207	VIH_200

Tabla 7. Clases reagrupadas del grupo VIH

4.7. Modelo usando las secuencias fragmentadas en el 100% de las clases (2079).

En la figura 25A, se observa como la exactitud del entrenamiento se aproxima al 0.87 un punto más de precisión que el modelo de 2092 clases y la exactitud en el conjunto de validación se aproxima al 0.91. En la figura 25B que muestra la perdida tiene la misma tendencia que la del modelo de la sección 4.6, ya que al remover 13 clases del modelo no afecta ni contribuye al desempeño global de la red. Derivado de esto se decidió utilizar este modelo como final para evaluar los resultados finales de este trabajo de tesis

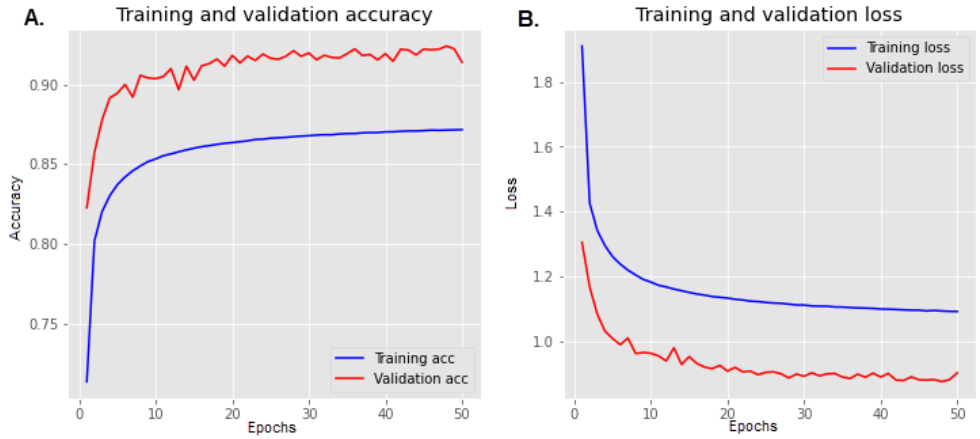

Figura 25. Resultados del modelo usando secuencias fragmentadas de longitud K=100 y salto=10 con el total de las clases (2079). A) Precisión del conjunto de datos de entrenamiento con valor de 0.87 y validación en 0.91 en la época 50. B) Perdida (loss) del conjunto de datos de entrenamiento con valor de 1.09 y validación de 0.91 en la época 50

4.8. Resumen de los resultados de los modelos

La tabla 8, presenta en resumen los resultados de los modelos anteriormente descritos para el conjunto de datos de entrenamiento, validación y pruebas. Como se puede observar, y como se mostró en la gráfica anterior, el modelo que utiliza proteínas completas alcanza una precisión del 96%. Sin embargo, dicho modelo no puede ser usado en fragmentos de secuencias de proteínas, lo cual es lo que se obtiene en datos metagenómicos reales. En cuanto al modelo propuesto en este trabajo de tesis, alcanza un 87% de precisión en el conjunto de entrenamiento y un 90% en el de prueba. En concordancia con la figura 25, en este modelo tiene un recall del 78% lo que nos sugiere que solo el 78% de estos fragmentos fueron bien clasificados.

Modelo	Precisión conjunto entrenamiento	Precisión conjunto validación	Evaluación del conjunto de Prueba (Test)		
			Precisión	Recall	MCC
Modelo base usando secuencias completas de longitud de 3000 bases (sección 4.4)	0.96	0.90	0.90	0.88	0.90
Modelo usando secuencias fragmentadas longitud 100, en el 10% de las clases (sección 4.5)	0.89	0.93	0.93	0.80	0.92
Modelo usando las secuencias fragmentadas en el 100% de las clases (2092) (sección 4.6)	0.86	0.92	0.89	0.79	0.90
Modelo usando las secuencias fragmentadas en el 100% de las clases (2079) (sección 4.7)	0.87	0.91	0.90	0.78	0.90

Tabla 8. Resumen de los resultados de los modelos implementados mostrados en las gráficas de validación/entrenamiento del modelo y por otra parte la evaluación del conjunto de prueba que representa el conjunto no visto por el modelo, en él se detallan la métrica de precisión, recall y el MCC (Coeficiente de Mathew)

4.9. Posprocesamiento de la clasificación

Como se había mencionado en la metodología (sección 3.1), en este trabajo de tesis no se cuenta con una clase negativa. Lo que significa que cualquier secuencia que se introduzca al modelo será asignada a una clase viral, aunque no lo sea. Derivado de esto, fue necesario definir un umbral para clasificar entre una clase de proteína viral y una no-viral. En la sección 4.9.1., se describen los resultados obtenidos de dicho análisis. Para esto, se obtuvieron los valores de probabilidades dados por la red, al asignar cada fragmento de las secuencias a una clase. Esto se realizó en varios conjuntos de datos que se definieron con secuencias completas virales y no virales (bacterias y humanas), que se describen en la tabla 9. Esto con la finalidad de ver si había una diferencia en los valores de probabilidad que genera la red con este tipo de datos de secuencias virales y no virales. Por otra parte, al fragmentar la secuencia en trozos, pudiera ser que diferentes partes de una sola secuencia sea clasificado por la red a diferentes clases, por lo que se realizó un posprocesamiento para asignar la secuencia total a la clase mayoritaria de los fragmentos que la componen, como se explicó en la sección 3.4. Los resultados de dicho análisis se muestran en la sección 4.9.2.

No.	Nombre	Secuencias	Fragmentos	Descripción
1	SetCompleto	144,121	5,545,89	Representa las secuencias virales que se utilizaron como conjunto de datos de entrenamiento, prueba y validación del modelo.
2	BacteMeta	10,000	104,944	Son secuencias de contigs de datos metagenómicos reales anotados como de bacterias.
3	BacteriaGenBank	10,000	221,389	Secuencias de genomas de bacterias ya anotados en GenBank.
4	EukaMetaNoHuman	10,000	84,424	Secuencias de contigs de datos metagenómicos (anotados en eucariotas) que **no** son de humanos.
5	EukaMetaHuman	2,248	13,175	Secuencias de contigs de datos metagenómicos anotados como de humano.
6	FagosGenBank	10,000	186,802	Secuencias de genomas de fagos ya anotados en GenBank.
7	VirusFagosMeta	10,000	137,195	Secuencias de contigs de datos metagenómicos anotados como fagos.
8	VirusMeta	1,184	23,319	Secuencias de contigs de datos metagenómicos anotados como virus.
9	FLU-15	409	18,418	Secuencias de los 197 grupos de FLU que se excluyeron del modelo por tener menos de 15 elementos.
10	VIH-15	789	17,965	Secuencias de los 247 grupos de VIH que se excluyeron del modelo por tener menos de 15 elementos
11	SIN-15	1914	618,044	Secuencias virales 1000 grupos de SIN que se excluyeron del modelo por tener menos de 15 elementos.

Tabla 9. Conjuntos de datos que se analizaron para definir un umbral para clasificar entre una clase proteína viral y una clase de no-virus. El Set Completo también se introdujo como conjunto de entrada (querie) al modelo entrenado para analizar las probabilidades. Los conjuntos numerados BacteriaMeta, BacteriaGenBank, EukaMetaNoHuman y EukaMeta, representan las secuencias no virales que no se introdujeron al modelo. El resto de los conjuntos representa secuencias que se conocen que son virales.

4.9.1. Definición del umbral

Como se mencionó anteriormente, definir un umbral para clasificar entre una clase de proteína viral y una clase de no-virus, a partir de los conjuntos que se definen en la tabla 9. Estos conjuntos se procesaron para introducirlos como consulta al modelo entrenado y analizar los resultados de las probabilidades de asignación a la clase dadas por la red residual que se muestra en la figura 25. De los conjuntos que definimos como no virales que son BacteriasMeta, BacteriaGenBank, EukaMetaNoHuman y EukaMetaHuman (figura 25A), podemos apreciar lo siguiente: a) De los conjuntos de BacteriasMeta y BacteriaGenBank la mayoría de sus valores se encuentran dentro de las probabilidades muy bajas que van del 0.05 al 0.25%, mientras que la frecuencia de los fragmentos anotados con probabilidad mayor al 0.95 es muy baja, lo cual indica que efectivamente los está clasificando como no virus. El 5% que clasifica como virus podría ser debido a que algunas bacterias contienen genomas de fagos incrustados en su propio genoma, existiendo un sobre lape de secuencias en ambas clases. Esto se da porque, cuando el fago (virus) infecta a la bacteria de forma lisogénica, integra completamente su genoma dentro del genoma de la bacteria *(Hatfull, 2008). b)* El conjunto EukaMetaNoHuman, se está clasificado correctamente ya que estas secuencias no son virales y como se puede observar en la gráfica se clasifican con bajos valores de probabilidad que van de valores entre 0.05 a 0.25. Por otra parte, del conjunto de EukaMetaHuman, se esperaba valores entre 0.05 a 0.25, aunque se logran clasificar algunos fragmentos entre estos valores y se observó un comportamiento similar al de Bacterias, son menos los fragmentos con este rango de probabilidad y posiblemente se deba a que contienen contigs de humanos que realmente no son virus sino retrovirus, herpes, papiloma o aquellos que se nombran repeticiones, los cuales están incrustados o se parecen mucho al genoma humano y que por lo mismo es muy común que se confunda como virus al realizar la anotación de este tipo de datos o viceversa. En cuanto a los conjuntos de secuencias de virus, para el conjunto de Fagos, se identificó que se está clasificando mucha información con muy baja probabilidad de 0.05 a 0.2%, lo cual no debería haber pasado. El modelo actualmente cuenta con 773 clases caracterizadas como fagos (del total de 2079), pero son grupos con muy pocos elementos, el mínimo es 15 y el mayor grupo contiene 61 secuencias. Posiblemente, se requiera más ejemplos de estos, pues se descartaron 213,140 secuencias y casi el 70% de estas secuencias se encuentran en grupos con un solo elemento, por lo cual posiblemente sea mejor un clasificador que se dedique básicamente a la identificación de fagos. Por último, los virus de eucariontes de muestras metagenómicas que corresponde al grupo de la gráfica nombrado como VirusMeta, los cuales pueden tener en algunos casos muy baja homología con los genomas de referencia, tienen el comportamiento de clasificación con rangos de probabilidad de 0.05 a 0.2 como si se tratará de no viral pues el modelo no detecta completamente la mayoría de las secuencias.

De la figura 25B, que muestra los datos de los grupos de las categorías FLU, VIH y SIN que se excluyeron del conjunto de datos para el entrenamiento del modelo por estar en grupos con menos de 15 elementos. Se observó, que el modelo tiende a clasificar la mayoría con probabilidad de 0.05 a 0.2, lo cual indica que es poca probabilidad de que sea proteína viral, lo cual no debería ocurrir, ya que estos grupos, aunque son los menos anotados en la base de datos de referencia (GenBank) corresponde a secuencias de proteína viral.

En conclusión y de acuerdo con el análisis visual que se hizo de las gráficas de frecuencias de probabilidades, se determinó que a partir de una probabilidad de 0.80 se podría considerar como viral y todo lo que esté por debajo de ese porcentaje se considere como no viral, esto para propósitos de realizar el posprocesamiento de los fragmentos.

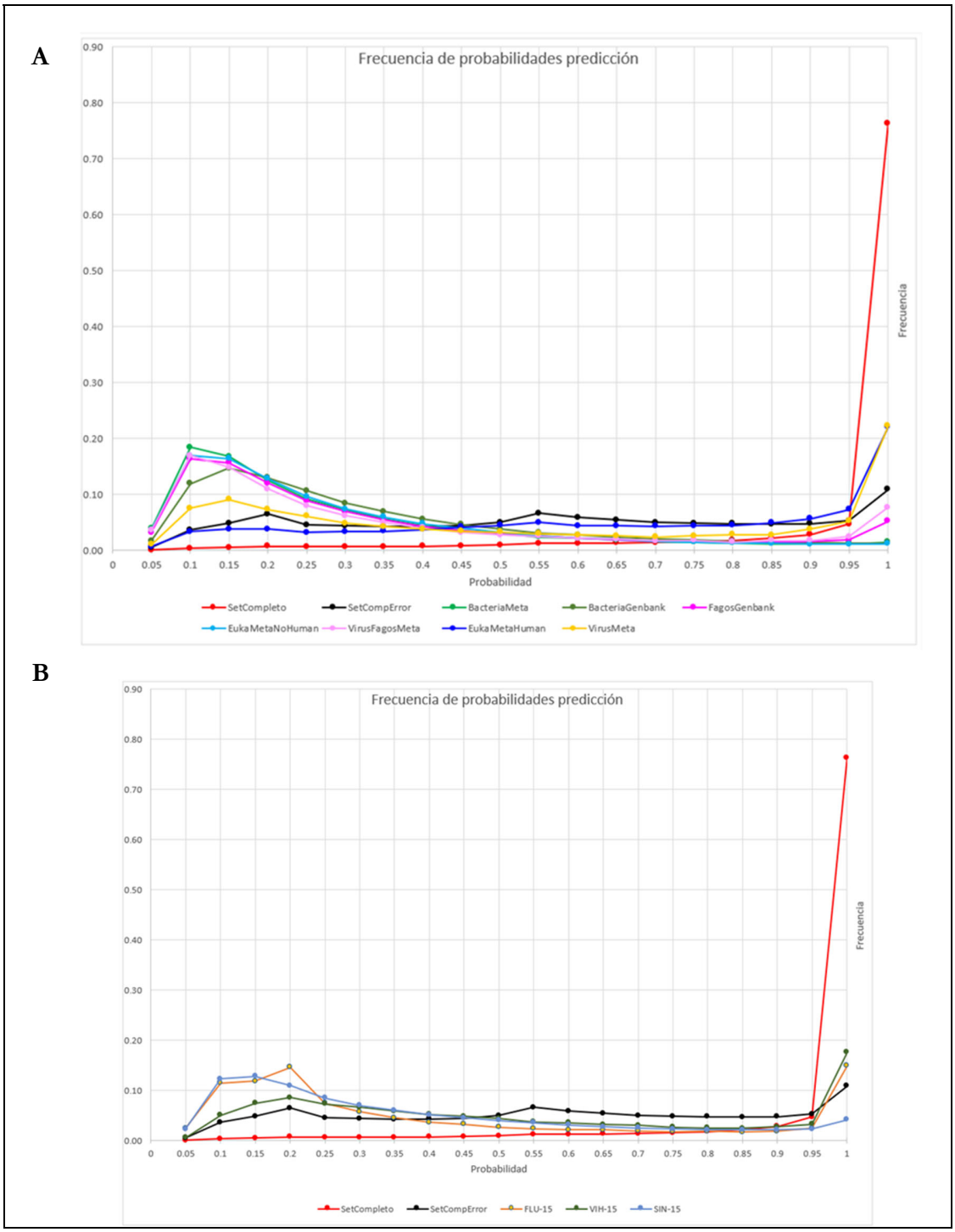

Figura 26. Gráfica de distribución de valores de probabilidad de datos de metagenómica reales. Grafica A. Muestra las probabilidades de los conjuntos de metagenómica. Gráfica B. Muestra las probabilidades de los grupos de VIH, FLU y SIN menores a 15 elementos que no se consideraron para el entrenamiento del modelo.

4.9.2. Análisis de las secuencias no virales

En esta sección, se describen los resultados obtenidos del posprocesamiento que se explicó en la sección 3.4 para asignar la secuencia completa a la clase mayoritaria asignada por el modelo, lo cual se realizó con los conjuntos de datos que se describieron en la tabla 9.

El conjunto BacteriaMeta, que se muestra en la tabla 10, representa un conjunto no viral. Para este conjunto se introdujeron al modelo entrenado 10,000 secuencias de las cuales el 5% que corresponde a un total de 557 secuencias, fue clasificado erróneamente como viral.

Conjunto Bacteria Meta						
Secuencias totales: *10,000* **Mal clasificadas:** *557 (5%)*						
Descripción de la clase donde se asignaron						**Núm. mal clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6,672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	247
SIN_183	138	51,017	4,261	BAW33238.1	polyprotein [Bovine viral diarrhea virus 1]	54
SIN_10	36	23,379	7,467	ATP66731.1	ORF1ab polyprotein [Rodent coronavirus]	29

Tabla 10. ***Bacteria Meta (son secuencias de contigs de datos metagenómicos reales anotados como de bacterias).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

El conjunto BacteriaMeta, que también es un conjunto no viral. De este conjunto se introdujeron al modelo 10,000 secuencias, de las cuales el 6% que corresponde a un total de 665 secuencias, fue mal clasificado como viral (tabla 11).

Conjunto BacteGenBank						
Secuencias totales: *10,000* **Mal clasificadas:** 665 *(6%)*						
Descripción de la clase asignada más probable						**Núm. mal clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	144
SIN_657	1605	462,067	3103	AQW44516.1	polyprotein [Hepacivirus C]	92
SIN_833	29	7,034	2942	AAD31543.1	polyprotein [GB virus C variant troglodytes]	67

Tabla 11. ***Bacteria GenBank (secuencias de bacterias ya anotadas en GenBank).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

EukaMetanHuman es otro conjunto que no representa virus. Como se puede observar en la tabla 12, existe el 29% de las secuencias mal clasificadas que representa un total de 862 secuencias de las 10,000 que se introdujeron al modelo.

Conjunto EukaMetaHuman						
Secuencias totales: *2,887* **Mal clasificadas:** *862(29%)*						
Descripción de la clase donde se asignaron						**Núm. mal clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6,672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	215
SIN_46	48	28,034	6,709	sp|Q98VG9.2|R1AB_FIPV	RecName: Full=Replicase polyprotein 1ab; Short=pp1ab;	169
SIN_10	36	23,379	7,467	ATP66731.1	polyprotein [Foot-and-mouth disease virus - type A]	84

Tabla 12. ***EukaMetaHuman (secuencias de contigs de datos metagenómicos de humano).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

El conjunto EukaMetaNoHuman, los cuales también no son virus, se puede observar en la tabla 13 que solo el 8.6% de las 10,000 secuencias se anotaron como virus y el 91.4% de las demás fueron descartadas como virales, lo cual es correcto ya que este tipo de secuencia no corresponde a virus sino a animales (puerco, murciélago, aves, etc.).

Conjunto EukaMetaNoHuman						
Secuencias totales: 10,000 **Mal clasificadas:** *(8%)*						
Descripción de la clase donde se asignaron						**Núm. mal clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6,672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	186
SIN_10	36	23,379	7,467	ATP66731.1	ORF1ab polyprotein [Rodent coronavirus]	104
SIN_347	138	39,722	3,535	AXU24940.1	polyprotein [Watermelon mosaic virus]	42

Tabla 13. ***EukaMetaNoHuman (Secuencias de contigs de datos metagenómicos (anotados en eucariotas) que no están anotados como de humanos).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

4.9.3. Análisis de las secuencias virales

Para el conjunto de datos FagoGenBank, como se puede apreciar en la tabla 14, se logró recuperar solo el 10.9% como virus de las 10,000 secuencias introducidas al modelo y el 89.1% se desechó lo cual no es un resultado satisfactorio ya que están secuencias representan virus.

Conjunto FagosGenBank						
Secuencias totales:10,000 **Clasificadas correctamente:** *1090(11%)*						
Descripción de la clase donde se asignaron						**Núm. bien clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	174
SIN_183	138	51,017	4261	BAW33238.1	polyprotein [Bovine viral diarrhea virus 1]	39
SIN_657	1,605	462,067	3103	AQW44516.1	polyprotein [Hepacivirus C]	31

Tabla 14. ***FagosGenBank (secuencias de fagos ya anotados en GenBank).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

En el conjunto de datos proveniente de metagenómica que corresponde a virus fagos VirusFagoMeta, como se muestra en la tabla 15 se logró recuperar el 16.12% como virus y no logró identificar el 83.38%; cómo se puede observar la anotación mejoró ligeramente en este conjunto con respecto al conjunto previamente descrito en la tabla 14 de FagosGenBank. Lo cual podría ser un indicador de que el modelo no está generalizando.

Conjunto VirusFagoMeta						
Secuencias totales:10,000 **clasificadas correctamente:** *1,612 (16%)*						
Descripción de la clase donde se asignaron						**Núm. bien clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6,672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	176
SIN_103809	40	140	250	YP_009597346.1	putative tail protein [Escherichia phage K1-ind (2)]	87
SIN_15620	25	1,619	775	YP_009113186.1	hypothetical protein LSPA1_39 [Salmonella phage LSPA1]	87

Tabla 15. ***VirusFagosMeta (secuencias de contigs de datos metagenómicos anotados como fagos).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

Del conjunto de VirusMeta que corresponde a virus de eucariontes como se muestra en la tabla 16, se logró recuperar como correctamente clasificado un 27%, es decir 319 de un total de 1,177 de secuencias que se introdujeron al modelo.

Conjunto VirusMeta						
Secuencias totales: 1,177 **Clasificadas correctamente:** 319 *(27%)*						
Descripción de la clase donde se asignaron						**Núm. bien clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_3729	34	4,624	1,617	AAZ81884.1	RdRp readthrough protein [Odontoglossum ringspot virus]	57
SIN_49668	65	593	423	CEZ26293.2	movement protein [Tomato mosaic virus]	41
SIN_1317	77	15,612	2,438	BAG82822.1	polyprotein [Enterovirus A71]	27

Tabla 16. **Virus de Eucariontes (secuencias de contigs de datos metagenómicos anotados como virus de eucariontes).** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

Los resultados que se muestran en la tabla 17 representan al grupo de FLU-15, es decir los grupos de proteína viral de la familia de virus de influenza que se excluyeron como clases del modelo por tener menos de 15 elementos, los cuales representan proteínas virales. De este conjunto se logró recuperar únicamente el 27%, esto es 109 de un total de 409 secuencias.

Conjunto FLU-15						
Secuencias totales: *409* **Clasificadas correctamente:** 109 *(27%)*						
Descripción de la clase donde se asignaron						**Núm. bien clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
FLU_101	1,192	55,842	586	pdb|5Z88|A	Chain A Hemagglutinin	18
FLU_1	277	18,173	809	pdb|6EVJ|F	Chain F Polymerase basic protein 2	16
FLU_63	760	28,735	738	pdb|4WSB|A	Chain A Polymerase PA	10

Tabla 17. **FLU-15 (secuencias de los 197 grupos de FLU que se excluyeron del modelo por tener menos de 15 elementos).** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

En la tabla 18, que representa al conjunto VIH-15, se puede ver que se logró recuperar únicamente el 23% de las secuencias, es decir 183 de un total de 789 secuencias. Al igual que el grupo anterior, estas secuencias son virales.

Conjunto VIH-15						
Secuencias totales: *789* **Clasificadas correctamente:** *183 (23%)*						
Descripción de la clase donde se asignaron						**Núm. bien clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
VIH_15	15,615	1,186,979	929	AAW83579.1	vpu protein [Human immunodeficiency virus 1]	64
VIH_2	10,707	445,885	1,499	CAD48448.1	gag-pol fusion polyprotein precursor [Human immunodeficiency virus 1]	28
VIH_227	3,837	8,013	130	AFB39601.1	rev protein [Human immunodeficiency virus 1]	19

Tabla 18. ***VIH-15 (secuencias de los 247 grupos de VIH que se excluyeron del modelo por tener menos de 15 elementos).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

Para el conjunto SIN-15, que representan proteínas virales, como se aprecia en la tabla 19, se logró recuperar sólo un 11% (206 de 1914 secuencias).

Conjunto SIN-15						
Secuencias totales: 1914 **Clasificadas correctamente:** 206 (11%)						
Descripción de la clase donde se asignaron						**Núm. mal clasificadas**
Clase	**Sec.**	**Frag.**	**Amino**	**ID GenBank**	**Descripción**	
SIN_48	110	65,172	6,672	QDY92358.1	non-structural polyprotein [Avian coronavirus]	72
SIN_347	138	39,722	3,535	AXU24940.1	polyprotein [Watermelon mosaic virus]	29
SIN_411	42	12,392	3,429	YP_009222008.1	polyprotein [Spondweni virus]	28

Tabla 19. ***SIN-15 (secuencias de 1000 grupos de SIN que se excluyeron del modelo por tener menos de 15 elementos).*** *Indica el nombre, el número de secuencias y fragmentos de las 3 clases mayoritarias asignadas a la secuencia, así como el número de aminos, ID de GenBank, y descripción de la secuencia representativa de la clase.*

CAPITULO 5. CONCLUSIONES

En este trabajo se propuso un modelo de red convolucional residual para clasificar proteínas virales en datos metagenómicos. Derivado de este trabajo de investigación se concluye que para mejorar la precisión de la asignación de secuencias se requiere tomar en cuenta varios elementos como lo son: i) Contar con un conjunto de datos de entrenamiento apropiado, donde se pueda confirmar que no exista duplicidad entre clases como lo que pudimos evaluar en el modelo con 2092 clases descrito de la sección 4.7 y probablemente el modelo de 2079 clases de la sección 4.8. ii) Verificar los elementos de reglamentación biológicos que puedan ser eficaces, para con que base a eso hacer más robusto el modelo, un ejemplo de esto podría ser: Que recientemente se ha reportado que el ensamblaje metagenómico produce contigs quiméricos que pueden contener regiones muy similares, pero de diferentes especies *(Shan et Sun, 2020),* lo cual podría agregar un sesgo de confusión para el modelo y podría afectar su desempeño. En este sentido, se puede concluir que también es necesario dotar al modelo del contexto de los millones de fragmentos que pueden ser homólogos entre distintas clases, para ello se podría usar una técnica de embedding-network o Skip-gram *(Guthrie et all.2006)* que es capaz de aprender automáticamente la relación aproximada de las palabras y de este modo ayudar a encontrar la relación y el orden de los fragmentos en las secuencias, en lugar de la empleada que fue one-hot enconded. iii) Verificar a profundidad los hiperparámetros de la arquitectura del modelo como lo son tasa de aprendizaje, agregar o disminuir capas ocultas, número de filtros, etc. iv) Por otra parte, de acuerdo con los resultados analizados, se detectó que para mejorar la clasificación de fagos posiblemente sea necesario desarrollar un clasificador específico de fagos, y que en lugar de fijar un umbral para descartar proteínas virales y no virales realizar un clasificador binario que realice el filtrado inicial de estas secuencias, y de este modo asegurar que solo se introduzcan datos de tipo viral al segundo modelo. v) Por último, en el análisis de secuencias metagenómicas derivadas de virus eucariontes, las cuales son de muy baja homología, observamos que este método de aprendizaje profundo no puede reemplazar a BLAST, que es la herramienta comúnmente usada, debido al número limitado de ejemplos que tuvimos en el conjunto de datos de entrenamiento.

REFERENCIAS

Albelwi, S., & Mahmood, A. (2017). A framework for designing the architectures of deep convolutional neural networks. Entropy, 19(6), 242.

Angermueller, C., Pärnamaa, T., Parts, L., & Stegle, O. (2016). Deep learning for computational biology. Molecular systems biology, 12(7).

Angly, F. E., Willner, D., Rohwer, F., Hugenholtz, P., & Tyson, G. W. (2012). Grinder: a versatile amplicon and shotgun sequence simulator. Nucleic acids research, 40(12), e94-e94.

Barrientos-Somarribas, M., Messina, D. N., Pou, C., Lysholm, F., Bjerkner, A., Allander, T., & Sonnhammer, E. L. (2018). Discovering viral genomes in human metagenomic data by predicting unknown protein families. Scientific reports, 8(1), 28.

Beerenwinkel, N., Günthard, H. F., Roth, V., & Metzner, K. J. (2012). Challenges and opportunities in estimating viral genetic diversity from next-generation sequencing data. Frontiers in microbiology, 3, 329.

Bileschi, M. L., Belanger, D., Bryant, D. H., Sanderson, T., Carter, B., Sculley, D., ... & Colwell, L. J. (2019). Using Deep Learning to Annotate the Protein Universe. bioRxiv, 626507.

Ekblom, R., & Wolf, J. B. (2014). A field guide to whole-genome sequencing, assembly and annotation. Evolutionary applications, 7(9), 1026-1042

Guthrie, D., Allison, B., Liu, W., Guthrie, L., & Wilks, Y. (2006, May). A closer look at skip-gram modelling. In LREC (Vol. 6, pp. 1222-1225).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

Hatfull, G. F. (2008). Bacteriophage genomics. Current opinion in microbiology, 11(5), 447-453.

Kumar, V., Maitra, S. S., & Shukla, R. N. (2015). Environmental metagenomics: the data assembly and data analysis perspectives. Journal of The Institution of Engineers (India): Series A, 96(1), 71-83.

Kulmanov, M., Khan, M. A., & Hoehndorf, R. (2017). DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. Bioinformatics, 34(4), 660-668.

Lao, Y. O., Rivas-Méndez, A., Pérez-Pravia, M. C., & Marrero-Delgado, F. (2017). Procedimiento para el pronóstico de la demanda mediante redesneuronales artificiales. Ciencias Holguín, 23(1), 1-18.

Li, W., Jaroszewski, L., & Godzik, A. (2001). Clustering of highly homologous sequences to reduce the size of large protein databases. Bioinformatics, 17(3), 282-283.

Mande, S. S., Mohammed, M. H., & Ghosh, T. S. (2012). Classification of metagenomic sequences: methods and challenges. Briefings in bioinformatics, 13(6), 669-681.

Paez-Espino, D., Eloe-Fadrosh, E. A., Pavlopoulos, G. A., Thomas, A. D., Huntemann, M., Mikhailova, N., ... & Kyrpides, N. C. (2016). Uncovering Earth's virome. Nature, 536(7617), 425.

Punta, M., Coggill, P. C., Eberhardt, R. Y., Mistry, J., Tate, J., Boursnell, C., ... & Heger, A. (2012). The Pfam protein families database. Nucleic acids research, 40(D1), D290-D301.
Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., ... & Sun, F. (2018). Identifying viruses from metagenomic data by deep learning. arXiv preprint arXiv:1806.07810.

Shang, J., & Sun, Y. (2020). CHEER: hierarCHical taxonomic classification for viral mEtagEnomic data via deep leaRning. Methods.

Scholz, M. B., Lo, C. C., & Chain, P. S. (2012). Next generation sequencing and bioinformatic bottlenecks: the current state of metagenomic data analysis. Current opinion in biotechnology, 23(1), 9-15.

Tampuu, A., Bzhalava, Z., Dillner, J., & Vicente, R. (2019). ViraMiner: deep learning on raw DNA sequences for identifying viral genomes in human samples. BioRxiv, 602656.

Wood Tomas, DeepAI: The front page of A.I. https://deepai.org/machine-learning-glossary-and-terms/softmax-layer

Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122.

Zou, J., Huss, M., Abid, A., Mohammadi, P., Torkamani, A., & Telenti, A. (2019). A primer on deep learning in genomics. Nature genetics, 51(1), 12-18.