



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CONCIENTIZACIÓN A NIÑAS Y NIÑOS CHILENOS DE 10 A 12 AÑOS SOBRE
CLASIFICACIÓN Y ALGORITMO BAYES INGENUO UTILIZANDO UNA
EXTENSIÓN AL LENGUAJE SCRATCH

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN

PATRICIO MANUEL QUIROZ VARGAS

PROFESOR GUÍA:
FRANCISCO GUTIÉRREZ FIGUEROA

MIEMBROS DE LA COMISIÓN:
PABLO GONZÁLEZ JURE
NELSON BALOIAN TATARYAN
FEDERICO MEZA MONTOYA

Este trabajo ha sido parcialmente financiado por Proyecto FONDECYT N° 11190248

SANTIAGO DE CHILE
2022

RESUMEN DE LA TESIS PARA OPTAR
AL GRADO DE MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN
POR: PATRICIO MANUEL QUIROZ VARGAS
FECHA: 2022
PROF. GUÍA: FRANCISCO GUTIÉRREZ FIGUEROA

CONCIENTIZACIÓN A NIÑAS Y NIÑOS CHILENOS DE 10 A 12 AÑOS SOBRE
CLASIFICACIÓN Y ALGORITMO BAYES INGENUO UTILIZANDO UNA
EXTENSIÓN AL LENGUAJE SCRATCH

Una de las áreas que ha tenido mayor éxito en Inteligencia Artificial (AI, por sus siglas en inglés) es Aprendizaje de Máquinas o Machine Learning (ML), debido a su utilidad para construir sistemas que detectan patrones. Más que solo automatizar procesos, los sistemas de AI están aumentando las capacidades humanas complementando labores frecuentes. Hay estimaciones para el 2022 [10], señalando que uno de cada cinco trabajadores dedicado a tareas no rutinarias, se apoyará en AI para realizar un trabajo. Esto implica que la mayoría de estudiantes actuales, en Chile de enseñanza básica o media, ingresarán al campo laboral cuando la AI esté aún más establecida. En este sentido, países como China han concretado iniciativas para educar a niñas y niños en el área [26], decretando que todos los estudiantes de secundaria —o enseñanza media en Chile— aprendan sobre AI. Otros países como EEUU, Alemania o España; también han comenzado a investigar en esta emergente área de conocimiento. Incluso, este fenómeno no es de exclusivo interés académico o científico sino que hay herramientas disponibles desde el ámbito privado, como TecheableMachine de Google y ML4Kids de IBM, para que niños y niñas comiencen a explorar conceptos como clasificación y ML.

Luego de revisar el estado del arte actual, fue posible identificar como problema la escasez de conocimiento sobre alternativas que permitan a niñas y niños adquirir nociones del tópico clasificación de texto, mediante el algoritmo Bayes ingenuo y utilizando el lenguaje de programación Scratch para su codificación. Frente a esto, el actual trabajo de tesis presenta una solución educativa que pretende justamente concientizar a escolares chilenos, sobre tales conceptos fundamentales de AI. Luego, para validar formalmente esta solución se realizó una experiencia con un grupo de niños y niñas de 10 a 12 años. Gracias a la evidencia recolectada, uno de los aprendizajes a mencionar es que, la mayor parte de las y los participantes logran programar la secuencia de pasos del algoritmo Bayes ingenuo en Scratch, no obstante un tercio aproximadamente logra describir con sus palabras las principales fases del algoritmo al clasificar texto. Esto, implica considerar esta solución como opción válida para concientizar a niñas y niños, de 10 a 12 años, sobre clasificación de texto mediante el algoritmo Bayes ingenuo utilizando Scratch.

Este trabajo está dedicado a todas y todos los niños, en especial a mi hijo León.

Agradecimientos

Comienzo agradeciendo a León, por ser mi fuente de virtud. También a Jessica, Matías, Ignacio y Francisco, mamá y hermanos respectivamente, por ser mi hogar a pesar de la distancia. En general, a toda mi familia: abuela, tios, tias, primos, primas y seres queridos; por su preocupación y aliento durante todo este periodo.

Igualmente, agradezco al Departamento de Ciencias de la Computación, en particular a la profesora Nancy Hitschfeld por su apoyo para retomar y finalizar el programa de Magíster. También, a Francisco Gutiérrez por su orientación y paciencia, junto a mis compañeros y compañeras colaboradores con este trabajo: Camila Brito, Sebastian Toro, Pablo Jaramillo, Monserrat Prado, Francisco Muñoz, Gabriel Iturra, Karina Parga y Juan Saez. Sencillamente, sin ellas y ellos no hubiese sido posible lograr este proyecto. Finalmente, este trabajo de tesis fue parcialmente financiado por el proyecto Fondecyt número 11190248, así que agradecimientos también a ANID.

Tabla de Contenido

1. Introducción	1
1.1. Problema Abordado	2
1.2. Hipótesis	4
1.3. Preguntas de Investigación	4
1.4. Objetivos	4
1.4.1. General	4
1.4.2. Específicos	4
1.5. Solución	5
1.6. Metodología	6
1.6.1. Espira 1 - Desarrollo de la Solución y Diseño de Caso	6
1.6.2. Espira 2 - Validación Preliminar	8
1.6.3. Espira 3 - Evaluación Formal con Usuarios	8
1.7. Estructura del Documento	8
2. Trabajo Relacionado	10
2.1. Iniciativas Educativas sobre AI	10
2.2. Programas Educativos para Niveles K-12	12
2.3. Plataformas y Herramientas	14
2.4. Síntesis del Estado del Arte	16
3. Desarrollo de la Solución y Diseño del Caso	18
3.1. Material Instruccional y Práctico	18
3.1.1. Análisis de Conceptos o Tópicos	18
3.1.2. Material Instruccional y Práctico	22
3.2. Extensión a Scratch	26
3.2.1. Aspectos técnicos	26
3.2.2. Desarrollo de la extensión	27
3.2.3. Descripción de bloques	27
3.2.4. Despliegue en Servidor	30
3.2.5. Comparación con otras soluciones	31
3.3. Taller como Unidad de Análisis	31
3.3.1. Propositiones del Caso	31
3.3.2. Métodos para Recolectar Evidencia	32
3.3.3. Planificación del taller	34
4. Validación Preliminar	39

4.1.	Preparativos del Caso de Estudio	39
4.2.	Primera Ejecución del Caso de Estudio	41
4.2.1.	Observación a Registros de Videos	41
4.2.2.	Respuestas en Cuestionarios	47
4.2.3.	Programas Scratch	48
4.2.4.	Aprendizajes	48
5.	Evaluación Formal	52
5.1.	Recolección de Evidencia	52
5.1.1.	Protocolo del Taller Versión 3	52
5.1.2.	Participantes	54
5.1.3.	Materiales y métodos	54
5.2.	Análisis	54
5.2.1.	Videos	54
5.2.2.	Cuestionarios	63
5.2.3.	Programas Scratch	72
5.2.4.	Análisis con <i>Pattern Matching</i>	75
6.	Discusión	84
6.1.	Aprendizajes	84
6.1.1.	Sobre Taller	84
6.2.	Implicancias de los Aprendizajes	90
6.2.1.	Teóricas	90
6.2.2.	Prácticas	93
6.2.3.	Sobre Amenazas a la Validez	94
6.3.	Revisitando las Preguntas de Investigación	97
7.	Conclusión	99
	Bibliografía	102
	Anexos	105
	A. Formulario de Consentimiento	106
	B. Formulario de Asentimiento	110
	C. Cuestionario Inicial	113
	D. Trivia con Kahoot	117
	E. Instructivo de Realización - Método CI con ML4Kids	122
	F. Instructivo de Realización - Método CI con Extensión Bayes Ingenuo	126
	G. Cuestionario para Facilitadores	131
	H. Cuestionario Verbal Probing	136
	H.1. Post CI - ML4Kids	137

H.2. Post CI - ExtBI	139
I. Conjunto de Entrenamiento Clasificador - V1	144
J. Carta Aprobación Estudio - Comité de Ética en FCFM	146
K. Cuestionario - Piloto de Exploración	149
L. Formulario de Inscripción	154

Índice de Tablas

3.1. Planificación Instruccional. Versión 1	23
3.2. Comparación Soluciones Escolares sobre AI	31
3.3. Planificación del Taller. Versión 1	35

Índice de Ilustraciones

1.1. Metodología con Action Research [8]– Espiras	7
3.1. Ilustración Árbol de Decisión con ID3 - Figura 3.1 en [15]	20
3.2. Agregar Extensión en Scratch 3.0	26
3.3. Extensiones por defecto	27
3.4. Archivos fuente creados y/o modificados	28
3.5. Diseño Bloques Scratch	28
3.6. Bloque clasificador	29
3.7. Bloque entrenar categoría	29
3.8. Bloque bayes categoría	29
3.9. Bloque maxima	30
3.10. Clasificador de Texto con Bloques Bayes ingenuo	30
3.11. Inducción Facilitadores - Primera Parte	36
3.12. Inducción Facilitadores - Segunda Parte	37
3.13. Taller - Protocolo Versión 1	38
4.1. Protocolo CI - Diagrama de actividades	40
4.2. ML4Kids - Clasificador Noticias	45
4.3. Bloque Clasifica - Versión 2	49
4.4. Clasificador de Sentimiento - Version 2	50
5.1. Taller - Protocolo Versión 3	53
5.2. ML4Kids - Clasificador Noticias Incompleto	73
5.3. ExtBi - Clasificador Noticias de una Participante	74
A.1. Formulario Consentimiento - Primera Parte	107
A.2. Formulario Consentimiento - Segunda Parte	108
A.3. Formulario Consentimiento - Tercera Parte	109
B.1. Formulario Asentimiento - Primera Parte	111
B.2. Formulario Asentimiento - Segunda Parte	112
C.1. Cuestionario Inicial - Primera Parte	114
C.2. Cuestionario Inicial - Segunda Parte	115
C.3. Cuestionario Inicial - Tercera Parte	116
D.1. Kahoot - Primera Pregunta	117
D.2. Kahoot - Segunda Pregunta	118

D.3. Kahoot - Tercera Pregunta	118
D.4. Kahoot - Cuarta Pregunta	119
D.5. Kahoot - Quinta Pregunta	119
D.6. Kahoot - Sexta Pregunta	120
D.7. Kahoot - Séptima Pregunta	120
D.8. Kahoot - Octava Pregunta	121
D.9. Kahoot - Novena Pregunta	121
E.1. Instructivo Ejecución CI ML4Kids - Primera Parte	123
E.2. Instructivo Ejecución CI ML4Kids - Segunda Parte	124
E.3. Instructivo Ejecución CI ML4Kids - Tercera Parte	125
E.4. Instructivo Ejecución CI ML4Kids - Cuarta Parte	125
F.1. Instructivo Ejecución CI ExtBi - Primera Parte	127
F.2. Instructivo Ejecución CI ExtBi - Segunda Parte	128
F.3. Instructivo Ejecución CI ExtBi - Tercera Parte	129
F.4. Instructivo Ejecución CI ExtBi - Cuarta Parte	130
G.1. Cuestionario Post Inducción - Primera Parte	132
G.2. Cuestionario Post Inducción - Segunda Parte	133
G.3. Cuestionario Post Inducción - Tercera Parte	134
G.4. Cuestionario Post Inducción - Cuarta Parte	135
G.5. Cuestionario Post Inducción - Quinta Parte	135
H.1. Verbal Probing ML4Kids - Primera Parte	137
H.2. Verbal Probing ML4Kids - Segunda Parte	138
H.3. Verbal Probing ExtBi - Primera Parte	139
H.4. Verbal Probing ExtBi - Segunda Parte	140
H.5. Verbal Probing ExtBi - Tercera Parte	141
H.6. Verbal Probing ExtBi - Cuarta Parte	142
H.7. Verbal Probing ExtBi - Quinta Parte	143
J.1. Carta Aprobación Estudio - Primera Página	147
J.2. Carta Aprobación Estudio - Segunda Página	148
K.1. Cuestionario Estudiantes de 1er Año - Primera Parte	150
K.2. Cuestionario Estudiantes de 1er Año - Segunda Parte	151
K.3. Cuestionario Estudiantes de 1er Año - Tercera Parte	152
K.4. Cuestionario Estudiantes de 1er Año - Cuarta Parte	153
L.1. Formulario de Inscripción - Primera Parte	155
L.2. Formulario de Inscripción - Segunda Parte	156

Capítulo 1

Introducción

En la actualidad, la masificación de computadores personales y distintos servicios que generan un gran volumen de datos, ha permitido incrementar el empleo de Inteligencia Artificial (AI, por sus siglas en inglés) para diseñar soluciones. Una de las sub-áreas de AI que ha tenido mayor éxito es Aprendizaje de Máquinas o Machine Learning (ML), dada su utilidad para construir sistemas que detectan patrones. Más que solo automatizar procesos, los sistemas de AI están aumentando las capacidades humanas complementando labores frecuentes. Hay estimaciones para el 2022 [10], señalando que uno de cada cinco trabajadores dedicado a tareas no rutinarias, se apoyará en AI para realizar un trabajo. Esto implica que la mayoría de estudiantes actuales en enseñanza básica o media, ingresarán al campo laboral cuando la AI esté aún más establecida. Con esta proliferación, crece también la necesidad de desarrollar habilidades y fundamentos conceptuales mínimos sobre AI, en personas de distintas edades. Como consecuencia, se hace necesario que aprendan cómo diseñar, manipular y trabajar con máquinas que usan AI.

Para que niñas y niños hoy en día aprendan cómo emplear y crear correctamente herramientas con AI, se hace imprescindible pensar cómo debería ser la enseñanza de AI en la educación pre-básica, básica y media de Chile. Esto último con la intención de contribuir, no solo a que la población esté más informada, sino que a comprender las tecnologías con que interactúan día a día e inspirar a próximas generaciones de investigadores en AI y desarrolladores de software. En este sentido, países como China han concretado iniciativas para educar a niñas y niños en el área [26], decretando que todos los estudiantes de secundaria, que en Chile corresponde a la enseñanza media, aprendan sobre AI. Esta propuesta contrasta la enseñanza tradicional, donde métodos comunes en AI como *clasificación* o *regresión* son introducidos por primera vez en la educación universitaria.

Asimismo, en Estados Unidos es posible destacar iniciativas educativas en el área como las realizadas por la organización AI4All [1]. Esta agrupación sostiene la visión de que la diversidad de perspectivas, voces y experiencias colaborando en la creación de sistemas con AI, tendría resultados que reflejan mejor a la sociedad en general. En base a esto, AI4All dispone de programas educativos sobre AI centrada en el ser humano, orientados a personas sin experiencia en AI y con bajos ingresos económicos, que históricamente han sido excluidas en el ámbito tecnológico o científico, como por ejemplo mujeres, indígenas, latinos/as,

asiáticos/as, no caucásicos/as, y quienes se autoidentifican en alguna diversidad LGTBIQ+.

No solo EEUU y China están comenzando a invertir en alternativas educativas, sino que también otros países de Europa como Alemania y España están explorando educación en AI. Así, la diversificación de trabajos en el área además de ser geográfica también es de enfoque, con propuestas que buscan acercar a niños y niñas a técnicas de ML como Aprendizaje Supervisado o Supervised Learning (SL) [19] y Aprendizaje Reforzado o Reinforcement Learning (RL) [11]. De estas iniciativas para niños y niñas, también hay variaciones respecto a la profundidad del tópico que se desea enseñar, en específico ML. Por ejemplo el trabajo de Rodríguez-García et al. [19], propone que los y las participantes comprendan la utilidad y configuración de sistemas ML, abordando en forma general las etapas de entrenamiento, aprendizaje y predicción. Mientras que Jatzlau et al. [11] intentan profundizar en aspectos de más bajo nivel computacional, revelando el algoritmo específico en un sistema con AI. Respecto a este último planteamiento, sin embargo, a la fecha no hay evidencia empírica que permita saber si niñas y niños pueden adquirir nociones sobre algoritmos considerados en ML.

Pareciera ser que la noción de educación en AI dirigida a quienes aún no poseen formación en computación, no solo está siendo considerada por instituciones educativas, sino que también se está promoviendo desde el ámbito empresarial privado. Por ejemplo Google ha disponibilizado la herramienta *Techeable Machine*¹, permitiendo la creación de modelos ML para reconocer imágenes, sonidos y posturas, generando el código de dicho modelo para luego poder utilizarlo en otros sitios o aplicaciones. En este contexto, también es posible mencionar *Machine Learning For Kids*² (ML4Kids), herramienta con el propósito de acercar a niñas y niños conceptos de SL. El sitio ML4Kids permite la construcción de un clasificador de texto, números, imágenes o sonido, generando código Scratch, App Inventor o Python.

1.1. Problema Abordado

Con la intención de estudiar y determinar los límites en el estado del arte sobre educación en AI, se definió como alcance los niveles Kindergarten hasta el nivel duodécimo (K-12) o cuarto medio. Junto a esto, y conforme a la clasificación “*Computer Science Education*” (CSE) según la *Association for Computing Machinery* (ACM), se realizó una búsqueda delimitada en las bases de datos bibliográficas *ACM Digital Library* e *IEEEXPlore*. Dada su especialización al dominio de estudio, se examinaron actas de conferencias, workshops y journals publicados en los últimos cinco años (2016 – 2021). No se recurrió a examinar publicaciones en otras editoriales, ya que a priori no se pretende realizar una revisión sistemática profunda. Definido esto, las actas de conferencias escogidas fueron:

- ACM Technical Symposium on Computer Science Education (SIGCSE);
- ACM International Computing Education Research Conference (ICER);
- ACM Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE);
- Workshop in Primary and Secondary Computing Education (WiPSCE); y

¹<https://teachablemachine.withgoogle.com>

²<https://machinelearningforkids.co.uk/#!/welcome>

- IEEE Frontiers in Education Conference (FIE).

Asimismo, se buscó en los siguientes journals:

- ACM Transactions on Computing Education; y
- IEEE Transactions on Education.

Con la definición de las fuentes, la búsqueda se hizo en base a las siguientes palabras clave o *keywords*: “*AI Education*”, “*CS Education*”, “*K-12*”, “*Artificial Intelligence*”, “*Machine Learning*”, “*CS0*”, “*CS1*” y “*Computational Thinking*”. Los criterios de inclusión específicos apuntaron a conocer objetos de estudio, tales como: currículo educativo, métodos de enseñanza, herramientas para facilitar el aprendizaje, e instrumentos para medir habilidades en AI o ML. Se encontró un conjunto de 272 artículos de forma preliminar.

El siguiente paso fue analizar el título del trabajo y *keywords* propuestos por sus autores/as. En aquellos casos donde coincidían con los criterios de inclusión específicos, se procedió a la lectura del resumen. De esta forma, se definieron las publicaciones candidatas para su estudio en mayor profundidad, de las cuales se analizó la introducción y conclusión con tal de conocer qué hicieron y sus principales hallazgos. Identificadas 101 investigaciones donde los aportes apuntaban claramente a un avance en la enseñanza o formación de niñas y niños en tópicos de AI o ML, se revisó la publicación en detalle enfocándose en su metodología y experiencia durante el desarrollo de su investigación.

Luego de leer los artículos seleccionados, se identificó la insuficiencia de evidencia empírica relacionada al desarrollo de habilidades en niñas y niños sobre algoritmos, como árbol de decisión o Bayes ingenuo, empleados frecuentemente en tareas de clasificación. Si bien, hay alternativas que —utilizando Scratch— buscan introducir en la construcción de aplicaciones con AI, no profundizan en como implementar un algoritmo usado en ML. Esta carencia de saberes se mantiene a nivel país, ya que si bien se encontró un curso —*offline*— de introducción a AI para escolares, tampoco profundiza en la definición de algoritmos usados en el área. Con estos antecedentes, en esta tesis el problema abordado es **el escaso conocimiento científico sobre alternativas educativas que permitan a niñas y niños adquirir nociones sobre clasificación de texto con el algoritmo Bayes ingenuo, usando el lenguaje de programación Scratch para su codificación.**

Este problema resulta importante, en particular para la comunidad de investigadores y profesionales en CSE, ya que contribuye a la comprensión del área. Además, brinda evidencia para al desarrollo de nuevas iniciativas, programas educativos y herramientas, útiles para la formación de niñas y niños en AI o ML. En concreto, resolver tal problemática aporta en esclarecer la factibilidad de aprendizaje sobre algoritmos en AI desde edades tempranas.

Dado esto, los resultados de esta tesis dan luces sobre cómo niñas y niños pueden desarrollar habilidades de resolución de problemas y diseño de sistemas utilizando AI desde los primeros niveles educativos. Además, resulta útil para acrecentar la inspiración en próximas generaciones de especialistas en AI, o profesionales de la computación en general. Finalmente, aporta a la comprensión de tecnologías con que la población interactúa día a día, contribuyendo a aumentar así el nivel cultural de la comunidad completa.

1.2. Hipótesis

En base a lo expuesto, el presente trabajo de Tesis definió las siguientes hipótesis:

H1 - Niñas y niños chilenos en edad escolar pueden adquirir nociones fundamentales de AI como clasificación, en forma sintetizada y en un lenguaje no-técnico, utilizando una extensión al lenguaje de programación Scratch.

H2 - Implementar aplicaciones utilizando una extensión para Scratch, permite que niñas y niños chilenos en edad escolar logren concientizarse sobre clasificación y el algoritmo Bayes ingenuo.

Dada las características de Scratch, las aplicaciones posibles de construir con este lenguaje son principalmente videojuegos, animaciones o historias interactivas.

1.3. Preguntas de Investigación

Las preguntas de investigación que permitieron abordar las hipótesis son:

PI - 1. ¿Cómo abordar el tópico de clasificación con algoritmo Bayes ingenuo, y ser entendible por niñas y niños de 10 a 12 años?. Responder esta pregunta permitió recopilar antecedentes para estudiar la validez de las hipótesis H1 y H2.

PI - 2. ¿Cómo diseñar experiencias prácticas, a desarrollar por escolares, que faciliten la comprensión del algoritmo Bayes ingenuo y programarlo con la extensión para Scratch?. Responder esta pregunta permite dar antecedentes para validar la hipótesis H2.

PI - 3. ¿Cuáles son las principales limitaciones de la extensión para Scratch al implementar el algoritmo Bayes ingenuo?. La respuesta a esta pregunta dio antecedentes para el estudio de validez sobre la hipótesis H2.

1.4. Objetivos

Para responder a las preguntas de investigación, se definieron los siguientes objetivos:

1.4.1. General

Concientizar a niñas y niños chilenos escolares sobre conceptos de AI, en particular sobre clasificación con el algoritmo Bayes ingenuo, utilizando el lenguaje de programación Scratch.

1.4.2. Específicos

OE - 1: *Determinar y analizar qué conceptos sobre clasificación con SL son prácticos y factibles de adquirir por niñas y niños en etapa escolar.* La realización de este objetivo se logró estudiando en profundidad los artículos seleccionados, y además la revisión del material y bibliografía de cursos ad-hoc como CC5206 Introducción a la Minería de Datos y CC5509 Reconocimiento de Patrones, dictados por el Departamento de Ciencias de la Computación

(DCC) de la Universidad de Chile. Con este objetivo, se aportaron antecedentes para responder a la pregunta de investigación PI - 1 y 2.

OE - 2: *Construir una extensión para el lenguaje Scratch, que incorpore nuevos bloques, útil para la implementación del algoritmo Bayes ingenuo.* El cumplimiento de este objetivo se logró extendiendo el código fuente de la última versión de Scratch 3, publicado por el grupo del MIT Lifelong Kindergarten (LLK) en su repositorio GitHub. Para la codificación de la extensión se consideraron los lineamientos oficiales de LLK, y luego se desplegó en un servidor del DCC. Con este objetivo, se aportaron antecedentes para responder a la pregunta investigativa PI - 3.

OE - 3: *Diseñar actividades teórico-prácticas que permita a escolares implementar el algoritmo Bayes ingenuo utilizando la extensión a Scratch.* Este objetivo se alcanzó con el desarrollo del material instruccional del taller como slides, guías de ejercicios y actividades, el cual fue además validado por profesionales del Centro de Modelamiento Matemático (CMM) y del DCC, ambos de la Universidad de Chile. También, se diseñaron actividades en grupos a desarrollar por niños y niñas. Cabe destacar que estas actividades colaborativas además fueron diseñadas para obtener datos cualitativos con tal de validar la solución. Más detalles sobre las actividades se describen en el Capítulo 3. Finalmente con este objetivo, se aportaron antecedentes para responder a todas las preguntas de investigación.

OE - 4: *Realizar un taller para promover a niñas y niños un acercamiento a AI, en particular sobre clasificación.* Este objetivo, se consiguió al llevar a cabo dos talleres a niñas y niños de 10 a 12 años, en los meses de enero y julio de 2021 respectivamente. Así, fue posible contribuir con antecedentes para responder ambas preguntas de investigación.

1.5. Solución

Este trabajo de tesis presenta una serie de actividades educativas secuenciales que buscan acercar y concientizar sobre AI, en particular al tópico de clasificación mediante el algoritmo Bayes ingenuo, a niñas y niños chilenos en etapa escolar. Así, esta solución se compone de: (1) una extensión para el lenguaje Scratch, (2) material educativo teórico y práctico, y (3) un taller diseñado para que, además de considerar el uso de ambas componentes, sea la unidad de análisis o caso para el estudio del comportamiento de participantes frente a la solución propuesta.

En primera instancia, y por razones de conveniencia, el alcance comprende a niñas y niños chilenos entre 10 y 12 años, y que además posean experiencia previa usando Scratch. Las razones para este campo de acción es porque niños y niñas deben tener al menos habilidades de lectura, aritmética y fracciones, así como también una noción básica en programación como el manejo de bloques tipo sensores, variables, listas y estructuras condicionales.

La extensión para Scratch incorpora nuevos bloques al entorno. Estos, denominados bloques Bayes ingenuo, además interoperan con los bloques que ofrece Scratch por defecto, posibilitando así construir un clasificador de texto. Dado que Scratch es útil para introducir la programación, los nuevos bloques además promueven un acercamiento al algoritmo Bayes ingenuo. Con esto se busca favorecer la comprensión de la técnica de Aprendizaje Supervisado

(SL, por sus siglas en inglés), empleada en ML.

El segundo elemento de la solución, las experiencias teórico-prácticas, contemplan actividades que favorecen la comprensión sobre qué es AI, clasificación con SL y finalmente el algoritmo Bayes ingenuo. No solo consideran el uso de los nuevos bloques propuestos, sino que también herramientas disponibles en el mercado como ML4Kids. Para lograr un adecuado uso de la extensión y así aplicar el algoritmo Bayes ingenuo por las y los participantes, se diseñaron sesiones sobre conceptos básicos de probabilidad, probabilidad condicional y el *Teorema de Bayes*.

Por último, el tercer componente de la solución consideró la realización de un taller diseñado para llevar a cabo las sesiones teóricas y prácticas utilizando los bloques Bayes ingenuo, y con esto ser el escenario de análisis mediante el método de investigación estudio de caso según Yin [29]. Un estudio de caso investiga un fenómeno contemporáneo (el “caso”) en su contexto real, especialmente cuando los límites entre fenómeno y contexto tal vez no son evidentemente claros. Con el taller considerado como el caso de estudio, el registro sobre el comportamiento y grado de concientización de los y las participantes en relación a los componentes de la solución, se consideró como evidencia empírica útil para confirmar — o rechazar — las hipótesis planteadas. Finalmente, con esta solución se intenta contribuir al progreso del dominio de estudio, en un contexto socio-cultural que no ha sido ampliamente explorado en la literatura específica del área.

1.6. Metodología

Para lograr los objetivos específicos, el desarrollo de los componentes fue de forma incremental aplicando la metodología Action Research [8]. En dicha metodología, se plantea un flujo de trabajo guiado por una planificación para la resolución de un problema específico, seguido por una intervención considerando una serie de acciones (dada la planificación), luego una reflexión de los efectos dadas las acciones, finalizando la iteración con una redefinición del problema para así comenzar nuevamente. De este modo, y en base a los Objetivos Específicos (OE) mencionados anteriormente, se definieron las iteraciones presentadas en la Figura 1.1.

1.6.1. Espira 1 - Desarrollo de la Solución y Diseño de Caso

La planificación de la primera espira consistió en determinar qué conceptos sobre ML, en concreto de clasificación con SL, son posibles de abstraer y considerarlos así en la solución. También se proyectó implementar la extensión para Scratch, acorde a la conceptualización sobre los tópicos escogidos. Es decir, esta iteración tuvo foco en recopilar antecedentes para los OE 1, 2 y 3. Las acciones entonces, en la primera iteración de la espira correspondieron a la lectura, análisis y triangulación de la información desde el material científico-educativo seleccionado, con énfasis sobre los tópicos AI, ML, SL y clasificación con algoritmos de árbol de decisión y Bayes ingenuo.

Una vez concretadas las tareas, la reflexión posterior determinó cuáles de estos conceptos se intentarían abstraer, y hasta qué nivel de profundidad, con el fin de ser adquiridos por niños y niñas. Finalmente se planificó comenzar la implementación de la extensión para Scratch, incorporando nuevos bloques para construir un clasificador con el algoritmo Bayes ingenuo.



Figura 1.1: Metodología con Action Research [8]– Espiras

Las acciones esta vez consistieron en implementar el algoritmo con el framework Node JS, según el repositorio oficial de Scratch.

Las acciones siguientes correspondieron entonces a definir el taller y su planificación, diseñándolo como unidad de investigación para el estudio del caso. Con la extensión Scratch implementada, se finalizó el diseño y construcción del material instruccional del taller, las actividades prácticas a realizar por las y los participantes, y se escogieron los métodos para recolectar información útil que permitiera confirmar o rechazar las hipótesis. Una vez generado todos los artefactos, se reflexionó sobre estos y se decidió hacer una primera experiencia con usuarios.

1.6.2. Espira 2 - Validación Preliminar

Se consideró probar el diseño inicial de la solución y así detectar mejoras en el material instruccional, la extensión de bloques Bayes ingenuo, y tareas operacionales —o planificación— del taller. Así, se llevó a cabo la exploración del caso con usuarios finales, escolares chilenos de 10 a 12 años. En consecuencia, ejecutar el caso no consta de solo exponer el material instruccional y ejecutar actividades prácticas, sino que también las acciones consideraron el reclutamiento de participantes, inducción a facilitadores, preparar logística según sesión del taller, coordinación con el grupo de facilitadores, y el posterior análisis al material generado por los participantes. Con esto se buscó principalmente generar antecedentes en relación al OE 4. Dado que se llevó a cabo en un periodo restringido por la pandemia COVID-19, se tomaron las acciones para que el taller fuese dictado mediante la plataforma Zoom entre los días 12 y 15 de enero del 2021, con una ceremonia de cierre donde participaron además de tutores, madres y padres de los participantes. La reflexión posterior permitió validar la metodología de instrucción del taller y alcance de los tópicos, identificando mejoras sobre cómo abordar los conceptos seleccionados, desde matemáticos hasta el algoritmo Bayes ingenuo, intentando así reducir aún más la barrera de entrada a dicho conocimiento.

1.6.3. Espira 3 - Evaluación Formal con Usuarios

En la última espira se llevó a cabo la evaluación formal de la solución, según la metodología de trabajo propuesta por Yin [29]. Se planificó la instrucción del taller, instancias de evaluación a la solución, utilizando métodos como cuestionario, observación y *Constructive Interaction* (CI) [12]. Este último, ayuda a determinar si los usuarios comprenden bien los conceptos básicos que subyacen a un sistema y si su implementación, usabilidad y utilidad son satisfactorias. La reflexión sobre los resultados del taller, así como de la evidencia generada por las y los participantes, permitió analizar el comportamiento que tuvieron en concreto con la extensión Scratch y su concientización sobre AI, ML y en particular sobre algoritmo Bayes ingenuo. Así, nuevamente se buscó aportar evidencia en torno al OE 4.

1.7. Estructura del Documento

Para describir en mayor profundidad al trabajo de tesis, el documento comprende la siguiente estructura:

- Capítulo 2: Describe el estado del arte y discute trabajos relacionados como currículos educativos, herramientas y programas para la enseñanza de AI a escolares.
- Capítulo 3: Presenta el análisis, diseño y construcción de la solución como caso de estudio.
- Capítulo 4: Explica la validación preliminar de la solución. Explora la primera intervención con niñas y niños.
- Capítulo 5: Detalla la validación formal de la solución o caso de estudio.
- Capítulo 6: Discute los resultados obtenidos e implicancias.
- Capítulo 7: Finalmente se presentan los principales hallazgos, estudiando la validez de las hipótesis, y proponiendo posibles trabajos futuros.

Capítulo 2

Trabajo Relacionado

El estado del arte relacionado a este trabajo de tesis contempla iniciativas escolares sobre AI y trabajos sobre currículos o programas educativos para niveles K-12, escuelas de verano y herramientas o aplicaciones para promover la educación de niñas y niños en AI. A continuación se presenta el cuerpo de literatura analizado.

2.1. Iniciativas Educativas sobre AI

Recientemente, la comunidad científica ha comenzado a manifestar interés en el desarrollo de estrategias para motivar el acercamiento a técnicas de AI y ML en comunidades escolares. Por ejemplo, en el año 2019, Touretzky et al. [27] de forma inédita dirigieron una sesión especial en SIGCSE, la conferencia internacional más importante sobre Educación en Ciencias de la Computación, para interactuar con investigadores en educación y AI. Además, los autores lideran el Grupo de Trabajo AI4K12¹, conformado en agosto 2018 entre la Asociación Americana para el Avance en Inteligencia Artificial (AAAI, por sus siglas en inglés) y la Asociación de Profesores en Ciencias de la Computación (CSTA, por sus siglas en inglés). La agrupación AI4K12 pretende definir las directrices sobre qué deberían conocer y ser capaces de hacer las y los estudiantes en los niveles K-12 en relación a AI, ML y robótica. Por ahora, uno de los resultados de su trabajo son las “Grandes Ideas en AI” que niños y niñas deberían conocer y manejar [26]. Las temáticas que enmarcan las Grandes Ideas son:

- 1 - *Percepción*: Las y los estudiantes deberían saber que los computadores perciben el entorno usando sensores y esto requiere de un dominio de conocimiento extenso. Por ejemplo, en sistemas de reconocimiento del habla, además de manejar sonidos, deben considerar el vocabulario, gramática y uso de patrones.
- 2 - *Representación y Razonamiento*: Las y los estudiantes deberían entender cómo representan el mundo los sistemas de AI o agentes inteligentes, y cómo en base a su percepción estos deliberan generando salidas que afectan su entorno. Además, es necesario que sepan cómo agentes inteligentes usando datos construyen su representación y las limitaciones de este modelo en su razonamiento. Por ejemplo, a pesar de que estos sistemas pueden razonar sobre problemas complejos, aún no piensan tal y como lo hacen

¹<https://ai4k12.org>

los humanos.

- 3 - *Aprendizaje*: Las y los estudiantes deben saber que los agentes inteligentes aprenden gracias a datos de entrenamiento, los cuales pueden ser facilitados por personas o adquiridos por el sistema mismo. También, deben conocer que ML es un tipo de inferencia estadística que detecta patrones en los datos, por ende es necesario que además entiendan las implicancias de sesgos en los mismos.
- 4 - *Interacción Natural*: Las y los estudiantes deben saber que algunos sistemas pueden entender lenguaje natural, en un ámbito limitado, pero carecen de capacidades conversacionales. Entender a las personas es una de las mayores tareas que enfrentan agentes inteligentes. Por ejemplo, reconocer estados emocionales e inferir intenciones desde el comportamiento observado.
- 0 - *Impacto Social*: Corresponde a una idea transversal a las anteriores, por ello su numeración. Se menciona individualmente dados los aspectos éticos y sociales que involucra. Las niñas y los niños deben ser conscientes sobre las formas que la AI está contribuyendo en sus vidas. Por ejemplo, las y los estudiantes deben entender que la ética para la construcción de agentes inteligentes, los cuales toman decisiones que afectan las vidas de las personas, requiere atención en asuntos de transparencia e imparcialidad.

Es posible encontrar otras clasificaciones o principios que deberían poseer niñas y niños. Sin embargo, los autores justifican que de este modo se abarca de mejor manera las necesidades de estudiantes y profesores de educación básica y media, o K-12.

Respecto al panorama en Oriente, Liu, Gong et. al [7] analizan cómo China ha impulsado proyectos educativos estratégicos para el desarrollo de AI. Estas iniciativas han comenzado en escuelas primarias y secundarias, con objetivo de mejorar el pensamiento creativo y la capacidad práctica en los estudiantes, fomentando la comprensión y análisis de AI. En concreto, buscan que estudiantes logren diseñar y utilizar AI con el fin de contribuir en la colaboración humano-máquina y división del trabajo, acorde al desarrollo de los tiempos. Los autores presentan el diseño de un sistema curricular considerando el modo de pensamiento, alfabetización y formación de habilidades, incluyendo proyectos básicos y prácticos. Se basa en STEAM, enfoque educativo que utiliza la ciencia, tecnología, ingeniería, artes y matemática como puntos referentes para guiar a estudiantes en su proceso. También considera y adapta el enfoque CDIO, marco conceptual educativo que enfatiza en los fundamentos de la ingeniería sobre concebir, diseñar, implementar y operar sistemas y productos del mundo real. El trabajo no presenta resultados empíricos comparables, sin embargo hace mención a los obstáculos para el desarrollo de la educación en AI, como planes de estudios fragmentados, herramientas de enseñanza ineficaces y recursos educativos desiguales en las zonas urbanas y rurales.

El año 2020, nuevamente Liu et. al. [6] presentan un nuevo estudio analizando el estado actual de la educación en AI desde Kindergarten hasta el noveno grado en Qingdao, China. Recopilaron datos mediante cuestionarios, discusiones de expertos y visitas de campo, analizando esta información desde las perspectivas de gobierno, colegios, profesores, estudiantes y padres. Los resultados muestran que, si bien en la zona geográfica de estudio se han llevado a cabo actividades educativas relacionadas con la AI para mejorar los conocimientos de los estudiantes en esta materia, aún hay por hacer para que esa educación sea más sistemática, estandarizada y personalizada. A pesar de las iniciativas promovidas por el país, como las

mencionadas previamente por el autor [7], la enseñanza de AI en los niveles K-9 se encuentra todavía en sus primeras etapas. La falta de planes de estudio y sistemas de evaluación sistemáticos de AI hace que su enseñanza y aprendizaje no sean tan eficaces como se esperaba.

Por otro lado, los autores mencionan acciones gubernamentales que establecen agencias especiales de educación en AI con el propósito de promover la educación en el área, incrementando los fondos para proyectos que apoyen proyectos de investigación y de enseñanza en AI; y también fortaleciendo el desarrollo de plataformas para actividades de estudiantes y disponibilizar recursos de AI. Finalmente, algo importante que realizan estas agencias, es estimular a universidades, institutos de investigación científica, y museos de ciencia y tecnología, a ofrecer ciclos de conferencias exposiciones y jornadas de puertas abiertas para estudiantes y así enriquecer sus experiencias con la AI. El estudio concluye con una serie de recomendaciones políticas, sobre construcción de infraestructura, entrenamiento de profesores, desarrollo de curriculums o planes de estudio, desarrollo de plataformas, consideraciones éticas y propiedad intelectual.

2.2. Programas Educativos para Niveles K-12

Respecto a innovación en currículos educativos sobre AI, Lee et al. [14] desarrollaron un taller de 30 horas con el fin de contribuir a formar ciudadanos informados y consumidores críticos de tecnología con AI. Proponen la concientización sobre la adopción de AI en los trabajos futuros y un cuestionamiento ético en AI. Las autoras presentan sus hallazgos al realizar un taller, online debido a la pandemia, denominado “*Developing AI Literacy*” o DAILY basado en su propuesta de curriculum, con estudiantes subrepresentados en STEM y computación entre 10 y 14 años.

El taller DAILY tuvo 31 participantes, de los cuales las autoras entrevistaron a 11 voluntarios, determinando que la comprensión conceptual de los estudiantes fue significativo sobre AI, asimilando los posibles prejuicios de AI y la capacidad de adaptarse a los futuros trabajos potenciados por esta área. Los y las participantes pudieron reflexionar sobre como stakeholders en sistemas de AI pueden diseñarlos basados en sus valores y necesidades, identificando así cómo y por qué las aplicaciones de AI pueden ser injustas para ellos mismos y para los demás. A pesar de una alta tasa de asistencia por cada unidad (85 % - 95 %), los porcentajes de estudiantes que entregaron el trabajo terminado oscilaron entre el 80 % y el 30 %, pudiendo justificarse por la modalidad del taller, limitaciones con las herramientas usadas como Techeable Machine y Scratch, o la potencia de sus computadores.

Continuando con programas educativos en AI, Kandlhofer et al. [13] presentan un modelo desde Kindergarten hasta el ultimo nivel obligatorio (K-12), que en Chile correspondería hasta cuarto medio. Una característica de este trabajo es la agrupación en tres bandas o módulos. El primero desde Kindergarten hasta quinto nivel; el segundo desde sexto a octavo; y el tercero lo que en Chile sería la enseñanza media completa. En el primer y segundo módulo, la propuesta de los autores se basa principalmente en la enseñanza de conceptos fundamentales en Computer Science (CS) como grafos, estructura de datos y algoritmos de búsqueda. Para el tercer módulo, además de continuar con la definición conceptos básicos en CS, se incorpora la enseñanza de agentes inteligentes y autómatas. En términos prácticos solo considera una discusión grupal sobre ML sin ahondar sobre conceptos propios, por ejemplo

clasificación. Es interesante que el currículo educativo presentado comprenda todos los niveles escolares obligatorios; sin embargo, podría mejorarse ahondando en tópicos propios de AI en todos los módulos, no solo en los superiores.

Vachovsky et al. [28] diseñaron un currículo educativo y una escuela de verano en AI destinada a niñas, de forma novedosa y con gran proyección aliándose con universidades estadounidenses como Stanford, Columbia, Princeton y otras. Con una perspectiva de diversidad de género, en esta investigación las autoras proponen alcanzar tres objetivos: (1) incrementar el interés en AI, (2) contextualizar y adoptar conceptos técnicos a través del impacto social, y (3) detectar barreras que podrían desalentar a niñas de 15 años en CS. Para esto, el programa educativo no solo considera la instrucción sobre tópicos en salas de clase, sino que actividades de sociabilización-educativa como desayunos diarios con un profesor que luego expone sobre un tópico en particular y posibles proyectos; sesiones para un desarrollo personal y profesional, tales como identificar y reaccionar a microagresiones, comunicación de equipo y diseño de sistemas interactivos. También, diariamente las participantes trabajaban en un proyecto grupal cada uno liderado por dos estudiantes graduados, en temáticas como visión por computador, robótica, procesamiento de lenguaje natural y biología computacional; teniendo como objetivo aumentar la comprensión de las estudiantes sobre los métodos técnicos utilizados en AI. La codificación de todos los proyectos se realizó en Python usando el entorno de programación Jupyter.

Como resultado, destacan que se logró cambiar la percepción de las estudiantes sobre AI y CS. Mediante los proyectos técnicos grupales y las actividades de generación de apoyo y crecimiento personal, las participantes lograron entender que una amplia variedad de personas se pueden desenvolver con éxito en el área, desarrollando así sus iniciativas propias. A pesar de esto, no fue posible apreciar el incremento de interés en las participantes y las/os autores especulan que las estudiantes ya venían con un alto interés dado su criterio de admisión. Su trabajo logró ser un camino efectivo para educar a estudiantes en AI y el impacto social que tienen las CS. Luego de los positivos resultados en esta experiencia, es viable pensar en diseñar y ejecutar una iniciativa similar, adecuada a niñas y niños chilenos en edades similares.

Otro proyecto a mencionar, ocurrió en Chile durante la escritura de esta tesis. Fundación Telefónica ofrece un curso web *offline*², permitiendo a participantes crear videojuegos considerando reconocimiento automatizado de objetos. El programa propuesto por dicha organización aborda el concepto de SL usando la plataforma Code.Org³. Finalmente, también ofrece un tutorial en Scratch, en específico para ML4Kids, con tal de construir un programa que permita reconocer imágenes y jugar “*piedra, papel y tijera*”. Se podría decir que este curso *offline*, dispone de videos y tutoriales sobre SL con ML4Kids, sin embargo no profundiza sobre la definición de algún algoritmo en ML.

²<https://campus.fundaciontelefonica.movistar.com/aula-virtual-profuturo/creando-videojuegos-con-inteligencia-artificial-autoaprendizaje/>

³<https://studio.code.org/s/oceans/lessons/1/levels/1>

2.3. Plataformas y Herramientas

Según Resnick et al. [17], las experiencias con objetos físicos son útiles para facilitar el aprendizaje de conceptos abstractos. En esta línea, Hitron et al. [9] proponen la adopción de tópicos relacionados a ML en niños entre 10 y 12 años utilizando Scratch Node⁴, un dispositivo programable para su uso al aire libre. Para esto, utilizaron el esquema definido por Sun [23], quien descompone ML en seis categorías de conceptos: (1) etiquetado de datos, (2) extracción de características, (3) selección del modelo, (4) ajuste de parámetros, (5) evaluación y (6) aplicación en el mundo real. En particular, enfocaron esfuerzos en medir la adquisición de (1) y (5) considerándolos como claves para la experiencia de aprendizaje. Dentro de los hallazgos destacan que a los participantes les fue posible explicar con propias palabras qué es clasificación y asociarlo a nuevos contextos, no solo el experimentado. Además, indican que algunos elementos claves de (1) fueron simples de entender, como por ejemplo el tamaño de la muestra; mientras que otros fueron complejos en su adquisición, por ejemplo, versatilidad y ejemplos negativos. En los casos donde a los participantes les fue difícil entender, hallaron que en la mayoría una retroalimentación o *feedback* articulado les permite finalmente comprender el tópico. Con esto último, destacan que los tipos de retroalimentación apropiados pueden mejorar enormemente el aprendizaje de niñas y niños, por lo cual los diseñadores de sistemas deberían introducir tal característica para apoyar gradualmente el entendimiento.

Existe una estrecha relación entre AI y Ciencia de Datos (DS, por sus siglas en inglés). Por ejemplo, ambas disciplinas se basan en grandes cantidades de datos como insumo principal, compartiendo tópicos y métodos claves en cada área. En este contexto, Dasgupta y Hill [3] presentan “Scratch Community Blocks”, una extensión para Scratch que permite utilizar los datos de su comunidad online, con el propósito que niñas y niños realicen actividades de DS, como visualización y análisis. En este trabajo se proponen dos objetivos principales: (1) apoyar al aprendizaje con actividades creativas en un contexto social (utilizando la comunidad de Scratch); y (2) apoyar la autorreflexión y conocimiento debido al propio aprendizaje. Así como el lenguaje Logo, propuesto por Papert et al. [16] fue diseñado para dar foco al aprendizaje de las y los niños más que solo intentar convertirlos en programadores, “Scratch Community Blocks” fue diseñado no solo para hacer DS sino que puedan visualizar, representar y responder a propias preguntas usando datos reales. Si bien en esta trabajo no incursionan directamente en AI, la cercana relación entre esta disciplina y DS permite rescatar la forma y cómo abordan el problema.

Asimismo, Dhariwal en su trabajo de tesis [4], para optar a Master en Artes y Ciencias de la Comunicación del MIT Media Lab, presenta la herramienta *Let’s Chance* la cual promueve el pensamiento probabilístico en niñas y niños. *Let’s Chance* es un micromundo de Scratch, es decir una adaptación al lenguaje de programación con nuevos tipos de bloques y representaciones, que facilita y promueve la experimentación con ideas fundamentales del pensamiento probabilístico. Con esta herramienta, niños y niñas pueden codificar proyectos probabilísticos generativos sobre arte, música o texto. Por ejemplo, es posible construir juegos e historias basadas en el azar, visualizaciones y simulaciones interactivas, e incluso proyectos avanzados para hacer que el computador aprenda a partir de datos de entrada utilizando modelos simples de Markov. Así, es posible considerar a *Let’s Chance* como una alternativa para facilitar el aprendizaje de conceptos probabilísticos, pudiendo entonces ser útil para

⁴<http://milab.idc.ac.il/teaching/projects/scratch-nodes/>

abordar tópicos matemáticos requeridos en algoritmos de AI.

Por otro lado, La herramienta ML4Kids⁵ es considerada una alternativa educativa para introducir a ML y SL, mencionada en distintos trabajos [11, 14, 19]. Sin embargo, en las actas de conferencias escogidas no se encontró alguna publicación de los creadores o autores de esta herramienta. A pesar de esto, se puede decir que es un sistema web útil para introducir en ML a través de experiencias prácticas. Los usuarios pueden entrenar y construir clasificadores de texto, números, imágenes o sonidos. Para esto, ofrece en una interfaz de usuario con los tres principales pasos de ML supervisado: entrenamiento, aprendizaje y evaluación, generando bloques en Scratch, AppInventor⁶ o código Python. Se podría indicar entonces que esta herramienta consta de dos componentes: el sitio donde se define el modelo con SL, y otro sitio (Scratch 3 por ejemplo) donde se puede hacer uso de este modelo dado los bloques o código generado. Además, internamente esta plataforma utiliza IBM Watson⁷ como motor o API de AI, soportando así el código generado. Otra de las funcionalidades de ML4Kids es que posibilita agrupar un conjunto de perfiles de usuarios con un moderador, permitiendo así ser utilizado por escuelas, clubes de código y familias.

Una plataforma que apoya la enseñanza y aprendizaje sobre fundamentos de ML, semejante a ML4kids, mediante la creación de proyectos prácticos con Scratch, es LearningML [19]. Sus autores exploraron cómo un seminario web sobre AI y la creación de proyectos de programación con LearningML impacta en los conocimientos sobre AI de estudiantes de entre 10 y 16 años. LearningML presenta dos diferencias con ML4Kids, (1) no depende de ningún servicio externo de ML o API, ejecutando localmente el algoritmo de ML; y (2) tanto el conjunto de datos usados en un proyecto como el código creado pueden descargarse localmente y recuperarse de nuevo para un uso posterior. Respecto a los bloques Scratch utilizados, se puede mencionar que hay un grupo destinado para clasificar texto, y otro conjunto de bloques para clasificar imágenes.

Los autores de LearningML también proponen un test que incluye diferentes tipos de preguntas basadas en investigaciones y publicaciones anteriores, para medir si los y las participantes sin conocimientos previos sobre AI o ML pueden aprender los fundamentos del ML a través de actividades prácticas con la herramienta propuesta. El test o cuestionario lo aplicaron pre y post webinar. Así, los autores analizaron las respuestas de 135 estudiantes, reconociendo que al ser online muchos aspectos del proceso no podían ser controlados. Sin embargo, aseguran que los resultados demuestran que los y las participantes son capaces de aprender sobre AI. Los autores también concluyen, que mientras las ideas previas sobre AI giran en torno al término robot, después de la experiencia lo hacen en torno a la solución y al problema. Finalmente los investigadores destacan que la característica más interesante de LearningML quizás sea la posibilidad de compartir los modelos y proyectos creados por los usuarios, permitiendo así la creación de un repositorio a gran escala de la actividad, pudiendo ser útil para otras investigaciones.

Una propuesta que apunta a comprender en mayor grado sobre computación y ML, particularmente en Aprendizaje Reforzado (RL, por sus siglas en inglés), es la que presentan

⁵<https://machinelearningforkids.co.uk/#!/welcome>

⁶<https://appinventor.mit.edu>

⁷<https://www.ibm.com/watson>

Jatzlau et al. [11]. RL es un paradigma inspirado en la psicología donde el agente aprende mediante la recompensa y el castigo, logrando dominar una determinada tarea de forma autónoma a través de la interacción con su entorno intentando maximizar la recompensa total. Los autores señalan que los frameworks existentes dirigen a los principiantes y abren una vía sencilla e intuitiva para que comprendan los conceptos de ML sin el obstáculo que supone la programación. Sin embargo, estos enfoques existentes comparten un rasgo común: la implementación del marco de AI subyacente sigue siendo una caja negra. Con esto último, el grado en que un usuario puede “mirar dentro” es extremadamente limitado. Como resultado, los y las aprendices a menudo no pueden obtener una comprensión más profunda del proceso real de ML.

En base a esto, los autores en el mismo trabajo proponen cuatro principios de diseño sobre ML adecuado para su aprendizaje en aulas, y también presentan SnAIp una herramienta destinada para la enseñanza de RL desarrollada completamente en el entorno *Snap!*⁸. SnAIp está pensado para su uso desde los 14 años, y se probó en una clase del nivel décimo. En particular SnAIp implementa el algoritmo Q-Learning, y todo el comportamiento se implementa en bloques de Snap!, es decir no hay wrapper de código JavaScript externo o llamadas a algún servidor. Finalmente, los autores destacan que aunque SnAIp puede utilizarse individualmente, también puede incorporarse a un plan de estudios completo, sin embargo los autores no presentan resultados empíricos sobre esto.

2.4. Síntesis del Estado del Arte

Dado todo lo anterior, la inspiración para el presente trabajo de Tesis se basa en que las alternativas sobre educación en AI mencionadas permiten una comprensión superficial sobre fundamentos computacionales en ML y clasificación, manteniendo así cierto nivel de hermetismo sobre CS. Con las soluciones existentes, no sería viable que niñas y niños comprendan los fundamentos de SL o clasificación, como por ejemplo el funcionamiento, pasos o acciones, que considera un determinado algoritmo empleado en el área.

Por ejemplo, en distintos planes de estudios sobre AI para escolares, los objetivos de aprendizaje se limitan a que niños y niñas comprendan, a un alto nivel de abstracción, conceptos sobre AI o ML. Los resultados muestran que efectivamente las y los participantes logran adquirir nociones sobre tales tópicos, como por ejemplo comprender la importancia de un dataset o conjunto de entrenamiento en ML y cómo utilizarlo en un clasificador; pero tales habilidades adquiridas podrían considerarse como ambientación o configuración de una herramienta, manteniendo la reserva lo que ocurre internamente en esta. El currículo usado en el taller DAILY [14] considera la enseñanza del algoritmo árbol de decisión para clasificar, pero no incluyen actividades sobre su implementación en un lenguaje de programación como Scratch; y este trabajo no presenta evidencia en relación si los y las participantes lograron comprender dicho algoritmo. Por otro lado el trabajo de Vachovsky et al.[28] y su programa de escuela de verano para niñas de nivel decimo, considera la participación de un grupo en un proyecto de procesamiento de lenguaje natural, no evidencian si ellas logran comprender el funcionamiento del algoritmo utilizado y programado Python. Es probable que conceptos relacionados con AI puedan resultar complejos de comprender en determinados contextos,

⁸<https://snap.berkeley.edu>

pero comenzar con su educación de forma temprana aumenta la posibilidad de adquirirlos.

En relación a herramientas, como LearningML⁹, ML4Kids¹⁰ o Teachable Machine¹¹, estas resultan ser útiles para presentar la utilidad de ML y SL, permitiendo configurar un clasificador con bloques en Scratch. Sin embargo, no transparentan cómo y por qué se predice una determinada clase o categoría. Por otro lado SnAIP [11] permitiría a las y los usuarios comprender los conceptos algorítmicos subyacentes, sin embargo los resultados de este trabajo, no evidencian si es posible que niñas y niños comprendan el funcionamiento del algoritmo-Q empleado en RL.

Con estas limitaciones actuales no es posible solucionar de manera completa el problema presentado en esta tesis, en relación a la carencia de alternativas educativas que permitan a niñas y niños comprender a nivel algorítmico cómo se logra una predicción, o clasificación, de un determinado elemento. Por esto, contar con una solución educativa que aborde los tópicos de AI y ML con SL concientizando sobre su uso, y a la vez profundizando en conceptos fundamentales, aportaría a que niños y niñas comprendan de forma más amplia el área. Así, en este trabajo se presenta como alternativa educativa un taller con Scratch, de tal forma que logren codificar con bloques el algoritmo Bayes ingenuo para clasificar texto, permitiendo a niñas y niños crear aplicaciones significantes para ellas y ellos como videojuegos o animaciones. Esta solución permite complementar las soluciones en la actualidad, aportando particularmente a transparentar lo que acontece cuando se realiza clasificación en ML. Por consiguiente, también aportaría a equiparar el trabajo realizado en países desarrollados respecto a prácticas y actividades educativas, pudiendo posicionar al país como pionero en la región sobre educación en AI.

⁹<https://web.learningml.org>

¹⁰<http://machinelearningforkids.co.uk/>

¹¹<https://teachablemachine.withgoogle.com>

Capítulo 3

Desarrollo de la Solución y Diseño del Caso

En este capítulo se presenta el análisis, diseño y construcción del taller y la extensión a Scratch. Con esta solución, se intenta solventar la carencia de iniciativas educativas que permitan concientizar sobre clasificación con el algoritmo Bayes ingenuo, y por consiguiente adquirir nociones sobre AI y ML comprendiendo parte de su potencial y limitaciones. De este modo, y dada la metodología de investigación presentada en la sección 1.6.1, se describe la primera espiro y sus iteraciones para la obtención de cada componente en la solución.

3.1. Material Instruccional y Práctico

Dada la planificación, se comenzó con la profundización de los conceptos técnicos sobre clasificación con los algoritmos Árbol de decisión y Bayes ingenuo. Determinado los tópicos y su profundidad, se analizaron y diseñaron actividades que permitirían su comprensión.

3.1.1. Análisis de Conceptos o Tópicos

Además de considerar las soluciones y trabajos relacionados mencionados en el Capítulo 2, se consideró el material educativo de los cursos CC5206 Introducción a la Minería de Datos y CC5509 Reconocimiento de Patrones, dictados por el DCC en la Universidad de Chile. También, en base a la bibliografía de los cursos señalados, se revisaron los libros *Artificial Intelligence A Modern Approach*, 3era edición de Russell y Norvig [20]; *Machine Learning*, 1era edición de Mitchell [15]; *Data Mining, Practical Machine Learning Tools and Techniques*, 2da edición de Tan et al.[24]; y *Pattern Recognition*, 4ta edición de Koutroumbas y Theodoridis [25]. De esta forma, el análisis de conceptos con triangulación de la información desde las distintas fuentes, permitió determinar los siguientes tópicos a ser considerados en la solución:

1. Qué es AI. Definición desde las ciencias de la computación, estructura general de algoritmos de AI y aplicaciones cotidianas.
2. Qué es ML. Cómo se relaciona con AI.

3. Técnica SL. Proceso y utilidad para la toma de decisiones.
4. Clasificación. Empleo de algoritmo con SL para determinar una categoría determinada.
5. Fundamentos de programación con Scratch y tipos de bloques.

Adquirido un dominio en los conceptos, el enfoque estuvo en estudiar los algoritmos árbol de decisión y Bayes ingenuo, e identificar posibles puntos de abstracciones en estos con tal de que pudiesen ser comprendidos por personas entre 10 y 12 años. Para esto, se implementaron dichos algoritmos en los lenguajes Java y luego Javascript, siendo este último la base de Scratch 3.

Algoritmos de Clasificación

Una técnica de clasificación —o clasificador— es un enfoque sistemático para construir modelos de clasificación, es decir determinan una determinada clase o categoría, a partir de un conjunto de datos de entrada [24]. Algunos ejemplos de algoritmos son los clasificadores de árbol de decisión, redes neuronales y Bayes ingenuo. El modelo generado por un algoritmo de aprendizaje debe ajustarse bien a los datos de entrada y predecir correctamente las etiquetas de los registros que no ha visto antes. Por lo tanto, un objetivo clave del algoritmo de aprendizaje es construir modelos con buena capacidad de generalización; es decir, modelos que predigan con precisión la etiqueta o clase para registros desconocidos.

Árbol de Decisión ID3. El algoritmo con árbol(es) de decisión permite una aproximación de funciones objetivo de valor discreto, en el que la función aprendida está representada por un árbol de decisión [15]. Hay variaciones para este algoritmo, entre las que se puede encontrar ID3, C4.5 y CART. Estos algoritmos se han aplicado con éxito a una amplia gama de tareas, desde el diagnóstico en casos médicos, hasta la evaluación de riesgo crediticio para solicitantes de préstamos, según Mitchell [15].

Los árboles definidos pueden representarse como conjuntos de reglas *si-entonces*. Dada esta ilustración que facilita la legibilidad humana, se escogió como primera opción para intentar concientizar sobre clasificación a niñas y niños. En esta tesis, se consideró el algoritmo ID3 propuesto por J.R. Quinlan, publicado en el libro *Machine Learning* de Mitchell [15]. Por ejemplo en la Figura 3.1, se representa un árbol de decisión generado por el algoritmo ID3, en donde se intenta clasificar si es un buen día para *Jugar Tenis* dado el pronóstico climático en términos del *Viento*, *Humedad* y *Nivel de Nubosidad*. La figura aquí, es una traducción de la Figura 3.1 del libro citado.

ID3 no es factible de implementar con Scratch. Con la elección del algoritmo ID3 se analizó cuáles componentes y funciones de este podrían abstraerse con tal de ser implementado mediante bloques de Scratch. La principal dificultad que se encontró fue que el algoritmo ID3 realiza recursión, es decir, internamente hace invocación a sí mismo. Dada la edad de niñas y niños a quienes va dirigido este trabajo, que probablemente están comenzando a introducirse en programación, es complejo abordar el tópico de recursión ya que como mínimo requiere tener una base en manejo de funciones y retorno de valores. En concreto, Scratch es un lenguaje donde las funciones no son de primera clase y por ende no es posible tratarlas como variables. Si bien Scratch 3.0 está construido con Javascript, en específico el framework NodeJS, el entorno abstrae las características de las funciones. En Scratch es

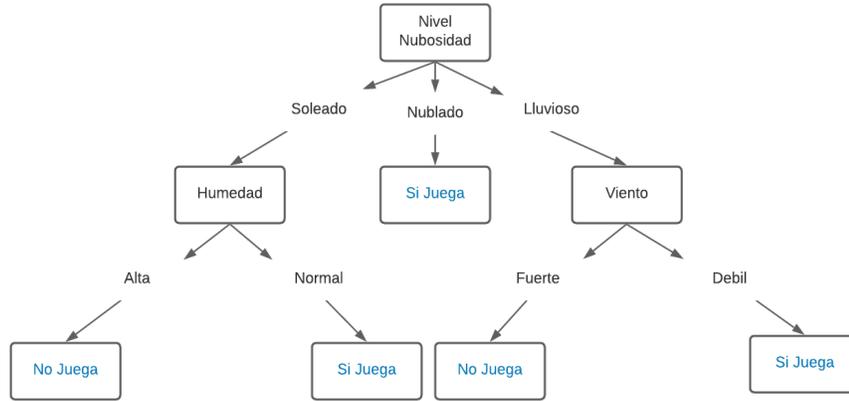


Figura 3.1: Ilustración Árbol de Decisión con ID3 - Figura 3.1 en [15]

posible crear bloques que agrupan un conjunto de instrucciones, pero no permite retornar un bloque función, o considerar como parámetro otra función. Probablemente en otro entorno de programación, como por ejemplo Snap! donde las funciones sí son de primera clase, podría implementarse algoritmo ID3 o C4.5. Luego, en este trabajo de tesis se decidió continuar con el empleo de Scratch, dado que es idóneo para introducir en la programación a niñas y niños. De este modo, el algoritmo árbol de decisión fue descartado y por ende, para cumplir con el objetivo general de concientizar a niñas y niños en algoritmos de clasificación, se optó por considerar exclusivamente el algoritmo Bayes ingenuo.

Bayes Ingenuo. Este algoritmo está estrechamente relacionado con el *teorema de Bayes*, proposición matemática planteada por Thomas Bayes que determina la probabilidad de un *evento A* dado un *evento B* en términos de la probabilidad del *evento B* dado el *evento A*. El teorema de Bayes, también conocido como regla o ley de Bayes, se define en la Ecuación 3.1.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (3.1)$$

De esta forma, el algoritmo Bayes ingenuo provee un enfoque probabilístico de inferencia. En clasificación, a menudo se emplea para determinar la mejor hipótesis (clase o categoría) de un espacio H , dados los datos de entrenamiento D . Con la mejor hipótesis se refiere a la hipótesis más probable dado D . El teorema de Bayes proporciona un método directo para calcular tales probabilidades, como se presenta en Ecuación 3.2. En clasificación se puede considerar a cada h como la categoría o clase indicada en la fase de entrenamiento, aplicando así la técnica SL.

$$P(h|D) = \frac{P(D|h) * P(h)}{P(D)} \quad (3.2)$$

En Ecuación 3.2, $P(h)$ denota la probabilidad inicial de que la hipótesis h se cumpla, antes de haber observado los datos de entrenamiento. El término $P(D)$ se refiere a la probabilidad

previa de que se observen los datos de entrenamiento D (es decir, la probabilidad de D sin saber qué hipótesis se cumple). Luego, $P(D|h)$ se refiere a la probabilidad de observar los datos D dado un escenario donde se cumple hipótesis h , es decir, la probabilidad condicional. En problemas de ML se busca determinar la probabilidad $P(h|D)$, o sea, la probabilidad de h dados los datos de entrenamiento considerados D . A la probabilidad $P(h|D)$ también se le denomina como *probabilidad a posteriori* de h . Finalmente, el algoritmo Bayes ingenuo se define de forma general con el siguiente pseudocódigo:

1. Por cada h en H , se calcula la probabilidad a posteriori
2. Seleccionar la h con mayor probabilidad a posteriori

Entonces, el resultado de la clasificación con Bayes ingenuo, corresponde a la categoría o clase con mayor probabilidad para un nuevo valor o registro. En algunos dominios, el rendimiento del clasificador con Bayes ingenuo ha demostrado ser comparable al de redes neuronales y árboles de decisión [15]. Además, es útil para clasificar tipos de datos como cadena de texto, o datos con alguna estructura o tabulación.

Clasificando texto con Bayes Ingenuo. Por ejemplo, el algoritmo Bayes ingenuo en clasificación de texto, define cada ítem —documento de texto o “*doc*”— según las palabras que contiene este, es decir en sus atributos (a_1, a_2, \dots, a_n) . Así, el primer paso del algoritmo “*por cada h en H calcular probabilidad a posteriori*”, queda:

$$\begin{array}{ll}
 P(h_j|doc) & h_j \in H \\
 P(h_j|a_1, a_2, \dots, a_n) & h_j \in H \\
 \frac{P(a_1, a_2, \dots, a_n|h_j) * P(h_j)}{P(a_1, a_2, \dots, a_n)} & h_j \in H \\
 P(a_1, a_2, \dots, a_n|h_j) * P(h_j) & h_j \in H
 \end{array}$$

En la última expresión, se hace una simplificación del teorema de Bayes quitando el denominador. Este valor corresponde a la probabilidad del documento, factor común en la probabilidad a posteriori para cada h o categoría.

$$\begin{array}{ll}
 P(a_1, a_2, \dots, a_n|h_j) * P(h_j) & h_j \in H \\
 P(a_1|h_j) * P(a_2|h_j) * \dots * P(a_n|h_j) * P(h_j) & h_j \in H
 \end{array}$$

Ahora, se aplica la noción de ingenuidad al considerar que cada probabilidad, para una palabra en el texto, es independiente de las posiciones del resto de palabras. Esta suposición en la práctica funciona bien, pero notar que en la realidad esta apreciación es incorrecta. Por ejemplo, la probabilidad de observar la palabra “*learning*” en alguna posición puede ser mayor si la palabra precedente es “*machine*”. Finalmente, el segundo paso “*seleccionar la h con mayor probabilidad a posteriori*” se puede definir:

$$\begin{aligned}
 \max_{h_j \in H} & P(a_1|h_j) * P(a_2|h_j) * \dots * P(a_n|h_j) * P(h_j) \\
 \max_{h_j \in H} & P(h_j) * \prod_i P(a_i|h_j)
 \end{aligned}$$

Abstracción de funciones en Bayes Ingenuo. Cómo se puede apreciar, clasificación con Bayes ingenuo requiere de operaciones matemáticas, en concreto nociones de probabilidad condicional y tablas de contingencia, tópicos que se abordan en cursos de enseñanza media en Chile. Una opción para que niños y niñas de 10 a 12 años logren comprender la secuencialidad de pasos del algoritmo, es descomponer en porciones a Bayes ingenuo. Con esto, se puede aislar y abstraer la complejidad de cálculo matemático en funciones, posibilitando comprender la utilidad de cada una de estas y rol en la secuencia de Bayes ingenuo. Así, se definieron los siguientes procedimientos, base de los bloques Scratch desarrollados:

entrenar. Construye la tabla de contingencia, manteniendola en memoria, para una categoría en base a ejemplos dados. Tal estructura queda en memoria. Esta función requiere dos parámetros, “*valor_categoria*” correspondiente al identificador o *id* de la categoría que se está entrenando, y el parámetro “*ejemplos*” es la lista con textos ejemplos que entrenará dicha categoría:

```
entrenar(valor_categoria , ejemplos) -> void
```

bayes. Esta función calcula la probabilidad a posteriori de una categoría para un nuevo texto. En relación a los parámetros necesarios, además de necesitar el *id* o “*valor_categoria*”, requiere de “*nuevo_valor*” correspondiente al nuevo texto. Esta función considera la simplificación del teorema de Bayes (no considerando el denominador) y la suposición de independencia —o ingenuidad— en los componentes del nuevo texto:

```
bayes(valor_categoria , nuevo_valor) -> numero
```

maxima. Función que determina la categoría con probabilidad a posteriori mayor, según las indicadas en el entrenamiento. Con tal de facilitar la comprensión del algoritmo, se define un parámetro “*lista_resultados_bayes*” el cual corresponde a los resultados calculados para cada probabilidad a posteriori.

```
maxima(lista_resultados_bayes) -> string
```

3.1.2. Material Instruccional y Práctico

Dada las características de Bayes ingenuo, para concientizar a escolares sobre el tópico de clasificación con este algoritmo, se determinó incorporar una introducción sobre los siguientes conceptos:

- Probabilidades
- Probabilidad Condicional
- Fórmula del Teorema de Bayes

Con los tópicos sobre CS, AI y matemática ya acotados, se intentó hacer una abstracción de estos de manera tal de ser comprensibles por personas de 10 a 12 años. Lo primero entonces fue buscar material educativo existente, analizarlo y determinar cuál podía ser útil, y cuál habría que crear completamente. Para las actividades matemáticas fueron seleccionadas

Tabla 3.1: Planificación Instruccional. Versión 1

AI y SL	Fundamentos Programación	Probabilidades y Bayes	Algoritmo Bayes Ingenuo
Introducción a AI	Algoritmos	Probabilidad	Algoritmo Bayes Ingenuo
Introducción a SL y Clasificación	Scratch	Probabilidad Condicional	Crear Clasificador con Scratch
		Fórmula de Bayes	

las publicadas en los textos oficiales para los niveles 5^o Básico hasta 4^o Medio, según el Currículum Nacional¹ definido el Ministerio de Educación (MINEDUC) chileno. Respecto al material sobre los tópicos de CS y AI, se profundizó en la bibliografía de los artículos mencionados en el Capítulo 2 y los libros presentados a inicio del actual Capítulo. De esta forma la primera versión de la planificación se presenta en la tabla Tabla 3.1.

Material Instruccional

Tópicos de AI y SL. El material para intentar explicar a niñas y niños los conceptos de AI y SL se basó en las primeras unidades del currículo DAILY [5], “¿Que es AI?” e “Introducción a Aprendizaje Automático Supervisado” disponibles en su sitio oficial². Al analizar este material, se filtró y adaptó en base al objetivo propuesto en este trabajo de tesis. Por ejemplo, se consideró instruir sobre ML4Kids para que las y los participantes practicaran la técnica de SL. El resultado fue validado con una experta en el dominio.

Fundamentos de Programación. Respecto a este tópico, el material fue construido completamente. Así, la actividad de instrucción tuvo el propósito de definir qué es un algoritmo y su posible ejecución por personas y/o computadores. Para este último caso, se explicó la necesidad de utilizar un lenguaje de programación, definiendo conceptos como valores, operadores, estructuras de condición, variables y listas. Para intentar enraizar estas ideas en las y los participantes, se utilizó la “sintaxis” de Scratch, es decir la definición de un programa con bloques, para lo cual fue necesario además explicar las características del entorno y modo de uso. El resultado fue presentado a un experto, en el área de CSE particularmente en Pensamiento Computacional, y finalmente validado por él.

Probabilidad y Fórmula de Bayes. Para la instrucción de este tópico, el principal desafío fue intentar abstraer conceptos que son impartidos en etapas escolares superiores a las que se considera en esta tesis. Conceptos de probabilidad condicional y teorema de Bayes son impartidos en los niveles de enseñanza media y universitario, respectivamente. Para esto, se determinó que no se realizaría la justificación de cada uno, sino que se explicaría en qué consiste y cómo poder calcular o definir su respectivo valor. Por ejemplo, el teorema de Bayes se abordó de manera que entendieran su utilidad para determinar $P(\text{causa}|\text{efecto})$ conociendo $P(\text{efecto}|\text{causa})$. Dado esto, para no caer en la explicación matemática de qué es un teorema, se enfatizó en la ecuación o fórmula de Bayes y su aplicación. El material generado

¹<https://www.curriculumnacional.cl/portal/>

²<https://aieducation.mit.edu/daily/>

fue revisado por investigadores de educación en matemática del CMM, considerándolo como válido para introducir a tales conceptos matemáticos; a niños y niñas de 10 a 12 años.

Algoritmo Bayes Ingenuo. En esta última serie instruccional es donde convergen todos los conceptos previos. Ejemplificando con casos cotidianos donde se realiza clasificación, se presenta el algoritmo Bayes ingenuo. Dado esto se precisa su definición identificando los pasos o instrucciones necesarios, y luego se expone los nuevos bloques Bayes ingenuo relacionados. Finalmente, se codifica el clasificador en Scratch siguiendo la receta o secuencia de pasos del algoritmo Bayes ingenuo. Este material, también fue validado con expertos en CSE y DS del DCC.

Actividades Prácticas

Respecto a las actividades prácticas, para complementar a las sesiones de instrucción, se definieron ejercicios a realizar mediante computador, así como también actividades desenchufadas o *unplugged*. Estas actividades se refieren a tareas educativas sobre conceptos de CS que no dependen de algún computador para realizarse [2]. De esta forma, se priorizó analizar actividades desenchufadas y luego aquellas mediante computador, para definir entonces la incorporación de alguna(s) en la solución y finalmente aportar antecedentes para los OEs 1, 2 y 3.

Actividades Desenchufadas. En los artículos seleccionados y mencionados en el Capítulo 2, en particular los relacionados con currículos educativos, es posible identificar actividades que no necesitan de un computador para ejercitar un concepto sobre CS o AI. Se comenzó analizando las actividades propuestas en el proyecto *CSUnplugged*³, fundado por Bell, Witten y Fellows [2], con el objetivo de promover las CS (y la computación en general) entre los y las jóvenes como una disciplina interesante, atractiva e intelectualmente estimulante. Se encontró una actividad nombrada “*Inteligente pieza de papel*” que trata de evidenciar cómo un computador sigue reglas (pasos del algoritmo) y estas pueden ser consideradas inteligentes dado su uso, por ejemplo para jugar a “ceros y cruces”, conocido en Chile como “el gato”. Si bien esta actividad permite hacer reflexionar a las y los participantes sobre qué es inteligencia y si algo que sigue reglas puede considerarse como tal, no se consideró incluir en la solución ya que no ahonda en conceptos como aprender patrones ni toma de decisiones basada en datos. Luego, se estudió en profundidad la publicación de Lee et al. [14] sobre el taller *DAILY* y el currículo sobre ética de la AI del *MIT Media Lab*⁴, encontrando actividades útiles para un taller presencial denominadas “*Bingo AI*”, “*Receta para la mejor pizza*” y “*Matrices éticas*”. Considerando que se adaptó el taller para ser remoto y *online*, requiriendo principalmente de un computador, las actividades desenchufadas encontradas en la literatura fueron descartadas. Sin embargo, igualmente se definieron dos ejercicios con lápiz y papel; en el primero definirían el algoritmo necesario para lavarse los dientes, mientras que en el segundo para ejercitar probabilidades.

Actividades Mediante Computador. Las actividades prácticas con uso de computador se diseñaron con la intención de completar finalmente la instrucción de un tópico específico.

³<https://classic.csunplugged.org/artificial-intelligence/>

⁴<https://docs.google.com/document/d/1e9wx9oBg7CR0s507YnYHVmX7H7pnITfoDxNdrSGkp60/edit?usp=sharing>

Para esto, se encontró actividades conectadas que fueron adaptadas al objetivo del taller, mas no para construir un clasificador de texto con Bayes ingenuo y Scratch. También, no solo se consideró codificar con Scratch, sino que se incorporó el uso de otras herramientas como Kahoot⁵ para *quizzes* o trivias, y hojas de cálculo para ejercitar conceptos matemáticos. De esta forma, las actividades seleccionadas fueron:

Trivia con Kahoot. Esta práctica fue diseñada para intentar hacer reflexionar a los y las participantes, con preguntas sobre generales sobre AI, entrenamiento y predicción en ML. Cada pregunta tenía entre tres y cuatro respuestas, donde una era la correcta y se asignaba puntaje dependiendo de la velocidad de respuesta y acierto. La ventaja de usar una herramienta como Kahoot, es que permite gamificar la actividad fomentando la participación y sano desafío. El material final se encuentra en el Apéndice D.

Clasificador de texto con ML4Kids. Esta actividad fue diseñada con el objetivo de introducir a las y los participantes en SL y la construcción de un clasificador con Scratch. Así, el principal objetivo es indicar que el programa realiza la clasificación de un nuevo texto (no considerado en el conjunto de entrenamiento). Sin embargo, usando esta herramienta no es posible saber cómo determina tal decisión. Con esto, se busca que las y los participantes logren reflexionar sobre el proceso de predicción, y aclararles a ellas y ellos que en las siguientes actividades es posible transparentar un poco más esta determinación.

Probabilidades con hoja de cálculo. Luego, se buscaron actividades que permitieran ejercitar probabilidad condicional mediante tablas de contingencia. Una vez seleccionadas, se adaptaron a una hoja de cálculo con tal de facilitar el manejo numérico. Se creó un archivo en la nube, y cada hoja sería destinada a un/a participante en particular con el enunciado del ejercicio, la tabla de contingencia pre-llenada y las preguntas a responder. Esta actividad fue diseñada con la intención de conocer el nivel de comprensión de este tópico, y su relación posterior con el manejo del algoritmo Bayes ingenuo.

Clasificador de texto con bloques Bayes Ingenuo. Finalmente, se diseñaron las actividades que permiten a las y los participantes construir el clasificador con el algoritmo Bayes ingenuo. Un criterio de diseño fue priorizar clasificar texto, es decir con la solución promover a que niñas y niños construyan más de un programa para profundizar así en este tipo de clasificación; y luego, opcionalmente realizar un clasificador de datos tabulados. Entonces, el resultado fue actividades prácticas para clasificar texto y datos con estructura o tabulación; y además posibilitando la colaboración entre pares.

Respecto a las actividades para clasificación de texto se diseñaron dos: una para identificar el sentimiento o emoción —alegre o triste— expresada en un mensaje, mientras que la segunda para determinar el tipo de noticia entre videojuegos y animales. Por otro lado para clasificar datos tabulados, se consideró una actividad que permitiera predecir si el pronóstico de un día podría considerarse como bueno o malo para jugar, dado el viento, humedad, nubosidad, entre otros datos estructurados. Finalmente, con estas actividades se permitió a que las y los participantes se enfoquen en los pasos del algoritmo, para posteriormente pensar en el desarrollo de una animación o videojuego con Scratch.

⁵<https://kahoot.com>

3.2. Extensión a Scratch

Definidas las actividades sobre clasificación de texto, se prosiguió con el diseño y desarrollo de los bloques Scratch que permita a los y las participantes programar el algoritmo Bayes ingenuo. Estos nuevos bloques se incorporaron a Scratch desarrollando una extensión, denominando como ExtBi (abreviación de *extensión Bayes ingenuo*) a esta componente de la solución. Así, los bloques que incluye la ExtBi se definieron con la intención de abstraer determinados componentes del algoritmo, reservando principalmente el computo matemático involucrado, permitiendo comprender su principal secuencia de pasos.

3.2.1. Aspectos técnicos

Para el desarrollo de los nuevos bloques se consideró la última versión oficial de Scratch, número 3.0. Con este *release*, además de disponibilizar la versión de escritorio u *offline* del entorno de programación, se permite también su ejecución desde el navegador web y en dispositivos móviles. En otro aspecto, la versión 3.0 de Scratch además de considerar los bloques por defecto, ofrece una biblioteca de nuevos bloques extendiendo así las opciones de programación. Para utilizar estos nuevos bloques se utiliza la opción “Agregar Extensión”, ubicada en la esquina inferior izquierda del entorno, como se destaca en la Figura 3.2. Con esto es posible añadir extensiones que faciliten, por ejemplo, la construcción de un programa que traduzca un texto de un idioma a otro; o también para que los personajes respondan a movimientos corporales mediante la cámara web. Por consiguiente, los nuevos bloques Bayes ingenuo son incorporados al entorno de programación respetando el flujo de la funcionalidad de agregar extensión, como se indica en la Figura 3.3.

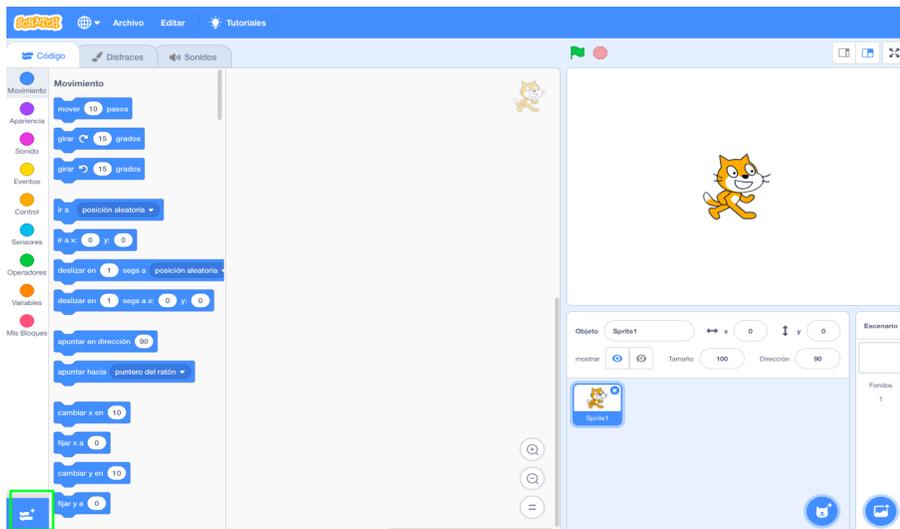


Figura 3.2: Agregar Extensión en Scratch 3.0

La arquitectura de Scratch 3.0⁶ se basa en una serie de proyectos desarrollados y mantenidos por el grupo LLK, donde los principales repositorios son denominados *scratch-gui*, *scratch-blocks*, *scratch-vm* y *scratch-render*. Para el desarrollo de los bloques Bayes ingenuo, se debe extender el código de los repositorios GitHub *scratch-gui*, interfaz gráfica de usuario

⁶<https://github.com/LLK>

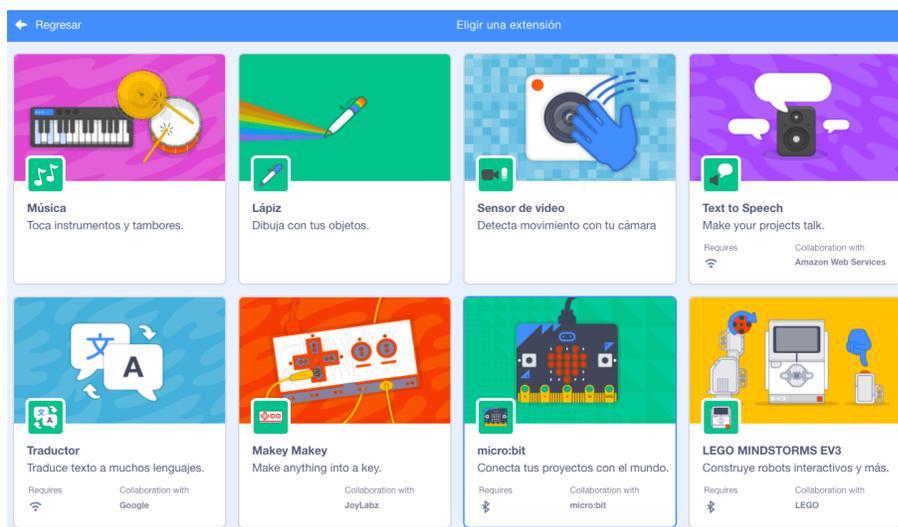


Figura 3.3: Extensiones por defecto

para la creación y ejecución de programas de Scratch; y *scratch-vm* maquina virtual utilizada para representar, ejecutar y mantener el estado de los programas de Scratch. Finalmente, se menciona que esta versión de Scratch está implementada principalmente en el framework NodeJS⁷, usando el lenguaje de programación Javascript en más del 90 % de su código.

3.2.2. Desarrollo de la extensión

Como es necesario extender los proyectos *scratch-gui* y *scratch-vm*, se comenzó haciendo un *git fork* de cada repositorio. Luego, siguiendo los lineamientos sugeridos por el grupo LLK⁸, se continúa la ambientación enlazando ambos proyectos mediante comandos *Node Package Manager* o NPM⁹. Así, para el desarrollo de extensiones según la documentación oficial¹⁰, en *scratch-vm* se encuentra la lógica de cada bloque y el modelo a definir según los datos de entrenamiento. Por consiguiente, para incorporar la nueva extensión con bloques Bayes ingenuo, se extiende y actualiza el código fuente de *scratch-vm* tal como indica la Figura 3.4. Aquí, en el archivo `naive-bayes.js` es donde está la lógica de cada bloque en la extensión Bayes ingenuo. La implementación se hizo siguiendo la metodología de desarrollo guiado por pruebas o *test driven development* (TDD), definiendo así los test unitarios de funciones en la ruta `test/unit/naive-bayes`. Por otro lado, para renderizar los bloques y poder utilizarlos en el entorno Scratch, en el repositorio *scratch-gui* se actualiza el archivo `src/lib/libraries/extensions/index.jsx`, agregando la configuración en relación a la nueva extensión.

3.2.3. Descripción de bloques

En Scratch hay una denominación o clasificación para cada tipo de bloque. En la Figura 3.5 se aprecia el diseño o forma de los bloques clásicos, y las características de estos son:

⁷<https://nodejs.org/en/about/>

⁸<https://github.com/LLK/scratch-gui/wiki/Getting-Started>

⁹<https://www.npmjs.com>

¹⁰<https://github.com/LLK/scratch-vm/blob/develop/docs/extensions.md>

```

1  scratch-vm
2  └─ src
3      .
4      .
5      .
6      └─ extensions
7          .
8          .
9          .
10         └─ scratch3_naive_bayes [AGREGA]
11             └─ index.js
12                 └─ naive-bayes.js
13                     └─ model.js
14             .
15             .
16             .
17         └─ extension-support
18             .
19             .
20             .
21         └─ extension-manager.js [ACTUALIZA]
22     └─ test

```

Figura 3.4: Archivos fuente creados y/o modificados

- Bloque Sombrero: inician la ejecución de un conjunto de bloques
- Bloque Pila o *Command*: ejecutan la instrucción que se indica, es un *statement*
- Bloque Reportador o *Reporter*: contienen o producen, un valor número o palabra (*strings*)
- Bloque Booleano: contienen o producen, un valor verdadero o falso
- Bloque C: controlan que los bloques ubicados en su interior se ejecuten hasta que la condición se cumpla

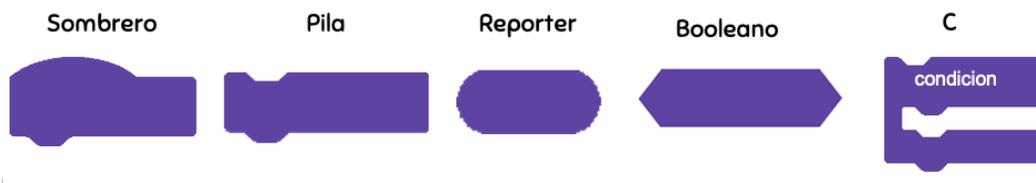


Figura 3.5: Diseño Bloques Scratch

En consecuencia, la extensión Bayes ingenuo incorpora bloques de tipo *command* y *reporter*. Estos interoperan con los demás bloques del entorno, es decir, pueden asociarse con los bloques por defecto para construir un programa. De hecho, los bloques Bayes ingenuo necesitan de bloques Scratch, como listas, para lograr construir el clasificador de texto. Así, los bloques Bayes ingenuo definidos y refinados luego de la validación preliminar —segunda y tercera espira según la metodología de la Sección 1.6— se describen en las siguientes subsecciones.

Bloque clasifica

Este bloque es el que permite inicializar el clasificador y el modelo de aprendizaje. Este bloque es de tipo *command* y se presenta en la Figura 3.6. Inicialmente se definió con tres

parámetros: (1) nombre del clasificador, (2) tipo de datos a clasificar, opción texto o estructurados; y (3) una descripción sobre los datos. Internamente, este bloque inicializa el objeto que contiene los métodos o funciones descritas en la subsección de abstracción de funciones en Bayes ingenuo, Sección 3.1.1. Este bloque es obligatorio por cada clasificador.



Figura 3.6: Bloque clasificador

Bloque entrenar

El siguiente bloque permite asociar ejemplos de entrenamiento para una categoría o clase. También es de tipo *command* y se presenta en la Figura 3.7. Requiere de tres parámetros: (1) el nombre de la categoría que se está entrenando, (2) la lista de ejemplos, y (3) el largo de la lista de ejemplos. Con este bloque, internamente se asocia el listado de ejemplos con su tabla de contingencia que luego servirá para el cálculo de las probabilidades a posteriori para cada categoría.

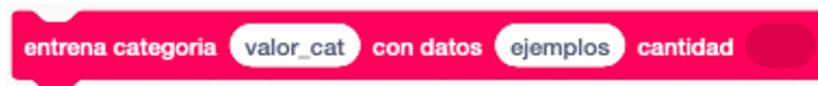


Figura 3.7: Bloque entrenar categoría

Bloque bayes

Con este bloque se calcula el valor de la probabilidad a posteriori para una categoría dado un nuevo ejemplo (no existente en el conjunto de entrenamiento). Es de tipo *reporter* como se muestra en la Figura 3.8, es decir retorna el valor calculado. Tiene dos parámetros: (1) el nombre de la categoría para la cual se calcula su probabilidad y (2) el nuevo valor.

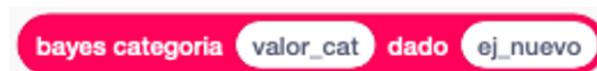


Figura 3.8: Bloque bayes categoría

Bloque maxima

Con el bloque *maxima* se determina la categoría con mayor probabilidad a posteriori. Corresponde a un bloque tipo *command*, retornando el nombre de la categoría con probabilidad mayor. Se presenta en la Figura 3.9. Necesita de un parámetro para su funcionamiento, a decir, el listado de los resultados de las probabilidades calculadas con el bloque *bayes*.

Cabe destacar que los bloques Bayes ingenuo se presentaron a expertos en las áreas CSE y AI, docentes del DCC, y sus comentarios apuntaron principalmente a que al explicar el funcionamiento de cada uno, se enfatizara en la utilidad de cada bloque. También, los bloques fueron refinados en base a los resultados de experiencias con usuarios finales, presentadas en los Capítulos 4 y 5. En definitiva, un clasificador de texto para determinar la categoría del



Figura 3.9: Bloque maxima

sentimiento expresado en este, codificado con los bloques Bayes ingenuo se aprecia en la Figura 3.10.

Con estos bloques, es posible construir un clasificador como el presentado en Figura 3.10. Describiendo el código, lo primero que se realiza es definir el nombre del clasificador. Luego, se entrena cada categoría `alegria` y `tristeza` con una lista de ejemplo, respectivamente. A continuación, se vacía la lista `valoresBayes`, la cual almacena valores de probabilidad a posteriori para cada categoría (el resultado del bloque `bayes categoria`). Luego, se solicita al usuario ingresar un nuevo mensaje, valor al cual se determinará la probabilidad a posteriori por cada categoría, agregando este resultado decimal a la lista `valoresBayes`. Finalmente, la predicción final es determinada por el bloque `maxima categoria` el cual, usando la lista `valoresBayes`, determina la categoría con probabilidad mayor.



Figura 3.10: Clasificador de Texto con Bloques Bayes ingenuo

3.2.4. Despliegue en Servidor

El entorno Scratch con la nueva extensión desarrollada se encuentra desplegada en un servidor del DCC. Las principales características de hardware son: procesador de 64 bits, 8 Gb en memoria RAM y 30 Gb en disco duro. Por otro lado, a nivel de software, el servidor tiene instalado el Sistema Operativo Debian, Git, NodeJS y Nginx. Así, Scratch y los bloques Bayes ingenuo se encuentran en el siguiente enlace: <https://scratch-naive-bayes.dcc.uchile.cl>.

3.2.5. Comparación con otras soluciones

Como se mencionó en el Capítulo 2, existen alternativas educativas que intentan introducir a niñas y niños en ML, y en la Tabla 3.2 se presenta una comparación entre estas. Además de identificar el tipo de clasificación que permite cada herramienta, también se señala la dependencia de cada una con terceras aplicaciones o servicios. Por ejemplo, aquellas soluciones que no dependen de otras para su funcionamiento son los bloques Bayes ingenuo y LearningML, mientras que ML4kids y TecheableMachine dependen de servicios de IBM y Google respectivamente. Finalmente, cabe mencionar que la extensión de bloques Bayes ingenuo es la única que permite apreciar el algoritmo involucrado en la tarea de clasificación, el resto mantienen en secreto las acciones involucradas en tal predicción.

Tabla 3.2: Comparación Soluciones Escolares sobre AI

Solución	Depende de Ser-vicios Externos	Clasifica Texto	Clasifica Imagen	Clasifica Audio	Algoritmo Visible
ExtBi	X	✓	X	X	✓
LearningML	X	✓	✓	X	X
ML4Kids	✓	✓	✓	✓	X
Techeable Machine	✓	X	✓	✓	X

3.3. Taller como Unidad de Análisis

Se diseñó un taller para niñas y niños escolares de Chile, escenario donde se aplica el material instruccional y permite el uso de la extensión Scratch Bayes ingenuo, definiendo así el caso a ser examinado mediante el método de investigación caso de estudio [29]. La principal unidad de análisis del caso es el grupo de participantes al taller, en concreto su comportamiento y concientización durante dicha experiencia educativa. De esta forma, se podría catalogar al caso como *único* y *holístico* ya que se intenta comprender la totalidad de la unidad de análisis y no solo un determinado aspecto. El taller se definió de manera presencial, pero en el segundo semestre del 2020 se optó por adecuar la iniciativa de forma remota *online* debido a la pandemia por COVID-19. Con esto, la planificación del taller se distribuyó en sesiones, una por día, de tres horas y treinta minutos aproximadamente cada una; considerando ventanas de recreo o *break* de treinta minutos en total.

3.3.1. Proposiciones del Caso

Según las hipótesis planteadas en este trabajo de tesis, las proposiciones que aborda el caso de estudio son:

1. Se puede reconocer las principales características genéricas de algoritmos en AI: conjunto de entrenamiento, acciones que aprenden un patrón y también predicción como resultado.

2. Con las actividades prácticas se permite reforzar la relevancia de los datos en el entrenamiento, identificar cuándo y cómo se aplican los pasos de reconocimiento de patrones, y por ende cómo se produce una predicción o toma de decisión.
3. A las y los participantes del taller se les permite reflexionar y comparar programas con AI y sin AI, al construir un clasificador de texto utilizando el lenguaje Scratch.
4. Al abstraer determinados módulos del algoritmo Bayes ingenuo con ExtBi, se les posibilita a las y los participantes una concientización sobre este algoritmo al implementarlo.
5. Participantes pueden reconocer distintas aplicaciones o servicios que utilizan AI, y en concreto aquellas que permiten comenzar a aprender sobre ML como ExtBi o ML4kids.
6. Los y las participantes pueden comparar la modalidad de cada herramienta, ExtBi versus ML4kids, al construir un clasificador de texto.
7. Es posible identificar las dificultades y complicaciones que tuvieron los y las participantes con la ExtBi, contribuyendo a definir las limitaciones de esta solución.

3.3.2. Métodos para Recolectar Evidencia

El diseño del taller incorpora métodos que permiten la obtención de evidencia generada por las y los participantes. Además de intentar conocer el comportamiento de los niños y niñas con la solución propuesta, también se intenta medir su comprensión sobre los conceptos instruidos. Así, estos mecanismos fueron planificados para realizarse en distintas sesiones, ad-hoc al avance del taller.

Antes del comienzo del taller, se le solicita a las y los apoderados de cada participante responder el formulario de consentimiento, el cual tiene como propósito autorizar o no el uso de los resultados de aprendizaje de su pupilo/a (tales como actividades parciales, avances de proyectos, proyecto final). En el Apéndice A se presenta el formulario firmado por apoderados/as. También, se solicitó a cada participante completar el formulario de asentimiento para saber si autorizan o no la utilización de sus resultados de aprendizaje. En el Apéndice B se presenta el formulario completado por participantes. En la etapa de análisis de evidencia, se consideró solo los resultados de aquellos participantes que autorizaron el formulario de asentimiento, y consentimiento por sus apoderados. Este estudio fue aprobado por el Comité de Ética de la FCFM, según se aprecia en el Apéndice J.

Observaciones

El método de observación hacia los participantes se aplicó usando como insumo los registros de video, recolectados durante la realización del taller. Así, se revisó los registros sobre la exposición de conceptos, como también los archivos de video sobre las actividades prácticas desarrollados por participantes. Finalmente, con los tutores posterior a cada sesión se realizó una reunión para conocer las principales apreciaciones de cada uno/a sobre las actividades del taller y los y las participantes. Esta evidencia es útil para analizar el comportamiento de los participantes frente a la solución, como también las primeras impresiones sobre los conceptos presentados. Es esperable que no todos los participantes hablen o pregunten durante el taller y por ende no se tenga registros de su comportamiento. Sin embargo, con este método se logra una noción general sobre el grupo, permitiendo así tener una referencia con la cual comparar las respuestas individual de cada niño o niña en los cuestionarios.

Cuestionarios

Este instrumento fue definido para formar un diagnóstico de los y las participantes al inicio del taller sobre conceptos básicos de AI, ML, SL y clasificación, para luego contrastar los resultados aplicando nuevamente el instrumento al finalizar el taller. Se constituyen principalmente por preguntas abiertas. Al momento de recolectar datos con el cuestionario, se indica a las y los participantes que no es un mecanismo para calificar sus conocimientos, sino para tener una idea sobre sus nociones de los conceptos o tópicos. Se utilizó un formulario de Google como mecanismo para recolectar las respuestas, y se presenta el cuestionario inicial en el Apéndice C. Otro cuestionario diseñado, para aplicar posterior a la sesión de SL fue una trivia Kahoot, presentada en Apéndice D, con la intención de poder medir el nivel de comprensión de los y las participantes sobre conceptos de clasificación con SL. A diferencia de los otros cuestionarios, las preguntas son con alternativas o respuestas cerradas. Con este instrumento es posible medir el progreso de las nociones individual de cada niña o niño, en relación a los principales tópicos o conceptos que considera el caso; y también conocer su apreciación sobre la extensión Bayes ingenuo y el taller en general. En el caso se destinó tiempo para responder cada cuestionario, y se avanzó a la siguiente actividad una vez todas/os lo respondieron.

Constructive Interaction, CI

También se diseño una actividad con el método *constructive interaction* (CI) [12], el cual ayuda a determinar si los usuarios comprenden de manera correcta los conceptos básicos que subyacen a un sistema y si su implementación resulta útil. Además, el CI tiene dos características principales: (1) está orientado para desarrollar tareas colaborativas, y (2) quienes realizan las tareas van verbalizando en voz alta lo que hacen. Por esto, se definen grupos de participantes que construyen en forma conjunta un clasificador de texto, turnándose de a uno/a para realizar una tarea mientras comparte su pantalla verbalizando lo que va haciendo hasta completar lo solicitado; luego se da el turno a otro/a compañero/a para que realice la tarea siguiente. Finalmente, se definió hacer dos CI: (1) usando ML4kids en la segunda sesión y (2) empleando ExtBi la cuarta sesión.

Cada grupo está guiado por un/a facilitador/a quien, además de listar las tareas a ejecutar y velar por su cumplimiento, está encargada/o de grabar la actividad. Se definió un instructivo, detallado en el Apéndice F, donde se explica cómo realizar el CI usando ExtBi; y otro con la herramienta ML4Kids en el Apéndice E. En concreto y relativo a la solución propuesta en este trabajo de tesis, se utilizó CI para recolectar evidencia sobre cómo los y las participantes interactuaron con la ExtBi al construir un clasificador de texto. Y dado que se definieron grupos, se analizaron las respuestas de cada participante considerando también el equipo y tutor que los guió.

Verbal Probing

Este instrumento es un tipo de cuestionario con la característica que está compuesto por preguntas de respuestas libres o abiertas. Los cuestionarios tipo *verbal probing* aplicados se presentan en Apéndice H. Se utilizaron principalmente para conocer la noción o entendimiento de la o el participante sobre un tópico en particular, dándole el espacio para que lo explique con palabras propias. En el caso o taller se realizan posterior a cada CI, sesión 2 y 3, y estos

instrumentos se utilizan para contrastar y confirmar lo respondido por los participantes en cuestionarios previos. Al igual que los registros al realizar CI, las respuestas de *verbal probing* se analizaron de forma individual pero considerando el grupo en el cual participaron y el tutor a cargo.

Programas Scratch

Finalmente, como evidencia a recolectar se consideran los programas `.sb3` implementados por las y los participantes al construir el clasificador de texto. Con esta evidencia documental, se busca analizar el manejo de los bloques que incorpora la ExtBi, y también contrastar con el uso del sitio ML4kids y sus bloques Scratch. Para apoyar a las y los participantes en la construcción del clasificador, se les entregó archivos `.txt` como ejemplos de conjunto de entrenamiento, presentados en el Apéndice I. Dado que estos archivos fueron generados principalmente durante el CI, se consideraron cada grupo para la revisión documental. Para considerar si efectivamente un/a participante que finalizó correctamente su programa comprende el algoritmo implementado, es imperativo además contrastar con la evidencia recuperada desde otros métodos, como cuestionarios y/u observación.

3.3.3. Planificación del taller

La planificación para el taller se presenta en la Tabla 3.3, indicando la distribución de sesiones de instrucción y la aplicación de los métodos para recolección de evidencia. En la primera sesión, se definió introducir a las y los participantes en definiciones fundamentales de AI y SL —desde las ciencias de la computación—; para luego realizar una introducción al entorno de Scratch, acotado a un mínimo de bloques a ser utilizados posteriormente en la construcción de un clasificador de texto. En la siguiente sesión, se presentó la herramienta ML4Kids y su ventaja para realizar un clasificador con SL, y se definió realizar la actividad grupal CI con ML4Kids. La tercera sesión, se enfocó en presentar las limitaciones de ML4Kids para comprender los pasos o algoritmo que permiten realizar la clasificación de texto. Así, se destinó esta sesión para abordar los tópicos necesarios en Bayes ingenuo: probabilidades, teorema de Bayes y una primera aproximación a dicho algoritmo. Finalmente, la sesión cuatro se inicia abordando en profundidad el algoritmo Bayes ingenuo, más el uso de la ExtBi. Además, una vez realizado por cada participante un clasificador de texto con la extensión, se realizó la actividad grupal CI usando ExtBi para construir otro clasificador.

Equipo de Facilitadores

Como se adelantó en la Sección 3.3.2, sobre *constructive interaction* o CI, se debe contar con un equipo de facilitadores para realizar el CI usando ML4Kids y ExtBi. Para esto, previo al taller se llevó a cabo la selección e inducción de candidatos a facilitadores. Esta actividad, consistió en presentar los principales tópicos del taller —como AI, SL, ML4Kids, Bayes ingenuo y ExtBi— además explicar en que consiste el método CI y cómo deberían aplicarlo. Dada la planificación del taller, la inducción se dividió en dos partes con una duración de tres horas cada una. Así, el protocolo de la inducción se presenta en Figura 3.11 y Figura 3.12, indicando las tareas realizadas por el profesor y por el candidato/a a facilitador.

Los y las candidatos a facilitadores, que en la primera inducción fueron estudiantes de ingeniería en computación en el DCC, además de poder comprender los tópicos que involu-

Tabla 3.3: Planificación del Taller. Versión 1

Días			
Sesión 1	Sesión 2	Sesión 3	Sesión 4
Cuestionario Inicial	ML4Kids	Probabilidades	Algoritmo Bayes Inge- nuevo II
Introducción a AI			ExtBi
Introducción a SL y Clasificación		Probabilidad Condi- cional	
Break 15 - 30 minutos			
Quiz Kahoot	CI ML4Kids	Teorema de Bayes	CI ExtBi
Scratch		Algoritmo Bayes Inge- nuevo I	

Para el taller, debían poseer la capacidad y motivación para trabajar con niñas y niños. De este modo, para validar si era apto para facilitador se le solicitó responder un cuestionario, presentado en el Apéndice G. Dado que cada facilitador debería llevar a cabo el CI, se le hizo entrega de un instructivo con tal de complementar lo presentado, donde además se indica la lista de tareas que deberían realizar por cada grupo.

Protocolo del Taller Versión 1

Finalmente y complementando la planificación del taller, el protocolo de la Figura 3.13 indica las actividades dirigidas por el profesor y facilitadores. También, se señala cuándo realizar las reuniones de equipo docente, en concreto al finalizar cada sesión. Este primer protocolo se validó con la revisión de un profesor del DCC, experto en el área CSE (*Computer Science Education*).

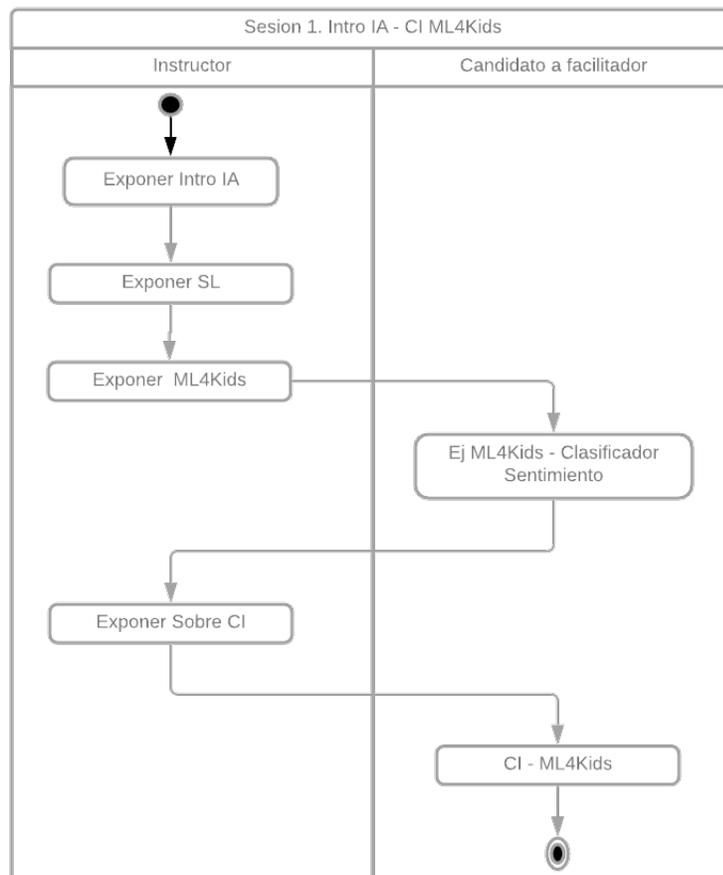


Figura 3.11: Inducción Facilitadores - Primera Parte

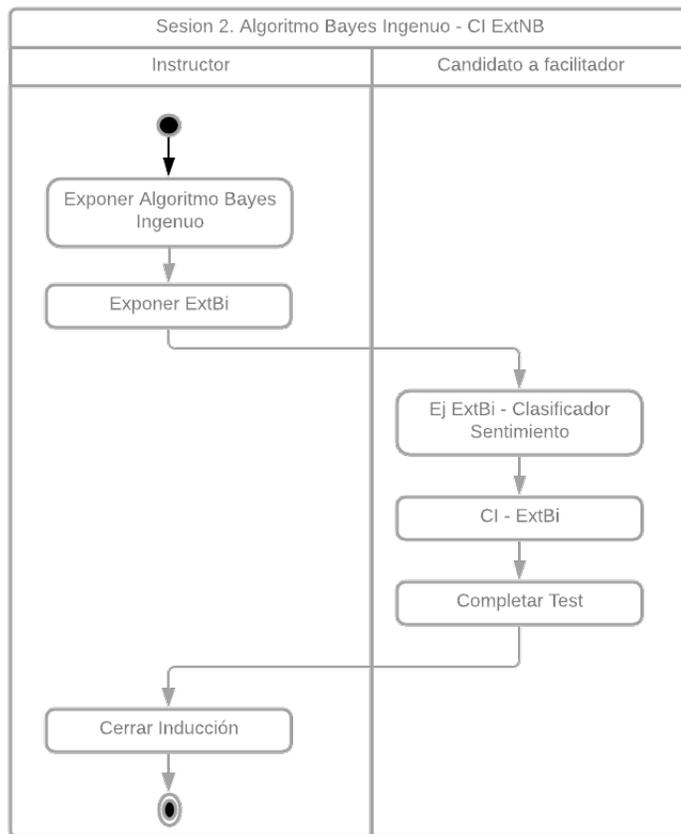


Figura 3.12: Inducción Facilitadores - Segunda Parte

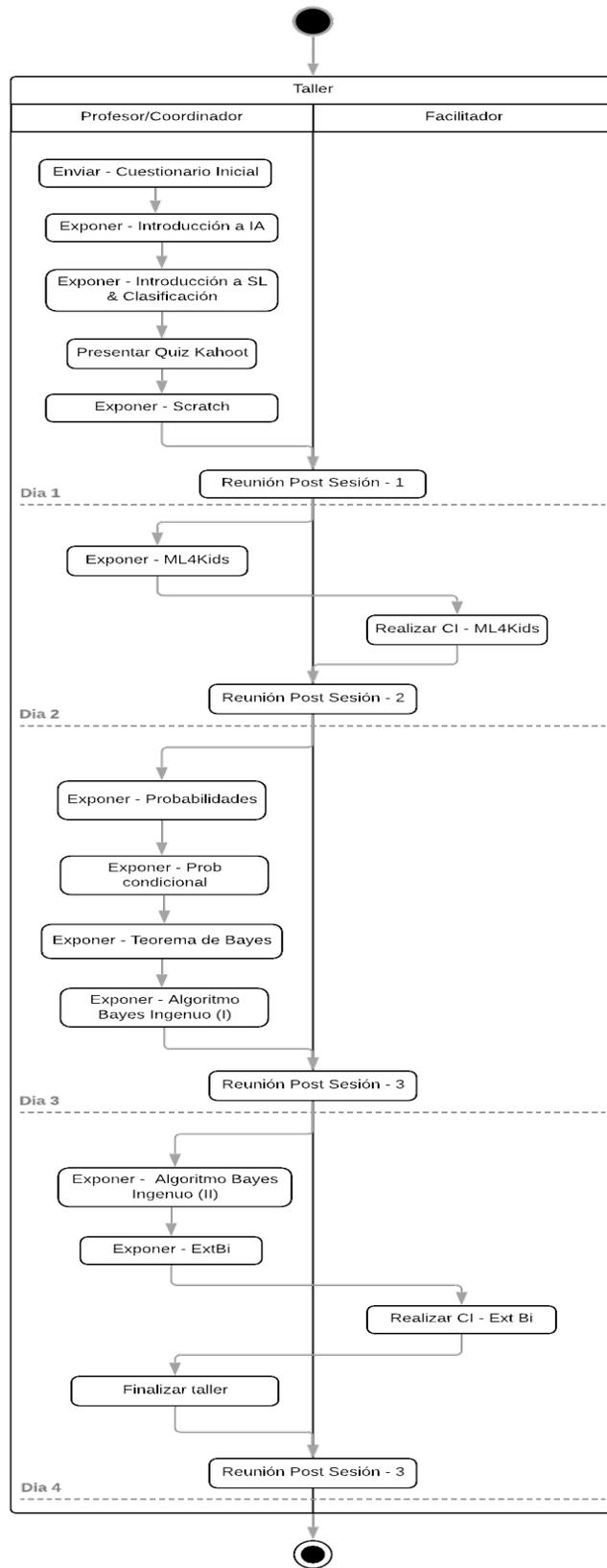


Figura 3.13: Taller - Protocolo Versión 1

Capítulo 4

Validación Preliminar

En este capítulo, se presenta el primer taller realizado con niñas y niños de 10 a 12 años de edad. En este caso de estudio, se buscó probar la metodología e instrucción del taller y la aplicación de los instrumentos para recolectar evidencia. Se piloteó el diseño de la solución, presentado en el capítulo previo, durante cuatro días seguidos en Enero 2021.

4.1. Preparativos del Caso de Estudio

Con el diseño del caso ya implementado, la solución pasó por una serie de validaciones principalmente por expertos, especialistas en ciencias de la computación y matemáticas. Luego de eso, se concretó el patrocinio del DCC para llevar a cabo el taller a niños y niñas. Finalmente, se difundió el formulario de inscripción —Apéndice L— a funcionarios de la FCFM. También, se llevó a cabo la validación operativa del taller. Considerando las cuatro sesiones diarias, de dos horas más treinta minutos de *break*, se configuraron las reuniones de videollamada. Además, se definió con el equipo de facilitadores el medio y forma de coordinación mientras se ejecutara las sesiones del taller.

El contenido del material expositivo, slides e instructivos, fue validado por expertos de la FCFM. En particular, respecto a los tópicos relacionados a las ciencias de la computación fue revisado por profesores del DCC —entendidos en CSE y AI—, mientras que el contenido matemático por expertos del CMM. De este modo, y según el diseño del taller, el producto fue un conjunto de siete diapositivas y tres instructivos.

En relación con el material práctico, es preciso señalar que también fue validado por los expertos mencionados. Luego, se estimó el esfuerzo de estas actividades, multiplicando por tres el tiempo utilizado por los facilitadores al realizar tales ejercicios. Así, la preparación de los ejercicios consistió principalmente en la elaboración de guías, credenciales para ML4Kids, despliegue de ExtBi, y archivos necesarios para los conjuntos de entrenamiento.

Respecto a los instrumentos, los cuestionarios inicial, kahoot, verbal probing (post CI) y final, también fueron revisados por un profesor del DCC, experto en realizar estudios de usabilidad y experiencia de usuario. Además, también se validó el protocolo para la ejecución de cada CI —usando ML4Kids y con ExtBi— según el diagrama de actividades, Figura 4.1. Este

protocolo, especifica las actividades realizadas por el facilitador y un participante seleccionado del grupo, destacando que cuándo este último no finaliza la tarea asignada correctamente se abre la opción a que el resto de compañeras/os puedan guiarle. Por consecuencia, se considera la colaboración entre pares para alcanzar el objetivo final, construir el clasificador de texto.

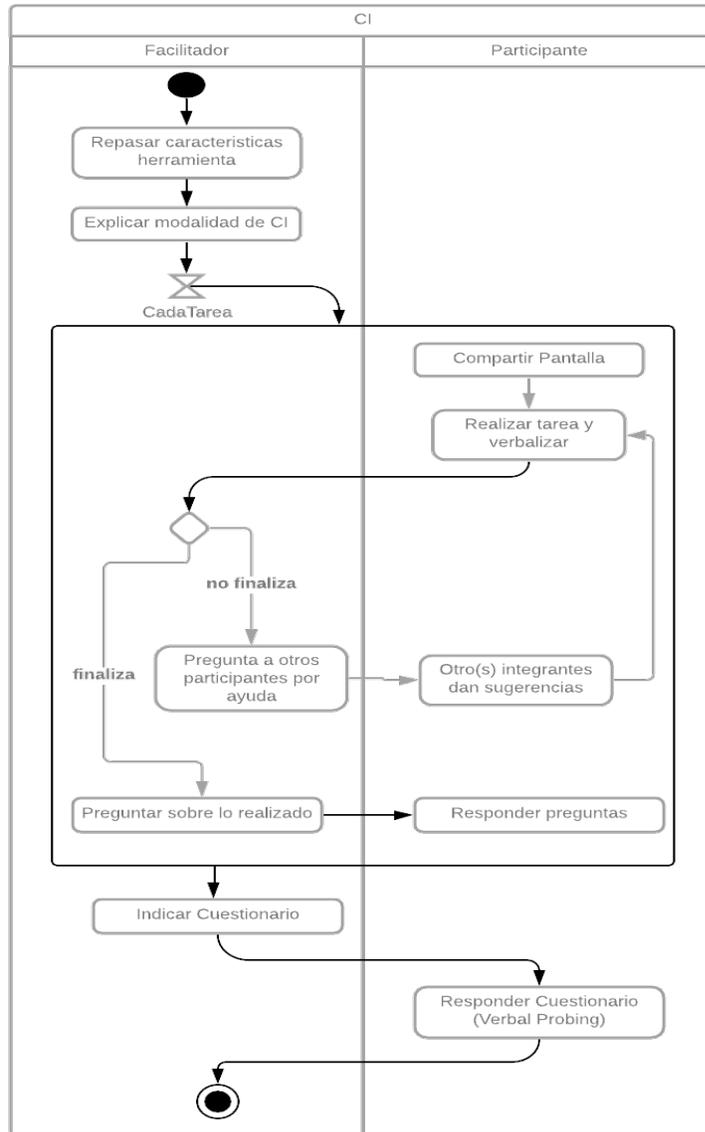


Figura 4.1: Protocolo CI - Diagrama de actividades

Previo al taller piloto con niñas y niños, se hizo una validación informal sobre usabilidad de la ExtBi con grupo objetivo distinto al considerado hasta ahora. Por esta vez, se definió como usuarios a estudiantes de primer año en la FCFM. La razón de preferencia por este grupo, fue con tal de realizar una validación inicial pronta con un perfil de participantes que —probablemente— no poseían experiencia en ML ni algoritmo Bayes ingenuo, tal como niños y niñas de 10 a 12 años. Con el público objetivo ya definido se publicó el formulario de inscripción, recibiendo 17 interesados/as en participar y de estos seleccionando a 10 dado

que indicaron preferencia de horario donde coincidió la mayoría.

Dado que para utilizar la ExtBi, fue también necesario presentar los conceptos fundamentales de AI, SL y Algoritmo Bayes ingenuo; esta vez fue la primera vez que se expuso el material del taller a un grupo de participantes. En esta primera presentación de la solución, se abordaron los conceptos para desarrollar los dos tipos de clasificador posible, de texto y con datos tabulados. En esta oportunidad también se piloteó el CI con la ExtBi, por ende también la inducción —previa— a facilitadores necesaria, y así además probar las actividades grupales diseñadas.

Luego de la experiencia con estudiantes de primer año, al revisar la evidencia se logró identificar mejoras en el hilo expositivo y argumentativo entre cada tópico. Respecto a las actividades prácticas, se decidió enfocar el taller solo a realizar clasificación de texto, no con datos tabulados. Así, se mantiene el foco de enseñar clasificación pero mediante casos que posiblemente son más cotidianos para ellas y ellos, usando frases u oraciones. También, otros aprendizajes gracias a las respuestas de los participantes, recolectadas al finalizar la experiencia con el formulario Apéndice K; fue que era posible mejorar el *layout* de los bloques en la ExtBi, sin embargo igualmente al menos la mitad de los participantes logró construir su clasificador.

4.2. Primera Ejecución del Caso de Estudio

Se llevó a cabo el caso de estudio a un grupo de 23 niños y niñas, con edad de 10 a 12 años. El horario fue de martes a viernes, desde las 10:00 AM - 12:30 PM. La principal dificultad fue exponer los conceptos y realizar las actividades prácticas en el horario definido, dos horas y treinta minutos. Además del coordinador y profesor del taller, habían 3 facilitadores apoyando en las actividades del taller. En términos generales, resultó una experiencia útil dado que permitió identificar situaciones que en la práctica complicaban la obtención de datos.

4.2.1. Observación a Registros de Videos

Los registros de video sobre cada sesión diaria y el CI con ML4Kids, fueron revisados posterior a la finalización del taller. No se hace mención al video sobre el CI con ExtBi —última actividad según la planificación del taller Tabla 3.3 y el protocolo Figura 3.13— dado que no fue posible realizarlo por razones de tiempo. A pesar de esto, la experiencia cumplió el objetivo de pilotear el taller, permitiendo ajustar la metodología y refinar los instrumentos para recolectar evidencia.

Sesión 1 - Introducción a IA, Clasificación y Scratch

En la primera sesión, se logró exponer los tópicos planificados. Sin embargo, la trivía con Kahoot fue postergada para la sesión del día 3. Además de la presentación del equipo docente, profesor y facilitadores, no se había previsto tiempo para la presentación individual de los participantes, produciendo así la primera alteración a la planificación. Otros inconvenientes, que produjeron retraso en lo programado, fueron dificultades técnicas —por parte de algunos participantes— para acceder a Zoom, Google Drive o su cuenta de correo.

En relación con la exposición de contenido, se aprecian dificultades para transmitir tópicos como la definición de algoritmo y diferencia con programación. Luego, al presentar las características fundamentales de algoritmos en AI, como empleo de conjunto de datos, acciones que aprenden y predicción, hubo preguntas y aportes que permiten identificar mejoras a incorporar para la explicación de cada uno de dichos componentes. Por ejemplo, para el elemento *dataset* se podría mejorar el material expositivo para mostrar las distintas categorías y ejemplos asociados a cada una. Luego, una pregunta de un participante fue “¿es posible que falle la predicción de una aplicación que usa un algoritmo de AI?”; evidenciando pronto interés en cómo se determina la toma de una decisión y posibles limitaciones de estos algoritmos.

Luego, al exponer sobre SL y clasificación se identifica una débil definición de cada concepto y su nexos con la definición previa de algoritmos en AI. En relación a la predicción, se podría profundizar en que el resultado es según las categorías que conoce, para valores que no están —completamente— en el entrenamiento. Así, se hace necesario también reforzar que se construye un modelo a partir de un algoritmo, basado en los ejemplos del entrenamiento, en concreto según los atributos o características de estos datos. De hecho, en este piloto se consideró un escenario ilustrativo para presentar la dinámica de cómo se realiza SL. Sin embargo el enunciado se podría mejorar y así ahondar que al entrenar se enseña las características de un ejemplo a una categoría específica, esto con el fin de marcar las diferencias de la fase de entrenamiento y aprendizaje. Finalmente, se podría fortalecer las diferencias de clasificadores con AI y sin AI, ya que en este taller no se produjo un contraste entre ambas alternativas, apreciando dificultad para indicar ventajas y desventajas de SL.

Luego se hizo una pausa y recreo, para posteriormente realizar la introducción al lenguaje Scratch. Si bien se presentaron los bloques principales a utilizar en el taller, faltó reforzar cómo se ejecuta y define un programa —instrucción por instrucción. También, faltó mencionar las zonas o secciones del entorno, en particular cómo se agrega una extensión de bloques. Esto, resulta importante dado que no se exigió poseer conocimientos previos en Scratch, por ende fue necesario establecer una base conceptual en todos y todas las participantes. Finalmente, se podrían mejorar los ejemplos y ejercicios planteados en esta clase, en particular tres: (1) al inicializar o poblar listas importando un archivo `.txt`, (2) agregar elementos a una lista de forma individual; y (3) construir un clasificador solo con los bloques presentados hasta el minuto, es decir sin AI. Este último ejercicio, sería útil justamente para hacer contraste con un programa que sí usa AI.

Sesión 2 - ML4Kids y Primer CI

La sesión comenzó con la exposición por parte de participantes voluntarios sobre su programa Scratch, en relación al último ejercicio planteado el día anterior sobre uso de condicionales. Luego de dos correctos programas presentados, que además de cumplir con la funcionalidad mejoraron la animación, el profesor del taller codificó el programa en vivo con la intención —nuevamente— de nivelar a los y las participantes sobre el funcionamiento de lenguaje. Esta actividad tomó aproximadamente 40 minutos, no fue considerada en la planificación inicial sino que surgió dado lo percibido por el equipo docente en la sesión del día 1 .

Luego se retomó el tópico SL, continuando ahora con la presentación de la herramienta ML4Kids y su utilidad para tareas de clasificación. Por esto, se preguntó a los y las partici-

pantes en que consistía SL, y una respuesta precisa fue “... *aprender a través de ejemplos*”. Se comenzó mostrando el sitio web y el proceso para definir el modelo del clasificador que posteriormente sería utilizado con Scratch. Esta actividad, se diseñó para que cada participante ingresara al sitio de ML4Kids con usuario y contraseña asignadas por el profesor. La ventaja de esta modalidad es que el profesor, al tener perfil administrador, tiene visibilidad de los proyectos de cada participante; criterio definido para —principalmente— revisar y analizar el clasificador de texto resultado del CI. Sin embargo, ocurrieron múltiples inconvenientes con las credenciales e ingreso al sitio. Se enviaron por correo, previo inicio de la segunda sesión, y hubo participantes que no tenían acceso al correo ya que algunos indicaron el de su madre o padre y en ese minuto no podían acceder a este. Por otro lado, tampoco fue entendible el formato de entrega de las credenciales, por ende no sabían cual correspondía al nombre de usuario y la contraseña. También, algunos/as no podían ingresar al sitio dado que el computador utilizado era limitado en *hardware*, impidiendo el uso simultáneo de la aplicación Zoom y navegador con ML4Kids.

A pesar de lo anterior, y luego de resolver inconvenientes a distintos participantes, finalmente se pudo presentar el primer ejercicio: construir un clasificador de sentimiento —alegre o triste— expresado en un mensaje. Aquí el profesor realizó un ejercicio guiado, resolviendo inconvenientes directamente de los y las participantes que requerían ayuda. Cabe señalar, que hasta el momento se mantenía una sola reunión para el total de los participantes, aún no se producía la separación en grupos con un facilitador a cargo. Una de las preguntas interesantes por un participante fue, qué pasa cuándo se entrena con mayor cantidad de ejemplos un categoría que la otra(s). Esta pregunta plantea el escenario de *overfitting* o sobre-ajuste al entrenar, y se respondió que en estos casos el clasificador inclinará su predicción por una categoría, produciendo malos aciertos en ejemplos que corresponden a la(s) otra(s) categoría(s). Por otro lado, respecto a la usabilidad del sitio, algunos participantes se perdían en los pasos no sabiendo como ir de una etapa a otra, por lo cual fue recurrente tener que volver a pasos anteriores para explicar como llegar a un punto determinado. Finalmente lograron construir el modelo y generaron los bloques Scratch de ML4Kids. Luego programando el clasificador, y como era de esperar, aparecieron dudas de cuáles bloques utilizar y por qué. También, al probar su primer clasificador en Scratch y apreciar que si bien el resultado es correcto en varios ejemplos, igualmente hubo casos donde falló; produciéndose así un momento reflexivo en los y las participantes de por qué se podría producir una mala clasificación, proponiendo también estrategias de como mejorarlo.

CI con ML4Kids. Luego de un receso de 10 minutos, comenzó la actividad CI usando ML4Kids. Se definieron previamente cuatro grupos, de 6 o 7 participantes más un facilitador. Las reuniones para cada equipo se hicieron en Google Meets, y a pesar de que fue informado previamente la asignación de cada participante —y el enlace de la reunión— igualmente hubo inconvenientes. Por ejemplo, algunos/as no tenían claro cual era su grupo, otras/os no habían usado Google Meets o tenían problemas con sus equipos. La decisión de usar Meets para el CI, fue porque se aspiraba a que todos los y las participantes en un grupo pudieran compartir pantalla simultáneamente, y Zoom solo permite de a uno a la vez. Esta evidencia, permitiría analizar de mejor forma el progreso y desempeño de las y los participantes mientras realizan las tareas del CI. A pesar de las dificultades, el equipo docente pudo reorganizar los grupos, determinando que un grupo usará Zoom y los otros tres Meets. Dado el retraso al inicio del CI, en los cuatro grupos solo faltó realizar la última tarea, completar el cuestionario *verbal*

probing, postergada a realizarse al inicio de la siguiente sesión.

Grupo A. En este grupo, al principio costó organización ya que algunos estaban también en la reunión Zoom, usada por el Grupo D y los participantes que no pudieron ingresar a Meets. Con esto, hubo dificultad para presentar el ejercicio y las instrucciones de cómo se llevaría a cabo. Dado que era el primer CI a niños y niñas ejecutado por el equipo docente, al principio los facilitadores no promovían la interacción con preguntas, sino que solo indicaban la tarea y seleccionaban al participante para que la realizara. Incluso en este grupo, el facilitador intervino en la ejecución al guiar a participantes —primeras dos tareas— para completar lo solicitado. Similarmente, este fenómeno también ocurriría en los otros grupos. Asimismo, al principio los facilitadores mencionaron sutilmente la importancia de verbalizar lo realizado por cada participante mientras llevaba a cabo la tarea asignada. En relación a la ejecución de las tareas número 3 y 4 —según el Apéndice E— relativo a entrenar las dos categorías del clasificador de noticias, finalizarlas en el Grupo A tomó aproximadamente la mitad del tiempo que duró el CI, es decir 30 de 65 minutos.

Grupo B. También se vio afectado por los inconvenientes iniciales. Comenzando con la actividad, al facilitador le faltó ser preciso en la metodología del CI e instrucciones para las y los participantes. Un participante usó tablet y tuvo dificultades con el *responsive* de ML4Kids para dispositivos móviles. Al realizar el entrenamiento, este grupo determinó hacerlo con un *dataset* propio —distinto al entregado como referencia para el CI—, donde cada participante pudo indicar un ejemplo para cada categoría. El motivo de hacer entrega del *dataset* fue justamente para agilizar la tarea de entrenamiento, sin embargo aquí se comprobó que no hubo diferencia considerable con definir uno propio ya que se tomó el mismo tiempo —aproximado— que en el Grupo A al completar esta fase. Si bien el facilitador tuvo alguna influencia en las tareas iniciales, se destaca que pudo plantear preguntas de manera correcta a los participantes durante el transcurso del CI.

Grupo C. Aquí, si bien pudieron en un momento compartir pantalla simultáneamente todos los participantes del grupo, finalmente se optó por hacerlo por turnos —igual que en los demás grupos—. Algunos participantes tuvieron complicaciones con el manejo de los bloques Scratch, tanto de bloques genéricos usados —operadores, condicionales y listas— como de los de ML4Kids; esperable dado que no tenían experiencia previa. Aquí, al facilitador faltó reforzar que el bloque *recognise* es quién hace la clasificación y es “caja negra” porque no es posible conocer las acciones internas que llevan a un determinado resultado. Igualmente, se aprecia que un participante logra verbalizar correctamente la utilidad del bloque *recognise*. En general, el facilitador promovió la interacción y reflexión conjunta en los participantes, además justificó lo realizado al terminar una tarea correctamente, intentando consolidar los conceptos teóricos adyacentes.

Grupo D. También crearon su propio *dataset*. Para esto, se asignó a un participante de la escritura de los ejemplos, mientras el grupo completo iban sugiriendo y seleccionando cuales agregar al entrenamiento de cada categoría. En este grupo, se percatan que cuando no hay una palabra —en el entrenamiento— el modelo no sabe a cual categoría pertenece e intenta asociarla a alguna de las disponibles. Dado que la mayoría no tenía experiencia previa en Scratch, algunos necesitaron ayuda puntual del facilitador para poder solucionar el

inconveniente y así avanzar. A pesar de esto, se aprecia que en términos generales entienden los pasos necesarios del clasificador, ya que se pudo validar por las respuestas a preguntas complementarias que hizo el facilitador.

Sesión 3 - Probabilidades y Teorema de Bayes

Como se mencionó al final de la sección anterior, el tercer día inició con la realización del cuestionario del CI con ML4Kids. A pesar de saber que tomaría tiempo de la planificación aplicar este cuestionario, y que no se podía extender el horario de las siguientes sesiones, se determinó realizarlo de todas formas. Además se hizo un repaso del programa construido con ML4Kids —como el presentado en la Figura 4.2— para evidenciar que el resultado del clasificador es dado por el bloque `recognise`, y que es caja negra ya que no es posible conocer internamente como determina el valor. Coincidentemente, este bloque es de color negro. Sin embargo, se destinó poco tiempo para profundizar en preguntas y comentarios de los participantes, lo cual fue lamentable porque se generó un constructivo espacio de ejemplos sugeridos por ellas y ellos, y hubiese sido positivo contar con mayor tiempo.



Figura 4.2: ML4Kids - Clasificador Noticias

Luego, se comenzó con los conceptos matemáticos necesarios para comprender el funcionamiento de la caja negra, secuencia interna de pasos en el bloque `recognise`, y así profundizar en cómo aplicar la técnica de SL. Para esto, el primer paso fue presentar el concepto de probabilidad. Luego de la exposición y ejercicios, al preguntar si alguno tenía preguntas o no entendían el concepto, la mayoría confiesa que logran captar la idea principal e incluso algunos ya lo conocían —algo— desde el colegio. No obstante al exponer sobre probabilidad condicional, ya se notó confusión del concepto. Se presentó utilizando tablas de contingencia, para hacer una representación de los datos, y también para determinar de forma simple probabilidades condicionales. Igual hubo participación, menos que en probabilidad con regla de Laplace, disminuyendo también comentarios o afirmaciones al respecto.

Luego de la realización del cuestionario Kahoot, el cual no consideró preguntas sobre ningún concepto matemático, se presentó una introducción al teorema Bayes ingenuo. Al presentar la ecuación principal, se evitó entrar en la justificación de por qué es así, y más bien se enfocó en presentar su utilidad mediante ejemplos guiados. Igualmente hubo solo algu-

nos participantes que indican comprender el concepto, por lo cual se sugiere aplicar mejoras en la exposición y material. Al ilustrar el beneficio del teorema de Bayes para determinar la probabilidad de que ocurra un evento dado otro, determinando la probabilidad en el caso de tener una enfermedad dado un síntoma, se podría considerar el comentario de un participante que ejemplifica esto con tener reflujo dado que se tiene tos. Luego vino un ejemplo, para esclarecer el concepto de ingenuidad e independencia de variables mediante tablas de contingencia; es decir, con data estructurada o expresable en un archivo separado por coma. Hay que resaltar, que no se les indicó el concepto de espaciado por algún delimitador, sino más bien una tabla con datos para predecir el estado de un día dado el clima; esto a fin de introducir al algoritmo Bayes ingenuo.

Sesión 4 - Algoritmo Bayes ingenuo y CI ExtBi

El último día se inició recapitulando las características generales de algoritmos en AI concerniente a, conjunto de ejemplos para entrenar, acciones que aprenden un patrón, y predicción como resultado. Luego al retomar el concepto de clasificación, mencionando que se logra con SL y por ende considera las particularidades de algoritmos en AI mencionadas al principio; se ilustra con un caso donde se determina si tiene o no una enfermedad —reflujo—, dado un test realizado. Si bien, al preguntar a los y las participantes si van entendiendo algunos declaran que “sí” y otros “*más o menos*”, aún la explicación de cómo se llega al resultado se apoya fuertemente en cálculos numéricos y abreviaciones que pueden ser difíciles de comprender para niños y niñas que por primera vez oyen del teorema de Bayes. Con el resultado de la clasificación ya determinado —en el ejemplo de tener o no una enfermedad dado un test—, se clarificó que a la secuencia de pasos usada se le conoce como algoritmo Bayes ingenuo, precisando así su definición. Luego, se abordó el concepto de ingenuidad usando datos tabulados, con el ejemplo sobre el estado de un día dado el clima. Se decidió usar este caso, dado el uso de tabla de contingencia para apreciar directamente el valor de la probabilidad —condicional— de cada componente —columna en la tabla— dado el valor a clasificar. Los comentarios de niños y niñas al preguntar si entendían o no el concepto, ahora fue de “*mas o menos*”, “*no*” o “*es mucha información*”. Era esperable recibir estos comentarios, pero quizás se debió también a que tal vez asumieron que debían encargarse directamente de los cálculos, ya que solo una vez presentado el ejemplo —de clasificación con datos tabulados— se aclaró que la extensión y nuevos bloques en Scratch los ayudarían con tales operaciones.

Finalmente se presentan los bloques ExtBi. Para esto, se presentó otra vez el caso de clasificación del sentimiento —alegría o tristeza— expresado en un mensaje, ahora con el fin de ilustrar como sería programar la secuencia de pasos del algoritmo Bayes ingenuo. Con este ejemplo, el propósito fue hacer hincapié en el orden de las acciones necesarias, más que profundizar en valores numéricos al determinar la probabilidad de cada categoría dado el nuevo mensaje —con el teorema de Bayes. Luego, se presentó de forma individual cada bloque y sus parámetros requeridos, además de la salida o resultado; para finalmente presentar el clasificador completo usando la ExtBi. Con el fin de reforzar aún más lo presentado, se hizo codificación —en vivo— del algoritmo Bayes ingenuo para clasificar el sentimiento en un texto. El profesor, durante la construcción del programa planteó preguntas sobre lo que iba desarrollando, y las y los participantes esta vez respondieron activamente con sugerencias, aportes, y a la vez generando contrapreguntas.

Posteriormente, las y los participantes implementaron individualmente el clasificador de sentimiento en un mensaje. Mientras codificaban, surgieron evidentes preguntas sobre los bloques de ExtBi, pero también sobre como poblar la lista de ejemplos importando el archivo `.txt`. Otra vez, a algunos les costó moverse en la carpeta Google Drive y encontrar el fichero. Por otro lado hubo quienes evidenciaron, al igual que con ML4Kids, que hay casos donde la predicción del clasificador no es correcta, lo que nuevamente sirvió para que ellas y ellos reflexionaran en cual podría ser la razón. De todos modos, hubo casos que finalizaron su clasificador con el resultado esperado. Dado el tiempo limitado, en esta última sesión no se pudo realizar la actividad CI con ExtBi ni el cuestionario *verbal probing* final.

4.2.2. Respuestas en Cuestionarios

Se recopilaron las respuestas, dadas por los participantes, a los distintos cuestionarios aplicados a lo largo de la intervención. En cada sesión, se aplicó al menos un cuestionario.

Inicial

Una vez iniciado el taller y presentado el equipo docente, la primera actividad fue completar este cuestionario, el cual estuvo compuesto por siete preguntas cerradas, es decir con alternativas. Luego de revisar esta evidencia, se podría concluir que las y los participantes responden lo que creen correcto pero no necesariamente lo que saben. Para ilustrar esto último, en la pregunta “*Escoge la definición precisa sobre Inteligencia Artificial (IA)*”, hubo más del 50% que escogió la alternativa correcta, sin embargo al observar los registros de video no hubo la misma abundancia respondiendo de forma correcta. Y respecto al contenido de las preguntas, faltó incorporar una para identificar a quién responde. Tampoco se planteó alguna pregunta para conocer si habían programado en Scratch previamente.

Verbal Probing - CI ML4Kids

Se llevó a cabo al inicio de la tercera sesión. Al considerar solo preguntas con respuesta libre o abierta, se pudo obtener mayor información del nivel de comprensión sobre los conceptos. Por ejemplo en la pregunta “*Menciona al menos una ventaja de Aprendizaje Supervisado*”, hubo aproximadamente un tercio de respuestas correctas (6 de 17) que referencian a mejorar el funcionamiento o desempeño en base a ejemplos; mientras que otras señalan la ventaja de aprender —en modo general— de forma más rápida o efectiva. En general, hubo mayor dispersión en las respuestas —a cada pregunta— encontrando correctas o cercanas a lo esperado, como también respuestas “*no sé*” o en blanco.

Por otro lado, igualmente hubo inconvenientes. Por ejemplo, el cuestionario estaba configurado para ser respondido solo por aquellos autorizados mediante el correo, y dado que no todos tenían acceso a su cuenta, significó invertir tiempo para finalmente ingresar al formulario. Dada esta contingencia finalmente se dejó público.

Kahoot

Este cuestionario finalmente se realizó en la sesión del día 3, posterior a la introducción al teorema de Bayes. Contó con ocho preguntas —todas cerradas— cada una con tres o cuatro alternativas. Las y los participantes demuestran gran interés en esta actividad, y

solo algunos no han utilizado previamente esta plataforma. Por ende, antes de comenzar se realizó igualmente una presentación de la herramienta y modalidad de juego. Luego de realizar Kahoot, podría considerarse que el quiz en 7 de 8 preguntas, se enfocó en consultar sobre la elección de conjunto de entrenamiento para construir alguna aplicación en particular.

4.2.3. Programas Scratch

Los programas que se pudieron revisar fueron los construidos con ML4Kids para el clasificador de noticias. Para hacer la revisión de estos, se les solicitó que guardaran su programa en su computador y luego lo subieran al directorio Drive indicado. Si bien, se insistió en la importancia de subir el programa, solo 6 participantes lograron cargarlo a la nube. Esto, podría deberse a que (1) no todos finalizaron su programa, (2) algunos no conocen como se descarga y sube a Drive, y (3) no tienen acceso a la carpeta en la nube. De todas formas, los programas disponibles fueron revisados y ejecutados en la plataforma. Aquí, se validó el uso de los bloques `recognise`, operador igualdad (o distinto a), condicional `si-sino`, y finalmente el bloque `decir` para presentar el resultado de la clasificación. Esto, ya que es posible considerar como significativas a tales secciones del código, al programar el clasificador con ML4Kids.

4.2.4. Aprendizajes

Dado los hallazgos mencionados recientemente, es posible contribuir a la proposición — número 7— que referencia a definir los límites de esta solución, planteada en el diseño del caso Subsección 3.3.1. Aunque, con el posterior análisis de la evidencia, igualmente es posible reconocer puntos de mejora relacionados con la metodología del taller, material instruccional y práctico, la ExtBi e instrumentos de evaluación. Una vez perfeccionada la solución, se volvió a realizar una validación por expertos —en ciencias de la computación y matemáticas— de la FCFM.

Metodología del Taller

Dado que la principal dificultad en el taller fue cumplir con el tiempo estimado, la medida tomada fue aumentar la cantidad de horas por sesión, de 2.5 a 4 horas. Se descartó la opción de eliminar contenidos o actividades del taller, ya que se consideran lo mínimo para lograr el objetivo principal de concientizar a niños y niñas sobre algoritmo Bayes ingenuo, y validar la solución. De hecho, se determinó incorporar un bloque al inicio de cada sesión para hacer un repaso del día anterior, responder dudas o atender aportes de las y los participantes. La modalidad *online* se mantiene y se utilizará únicamente Zoom para todas las actividades del taller.

Por otro lado, fue prudente hacer una reorganización de lo expuesto la primera sesión, en concreto planificar que luego de hacer la introducción a AI, se realice la clase de Scratch y finalmente plantear un ejercicio de clasificación sin AI. Además, se decidió que las clases de Scratch, ML4Kids y bloques ExtBi, fuesen en modalidad de grupos, a cargo de facilitador quién guía la exposición y a su grupo. Esto implica, que en la inducción a facilitadores también se incorporaron estos conceptos. Además se determinó que los grupos se mantendrían — participantes junto a su facilitador— durante todas las sesiones, con tal de focalizar de mejor

forma la enseñanza. De este modo, las sesiones grupales serían utilizando la funcionalidad *Breakout Rooms* de Zoom, donde el facilitador será responsable de grabar su sala o *room*, para su posterior análisis.

Material instruccional y exposición de tópicos. Las mejoras apuntaron, principalmente, a que la exposición entre cada contenido resultará con mayor conexión y fluidez. Por ejemplo, enlazar de manera evidente la definición general de algoritmo con las particularidades de algoritmos en AI. Luego, evidenciar las limitaciones al programar una aplicación de AI —clasificador— mediante reglas versus aprendizaje supervisado. También, al usar ML4Kids para implementar el clasificador indicar claramente las limitaciones de caja negra en el bloque *recognise*, y el algoritmo Bayes ingenuo como posible explicación a esta incógnita. Finalmente, una sugerencia indicada por una de las facilitadoras, fue que al hacer preguntas dirigidas a los y las participantes, sea más claro o explícito que se le está preguntando a ellas y ellos.

Para todo esto, se refinó —cuando fue posible— las definiciones de cada concepto y ejemplos. En particular al material matemático, se depuraron enunciados, notaciones y ejemplos; quitando algunas palabras o conceptos de la exposición, como hipótesis, causalidad y probabilidad total. También, se revisó la explicación de los ejemplos o casos, utilizados para evidenciar la utilidad del teorema de Bayes, por ejemplo al diagnosticar enfermedad —reflujo— dado un test. Luego de explicar los pasos del algoritmo Bayes ingenuo, y al ahondar en el concepto de ingenuidad, se sugiere clarificar que los valores numéricos, de probabilidades con el teorema de Bayes, serán calculados con los nuevos bloques Scratch. Así, se consideró enfatizar en que el objetivo de los y las participantes es definir el orden de los pasos del algoritmo, empleando los bloques necesarios y delegando la operación matemática en estos. Finalmente, se consideró necesario explicitar y reforzar la relación de cada bloque en ExtBi con las etapas del algoritmo AI, particularmente de Bayes ingenuo.

ExtBi. En relación a la extensión, la principal modificación fue al bloque *clasifica*. Se determinó rediseñar este bloque dado que el taller se enfocó a ejercicios prácticos para clasificar texto, quitando entonces la opción para clasificación de datos tabulados. El resultado final se presenta en Figura 4.3. En relación al resto de los bloques, no hubo cambios visibles sino que internamente se agregaron nuevas validaciones para robustecer su funcionamiento. Así, el clasificador de texto con la nueva versión de ExtBi se presenta en Figura 4.4.

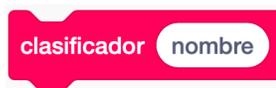


Figura 4.3: Bloque Clasifica - Versión 2

Sobre Recolección de Evidencia

Método CI. Para la ejecución del *constructive interaction*, la mejora principal fue especificar el modo y objetivo de este, en el instructivo y en la inducción a facilitadores. Por ejemplo, se enfatizó en que los y las participantes realicen la tarea colaborativamente mediante turnos asignados por el facilitador. La o el participante escogido para ejecutar una tarea comparte su pantalla y verbaliza lo que va haciendo. Así, se descartó la opción de compartir monitor

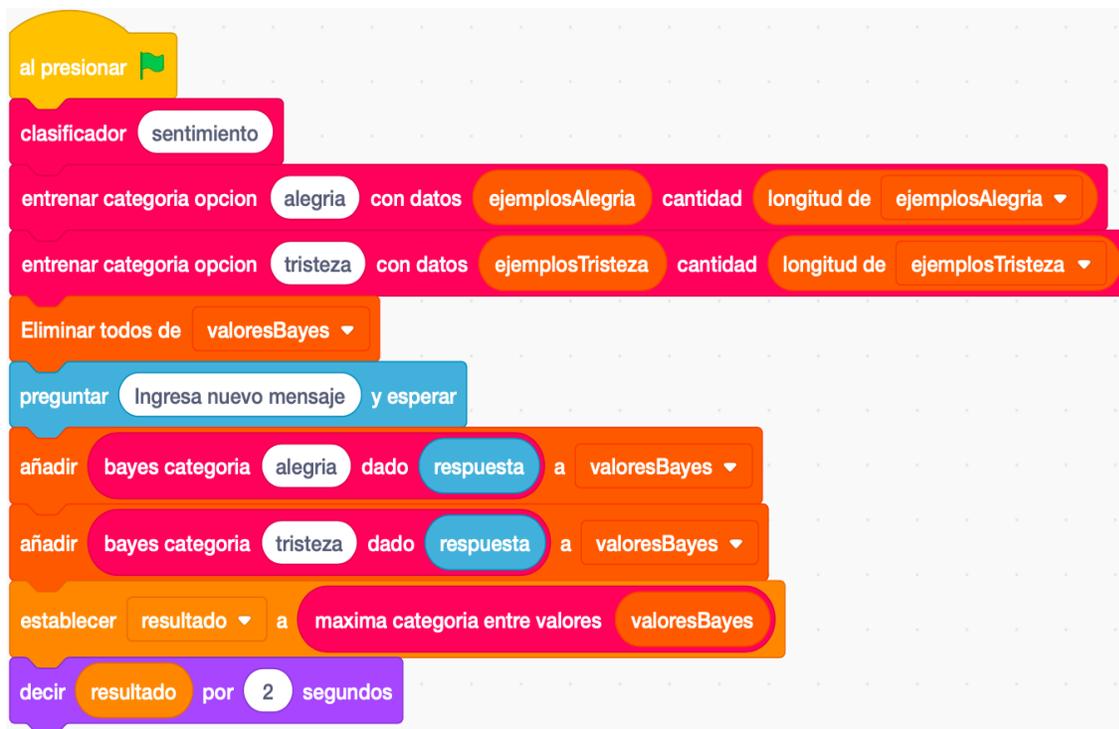


Figura 4.4: Clasificador de Sentimiento - Version 2

simultáneamente entre todo el grupo. También, se resaltó que los facilitadores eviten dar pistas o sugerencias de cómo hacer una tarea, y que cuando un participante no sepa debe preguntarse a los otros participantes y guiarlos para que en conjunto lo logren. También, dada la alta participación en los grupos donde crearon su propio conjunto de entrenamiento para el clasificador, se promueve esta idea a los facilitadores. De todos modos, se mantiene la entrega de un *dataset* como referencia. Finalmente, para subir programas a la carpeta compartida se insistió en que, soliciten descargar el programa —archivo .sb3— o tomar un *screenshot* de este, y supervisar hasta completar la carga de todos los archivos del grupo.

Cuestionarios. Se hizo revisión, y aplicaron mejoras, a cada formulario completado por las y los participantes. Una modificación transversal a cada instrumento, fue incorporar la pregunta sobre el dispositivo —móvil o de escritorio— usado. Además, en todos los cuestionarios se validó que pudiera identificarse quién responde, y fuera accesible para quienes tuviesen la dirección o *link*.

En el cuestionario inicial, la modificación más importante fue cambiar las preguntas cerradas a abiertas. También, se quitó una pregunta para saber si conocían el concepto de probabilidad condicional. Y, se agregó otra pregunta para indicar si posee experiencia en Scratch previamente.

Respecto al cuestionario Kahoot, se mantuvo orientado a preguntar sobre características de algoritmos en AI, sin embargo se quitó preguntas para identificar el conjunto de entrenamiento, y por otro lado se agregó consultas sobre predicción. También, se incorporaron preguntas de otros conceptos como programación y clasificación. Finalmente, las alternativas disponibles no fueron solo tres o cuatro, sino que se incorporó preguntas con verdadero

y falso. En relación a los cuestionarios *verbal probing*, posterior a cada CI, se precisaron algunos enunciados para preguntar específicamente sobre la actividad precedente realizada —clasificación de texto—, y conocer sobre su desempeño con la herramienta —ML4Kids o ExtBi—.

Validación con Expertos

Luego de incorporar las mejoras a la solución, el contenido relacionado con tópicos del área de computación y los instrumentos para recolectar evidencia, se validó con un docente del DCC experto en CSE —educación en ciencias de la computación—. Por otro lado, respecto al contenido matemático se validó nuevamente con una experta perteneciente al CMM. El principal aporte luego de esta última revisión, fue que la enseñanza del concepto de ingenuidad se puede lograr presentando probabilidad de eventos independientes. Esto implicó poder prescindir del concepto de tablas contingencia como se consideraba en el taller, permitiendo finalmente obviar casos o enunciados —por ejemplo con datos tabulados— y reducir en cierto grado la cantidad o carga de tópicos en el taller.

Capítulo 5

Evaluación Formal

El capítulo comienza presentando las consideraciones tomadas para la recolección de evidencia durante la realización del caso de estudio en julio del año 2021. Luego, se realiza una discusión preliminar de los hallazgos empíricos y posteriormente se describen con la técnica analítica *pattern matching*, según la metodología de Yin señalada en el capítulo 5 de su libro [29]. Los principales descubrimientos del caso son presentados y discutidos en profundidad en el Capítulo 6.

5.1. Recolección de Evidencia

Luego del primer taller con niñas y niños de 10 a 12 años, en Enero 2021, una de las principales mejoras consistió en ajustar el instructivo para la ejecución del método CI. De todas maneras, también se incorporaron mejoras para que la aplicación de cuestionarios fuese más eficaz, por ejemplo asegurando acceso al formulario independiente de la cuenta de correo. En concreto, para aumentar la probabilidad de éxito al aplicar los instrumentos fue necesario un refinamiento del taller y por ende actualizar su protocolo.

5.1.1. Protocolo del Taller Versión 3

Una vez finalizada la validación preliminar, descrita en el Capítulo 4, los aprendizajes y hallazgos sugirieron ajustar la metodología del taller y calibrar los instrumentos para recolectar evidencia. De este modo, en la Figura 5.1 se presenta el protocolo actualizado. Si bien se mantiene el taller de cuatro días, la duración de cada sesión diaria se aumentó a cuatro horas cronológicas, incluyendo *breaks* o recreos. También, se reorganizó la exposición de los tópicos en la sesión 1, decidiendo introducir a clasificación al final del día y presentando SL al inicio del siguiente. Por otro lado, se determinó que el material expositivo sobre la introducción a Scratch y ML4Kids, fuese en modalidad de grupos presentado por el facilitador a cargo. Esto último, con la intención de focalizar el aprendizaje, pudiendo atender dudas o comentarios de manera más directa a cada participante. También se determinó que los asignación de grupos, participantes y facilitador, se mantendrían durante todo el taller.

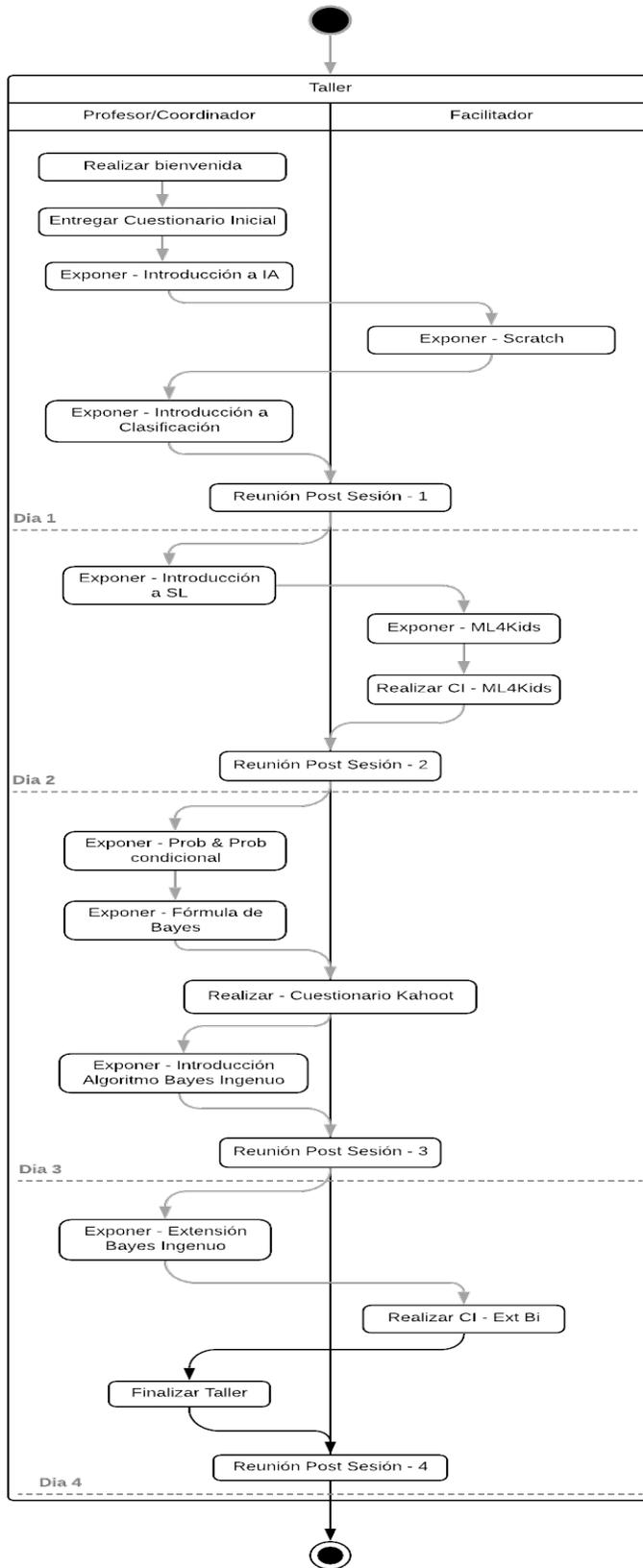


Figura 5.1: Taller - Protocolo Versión 3

5.1.2. Participantes

Para la selección de participantes, se mantuvo el criterio de ser escolares en Chile entre 10 a 12 años, sin requisito de dominio en Scratch. En esta ocasión, la convocatoria se destinó a familiares de funcionarios de la FCFM, dado que se buscó en primera instancia un grupo de participantes acotado y con características similares. Esta decisión se tomó para dar foco a la realización del taller y conocer el desempeño de los participantes con la solución. En la inscripción preliminar hubo 89 interesados/as, y por razones operativas finalmente se seleccionó a 46 para participar en el taller. Como era de esperar, posibles inconvenientes podían ocurrir sobre todo el primer día del taller, y lo más notorio, fue que algunos/as participantes tuvieron dificultades con el manejo de Zoom declarándose como usuarios de Google Meets. Así, manifestaron tener inconvenientes con la función *breakout rooms* para las actividades grupales; mientras que otros tuvieron algunas complicaciones con sus dispositivos utilizados (principalmente móviles) o manejo de Google Drive.

5.1.3. Materiales y métodos

Los materiales y herramienta utilizados para la recolección de evidencia fueron Google Forms y Kahoot para cuestionarios, Zoom para grabar evidencia, y notas escritas sobre las reuniones post-sesión con el equipo de tutores. La evidencia se almacenó en un repositorio, al cual solo tiene acceso el investigador principal del equipo. Aquí, se alojan las respuestas de los participantes a los cuestionarios y sus programas Scratch, más las grabaciones en videos de las sesiones.

5.2. Análisis

Siguiendo la estrategia general basada en las proposiciones, indicadas en la Subsección 3.3.1, el proceso analítico de la evidencia recolectada comenzó buscando ideas, conceptos prometedores o patrones, definiendo así en cuáles de los hallazgos profundizar y por qué. De esta forma, lo primero en revisar y analizar fueron los registros de video sobre la instrucción del taller y cada CI, luego los cuestionarios y posteriormente los programas creados por los y las participantes. También, se aplicó la técnica *pattern matching* la cual consiste en comparar un patrón observado empíricamente con uno basado en las proposiciones del caso.

5.2.1. Videos

Las grabaciones en video correspondieron a las sesiones diarias expositivas y a las actividades practicas CI por cada grupo. De esta forma, la revisión del registro audiovisual (*archival records*) se hizo siguiendo la planificación del taller. A consecuencia de esto, en las siguientes subsecciones se presentan las impresiones o primer análisis de cada sesión y CI realizado.

Sesión 1 - Introducción a IA y Scratch

Como contexto, antes de comenzar con la instrucción del día 1 se realizó la actividad para completar cuestionario inicial, la presentación del equipo de investigación y algunos participantes. El principal propósito educativo de esta sesión fue comenzar con la construcción de la primera versión para el clasificador de texto con Scratch. Dado que no hubo requisito sobre

saber programar para los y las participantes del taller, se realizó una nivelación desde la definición básica de algoritmos, para luego presentar características fundamentales de algoritmos en AI (consideraciones sobre fase de entrenamiento, aprendizaje y predicción). Así, luego de presentar ejemplos algoritmos, se planteó un ejercicio sobre este concepto con casos frecuentes y conocidos por ellos y ellas (como lavarse los dientes), y dada su participación se evidenció comprensión en definir un conjunto de pasos para resolver o solucionar algo. Avanzando en la sesión y posterior a la exposición de singularidades sobre algoritmos en AI, se plantearon ejercicios para identificar el conjunto de entrenamiento usado en algunas aplicaciones con AI, y debido a las respuestas de los y las participantes se evidenció que es posible adquirir noción sobre el propósito de un *dataset* y su relación con una determinada predicción.

Scratch. Posterior a un *break* en la instrucción, se realizó la clase de Scratch en grupos definidos aleatoriamente pero estable, es decir una vez definidos se mantuvieron para todas las actividades grupales. De este modo se definieron seis grupos, cada uno identificado con una letra desde la A hasta la F. Aquí, el fin fue principalmente presentar el conjunto (mínimo) de bloques a utilizar durante el taller, y además cómo incorporar nuevas extensiones al entorno. A pesar de que en el cuestionario inicial 19 participantes (65 %) declaran no tener nada de experiencia en Scratch, la sesión sobre este entorno y lenguaje se realizó correctamente en cinco grupos. El único grupo donde no se terminó, se alcanzó a instruir cómo importar listas desde archivo pero no se pudo plantear el ejercicio relacionado a este tópico. Un aspecto importante del cual se percataron algunos, fue que un programa con Scratch es similar a un puzzle, piezas o legos. Fue posible apreciar que crear listas en Scratch, importando un archivo `.txt`, se logra sin mayores complicaciones en la versión de escritorio pero en dispositivos móviles hay mayores dificultades para encontrar la funcionalidad. A pesar de que en los grupos (que finalizaron) declaran comprender ejercicio final sobre uso de condicionales y lo implementan por sí solos, a este minuto del análisis no se evidencia que los tópicos presentados sean dominados por los y las participantes. Por esto, igualmente se sugiere reforzar sobre secuencia de bloques y condicionales [18], en definitiva el uso de operadores y booleanos (verdadero o falso, más proposiciones y declaraciones), incorporando nuevos ejercicios sobre bloques que retornan, generan o almacenan un valor. También, se podría integrar una actividad sobre manejo de variables (y estructuras de datos en general) para múltiples personajes, con el objetivo posterior de abordar clasificación con algoritmo Bayes ingenuo entre escenarios o personajes de Scratch, por ejemplo sobre algún videojuegos o animé.

Sesión 2 - Aprendizaje Supervisado

El segundo día comenzó en una única sesión, no separados por grupos. Se retoma el ejercicio donde se practicó condicionales y listas con Scratch, intentando así nivelar a los participantes del grupo donde no se alcanzó a terminar la clase el día anterior. Con esto, se dio espacio para aquellos/as que hicieron los ejercicios compartieran sus programas, comparando posibles diferencias en el código de uno y otro.

Luego, se planteó un nuevo ejercicio solicitando a los participantes construir un programa para reconocer el sentimiento expresado en un mensaje, primer clasificador de texto. Dado que hasta ese momento del taller se instruyó sobre operadores booleanos y condicionales con Scratch, se propuso este ejercicio para que los y las participantes reflexionaran sobre lo difícil que es construir un clasificador solo con bloques `si-sino` y lo complicado de conocer todas las

reglas posibles en cada condicional. Para abordar este tipo de escenarios sobre clasificación, se presentó el método aprendizaje supervisado, SL, con ejemplos cotidianos, definiciones generales y finalmente reforzando las características de algoritmos en AI (indicadas el día 1 del taller) ahora con SL.

Luego de un recreo, el segundo bloque se hizo de forma grupal manteniendo la asignación del día anterior, con el fin de practicar el método SL usando ML4kids. El objetivo con esta actividad era que los/as participantes pudieran ejercitar las etapas de SL construyendo el clasificador de sentimiento en un mensaje. Al inicio se aprecia confusión con el propósito de esta plataforma, por ende se recomienda volver a mencionar ejemplos de servicios (e.g, etiquetado en redes sociales) y las ventajas de este sitio para implementar clasificadores (de texto, imagen o audio) versus programarlos solo con reglas *si-sino*.

Otras complicaciones al usar ML4kids fue acceder al sitio sin *login*, pero los obstáculos más significativos ocurrieron al entrenar sus clasificadores, pudiendo ser por dificultad de usar la combinación de teclas `ctrl+c` y `ctrl+v`, o debido al estar usando tres aplicaciones para realizar esta tarea: editor de notas, navegador y Zoom. Dado esto, es válido anticipar que ExtBi permite menor dificultad inherentes al cambio de contexto, dado que el conjunto de entrenamiento se puede definir importando un archivo `.txt`. Por otro lado, algunos participantes identifican determinadas complicaciones con la usabilidad del sitio: incompatibilidad con navegadores (e.g, Edge) y dispositivos móviles, también textos del sitio en español (e.g, “Rasguño 3” en vez de “Scratch 3”) y nombres de bloques en inglés. Además, se evidenció dificultades para usar Google Drive y acceder al material o subirlo por parte de algunos participantes; quizás un video explicativo podría abordar estas complicaciones y mostrar cómo solucionarlas.

CI - ML4kids. Luego de permitir a cada niño o niña construir su primer clasificador con ML4Kids, se realizó el CI con el propósito de que los y las participantes pudieran construir otro clasificador de texto, esta vez sobre noticias de animales y videojuegos. Para esto, se utilizó el listado de tareas indicado en el Apéndice E. Con esto, uno de los primeros aspectos observados fue que los y las participantes presentan perplejidad al mencionarles que la actividad se denomina *constructive interaction* o CI, por ende se sugiere indicar que se hará sencillamente un ejercicio colaborativo. Una complicación presente, nuevamente al definir el conjunto de entrenamiento, fue que algunos indicaron el listado del *dataset* completo en un solo ejemplo, lo que sugiere poco entendimiento sobre la unidad de ejemplo y la necesidad de varias para entrenar. Dada esta confusión, podría ser útil realizar un clasificador usando un párrafo como unidad de ejemplo de entrenamiento, y clasificar lo expresado en este. También en este sentido, podría ser útil solicitarle a un participante que lea la lista de cada categoría, y así reducir riesgo de rezagados/as. Se aprecia que entre participantes de cada grupo, hay quienes van siguiendo las indicaciones del tutor, mientras que otros se adelantan o van atrasados.

Sobre las tareas definidas para el CI y su ejecución, se procuró recordarles —y clarificarles— a los y las participantes la modalidad de la actividad: verbalizar lo que van haciendo y compartir pantalla. A medida que las y los participantes iban realizando tareas, los facilitadores les hacían preguntas abiertas a alguno del grupo sobre lo realizado. Por ejemplo, al consultarles “¿es posible en ML4kids obviar la etapa dos de ‘aprendizaje y probar’?”, hubo respuestas

correctas y erróneas como: “*es necesario que aprenda sino no identifica los ejemplos aún*” o “*solo para probar precisión o confianza al clasificar un nuevo ejemplo (en el sitio)*”. En un grupo, hizo sentido a los participantes el usar la metáfora de *estudiar ejemplos* para referirse a la etapa dos de ML4kids, *aprender o configurar* el modelo del clasificador. Finalmente podría ayudar a comprender la importancia de la etapa de aprendizaje a que se imaginen a ellos y ellas mismas cuando estudian lo visto en el colegio, ya que no solo basta que la profesora les presente un tópico y ejemplo para conocer algo.

Respecto a lo observado en cada grupo, 5 de 6 equipos finalizaron todas las tareas del CI, construyendo un clasificador de texto como el que se presenta en la Figura 4.2. Los y las participantes del Grupo A en general supieron qué pasos realizar y cuándo alguno no, un compañero/a lo ayudó. En el Grupo B, se aprecia que identificaron claramente la relación entre categorías y ejemplos en la fase de entrenamiento. Algunos del Grupo C mencionaron que el indicar ejemplos para entrenamiento es tedioso. El Grupo D manifiesta interés extra en comprender, luego de codificar el clasificador algunos/as preguntan si pueden ir avanzando tareas o hacer el clasificador más sofisticado como mejorar cuando se presenta el mensaje o resultado. El Grupo E varió y realizaron un clasificador entre animales aéreos y terrestres, generando su propio *dataset* mencionando propiedades de cada categoría de animal como ejemplos (e.g, tienen alas, tienen garras, etc). Sin embargo, al momento de probar la confianza de la predicción no hubo claridad cuál ejemplo usar, si alguno usado en el entrenamiento u otro nuevo. También algunos del Grupo E presentaron confusión con el proceso de clasificación en sí, en concreto con el bloque **recognise** de ML4Kids y la finalidad del parámetro que requiere dicho bloque. Esto último sugiere incompreensión en cómo se procesa un nuevo valor de texto, contenido en bloque **respuesta**, y cómo se llega a la predicción de la categoría. El grupo F llegó hasta la tarea definir categorías y ejemplos entrenamiento, no finalizando el CI.

A pesar de que la mayoría de los grupos realizaron todas las tareas, persisten dudas en algunos participantes sobre la relación con cada etapa del algoritmo de IA (entrenar, aprender, predecir). El tiempo promedio fue de 60 minutos, y comparando entre CI (usando ML4kids) de enero y julio se mantiene.

Sesión 3 - Probabilidades y Bayes ingenuo

Se inició la sesión tres presentando algunos clasificadores de noticias, resultado del CI con ML4Kids en la sesión anterior, e indicando que el bloque que realiza la predicción (bloque **recognise**) tiene la característica de ser “caja negra” ya que no posibilita saber el conjunto de pasos que sigue al determinar una categoría para un nuevo ejemplo. En base a esta problemática se planteó entonces como meta del taller indagar en esta “caja negra”, considerando el método SL para definir la serie de pasos —algoritmo— que determinan una categoría dado un nuevo valor. Finalizando el repaso, se volvió a mencionar que la clasificación puede ser entre N categorías y no solo dos, dependiendo de los conjuntos de entrenamiento.

Así, para definir el proceso de clasificación, la sesión se orientó en exponer los conceptos matemáticos involucrados en el algoritmo Bayes ingenuo, como probabilidades y probabilidad condicional, proponiendo algunos ejercicios básicos a realizar con hoja y lápiz. Por consiguiente, también se presentó la fórmula del teorema de Bayes y su utilidad para determinar la probabilidad de un *Evento A* dado un *Evento B*, en términos de la probabilidad de

un *Evento B* dado un *Evento A*. Ya que el teorema de Bayes es un t3pico que se ense1a en niveles superiores y Universidad, este se present3 mediante ejemplos de casos que permitieron evidenciar la utilidad del teorema, y los participantes no realizaron ejercicios o c1culos num3ricos espec3ficos. Antes de continuar con el segundo ejemplo sobre teorema de Bayes, se hizo un *break* y luego se llev3 a cabo la actividad con Kahoot. El an1lisis de este cuestionario se presenta en Secci3n 5.2.2 y en s3ntesis se pregunt3 sobre el contenido visto hasta la sesi3n 2, es decir no hubo preguntas sobre conceptos matem1ticos.

De este modo, luego de un segundo *break* se retom3 el 3ltimo ejemplo sobre el uso de la f3rmula de Bayes, 3til para presentar por primera vez clasificaci3n siguiendo los pasos del algoritmo Bayes ingenuo. A saber, se definieron los pasos para lograr clasificaci3n y luego se explic3 que a esta secuencia se le conoce como *algoritmo Bayes ingenuo*. En relaci3n a las acciones o pasos definidos, hubo preguntas y tambi3n comentarios indicando que se entend3a. Siguiendo la planificaci3n, se present3 un segundo ejemplo de c3mo se aplica el algoritmo para clasificar texto, esta vez sin presentar c1culos num3ricos sino que enfatizando en los particularidades del algoritmo, por ejemplo explicando la raz3n del nombre de “ingenuo”. Un participante muestra inter3s preguntando en qu3 otras aplicaciones, adem1s de clasificar texto, se puede aplicar este algoritmo. En estos casos, se recomienda mencionar ejemplos sobre clasificadores de im1genes, videos u otros de texto, como identificar el idioma en cual est1 escrita una oraci3n.

Finalmente, hubo inter3s tambi3n en los participantes por saber c3mo se implementar3a el clasificador Bayes ingenuo en Scratch, anticip1ndose a lo planificado. Sin embargo, y justamente, la siguiente actividad consisti3 en presentarles los bloques Bayes ingenuo para posteriormente presentar el clasificador con ExtBi. De este modo, se les ense13 los cuatro bloques Scratch que considera la extensi3n Bayes ingenuo: “clasifica”, “entrena”, “bayes categoria” y “maxima”. Una vez explicado el prop3sito de cada bloque y los par1metros que requiere, se present3 tambi3n los bloques auxiliares necesarios para clasificar texto, como el bloque —“respuesta”— donde se almacena el nuevo valor a clasificar, o la lista que mantiene los valores cuando se calcula la probabilidad con Bayes por cada categor3a. De este modo se present3 el clasificador completo, Figura 4.4. Las primeras impresiones de los participantes sobre el algoritmo implementado con Scratch, fueron de asombro y comentarios como “*es muy largo*”. Considerando que la mayor3a no tiene experiencia en Scratch antes del taller, por ende limitada pr1ctica en programaci3n, una posible interpretaci3n de sus opiniones sobre el clasificador puede ser que se refieren a la longitud del texto (*label*) en los bloques, o la cantidad de bloques utilizados. Finalmente recordar, que esta primera vez que se vi3 algoritmo Bayes ingenuo fue al final de la sesi3n, luego de ver los conceptos matem1ticos, y se aprecia cansancio en los y las participantes.

Sesi3n 4 - Algoritmo y bloques Bayes ingenuo

El 3ltimo d3a, la sesi3n inici3 con un resumen de los distintos clasificadores de textos programados con Scratch en el taller (sin ML, con ML4Kids y con bloques Bayes ingenuo), sintetizando lo que se ha hecho y por qu3 se ha hecho. Luego, se expuso por segunda vez el algoritmo Bayes ingenuo y los bloques Scratch para implementarlo. A continuaci3n, con tal de que los y las participantes pudieran construir su primer clasificador de texto con Bayes ingenuo, se volvi3 a la modalidad de grupos y as3 ejercitar con ayuda m1s focalizada mediante

la guía de un tutor/a. Durante esta etapa de exploración y práctica, los y las participantes demuestran interés en querer personalizar el resultado de la clasificación, por lo cual se podría considerar una actividad donde se les permita crear disfraces según las categorías de un clasificador.

CI - ExtBi. Luego de un *break*, en el segundo bloque se llevó a cabo el CI utilizando la ExtBi para construir un clasificador de texto, según el título de una noticia sobre videojuegos o animales. Se continuó en grupos, manteniendo los mismos participantes y tutora/es; y las tareas consideradas en la actividad se definen en el Apéndice F. Así, en el Grupo A se aprecia mayor claridad sobre qué hacer en cada tarea definida, al comparar con el CI de ML4kids, resultando en que seis de siete participantes logró construir el clasificador correctamente.

Mientras en el Grupo B, antes de iniciar el CI el tutor realiza breve repaso de cómo subir programas a Drive. En la etapa de entrenamiento del clasificador, el tutor define previo a la tarea “5 - *entrenar clasificador, categoría noticias animales*”, hacer la subtarea crear lista de ejemplos para cada categoría (no definida en el instructivo). Un hecho llamativo fue que el tutor no mencionó que esta vez se construía un clasificador de texto para “noticias” y algunos/as participantes nombraron su clasificador de “sentimiento”, pero rápidamente todo el grupo lo corrige rehaciendo solo la tarea 4. Este inconveniente en ML4Kids hubiese significado eliminar y crear un nuevo proyecto, rehaciendo desde la tarea 1. Luego, un participante describió correctamente cómo se agrega el valor retornado por el bloque `bayes categoria` a la correspondiente lista. Esta estructura de datos, al mantener los porcentajes de cada categoría, resulta clave en el funcionamiento del algoritmo ya que es el parámetro del bloque `maxima`. También, las y los participantes apreciaron diferencias en el código cuando se presentó directamente lo retornado por el bloque `maxima`, o cuando se usó una variable para almacenar el resultado e imprimiendo luego su contenido. Finalmente, la mayoría en este grupo indicó que les quedó claro cómo se construye el algoritmo; para medir estas percepciones más durabilidad o persistencia de los conocimientos adquiridos en el taller, se podría realizar una entrevista grupal a posteriori.

Para el Grupo C, en la tarea 10 un participante solicita ayuda del tutor, por ende podría considerarse que la tarea no es entendible. Para solventar esto, además intentar refinar el enunciado, podría considerarse subdividir la tarea en dos. Algunos se apoyan en el clasificador anterior para identificar qué hacer, cambiando los valores según el clasificador de noticias, lo que también sugiere incorporar una actividad donde se entregue un programa Scratch y ExtBi con el fin de que las y los participantes lo corrijan o mejoren. Mientras que, en relación al comportamiento o percepción de participantes usando ExtBi y ML4kids, una participante indica que la primera es más rápida y tiene todo el clasificador a la vista, haciendo referencia a que con ML4Kids necesita de entrenar y configurar el modelo en un sitio y luego programar en Scratch (otro sitio o página externa). Y sobre el algoritmo en sí, algunos participantes mencionan que para reconocer una nueva categoría habría que crear una lista de ejemplos, entrenar la categoría y luego calcular Bayes para esta, justificando así su comprensión en el orden de pasos en Bayes ingenuo y que la clasificación no está restringida solo entre dos categorías. Al igual que en otros grupos, mencionan que si tuviésemos más ejemplos de entrenamiento, el clasificador podría mejorar su predicción dado que tendría más información para dar un resultado.

Por otro lado en el Grupo D, las y los participantes enriquecen el mensaje respuesta del clasificador de sentimiento, para lo cual agregaron una comparación del resultado bloque *maxima*. Luego, una participante nombró a la lista de ejemplos para la categoría de animales como “*ClasificadorAnimales*” mientras que sus otros compañeros/as definen un nombre más preciso, como simplemente “*animales*”. De todas formas, esto sugiere que probablemente falta reforzar el propósito de la lista: entrenar una categoría específica. En general, se destaca sinergia en el grupo, donde un participante que no sabía cómo realizar una tarea, una compañera estaba dispuesta a guiarle.

En relación al Grupo E, un participante mientras realiza la tarea 2 va definiendo clasificación como “*poner cierta cosas en un grupo para que quede ordenado*” y luego menciona que por ende el siguiente paso es entrenar. Otro participante, al añadir la extensión ExtBi al entorno según la tarea 3, menciona que crea los bloques (en vez de indicar que se agregan). Continuando en el desarrollo de tareas, realizaron comparaciones entre ML4kids y ExtBi; en concreto indicaron que en ambas se usa una lista pero en la primera es necesario escribir cada ejemplo, mientras que en la segunda se puede hacer importando el archivo. Asimismo, que en ML4kids una vez finalizada la tarea de aprendizaje para usar el clasificador luego hay que ir a Scratch, mientras que con ExtBi se hace todo en un solo sitio. Luego, algunos al identificar las categorías en los bloques *entrenar* y *bayes* señalan nombres distintos. Por ejemplo, al *entrenar* indican “*videojuego*” y al calcular la probabilidad con Bayes indican “*videojuegos* o *video juegos*” refiriéndose a la misma categoría. Esto último, hecho que también pasó en el taller de enero, permite concluir que hay dudas con el parámetro del bloque donde se indica el nombre de la categoría, y que se debe reforzar la necesidad de consistencia en los *ids* en un programa. Finalmente, algunos evidencian dudas al presentar el valor final, resultado del bloque *maxima*, evidenciando su escaso manejo de variables (definidas por usuarios y bloque “*respuesta*”).

Respecto al Grupo F, que no finalizó el CI con ML4kids llegando solo a la tarea 4, tampoco terminó el CI con ExtBi quedando en la tarea 7 (ingresar una nueva noticia). Dado que no pudieron concluir ningún CI, se podría mencionar que posiblemente al tutor le faltó promover la interacción o colaboración entre pares. Pero también hubo otros hechos que dificultaron la actividad, como que algunos participantes no podían compartir pantallas, problemas de conexión, o que un participante no logró identificar cuál extensión agregar.

A modo de cierre se puede mencionar que, en cada grupo a pesar de haber voluntarios para realizar una tarea, los tutores escogieron quién la realiza y así evitaron que solo quienes sabían participaran. También, para el segundo CI se observó mayor fluidez en los participantes al verbalizar lo que van realizando y aumentó la resolución de problemas directamente entre pares. Otro aspecto a destacar, es que participantes atrasados en las tareas para construir su clasificador, lograron realizar lo pendiente de forma autónoma (los tutores no le indican cómo hacerlo). Respecto al manejo de los bloques, los participantes identificaron cuáles utilizar pero no los llaman por su nombre sino como “el cuadrado”, “el redondo”, etc. En este punto se aprecia su poco o nulo manejo en programación y Scratch, así que además de reforzar los fundamentos de programación con este lenguaje, se sugiere consolidar la relación entre cada etapa del algoritmo Bayes ingenuo y cada bloque, induciéndoles a que lean cada pieza e intenten comprender qué hace, cuáles parámetros requiere y por qué.

En general, no se podría mencionar que hubo diferencia de habilidades entre niños y niñas. En algunos grupos una niña evidencia mayor habilidad durante el CI mientras que en otro grupo es un niño. También, en distintos grupos se percataron sobre algunas malas predicciones del clasificador, lo que sugiere considerar posibles soluciones como refinar el conjunto de entrenamiento aumentando la cantidad y/o la calidad del texto de ejemplo. Finalmente, hubo dos participantes que no completaron el último cuestionario pero sí entregaron su programa; con esto resulta necesario reflexionar si agregar una sección al cuestionario —*verbal probing*— donde se pueda subir el archivo —.sb3— podría mejorar la recolección de evidencia.

Síntesis de Hallazgos en Videos

- Sesión 1:
 - Con la exposición de introducción a algoritmos, y sus respectivos ejercicios, se evidenció que captan la definición principal de este concepto: un conjunto de pasos para resolver o solucionar algo. Luego, se presentó las singularidades de algoritmos en AI, ahora con ejercicios para identificar el conjunto de entrenamiento usado en algunas aplicaciones con AI; y debido a las respuestas de los y las participantes se evidenció que es posible adquirir noción sobre el propósito de un *dataset* y su relación con una determinada predicción.
 - En la sesión de Scratch, algunos participantes se percataron de que un programa —en este lenguaje— es similar a un puzzle, piezas o legos. Esta sesión se realizó en grupos. En 5 de estos 6 grupos se pudo finalizar lo planificado. Mientras que en el que no, se abordó hasta el concepto de inicializar listas importando un archivo txt. A pesar de que en los grupos (que finalizaron) declaran comprender el ejercicio final sobre uso de condicionales y lo implementan por sí solos, a este minuto del análisis no se evidencia que los tópicos presentados sean dominados aún por los y las participantes.
- Sesión 2:
 - Al practicar el método SL usando ML4kids, se aprecia confusión con el propósito de esta plataforma. Por ende se recomienda volver a mencionar ejemplos de servicios (e.g, etiquetado en redes sociales) y las ventajas de este sitio para implementar clasificadores (de texto, imagen o audio) versus programarlos solo con reglas *si-sino*. Por ejemplo, y si bien algunas/os participantes manifestaron complicaciones para acceder al sitio sin login, los obstáculos más significativos ocurrieron al entrenar sus clasificadores. Esto, pudo deberse al estar usando tres aplicaciones para realizar esta tarea —editor de notas, navegador y Zoom— con el fin de copiar y pegar los ejemplos; considerando además la poca experiencia del atajo con las teclas `ctrl+c` y `ctrl+v`. Y también, se evidenció dificultades para usar Google Drive y acceder al material, o subirlo por parte de algunos participantes.
 - Por otro lado, algunos participantes identifican determinadas complicaciones con la usabilidad del sitio: incompatibilidad con navegadores (e.g, Edge) y dispositivos móviles, también textos del sitio en español (e.g, “Rasguño 3” en vez de “Scratch 3”) y nombres de bloques en inglés.
 - CI con ML4Kids:
 - * Nuevamente hubo complicaciones al definir el conjunto de entrenamiento, algunos/as indicaron el listado del *dataset* completo en un solo ejemplo. Esto

sugiere poco entendimiento sobre la unidad de ejemplo y la necesidad de varias para entrenar.

- * En 5 de 6 equipos finalizaron todas las tareas del CI, construyendo un clasificador de texto como el que se presenta en la Figura 4.2. El grupo F llegó hasta la tarea definir categorías y ejemplos entrenamiento, no finalizando el CI. En general, se aprecia que entre participantes de cada grupo hay quienes van siguiendo las indicaciones del tutor, mientras que otros se adelantan o van atrasados.
- * Un grupo varió y realizaron un clasificador entre animales aéreos y terrestres, generando su propio *dataset* mencionando propiedades de cada categoría de animal como ejemplos (e.g, tienen alas, tienen garras, etc). Sin embargo, al momento de probar la confianza de la predicción no hubo claridad cuál ejemplo usar, si alguno usado en el entrenamiento u otro nuevo. Mientras que, en otro grupo hizo sentido a los participantes el usar la metáfora de “estudiar” ejemplos para referirse a la etapa dos de ML4kids, aprender o configurar el modelo del clasificador. A pesar de que la mayoría de grupos realizaron todas las tareas, se perciben dudas en algunos participantes sobre como realizar cada etapa del algoritmo de IA (entrenar, aprender, predecir).

- Sesión 3:

- Al presentar, por primera vez, un ejemplo de clasificación empleando la fórmula de Bayes —e implícitamente siguiendo el algoritmo Bayes ingenuo—; hubo preguntas y luego comentarios indicando que se entendían las acciones o pasos definidos.
- Un participante muestra interés preguntando en qué otras aplicaciones, además de clasificar texto, se puede aplicar este algoritmo. En estos casos, se recomienda mencionar ejemplos sobre clasificadores de imágenes, vídeos u otros de texto, como identificar el idioma en cual está escrita una oración.
- Finalmente, hubo interés también en los participantes por saber cómo se implementaría el clasificador Bayes ingenuo en Scratch, anticipándose a lo planificado.
- En la exposición de ExtBi, sobre sus bloques y un clasificador implementado con estos, las primeras impresiones de los participantes sobre el algoritmo implementado con Scratch, fueron de asombro y comentarios como “es muy largo”. Estas opiniones, pudieron referirse a la longitud del texto (label) en los bloques o la cantidad de bloques utilizados.
- Recordar que, la primera vez que se vió algoritmo Bayes ingenuo fue al final de la sesión, y luego de ver los conceptos matemáticos; apreciando cansancio en los y las participantes.

- Sesión 4:

- Al exponer por segunda vez el algoritmo Bayes ingenuo y los bloques Scratch, esta vez no hubo opiniones de que fuese un algoritmo “muy largo” y algunos declaran comprender aún más.
- Durante la práctica, los y las participantes demuestran interés en querer personalizar el resultado de la clasificación, por lo cual podría incorporarse una actividad para crear disfraces según las categorías de un clasificador.
- CI ExtBi:

- * En un Grupo, nombraron su clasificador de “sentimiento” cuando iban a implementar el de “noticias”, pero rápidamente todo el grupo lo corrige rehaciendo solo la tarea 4. Este inconveniente en ML4Kids hubiese significado eliminar y crear un nuevo proyecto, rehaciendo desde la tarea 1.
- * Sobre el algoritmo en sí, un niño describió de forma precisa la etapa del algoritmo donde se calcula el valor la probabilidad de una categoría dado el nuevo valor (con el bloque `bayes`) y también cómo se agrega a la lista. Mientras que, por otro lado otros mencionan correctamente la modificación al programa con ExtBi, para reconocer una nueva categoría y que la clasificación no está restringida solo entre dos categorías.
- * En relación al comportamiento o percepción de participantes usando ExtBi y ML4kids, una niña señala que la primera es más rápida y tiene todo el clasificador a la vista, haciendo referencia a que con ML4Kids necesita de entrenar y configurar el modelo en un sitio y luego programar en Scratch (otro sitio). También al entrenar, otros participantes mencionan que en ambas soluciones se usa una lista, pero en ML4Kids es necesario escribir cada uno y con ExtBi se puede hacer importando el archivo.
- * En general, para el segundo CI se observó mayor fluidez en los participantes al verbalizar lo que van realizando y aumentó la resolución de problemas directamente entre pares.
- * Otro aspecto a destacar, es que participantes atrasados en las tareas para construir su clasificador, lograron realizar lo pendiente de forma autónoma (los tutores no le indican cómo hacerlo).
- * Respecto al manejo de los bloques, los participantes identificaron cuáles utilizar pero no los llaman por su nombre sino como “el cuadrado”, “el redondo”, etc.
- * También, en distintos grupos se percataron sobre algunas malas predicciones del clasificador, sugiriendo entonces refinar el conjunto de entrenamiento aumentando la cantidad y/o la calidad del texto de ejemplo.
- * Finalmente, además de sugerir reforzar los fundamentos de programación con Scratch, se recomienda consolidar la relación entre cada etapa del algoritmo Bayes ingenuo y cada bloque, induciéndoles a que lean cada pieza e intenten comprender qué hace, cuáles parámetros requiere y por qué.
- * No se podría mencionar que hubo diferencia de habilidades entre niños y niñas.

5.2.2. Cuestionarios

En los cuestionarios se revisó solo las respuestas de participantes quienes lo autorizaron expresamente, mediante el formulario de consentimiento y asentimiento. Para hacer un análisis del progreso de cada participante durante el taller, además de identificar el Grupo del cual fue parte en las sesiones focalizadas, también se analizó y comparó sus respuestas entre los distintos cuestionarios aplicados en el caso de estudio. Previo al análisis por pregunta, se definió un conjunto de posibles respuestas definiendo así una matriz de afinidad. Esta matriz, fue actualizándose por pregunta hasta poder categorizar la totalidad de las respuestas. De esta forma, la revisión de los cuestionarios se describe en los siguientes apartados.

Inicial

Fueron analizadas las respuestas de 28 participantes, 8 niñas y 20 niños. El tiempo medio en responder fue de 14 minutos. Respecto al dispositivo o equipo que declaran utilizar durante el taller, 19 indicaron computador, 5 señalaron tablets y 4 participantes ambas alternativas. En este cuestionario hubo preguntas sobre el perfil del participante, y preguntas de contenido como se indica en el Apéndice C.

En la primera pregunta sobre conceptos, “*P3: Describe qué entiendes por Algoritmo*”, 10 participantes (35 %) aciertan en la definición, mientras que 8 (28 %) responden que es una secuencia de pasos pero sin mencionar que tienen un fin o resuelven algo. También, hubo 3 participantes que lo definen como algo matemático y 2 indicaron un conjunto de algo (sistemas, números, etc). Solo 1 describe un ejemplo de algoritmo (videojuegos) y 4 indican que no saben o no responden. En síntesis, al iniciar el taller solo un tercio (aproximadamente) de los participantes conoce la definición de algoritmo.

En la siguiente, pregunta “*P4: ¿A qué te suena Inteligencia Artificial (IA)?*”, no hubo ninguna respuesta correcta o cercana a la definición usada en el taller, que define AI como el arte de crear programas (o máquinas) con funciones que requieren inteligencia si las hiciera un humano. La mayoría, 19 participantes o 65 %, asocian AI a robots o artefactos tecnológicos (como apps o servicios); mientras que 6 niños y niñas (21 %) responden que es la imitación de la inteligencia o comportamiento humano, y 3 como una solución a algo o información de los computadores. Luego, la pregunta “*P5: ¿Has oído sobre qué es un 'conjunto de entrenamiento'? De ser así comenta*”, la mayoría 19 participantes (65 %) indicaron que no. Solo una niña y un niño presentan una respuesta cercana a la correcta (conjunto de ejemplos para entrenar un modelo de AI), mientras que otros 2 lo definen como el conjunto para probar o validar un modelo de AI.

Continuando en la pregunta “*P6: Define con tus palabras qué es 'clasificación'*”, 21 participantes (75 %) lo definen como un ordenamiento o ponderación de alguien o algo, mencionando como ejemplo las calificaciones (que les dan en el colegio o en un Kahoot). Respecto a la respuesta correcta, de separar elementos en categorías en base a sus características, 7 participantes (25 %) responden acertadamente. En base a esto, se puede decir que aplicaciones o ejemplos de clasificación son cercanos a ellos y ellas, mas al definir este concepto no hubo precisión.

En la penúltima pregunta del cuestionario, que señala “*P7: ¿Has programado previamente con Scratch?. De ser así comenta brevemente o indica el link a algún proyecto que hayas creado*”, 19 participantes (67 %) declaran no haber programado antes. Mientras que 1 lo conoce pero no ha programado, y 7 restantes declaran tener alguna experiencia creando animación o videojuegos, pero no comparten detalles sobre el algoritmo usado o funcionamiento, sino solo una descripción de sus imágenes o personajes. Y finalmente la pregunta , “*P8: ¿Has oído del algoritmo Bayes ingenuo? De ser así, comenta con tus propias palabras de qué se trata*”, 26 participantes (92 %) no habían oído de Bayes ingenuo. Llama la atención que las otras 2 respuestas fueron idénticas y cercanas a lo correcto, refiriéndose solo a la utilidad de este para hacer clasificación, pero no se mencionan características del algoritmo o la secuencia de pasos que implica, lo que puede indicar que pudieron haberla investigado previamente, ya que en el afiche se hizo referencia a estos tópicos.

Verbal Probing - Post CI ML4Kids

El análisis fue a las respuestas de 23 participantes, quienes completaron el cuestionario en un tiempo aproximado de 20 minutos. Así, la primera pregunta solicitó indicar qué dispositivo utilizaron en la actividad práctica, donde 21 participantes (91 %) declaró usar computador y solo 2 tablet. Luego en la segunda pregunta, “*¿Cómo describirías el concepto de ‘Clasificación’ a algún compañero/a del colegio?*”, ya no aparece como respuesta lo referente a calificaciones (e.g. notas o Kahoot) según lo observado en la pregunta del cuestionario inicial. Ahora lo definen mayoritariamente como la separación o agrupación de objetos, utilizando palabras como categoría, clases y etiquetas, evidenciando así abstracción en sus definiciones.

En la siguiente pregunta “*El bloque ‘recognise’ determina la categoría del valor ingresado. Para lograr esto ¿cómo crees que trabaja internamente?*”, hubo 7 participantes que indicaron mediante una comparación con el conjunto de entrenamiento. También, hubo otros 3 participantes, 2 niños y 1 niña, que fueron un poco más precisos indicando que la clasificación es en base a la categoría donde hubo mayor similitud. Finalmente, hubo 6 participantes que señalaron no saber, mientras que en el resto hay una dispersión en las respuestas.

La penúltima pregunta: *Imagina que deseas mejorar tu clasificador agregándole nuevos ejemplos sobre noticias de videojuegos, ¿cuáles serían los pasos para lograrlo con ML4Kids?*, la mayoría (12 participantes) mencionan actualizar solo el conjunto de entrenamiento, faltando indicar las etapas restantes: aprender, crear y reabrir Scratch. De todos modos, 2 participantes fueron más precisos mencionando otra etapa además de entrenamiento, y otros 3 participantes respondieron de forma correcta completamente. Una respuesta llamativa fue la de un participante que describió de forma correcta la construcción del clasificador con los bloques necesarios, y que si bien no tributa a la pregunta sí aporta antecedentes sobre la comprensión al programar un clasificador con ML4Kids.

Finalmente, en la pregunta: “*Sin considerar que los bloques de ML4Kids están en inglés, ¿qué otra(s) dificultad(es) consideras que tiene esta herramienta?*”, algunos aprecian las restricciones de SL al requerir un volumen considerable de datos de entrenamiento para que funcione mejor. Respecto a los bloques o Scratch, se aprecia que 4 participantes hacen referencia a la utilidad o usabilidad de estos indicando que no es claro qué hace cada uno, ya que los consideran similares en aspecto. Asimismo, hubo comentarios sobre el sitio y excepciones ocurridas durante su uso, por ejemplo no poder crear una categoría durante el entrenamiento.

Kahoot

Con Kahoot se realizaron solamente preguntas con respuestas fijas o cerradas, entre dos y cuatro alternativas disponibles. Hubo algunos participantes que no habían usado esta herramienta y no conocían la modalidad de juego, por ende se introdujo brevemente cómo sería la actividad: presentación de las preguntas y la elección de las respuestas en un tiempo de 20 segundos para cada una. Se analizaron las respuestas de 25 participantes, y se indicó que los tutores también iban a participar (con el fin de promover la motivación) pero no se analizó las respuestas de estos. La actividad en total duró aproximadamente 40 minutos, esto porque luego de responder cada pregunta hubo unos minutos para discutir la respuesta correcta.

Hubo nueve preguntas en total y en la primera, “*P1: Inteligencia Artificial (IA) es*”,

respondieron correctamente 4 participantes. Para el análisis de estas respuestas se consideró comentarios de participantes como dificultad o no saber qué botones apretar, confusión en las alternativas, no poder ver directamente las preguntas, por mencionar algunos. Por ende, es probable que algunas/os no pudieron o supieron cómo responder, pero quizás tenían la respuesta. Nuevamente se procedió a explicar la modalidad de juego y cómo debían responder a la pregunta que iba apareciendo. De todos modos, comparando con las respuestas a la pregunta similar del cuestionario inicial, donde se solicita describir AI, en esta oportunidad hubo un mayor acierto.

Luego se continuó con una pregunta sobre un caso donde debían definir el conjunto de entrenamiento, en concreto “*P2: El reconocimiento facial ¿con cuál conjunto de entrenamiento se logra?*”, respondiendo correctamente el 96% de los participantes (solo uno no lo hizo). Continuando el análisis de preguntas sobre conjunto de entrenamiento, fue “*P6: Si tuvieses que construir un traductor de voz a texto ¿cuál conjunto de entrenamiento usarías?*”, con las opciones (a) películas subtituladas o dobladas, (b) canciones en mp3 o (c) libros en formato pdf. Aquí, la respuesta correcta (a) tuvo mayor cantidad de aciertos con 10 participantes, luego (c) con 7 participantes, y finalmente (b) con 6 participantes; mostrando así una mayor dispersión en las respuestas que en la pregunta anterior, pero igualmente el porcentaje de acierto es mayor que para las otras opciones. Otra pregunta sobre entrenamiento, fue “*P8: El detector de correo no deseado ¿con cuál(es) ejemplos se entrena para predecir?*” con las opciones (a) correos recibidos no deseados, (b) correos recibidos deseados y (c) todas las anteriores. En esta pregunta la mayoría, 19, indicó la alternativa (a); sin embargo, la correcta era la (c), y esto se explicaría dado que probablemente la mayoría no da un uso frecuente a su correo (como también se apreció en las complicaciones de uso de Google Drive al inicio del taller). Y la última pregunta sobre entrenamiento fue “*P9: ¿Cuál conjunto de entrenamiento es más importante para predecir el clima?*”, con las opciones de reporte sobre (a) temperatura, (b) humedad, (c) precipitaciones y (d) todas las anteriores; y a diferencia de la pregunta P8 esta vez acertó la mayoría, 17 participantes, indicando la opción (d) todos los reportes.

En otro tipo de preguntas, al consultar sobre la definición de algoritmo nuevamente, tal como se hizo en el cuestionario inicial y esta vez se solicitó indicar verdadero o falso a la siguiente afirmación “*P3: Hacer un algoritmo es lo mismo que programar*”; encontrando que 18 participantes (72%) de forma correcta indicaron que es falso, diferenciando así entre definir un algoritmo y programar. Otra pregunta fue sobre la predicción que realiza un sistema con AI, en particular “*P4: Google Maps te recomienda la ruta a tu destino ¿cómo es la ruta recomendada?*”, y las alternativas fueron la ruta: (a) con más sol, (b) más corta, (c) mejor paisaje o (d) menos policías. La mayoría (68%) respondió de forma correcta al indicar la alternativa (a), sin embargo se aprecia que algunas/os escogen (d) pudiendo haberse confundido con la predicción de otra aplicación como Waze.

La últimas dos interrogantes en analizar, fueron abordadas también en los cuestionarios previos. La pregunta “*P5: Algoritmos de IA tienen las siguientes características*”, la cual no resulta trivial y además las respuestas disponibles eran similares aumentando más la complejidad para escoger la correcta, según se aprecia en el Apéndice D. Aquí los y las participantes ya evidencian mejora en sus respuesta; 8 participantes escoge la correcta, una cantidad mayor comparando con la pregunta del cuestionario inicial. Y en la pregunta “*P7: Clasificación se refiere a*”, se aprecia que la mayoría acierta (15 participantes o 60%), manifestando un incre-

mento al comparar las respuestas de la pregunta similar en el cuestionario post CI-ML4kids.

Finalmente, dado este análisis al cuestionario Kahoot, se sugiere realizar una actividad previa con esta herramienta, quizás realizar algunas preguntas de prueba para asegurar que todos manejan la modalidad. Además, se recomienda asignar puntaje a todas las respuestas y no solo a la correcta. En general las niñas y niños mostraron mucho entusiasmo, declaran que fue una actividad que les gustó y se manifestó al querer comentar sobre las preguntas y las respuestas que escogieron. Se podría haber preguntado o registrado de alguna manera el dispositivo ocupado en Kahoot, porque hubo varias dificultades al jugar según lo indicado al inicio de esta sección. Y para cerrar, no se aprecian diferencias entre participantes de cada grupo, pudiendo decir que no haber finalizado el CI en un grupo no supuso efecto negativo de sus integrantes.

Verbal Probing - Post CI ExtBi

Este último cuestionario tuvo dos partes, la primera enfocada en conocer percepciones de los y las participantes sobre la ExtBi y el algoritmo Bayes ingenuo; mientras que la segunda sobre otros conceptos vistos en el taller, con algunas preguntas desde otros test. De tal modo, en la segunda parte se toman determinadas preguntas del cuestionario inicial, más algunas presentes en el test para usuarios del estudio de LearningML [19], trabajo mencionado en el Capítulo 2 como alternativa educativa para introducir a ML. Así, el cuestionario se compone por preguntas con tipo de respuesta libre o abierta, y algunas con alternativa o cerrada. Al final del test, se preguntó a las y los participantes su percepción, comentarios o sugerencias sobre el taller. La cantidad de respuestas analizadas fueron las de 24 participantes por cada pregunta, y para responder el cuestionario tomaron un tiempo de 30 minutos aproximadamente.

Primera Parte. En la pregunta “*P 1.2 - ¿Cómo le describirías «clasificación» a alguna compañera/o del colegio?*”, se evidenció mayor cantidad (9 participantes) de respuestas correctas —o cercanas— según lo dicho en el taller, demostrando progreso en la definición respecto a los cuestionarios previos. Luego al responder “*P 1.3 - En un clasificador ¿para qué es necesaria la etapa de entrenamiento?*”, hubo un 33% aproximadamente (8 participantes) que lo hacen de forma acertada, e.g. “*para aprender a separar bien*” o “*para que la IA sepa cual texto va a que tema*”. Luego, la siguiente mayoría (9 participantes), indicó una respuesta cercana a lo esperado pero faltó referenciar a la predicción como propósito. Otra respuesta a destacar fue “*porque así establece una probabilidad independiente*”, evidenciando que adquirió —también— nociones sobre los tópicos matemáticos expuestos en el taller. Y con la pregunta “*P 1.4 - La afirmación «Un clasificador Bayes Ingenuo permite reconocer solo dos posibles categorías» ¿es verdadera o falsa?*”, la mayoría de las respuestas (16 participantes) acierta en que la clasificación con este algoritmo no está restringida solo a dos categorías.

La pregunta consecutiva apunta directamente al manejo de los bloques ExtBi, solicitando escoger la alternativa con la secuencia de pasos o bloques que ellos/as escogerían para implementar un clasificador de texto, y la gran mayoría (87%) acertó correctamente. Esta es una pregunta que permite concluir que las y los participantes pueden indicar correctamente el orden de pasos del algoritmo Bayes ingenuo para construir un clasificador con los bloques ExtBi. Luego, se continuó con el objetivo de conocer las nociones adquiridas sobre el algorit-

mo y al solicitarles: “*P 1.6 - Describe las principales fases que componen al algoritmo Bayes ingenuo*”, a pesar de encontrar respuestas muy precisas (25% o 6 participantes) como “*entrenar, probabilidades condicional y máxima*”, también hubo 8 participantes que escogieron la alternativa correcta en la pregunta previa y ahora no supieron qué responder. Dado esto último, se podría indicar que a pesar de poseer una correcta idea de la secuencia con bloques ExtBi, resta mejorar la descripción en qué consiste cada uno y cómo se asocia con las etapas de un algoritmo en AI (entrenamiento, definición del modelo y predicción).

Las últimas dos preguntas de la primera parte, apuntaron a producir una comparación entre ExtBi y ML4kids, en particular cómo se realiza la fase de entrenamiento en cada solución. De este modo al plantear en la penúltima pregunta “*P 1.7 - Imagina que deseas mejorar tu clasificador agregándole nuevos ejemplos de noticias videojuegos, ¿cuáles serían los pasos para lograrlo?*”, se esperaba que las respuestas indicaran que para agregar ejemplos solo es necesario aumentar la lista, y no crear una nueva estructura ni tampoco ir a algún sitio, apreciable en las respuestas de 8 participantes (más de un 33%). Mientras que 7 participantes dan una respuesta cercana o levemente imprecisa. Hubo una respuesta interesante que, aunque comprendió mal el propósito de lo que se preguntaba, indicó los pasos del algoritmo completo de forma correcta mostrando comprensión en la definición del algoritmo Bayes ingenuo. Finalmente, y al consultarles “*P 1.8 - ¿Cuál de las dos opciones (Bloques Bayes Ingenuo o ML4kids) consideras mejor para construir un clasificador y por qué?*”, se aprecia preferencia del 58% (14 participantes) por ExtBi dada su rapidez, sencillez, por ser más específico, entendible y valorando que esté todo en Scratch. También hubo participantes que aprecian de manera positiva ExtBi sobre ML4Kids, en particular durante la fase de entrenamiento, y una respuesta que indica preferir ML4kids es porque en ExtBi es “*todo manual*”.

Segunda Parte. Comenzó, solicitando indicar cuál dispositivo usaron durante la experiencia práctica con ExtBi. Aquí, la mayoría (19 participantes) declaró usar computador personal y otros cinco tablets. Con esto, y si se considera la pregunta anterior —última de la primera parte— no se podría indicar que el dispositivo usado determinó preferencia por ML4Kids o ExtBi.

Las siguientes preguntas, se plantearon principalmente con el fin de evidenciar el progreso de los y las participantes sobre los tópicos vistos en el taller. Se volvieron a replantear seis preguntas desde los cuestionarios previos. De este modo, al solicitar “*P 2.2 - Describe qué entiendes por algoritmo*” —también planteada en el cuestionario inicial— y se observa que se mantiene la cantidad de respuestas acertadas (10 participantes), pero también se aprecia mayor precisión en la definición de las otras 14 respuestas, progreso que coincide al describir el algoritmo Bayes ingenuo por parte de niños y niñas. La pregunta número tres “*P 2.3 - ¿A qué te suena Inteligencia Artificial (IA)?*”, que en el cuestionario inicial tuvo respuestas cercanas pero no precisas, en esta ocasión sí se encontró una respuesta exacta según la definición usada en el taller. Además se aprecian definiciones más cercanas y abstractas a lo esperado, con menos ejemplos de alguna aplicación o servicio de AI.

Luego se continuó con preguntas en relación a la técnica de SL y entrenamiento. Así, en la consulta “*P 2.4 - ¿Cuáles ventajas ofrece la técnica Aprendizaje Supervisado?*” donde la respuesta esperada era mejorar el desempeño en base a ejemplos, se encontró 10 participantes respondiendo de forma precisa y otros 10 participantes con la segunda opción más cercana.

Esto demuestra que la mayoría de los y las participantes logran visualizar la importancia de los datos ejemplos al definir un programa. Luego, la pregunta 5 de la segunda parte, planteó un enunciado tomado del test en el estudio sobre LearningML [19], caso hipotético que presenta dos posibles estrategias de entrenamiento como alternativas: una con mayor cantidad de ejemplos para una categoría, y la otra con la misma cantidad de ejemplos para ambas categorías. Aquí, solo 6 participantes (o 25 %) optan por la alternativa con entrenamiento desnivelado en cantidad de ejemplos, mientras que la mayoría (18 participantes) seleccionan la alternativa que realiza un entrenamiento parejo para cada categoría. Esto evidencia comprensión en el 75 % de quienes participaron, que un adecuado entrenamiento contribuye a una correcta predicción.

Continuando con las preguntas —también presentes en otros cuestionarios— la siguiente solicitó a los y las participantes seleccionar la afirmación sobre SL, con la alternativa correcta “*es una técnica que permite entrenar a programas con datos de muestra para que realicen determinadas tareas, en lugar de tener que especificar un conjunto explícito de pasos y reglas a seguir*”. Aquí, con más del 70 % de participantes con respuesta correcta, se podría decir que la mayoría captó la ventaja de entrenar versus programar las reglas directamente. Y la última pregunta de conceptos, apuntó a la característica de la predicción —mediante SL— en particular sobre su utilidad para tomar decisiones dado un valor que no conoce o no han estado precisamente en su entrenamiento, se encontró que el 58 % de los participantes logra comprender esta característica.

Las dos últimas preguntas se plantearon para conocer la percepción de los y las participantes sobre el taller completo, es decir, considerando todas las sesiones y actividades prácticas. Así, en la pregunta “*¿Cómo evaluarías el taller?*” hubo 17 participantes (70 %) que lo evaluaron como “*muy divertido*”, y otro 20 % como “*divertido*”. Solo dos participantes indicaron que “*a veces divertido y otras no tanto*”. Y finalmente, al preguntar “*¿tienes algún comentario o sugerencia para mejorar el taller en próximas versiones?*”, 14 participantes (58 %) indicaron que se entretuvieron o les gusto mucho, no ahondando en otros comentarios. Luego, la segunda mayoría (5 participantes) indicaron que le agregarían más actividades practicas (como Kahoot) y “más aplicaciones”; sin embargo, para esto último no es posible concluir si se refieren a otras herramientas o más escenarios de clasificación con Bayes ingenuo. La única respuesta sobre quitar algo en el taller, hace referencia a eliminar contenido matemático del diseño y pertenece a quién indica en la pregunta previa que encuentra el taller a veces divertido y otras no tanto.

Síntesis de Hallazgos en Cuestionarios

- Inicial
 - Analizadas las respuestas de 28 participantes.
 - Al iniciar el taller solo un tercio (aproximadamente) de los participantes conoce la definición de algoritmo.
 - Al preguntarles a qué les suena Inteligencia Artificial (IA), no hubo ninguna respuesta correcta o cercana a la definición usada en el taller. La mayoría, 19 participantes, asocian AI a robots o artefactos tecnológicos (como apps o servicios).
 - Al solicitarles definir qué es “clasificación”, 21 participantes lo definen como un ordenamiento o ponderación de alguien o algo, mencionando como ejemplo las

calificaciones (que les dan en el colegio o en un Kahoot). En general, se podría decir que aplicaciones o ejemplos de clasificación son cercanos a ellos y ellas, mas al definir este concepto no hubo precisión.

- Cerrando el cuestionario, 19 participantes (67%) declaran no haber programado antes, y 26 participantes (92%) no habían oído de Bayes ingenuo.
- Post CI ML4Kids
 - Fueron analizadas respuestas de 23 participantes.
 - Al solicitarles describir qué es “clasificación” ya no aparece como respuesta lo referente a calificaciones (e.g. notas o Kahoot), evidenciado en el cuestionario inicial. Ahora lo definen como la separación o agrupación de objetos, utilizando palabras como categoría, clases o etiquetas; evidenciando así abstracción en sus definiciones.
 - Ya que en ML4Kids el bloque `recognise` determina la categoría del valor ingresado, se les pregunto a los y las participantes cómo creen que trabaja internamente tal bloque. Aquí, un tercio indica que la clasificación es en base a la categoría donde hubo mayor similitud segun el conjunto de entrenamiento, otro tercio señaló no saber, mientras que en el resto hubo dispersión en las respuestas.
 - Al plantear un escenario hipotético, donde se intentara mejorar el clasificador agregándole nuevos ejemplos sobre noticias, se les solicito indicar cuáles serían los pasos para lograrlo con ML4Kids. La mayoría (12 participantes) mencionan actualizar solo el conjunto de entrenamiento, faltando indicar las etapas restantes: aprender, crear y reabrir Scratch; y 3 participantes respondieron de forma correcta completamente. Una respuesta llamativa fue la de un participante que describió de forma correcta la construcción del clasificador con los bloques necesarios, y que si bien no tributa a la pregunta sí aporta antecedentes sobre la comprensión al programar un clasificador con ML4Kids.
 - Al preguntarles sobre la(s) dificultad(es) que presenta ML4Kids, se aprecia que 4 participantes hacen referencia a la utilidad o usabilidad de los bloques indicando que no es claro qué hace cada uno, ya que los consideran similares en aspecto. Por otro lado, sobre SL algunas(os) aprecian la restricciones de esté método de ML, asociado a la necesidad de un volumen importante de datos de entrenamiento para que funcione mejor.
- Kahoot
 - Fueron analizadas 25 respuestas de los participantes. El cuestionario fue compuesto por preguntas con alternativas y solo una ofrecía puntaje.
 - En general, en las preguntas de casos relacionadas a la definición de conjunto de entrenamiento, el mayor porcentaje fueron respuestas acertadas.
 - También se aprecia progreso en la noción de algoritmo y programar, más la diferencia entre ambos conceptos. Además, hubo avance respecto al cuestionario inicial, al identificar características de los algoritmos en AI.
 - Se distingue una mayor cantidad de participantes que definen de forma más precisa “clasificación”, evidenciando incremento al comparar las respuestas de la pregunta similar en el cuestionario post CI-ML4kids.
 - En general, no se aprecian diferencias entre participantes de cada grupo, pudiendo

decir que no haber finalizado el CI en un grupo no supuso efecto negativo de sus integrantes.

- Post CI ExtBi

- La cantidad de respuestas analizadas fueron las de 24 participantes.
- Al preguntarles para qué es necesaria la etapa de entrenamiento al construir un clasificador, una respuesta acertada y llamativa fue “*porque así establece una probabilidad independiente*”, evidenciando que adquirió también nociones sobre los tópicos matemáticos expuestos en el taller.
- La mayoría (16 participantes) acierta que, la clasificación con Bayes ingenuo no está restringida solo a dos categorías.
- Para indagar sobre el manejo de los bloques ExtBi, se solicitó escoger la alternativa con la secuencia de pasos o bloques que ellos/as escogerían para implementar un clasificador de texto, y la gran mayoría (87 %) acertó correctamente. Esto, permite concluir que las y los participantes pueden indicar correctamente el orden de pasos del algoritmo Bayes ingenuo para construir un clasificador con los bloques ExtBi.
- Luego al preguntarles como describen el algoritmo Bayes ingenuo —y considerando el punto anterior— se podría indicar que a pesar de poseer una correcta idea de la secuencia con bloques ExtBi, a algunos le falta mejorar la noción sobre en qué consiste cada uno y cómo se asocia con las etapas de un algoritmo en AI (entrenamiento, definición del modelo y predicción).
- Comparando ExtBi con ML4kids al plantearles un caso hipotético donde debían mejorar los conjuntos de entrenamiento, en general indicaron que en la primera solo es necesario aumentar la lista y no crear una nueva estructura ni tampoco ir a algún sitio, como en la segunda herramienta. Hubo una respuesta interesante que, aunque comprendió mal el propósito de lo que se preguntaba, indicó los pasos del algoritmo completo de forma correcta mostrando comprensión en la definición del algoritmo Bayes ingenuo.
- Finalmente al preguntar cuál opción —ExtBi o ML4kids— considera mejor para construir un clasificador y por qué, se aprecia preferencia del 58 % (14 participantes) por ExtBi dada su rapidez, sencillez, por ser más específico, entendible y valorando que esté todo en Scratch. En general, aprecian de manera positiva ExtBi durante la fase de entrenamiento y una respuesta que prefiere ML4kids es porque en ExtBi es “*todo manual*”.
- Respecto a la definición de algoritmo, y comparando las respuestas de este cuestionario con el inicial, se aprecia que se mantiene la cantidad de respuestas acertadas (10 participantes), pero también se aprecia mayor precisión en la definición de las otras 14 respuestas, progreso que coincide al describir el algoritmo Bayes ingenuo por parte de niños y niñas.
- Con preguntas sobre el método SL, se demuestra que la mayoría de los y las participantes logran visualizar la importancia de los datos ejemplos al definir un programa; y que un adecuado (nivelado para cada categoría) entrenamiento contribuye a una correcta predicción. También, la mayoría indentificó la ventaja de entrenar versus programar las reglas directamente, y la utilidad para tomar decisiones dado un valor que no conoce o no está precisamente en su entrenamiento.
- Finalmente, para conocer la percepción de las y los participantes sobre taller, al

preguntarles sobre cómo evalúan el taller, la primera mayoría indicó que fue “muy divertido” y la segunda mayoría como “divertido”. Luego, al solicitarles alguna sugerencia, en general señalaron que les gusto mucho el taller no ahondando en otros comentarios, y solo uno dijo que le quitaría contenido matemático.

5.2.3. Programas Scratch

Todos los programas desarrollados en el taller por las y los participantes fueron usando el lenguaje Scratch 3. El primer día no usaron ninguna extensión de bloques, pero sí el segundo y cuarto día: los bloques ML4Kids y ExtBi, respectivamente. Así, en el repaso o introducción a Scratch el día uno, los programas realizados consideraron el empleo de operadores booleanos, bloque *respuesta*, listas y condicionales (*si-sino*). Los códigos fuentes —archivos *.sb3*— no fueron cargados a la carpeta Google Drive, pero al inicio de la sesión 2 algunos presentaron sus programas, instancia provechosa también para comentar entre todos/as lo desarrollado y repasar lo visto el día anterior.

Una de las últimas tareas en el CI del segundo día, consistió en que los y las participantes cargaran su programa ML4Kids a Google Drive. Así, 16 participantes pudieron subirlo a la nube, de los cuales 6 programas no se pudieron volver a ejecutar en ML4Kids. La plataforma indica que la razón principal de este impedimento es que el proyecto ya no está disponible o fue eliminado, a pesar de rehacer las etapas respectivas (tareas 1 a 6 según el listado del Apéndice E) para ejecutar el programa *.sb3*. Cabe señalar que ML4Kids no advierte sobre posibles inconvenientes de este tipo al crear los proyectos.

De todos modos, en los programas revisados se validó el uso de los bloques *recognise*, operador igualdad (o distinto a), condicional *si-sino*, y finalmente el bloque *decir* para presentar el resultado de la clasificación. De los 10 programas, 2 fallaron por no utilizar el bloque *recognise*; el valor ingresado por el usuario (bloque *respuesta*) se comparó directamente con el valor de una categoría, tal como se muestra en la Figura 5.2. Notar diferencia en la condición del bloque *si-sino* con Figura 4.2. De los 8 programas completos, se podría decir que hubo dos versiones: (1) una donde se imprime directamente el resultado de la clasificación, y (2) otra donde se usa un condicional para mejorar el mensaje final.



Figura 5.2: ML4Kids - Clasificador Noticias Incompleto

En el cuarto día, 22 programas construidos con la ExtBi fueron subidos a la nube por los participantes. La totalidad se pudo volver a ejecutar, revisando así el doble de programas que con ML4Kids. Con esto, se comprobó el manejo de los nuevos bloques acorde al algoritmo Bayes ingenuo, esperando un resultado como el clasificador presentado en la Figura 5.3. En cada programa, por ejemplo, se validó que en el bloque **entrenar** la lista de ejemplos esté correctamente poblada, y que se indicara la categoría con un nombre o *id* sugerente. También, se validó la inicialización vacía de la lista donde se mantienen la probabilidad de cada categoría dado el nuevo valor, que en el caso de la Figura 5.3 la autora nombró como “*Valores Bayes*”. Por otro lado, se revisó que en el bloque **bayes categoria** (1) se indicara la categoría según su *id*, (2) el uso de variable **respuesta**, y (3) agregar el resultado a la lista (*Valores Bayes*). Para finalizar, en el bloque **maxima** se validó que se usará con el parámetro de la lista con las probabilidades previamente calculadas (*Valores Bayes*), y finalmente la presentación del valor retornado o calculado es decir la categoría con máxima probabilidad.

Producto de la inspección de los programas, se puede mencionar que 18 implementaron el algoritmo Bayes ingenuo con la ExtBi, y que 4 estuvieron incompletos o les faltó algún paso. También, se puede mencionar que 2 participantes inicializaron solamente una lista de entrenamiento a pesar de haber creado las dos, una para cada categoría; los 18 restantes lo hicieron correctamente. También, hubo 2 casos donde el *id* de la categoría es verboso, una frase en vez de un nombre, pero igualmente su programa funcionó. Otros 2 participantes nombraron a la lista de ejemplos como “*Clasificador_Animales*” (y “*Clasificador_Videojuegos*”), lo que puede indicar necesidad en reforzar que el propósito de la lista es para entrenar, que debería utilizarse un nombre más sugerente, y que para clasificar es necesario otros pasos adicionales no bastando definir una lista. Igualmente, el nombrar o identificar a estructuras usadas en un programa es una técnica que se mejora con la práctica de la programación en general, y como se mencionó anteriormente, una minoría de los participantes declara tener experiencia codificando. Solo hubo un programa de los revisados que no terminó su clasificador, llegando hasta la tarea 6 finalizando la etapa de entrenamiento.



Figura 5.3: ExtBi - Clasificador Noticias de una Participante

También, en los programas se evidencia una diversidad para referirse a la lista donde se almacenan las probabilidades de cada categoría dado el nuevo valor, como “*Bayes*” o “*Valores_Bayes*”. Por otro lado, hubo casos donde indican un *id* de categoría en el bloque *entrenar* y luego en el bloque *bayes categoría* un *id* similar refiriéndose a la misma categoría (agregando espacio o una letra de más), sugiriendo reforzar que esto debe realizarse con el mismo nombre o *id*. Hubo un caso donde se equivoca al agregar el resultado de *bayes categoría* añadiéndolo a una lista de entrenamiento; y otro participante usó el bloque *bayes categoría* solamente para una categoría, faltando el paso de determinar el cálculo de la probabilidad para la otra categoría e incorporarlo a la lista respectiva. Además faltó indicar a los participantes que cuando los valores son demasiado pequeños, Scratch los presenta con la notación exponencial (e^N). Igualmente se debería cambiar cómo se presenta este valor para ser comprendido por escolares de esta edad, como valores entre 0 y 1.

Continuando con el análisis, hubo un participante que no indicó la lista con valores en el bloque *maxima*, sino que dejó el valor por defecto. Para presentar el resultado final, algunos usaron una variable para guardar el valor y luego lo imprimieron (con el bloque *decir*), mientras que otros/as no usaron variables sino que presentaron directamente el resultado del bloque *maxima*. Se aprecia creatividad para mostrar el resultado final, ya sea utilizando una mejora del mensaje o modificando el escenario y disfraz del personaje. En esta línea hubo casos donde lograron mayor interactividad con el usuario, por ejemplo agregando bloques *preguntar* adicional en el programa, y lo interesante es que a pesar de las modificaciones igualmente lograron el clasificador y no introdujeron errores manteniendo la correcta secuencia del algoritmo. En general, la mayoría logra implementar correctamente el clasificador con Bayes ingenuo, indicando el resultado esperado.

Síntesis de Hallazgos en Programas

- ML4kids:
 - De los 16 participantes que entregaron su programa, 6 de estos no se pudieron volver a ejecutar en ML4Kids. La plataforma indica que la razón es porque el proyecto ya no está disponible o fue eliminado, a pesar de rehacer las etapas respectivas (desde la tareas 1 hasta la 6 según el Apéndice E) para ejecutar el programa `.sb3`.
 - Los 10 programas que sí corrieron, 2 fallaron al no manejar correctamente el bloque `recognise`. De los 8 programas completos, se podría decir que hubo dos versiones: (1) una donde se imprime directamente el resultado de la clasificación, y (2) otra donde se usa un condicional para mejorar el mensaje final.
- ExtBi:
 - 22 programas construidos con la ExtBi fueron entregados. La totalidad se pudo volver a ejecutar, revisando así el doble de programas que con ML4Kids.
 - 18 implementaron el algoritmo Bayes ingenuo con la ExtBi y 4 estuvieron incompletos o les faltó algún paso.
 - 2 participantes nombraron a la lista de ejemplos como “*Clasificador_Animales*” (y “*Clasificador_Videojuegos*”), identificando la necesidad de reforzar que el propósito de la lista es para entrenar, y que para clasificar es necesario otros pasos adicionales. Además, algunos nombran de forma distinta las categorías en los bloques `entrenar` y `bayes`; por ejemplo “*videojuego*” en un bloque y “*videojuegos o video juegos*” en el otro. Esto último, sugiere reforzar la consistencia de `ids` en un programa.
 - Dado que un valor decimal demasiado pequeño en Scratch se presenta con la notación exponencial (e^N), se debería cambiar el formato para ser comprendido por escolares de esta edad.
 - Respecto al resultado de la clasificación, algunos evidencian dudas al presentar el valor final —resultado del bloque `maxima`— evidenciando su escaso manejo de variables (definidas por usuarios y bloque “respuesta”). Sin embargo, en otros/as participantes se aprecia creatividad para mostrar el resultado final, mejorando el mensaje o modificando el escenario y disfraz del personaje; manteniendo la correcta secuencia del algoritmo y no introduciendo errores.

5.2.4. Análisis con *Pattern Matching*

Según Yin, esta técnica compara un patrón basado empíricamente —es decir, uno producto de los hallazgos del caso de estudio— con uno declarado antes a la recolección de evidencia [29]. En concreto, los patrones previos corresponden a lo expresado en las proposiciones del caso mencionadas en la Subsección 3.3.1; por ejemplo, la primera refiere a la posibilidad para niñas y niños de reconocer las principales características genéricas de algoritmos en AI —conjunto de entrenamiento, acciones que aprenden un patrón y predicción—. Otros ejemplos de proposición señalan que, también a los y las participantes se les permite comparar programas con AI y sin AI, y abstraer determinados módulos del algoritmo Bayes ingenuo con ExtBi les posibilita a las y los participantes una concientización sobre este algoritmo al implementarlo.

Patrones en Registros de Video

Sesión 1. En el primer bloque sobre introducción a AI, según los ejercicios planteados en clase y gracias a las respuestas de los y las participantes, se aprecia que captan la noción de algoritmo (conjunto de pasos para solucionar algo), pudiendo definir las acciones para lavarse los dientes o agregar un número al teléfono. También, dado otro grupo de ejercicios para identificar el conjunto de entrenamiento usado en algunas aplicaciones con AI, y gracias a las respuestas de los y las participantes se evidenció que es posible adquirir noción sobre el propósito de un *dataset* y su relación con una determinada predicción. Con esto, ya es posible apreciar una coincidencia con la proposición sobre el reconocimiento de características generales de algoritmos en AI, indicada en la sección Subsección 3.3.1.

En la sesión grupal Scratch, se logró finalizar en 5 de los 6 grupos definidos. Un aspecto importante del cual se percataron algunos, fue que un programa con Scratch es similar a un puzzle, piezas o legos. También, se aprecia que logran crear listas en Scratch, importando un archivo `.txt`, punto importante para posteriormente definir el conjunto de entrenamiento. Cabe recordar que 19 participantes declaran no haber programado o usado Scratch con anterioridad, y a pesar de esto en los grupos donde se finalizó la clase completamente señalan comprender ejercicio final sobre uso de condicionales y lo implementan por si solos. El segundo día comenzó dando espacio para que algunos/as que hicieron los ejercicios presentaran sus programas, apreciando gran interés en querer compartir lo realizado por ellos y ellas, y con buenos resultados según lo esperado. Dado que no se contaba con tiempo para que el total de participantes pudiera explicar lo realizado, no es evidente que el contenido de Scratch presentado sea dominado todos los y las participantes. Sin embargo, igualmente se podría decir que se logró el objetivo de abordar los conceptos técnicos mínimos para intentar construir un primer clasificador sin AI.

Sesión 2. El objetivo de este día fue construir un clasificador de texto, primero sin AI (solo bloques por defecto de Scratch) y luego con AI (en concreto con SL y ML4Kids), aportando antecedentes para la proposición que plantea la posibilidad de comparar clasificadores sin y con AI. De esta forma y antes de presentarles el método SL, se planteó el ejercicio sobre construir un clasificador de texto el cual podían realizar solo con bloques *si-sino*, donde los y las participantes manifestaron complicaciones con este estilo de programación. Ejercitando SL y utilizando ML4Kids por primera vez, al construir el clasificador de texto evidenciaron dificultades principalmente en la fase de entrenamiento. De todas maneras y a pesar de que en un grupo consideran tediosa esta tarea en ML4kids, 5 de los 6 grupos logran correctamente la fase de entrenamiento. Incluso, un grupo adaptó las tareas del CI con tal de clasificar entre animales terrestres y aéreos, definiendo su propio *dataset*. En general, se aprecia comprensión en los y las participantes sobre la necesidad de realizar cada etapa indicada en el sitio de ML4Kids. Sin embargo, a esta altura del taller no es posible establecer que los y las participantes asocian lo logrado (con esta herramienta) a algoritmos en AI y sus particularidades (entrenamiento, aprendizaje y predicción). Con esto, se puede establecer que el segundo día del taller se comienza con la contribución de evidencia a las proposiciones que señalan sobre la comprensión de la relevancia de los datos en el entrenamiento, identificar cuándo y cómo se aplican los pasos de reconocimiento de patrones, y por ende cómo se produce una predicción o toma de decisión.

Para finalizar, al momento de programar el clasificador hubo confusión con cuáles bloques se logra la clasificación, en particular dudas con el bloque *recognise* de ML4Kids y la finalidad del parámetro que necesita dicho bloque. También se produjo desorientación en los y las participantes debido a la usabilidad del sitio, como incompatibilidad con navegadores o la incompleta traducción del sitio y bloques al español, como se evidencia en la Figura 4.2. Si ML4Kids solventara estos inconvenientes, podría asistir en mayor grado a la proposición que referencia la posibilidad de comparar el código de un clasificador con AI y otro sin AI.

Sesión 3. Con el repaso de la sesión anterior, presentando principalmente algunos de los clasificadores construidos con ML4Kids, fue posible contrastar las diferencias entre un programa sin AI y otro con AI. Además, y primordialmente, se buscó problematizar sobre las limitaciones de los programas construidos con ML4Kids, en particular sobre el bloque *recognise* como caja negra. Con esto se contribuye, nuevamente, a la proposición que indica la posibilidad de reflexión sobre clasificadores con y sin AI; pero también se comienza a materializar otra proposición, principal del caso de estudio, que tiene relación con la concientización del algoritmo Bayes ingenuo al abordar el misterio que implica la caja negra, el bloque *recognise* de ML4Kids.

Dados los ejercicios de la sesión 3, se evidenció comprensión en probabilidades y probabilidad condicional. Luego, al presentar ejemplos sobre la utilidad del teorema de Bayes, y posteriormente indicando que a una determinada secuencia de pasos se le conoce como algoritmo Bayes ingenuo, hubo preguntas que al responder generaron comentarios de participantes indicando que se entendía a modo general. También, se evidenció interés en participantes por saber qué otras aplicaciones, además de clasificar texto, es posible de realizar con Bayes ingenuo. Además hubo anhelo por saber cómo sería posible programarlo con Scratch. Una vez presentado los nuevos bloques se mostró el código de un clasificador, y dadas las reacciones de los y las participantes a este, se podría considerar una concientización inicial sobre el algoritmo Bayes ingenuo usando la ExtBi, como indica una de las proposiciones del caso de estudio.

Sesión 4. El último día, comenzó analizando el código Scratch de los distintos clasificadores de texto construidos en el taller. Respectivamente, programas con y sin AI, basado en reglas y basado en datos, usando solo condicionales y con ML4Kids o ExtBI. Gracias a esta síntesis, se pudo justificar porqué se presentaron los tópicos en el taller. Asimismo, se produjo un calce con las proposiciones que se refieren a la posibilidad de comprender las características de un algoritmo en AI, contrastar programas sin o con AI, y además comparar las herramientas o soluciones que permiten practicar y aprender sobre ML (ExtBi y ML4kids).

Luego de exponer por segunda vez el algoritmo Bayes ingenuo y la ExtBi, se continuó en modalidad de grupos para ejercitar la construcción del clasificador de texto con ExtBi. Posterior a esto y una vez finalizado el CI con ExtBi, se percibió noción en las y los participantes sobre algoritmo, coincidiendo así con la proposición de generar concientización sobre Bayes ingenuo. En relación a esto, un participante pudo describir correctamente cómo se agregan elementos a la lista para los valores Bayes, estructura de datos que mantiene las probabilidades de cada categoría dado el nuevo valor, clave en el funcionamiento del algoritmo. Además, algunos participantes distinguen diferencias en el código al presentar el resultado de la predicción, valor retornado por el bloque *maxima*, o usar una variable auxiliar para

luego imprimir el contenido. Sobre el algoritmo en sí, algunos participantes mencionan que para reconocer una nueva categoría (extra a las de animales o videojuegos) habría que crear una lista de ejemplos, entrenar la categoría y luego calcular Bayes para esta, aludiendo clara comprensión en el orden de pasos en Bayes ingenuo y que la clasificación no está restringida solo entre dos categorías. Igualmente, al mencionar que si tuvieran más ejemplos de entrenamiento, el clasificador podría mejorar su predicción dado que tendría más información para dar un resultado. Un aspecto a destacar, es que participantes atrasados en las tareas para construir su clasificador con ExtBi, lograron realizar lo pendiente de forma autónoma (los tutores no le indican cómo hacerlo). Finalmente, se menciona que hubo mayor cantidad de programas construidos —de forma correcta— con ExtBi que con ML4kids.

Respecto a la proposición acerca de identificar las dificultades y complicaciones que tuvieron los y las participante con la ExtBi, se podría considerar cuando algunos al identificar la categoría en el bloque Bayes señalan un *id* distinto que el definido en el bloque `entrenar`, por ejemplo “*videojuego*” y luego “*videojuegos* o *video juegos*” refiriéndose a la misma categoría. Además, cuando un participante nombra a la lista de entrenamiento para categoría de animales como “*ClasificadorAnimales*”, sugiere que probablemente falta clarificar el propósito de la lista de ejemplos y como se asocia a una categoría (*id*). De esta forma, en la solución completa (taller y uso de la ExtBi) se debe abordar la importancia de cómo nombrar una categoría más su lista de ejemplos, y lograr consistencia en los *ids* usados en el programa.

Para cerrar, se puede mencionar otra evidencia que hizo calce con la proposición de que los y las participantes pueden comparar la modalidad de cada herramienta (ExtBi y ML4Kids) al construir un clasificador de texto. Por ejemplo, cuando una participante indica que ExtBi es más rápida que ML4Kids y además tiene todo el clasificador a la vista, se refiere a que (1) cuando se predice con ML4Kids hay una demora en obtener el resultado (probablemente por latencia de respuesta en la API IBM Watson), y (2) con esa plataforma realiza el entrenamiento y configuración del modelo en un sitio para luego programar con Scratch en una página externa. Otra comparación presente se refirió al entrenamiento, en concreto indicaron que en ambas se usa una lista pero en ML4Kids es necesario escribir cada ejemplo, mientras que con ExtBi se puede hacer importando el archivo. Una última ventaja de ExtBi versus ML4Kids, sobre tener todas las funcionalidades para construir el clasificador en un solo sitio, se pudo apreciar cuando en un grupo se equivocaron al nombrar el clasificador; con ExtBi bastó con actualizar el parámetro de un bloque, mientras que en ML4Kids hubiese significado eliminar el proyecto y crear uno nuevo.

Patrones en Cuestionarios

Inicial y Primer Verbal Probing El cuestionario inicial sirvió para hacer un diagnóstico de los y las participantes, y además comprobar que la mayoría no tenía experiencia en Scratch; tampoco conocían la definición de AI ni clasificación; y evidentemente la mayoría declaró no conocer el algoritmo Bayes ingenuo. Lo positivo de este estado inicial, es que posteriormente se pudo apreciar con mayor nitidez la adquisición de los contenidos ofrecidos por la solución de este trabajo de tesis. Ya en el primer cuestionario *verbal probing*, luego del CI con ML4Kids, se puede apreciar por ejemplo que al definir clasificación se refieren la separación o agrupación de objetos, utilizando palabras precisas como categoría, clases y etiquetas, evidenciando así abstracción en sus definiciones. Luego, con las preguntas sobre ML4Kids se les permitió

reflexionar sobre lo realizado con esta herramienta, pudiendo así calzar con el patrón que hace referencia sobre la posibilidad de analizar las herramientas que permiten aprender sobre ML.

Kahoot. En el cuestionario con Kahoot, se aprecia calce preciso con la proposición que indica la posibilidad de reconocer las características de algoritmos en AI, ya que se consultó directamente sobre aquello, habiendo un tercio de los y las participantes que respondieron correctamente. Otra pregunta que tributa a esta proposición se produjo al consultar qué es AI, donde la evidencia relacionada destaca mayor acierto que en el cuestionario inicial. Por otro lado, otra contribución a esta proposición ocurrió al preguntar si hacer un algoritmo es lo mismo que programar habiendo una amplia mayoría en indicar que falso; progreso evidente ya que al cuestionario inicial la mayoría no venía con experiencia previa en programación, y tampoco tenían claridad en qué es un algoritmo. Y finalmente, otro patrón apreciable que permite concordancia con tal proposición, en particular sobre la relevancia del conjunto de entrenamiento y su correlación con la predicción, se aprecia en las respuestas acertadas — en su mayoría— a cuatro preguntas sobre *datasets*. Solo en una —sobre cómo entrenar el sistema de detección de spam— la cantidad de aciertos fue minoría, pero en las otras tres —sobre sistemas de traductor de voz, reconocimiento facial y de predicción de clima— hubo clara predominancia en seleccionar la alternativa con el entrenamiento adecuado para cada sistema.

Segundo Verbal Probing En el cuestionario posterior al CI con ExtBi, hubo un tercio de los participantes que respondieron acertadamente a la pregunta “*en un clasificador ¿para qué es necesaria la etapa de entrenamiento?*”, produciéndose así un patrón coincidente a la proposición que sostiene que las actividades prácticas permiten reforzar la relevancia de los datos en el entrenamiento, identificar cuándo y cómo se aplican los pasos de reconocimiento de patrones, y por ende cómo se produce una toma de decisión. A modo de ejemplo, un participante en su respuesta indicó “... *si no lo entrenas no sabe que es la tristeza (ni) la alegría*”. Luego, en relación a la comprensión de clasificación y en particular con Bayes ingenuo, un 70% aproximadamente señala que con este algoritmo no se restringe solo a clasificar entre dos categorías, pudiendo relacionar este patrón nuevamente a la proposición que postula la posibilidad de reforzar las características de algoritmos en AI y la predicción —o clasificación— a través de la práctica.

Uno de los principales patrones ocurridos, es aquel que coincide con la proposición referida a la posibilidad de concientización sobre el algoritmo Bayes ingenuo, palpable cuando una gran mayoría (87%) de participantes seleccionó correctamente la secuencia de bloques para implementar un clasificador de texto con ExtBi. De igual modo, al preguntarles sobre cómo describirían las fases que componen el algoritmo Bayes ingenuo, un 25% de los y las participantes respondieron de manera excelente. Mientras que, en relación a la proposición que sostiene la posibilidad de comparar la modalidad de cada herramienta —ExtBi versus ML4Kids— al construir un clasificador de texto, se observó un patrón de preferencia por ExtBi de más del 50% cuando se preguntó directamente cuál de las dos opciones considera mejor para construir un clasificador. Cabe destacar que no se aprecia que el dispositivo —tablet o computador— usado durante las actividades prácticas haya influido en esta preferencia.

Respecto a la proposición que declara la posibilidad de reconocer las características de

algoritmos en AI, otro calce de patrón se presentó al responder “*Describe que entiendes por algoritmo*” —similar a lo preguntado en Kahoot pero ahora con respuesta de tipo libre o abierta— teniendo un tercio de respuestas acertadas (10 participantes) y mayor precisión en la definición de otras 14 respuestas. Este progreso, se asocia también al poder describir el algoritmo Bayes ingenuo, por parte de las y los participantes. Siguiendo con esta proposición, en la pregunta “*P 2.3 - ¿A qué te suena Inteligencia Artificial (IA)?*” ahora se encontró definiciones más cercanas y abstractas a la definición esperada —con menos ejemplos de alguna aplicación o servicio— e incluso una respuesta fue exacta. Luego hubo preguntas cerradas o con alternativas las cuales igualmente tributaron a esta proposición, con preguntas en relación a la fase de entrenamiento mediante el método SL. Así, al consultar las ventajas que ofrece SL se encontró un patrón de un 40 % respondiendo de forma precisa y otro 40 % con la segunda opción más cercana. Después al presentar dos posibles estrategias de entrenamiento con SL —una de forma sesgada y la otra no— un 25 % optan por la alternativa de entrenamiento sesgada, mientras que la mayoría (75 %) seleccionan un entrenamiento nivelado para cada categoría, evidenciando que un adecuado entrenamiento contribuye a una correcta predicción del programa.

Continuando con preguntas sobre SL —también presentes en otros cuestionarios— al solicitarles a los y las participantes seleccionar la afirmación que consideraran correcta sobre este método en ML, se observó un patrón del 70 % de respuestas que se refiere a la ventaja de usar datos para entrenar un programa versus definir las reglas directamente en el código. Esta pregunta también permite asociar tal patrón con la proposición de posibilitar a los participantes reflexionar y comparar programas con AI y sin AI. Finalmente al preguntar sobre la característica de la predicción con SL, el patrón observado señala que el 58 % de los y las participantes destacan su utilidad para la toma de decisiones, dado un valor que no conoce o distinto a los del entrenamiento.

Patrones en Programas Scratch

En la sesión 2, algunos participantes presentaron sus programas con los ejercicios del día anterior, y cuando se les solicitó realizar un clasificador de texto (sin AI); previo a la exposición sobre SL. Esto pone en evidencia un calce con la proposición que indica la posibilidad de reflexión en programas sin AI, proposición #3. Luego, respecto a los programas con ML4Kids y dadas las características de la herramienta en general, se puede apreciar también un calce con la proposición #2. Sin embargo, al programar tal clasificador no se podría decir que se logra noción de algún algoritmo AI ya que finalmente este bloque esconde la lógica interna.

Respecto a los programas implementados usando ExtBi, en general se aprecia una correcta implementación del algoritmo Bayes ingenuo. También se evidencia, un error en algunos programas relativo a la inconsistencia de nombre o *ids* de las categorías. Aún así, dado el análisis de los programas desarrollados usando la ExtBi, es posible declarar que los patrones confirmados abordan prácticamente las siete proposiciones. Sí en todas, aunque en mayor o menor grado en una que en otra proposición. Para finalizar el capítulo, el calce de los patrones predichos y los empíricos ayuda a la fortaleza interna del caso de estudio.

Síntesis de *Pattern Matching*

- Las proposiciones para el caso de estudio se presentan en la Subsección 3.3.1
- Patrones en Videos:
 - Sesión 1:
 - * Al apreciar que las y los participantes captan la noción de algoritmo (conjunto de pasos para solucionar algo), más el desempeño de ellas y ellos en ejercicios para identificar el conjunto de entrenamiento usado en algunas aplicaciones con AI; es posible apreciar el calce con la proposición #1, sobre el reconocimiento de las características generales de algoritmos en AI.
 - Sesión 2:
 - * Dado el objetivo del día, construir un clasificador de texto primero sin AI (mediante reglas, usando solo bloques **si-sino**) y luego con AI (mediante SL usando ML4Kids), se aportó antecedentes para la proposición #3 que plantea la posibilidad de comparar clasificadores sin y con AI.
 - * También con el ejercicio de ML4Kids, se contribuyó a la proposición #2 referente a la comprensión de la relevancia de los datos en el entrenamiento, identificar cuándo y cómo se aplican los pasos de reconocimiento de patrones, y cómo se produce una predicción o toma de decisión.
 - Sesión 3:
 - * Al iniciar esta sesión, presentando algunos de los clasificadores construidos por participantes con ML4Kids, se pudo nuevamente reforzar el contraste entre un programa sin AI y otro con AI (proposición #3). Con esto, y primordialmente, se buscó problematizar las limitaciones de los programas construidos con ML4Kids, relativo al uso del bloque **recognise** como caja negra, mencionando entonces que con ML4Kids no era posible apreciar como se llegaba tomar tal decisión; produciendo entonces un calce con la proposición #2. Y además, se produce una contribución inicial a la proposición #4, referente a la concientización en niños y niñas del algoritmo Bayes; al intentar abordar el misterio que implica la caja negra del bloque **recognise**.
 - * Así, presentado los bloques en ExtBi y un clasificador con estos, se podría considerar otro calce con la proposición #4; según las reacciones de los y las participantes.
 - Sesión 4:
 - * Se comenzó con el último repaso que, además de comparar programas sin y con AI, contrastó la implementación con ML4Kids versus ExtBi. Así, se produjo un nuevo calce con las proposiciones #2, #3, y por primera vez con la proposición #5 al comparar tales herramientas o soluciones que permiten practicar y comenzar el aprendizaje sobre ML (proposición #5).
 - * Mientras programaban usando la ExtBi, algunos participantes nombraron el parámetro del bloque **bayes** —identificador de la categoría— con un id distinto para referirse a la misma categoría en el bloque **entrenar**. Por ejemplo “videojuego” y luego “videojuegos” o “video juegos”. Así, reconocidas estas complicaciones de los y las participante con la ExtBi, se produjo un calce con la proposición #7. En esta misma línea, el que un participante nombre a la lista de entrenamiento —para la categoría de animales— como “Clasificado-

rAnimales”, sugiere que falta reforzar como y donde se asocia este conjunto de ejemplos a una categoría usando la ExtBi.

- * Después, al oír de una participante que ExtBi es más rápida que ML4Kids y además tiene todo el clasificador a la vista, se produce un calce con la proposición #6 relativo a comparar la modalidad de cada herramienta. Otra comparación presente, se refirió a la forma de realizar el entrenamiento, y que si bien en ambas se usa una lista, en ML4Kids es necesario escribir cada ejemplo mientras que con ExtBi se puede hacer importando un archivo.
- * Finalizado el CI, se percibe noción en algunos participantes sobre Bayes ingenuo, coincidiendo así con la proposición #4 de generar concientización a dicho algoritmo.

- Patrones en Cuestionarios:

- Dado el primer cuestionario, se aprecia que la mayoría no tenía experiencia en Scratch, tampoco sobre algoritmos y menos sobre Bayes ingenuo. Lo positivo de esto, es que se pudo apreciar con mayor nitidez la adquisición de los contenidos ofrecidos por la solución en este trabajo de tesis.
- De esta forma, dadas las respuestas en el cuestionario posterior al CI ML4Kids, se aprecia avance —respecto al cuestionario inicial— en las y los participantes al definir el concepto de “clasificación” de forma más precisa. Además, en base a las respuestas sobre preguntas que tenían directa relación con el uso de ML4Kids, se aprecia un calce con la proposición #5 y la posibilidad de reconocer aplicaciones que permiten iniciar el aprendizaje sobre ML.
- En el tercer cuestionario, mediante Kahoot, al preguntar sobre las principales características genéricas de algoritmos en AI y dada las respuestas —donde un tercio acierta correctamente— se aprecia calce preciso con la proposición #1. Otras preguntas que tributan a esta proposición, y donde se aprecia nuevamente progreso respecto al cuestionario inicial, fue al consultarles si hacer un algoritmo es lo mismo que programar, y qué es AI.

Además, con las preguntas relativas al uso del conjunto de entrenamiento y su correlación a una predicción, dada la predominancia de respuestas acertadas permite establecer concordancia con la proposición #2.

- En el cuestionario final —posterior al CI con ExtBi— se produce nuevamente un calce con la proposición #2, con un tercio de respuestas acertadas a la pregunta sobre la necesidad de la etapa de entrenamiento. Además, otro patrón que contribuye a esta proposición, fue al consultarles si la clasificación con Bayes ingenuo está restringida a dos categorías, donde la mayoría responde correctamente que no.
- Uno de los principales calce de patrones ocurrió con la proposición #4, cuando el 87 % de los y las participantes acertaron de forma precisa, al solicitarles elegir la secuencia de bloques para implementar un clasificador de texto con ExtBi. De igual modo, al preguntarles sobre cómo describirían las fases que componen el algoritmo Bayes ingenuo, un 25 % de los y las participantes respondieron de manera excelente.
- Luego, se produjo calce con la proposición #6 al comparar ML4Kids o ExtBi, y observando preferencia de las y los participantes por la segunda herramienta.

- En base a preguntas directamente relacionadas con SL, por ejemplo al presentar dos posibles estrategias de entrenamiento —una sesgada y la otra no—, se aprecia a la mayoría de las y los participantes escogiendo la alternativa correcta; calzando así con la proposición #2. Hecho que se reafirma al preguntarles sobre la característica de la predicción con SL, donde la mayoría destaca la utilidad para la toma de decisiones dado un valor que no conoce o distinto a los del entrenamiento.
- Finalmente, se produjo una coincidencia con la proposición #3, ya que la mayoría seleccionó correctamente la afirmación referente a la ventaja de usar datos para entrenar un programan versus definir las reglas directamente en el código.
- Patrones en Programas Scratch:
 - Considerando la exposición de algunos participantes el primer y segundo día, principalmente clasificador sin AI, es posible mencionar el calce con la proposición #3.
 - Respecto a los programas con ML4Kids, y dada las características de la herramienta en general, se puede apreciar un calce con la proposición #2.
 - Luego, los programas implementados usando ExtBi se aprecia una correcta implementación del algoritmo Bayes ingenuo. Así, los patrones confirmados prácticamente concuerda con las siete proposiciones —es decir la totalidad— aunque claramente en mayor o menor grado, en una que otra proposición.
- Dado este análisis, *pattern matching*, evidenciando concordancia de los patrones predichos y los empíricos, se aprecia la fortaleza interna del caso de estudio.

Capítulo 6

Discusión

A continuación se presentan los principales aprendizajes e implicancias —teóricas y prácticas— según los resultados dada la realización del caso de estudio. Con esto, es posible contrastar los resultados con el estado del arte, además de indicar limitaciones en la solución, y luego sugerir próximos pasos. Al final del capítulo se presentan respuestas a las preguntas de investigación planteadas en este trabajo de tesis.

6.1. Aprendizajes

Con la descripción de los resultados, descrita en el Capítulo 5, se desarrolló un análisis contrastando la evidencia previa y posterior a la realización del caso de estudio. Principalmente, se identificaron aprendizajes sobre el comportamiento de los y las participantes frente a la solución. Con esto, es posible evaluar el material expositivo y práctico, más la utilización de la ExtBi.

6.1.1. Sobre Taller

Dado el contexto en el cual se desarrolló el taller, mediante modalidad online, una de las primeras dificultades en las y los participantes fue el uso de la herramienta para videollamadas Zoom. Considerar que en sus actividades escolares —dada la pandemia— no todos los niños y niñas usaron Zoom, sino que algunos/as tenían mayor familiaridad con Google Meets y Classroom. Además, al emplear *Breakout Rooms* para reunir a cada grupo, hubo inconvenientes para mantener los subconjuntos de participantes. Por ejemplo, cuando alguno/a se reintegraba luego de haberse desconectado/a de la sesión principal. Además de esto, en el taller se presentaron dificultades por parte de algunos niños y niñas al usar Google Drive y Gmail. Y, a modo general también se apreció dificultades en algunos al utilizar las combinaciones de teclas para copiar y pegar texto, en tablets y computadores personales.

Sesión 1

Con el cuestionario inicial se pudo conocer que la mayoría de los participantes —19 de 28— no tenía experiencia previa en programación o con Scratch. También, se apreció que solo 10

participantes tenían noción sobre la definición de algoritmo. Luego, cuando se preguntó a qué asociaban AI, 19 participantes la relacionan con robots o artefactos tecnológicos (como apps o servicios), mientras que 6 respondieron que es la imitación de la inteligencia o comportamiento humano. Respecto al concepto de SL, la mayoría no había escuchado sobre este. En relación al concepto de clasificación, se puede decir que aplicaciones o ejemplos prácticos son cercanos a ellos y ellas, mas al definir este concepto la mayoría fue superficial ya que 21 lo definen solo como un ordenamiento o una ponderación. Además, 26 participantes declararon no haber oído de Bayes ingenuo. Así, y en síntesis, es posible mencionar que las y los participantes no tenían noción del algoritmo Bayes ingenuo, tampoco conocimientos sobre SL, ni tenían una definición de clasificación sino una aproximación a ejemplos cotidianos como separar o ponderar.

Ya con la primera exposición de los tópicos, en concreto con la introducción a AI, se aprecia comprensión en la definición de algoritmo y su aplicación en distintos escenarios. Luego, al especificar en algoritmos de AI y realizar ejercicios mediante escenarios frecuentes, las y los participantes evidenciaron adquirir nociones sobre el propósito de un conjunto de entrenamiento y su relación con una determinada predicción. Los inconvenientes con las plataformas Zoom, Gmail y Drive mencionados al inicio de esta sección, claramente fueron más notorios el primer día. Sin embargo —y de todos modos— fue posible finalizar la exposición para introducir Scratch, mediante subconjuntos de participantes en 5 de 6 grupos. Con esto, es posible señalar que el diseño de la primera sesión resulta adecuado para presentar la definición de AI y las características de los algoritmos en el área.

Sesión 2

En la segunda sesión, cuando los participantes intentaron hacer un clasificador de texto basado en reglas, algunos pudieron percatarse de las dificultades al considerar todos los escenarios posibles. Luego, en las sesiones grupales para practicar el método de SL mediante ML4Kids, los principales inconvenientes tuvieron relación con la usabilidad de esta herramienta. Por ejemplo, el *layout* de los bloques Scratch son en inglés a pesar de configurar el sitio en español; también textos del sitio tuvieron una inadecuada traducción (e.g. “Rasguño 3” en vez de “Scratch 3”); y además algunos/as participantes tuvieron incompatibilidad del sitio con navegadores (e.g, Edge) y dispositivos móviles (mal ajuste responsivo). Es posible indicar que, a pesar de las dificultades con ML4Kids, en la sesión las y los participantes pudieron ejercitar por primera vez el método de SL para clasificar texto, basándose así en datos y no en reglas explícitas en el código.

En el CI con ML4Kids para construir un clasificador de texto sobre noticias, una complicación presente fue al definir el conjunto de entrenamiento. Algunos ingresaron el listado completo del *dataset* de referencia como un solo ejemplo o ítem. Dado esto, es posible indicar que a esta altura del taller hubo confusión sobre la unidad de ejemplo, la necesidad de utilizar varias para entrenar, y cómo lograrlo con esta herramienta. Por otro lado, un grupo definió su propio conjunto de entrenamiento y al probar la confianza del modelo al clasificar, utilizaron ejemplos que ya habían sido utilizado en el entrenamiento. Esto sugiere que —a ese momento en el taller— hubo confusión sobre el propósito de un clasificador, es decir predecir la clase a valores que no conoce previamente.

Luego, al implementar el clasificador con los bloques ML4kids, hubo confusión sobre el

funcionamiento del bloque `recognise`, en particular lo que retorna este y el parámetro que requiere. Uno de los errores visibles de ML4Kids es que al cambiar el idioma del entorno Scratch se agregan —nuevamente— el conjunto de bloques para hacer uso del modelo, duplicando innecesariamente la cantidad de piezas de código. Finalmente, y a pesar de las complicaciones, se pudo apreciar en esta actividad que efectivamente se produce colaboración entre pares y finalizaron el clasificador en 5 de los 6 grupos. Si bien, algunos participantes pudieron aplicar el método SL con ML4Kids, es posible señalar que hubo dudas en algunos participantes sobre la metodología en SL, su aplicación con ML4kids, e interrogantes sobre la relación de este método con las etapas de un algoritmo en IA (entrenar, aprender y predecir).

En el cuestionario *verbal probing* post CI con ML4Kids, en la pregunta “¿Cómo describirías el concepto de clasificación a algún compañero/a del colegio?”, no apareció como respuesta lo referente a ponderación o calificaciones presentes en el cuestionario inicial. Ahora lo definen mayoritariamente como la separación o agrupación de objetos y utilizando palabras como categoría, clases y etiquetas, progresando en abstraer sus definiciones. Luego en preguntas sobre SL, al consultar cómo trabaja internamente el bloque `recognise`, hubo 7 participantes que indicaron mediante una comparación con el conjunto de entrenamiento, mientras que dos niños y una niña fueron un poco más precisos indicando que clasifica en base a la categoría donde hubo mayor similitud. Después, al preguntar cómo se podría mejorar el clasificador ML4Kids al agregar nuevos ejemplos, la mayoría (12 participantes) mencionan actualizar solo el conjunto de entrenamiento, faltando indicar las etapas restantes definir el modelo y bloques. Dado esto, es posible indicar que posterior al CI algunos/as participantes manifestaron nociones sobre SL, mientras que otros/as mantienen dudas sobre el método; y progreso en la definición sobre clasificación.

También, una opinión recurrente en los y las participantes fue que realizar entrenamiento es tedioso. Esto indica que, realizar esta etapa de SL en ML4Kids es poco usable para niños y niñas. Finalmente, mencionar que un participante describió en el cuestionario de forma correcta la construcción del clasificador y los bloques necesarios, lo que sugiere que hubo comprensión a nivel teórico y práctico de SL.

Respecto a la revisión de los programas durante el CI, de los 16 archivos `.sb3` cargados a Drive, 6 no se pudieron ejecutar en ML4Kids. A pesar de rehacer las etapas respectivas para cargar el archivo fuente en el entorno, la plataforma indicó error porque el proyecto ya no está disponible o fue eliminado. Con esto, es posible indicar que la plataforma no considera todos los escenarios posibles para volver a cargar un programa en esta. De todos modos, de los programas revisados, 2 fallaron al no asociar el bloque `respuesta` como parámetro del bloque `recognise`. Así, es posible indicar que estas/os participantes no tuvieron claridad en el funcionamiento del bloque `recognise` para predecir la categoría, dado un nuevo valor contenido en el bloque `respuesta`.

Sesión 3

En la tercera sesión, luego de haber presentado todos los conceptos matemáticos considerados en el taller y el cuestionario con Kahoot, se presentó un caso introductorio para definir el algoritmo Bayes ingenuo. Con esto, también hubo un preámbulo sobre cada bloque en ExtBi, empleado en un clasificador de texto. Las primeras impresiones de los participantes sobre el algoritmo implementado con ExtBi, fueron de asombro y comentarios como “*es muy*

largo”. Dada la limitada práctica en programación, una posible interpretación de sus opiniones sobre el clasificador puede ser que se refieren a la longitud del texto (*label*) en los bloques, o la cantidad de bloques utilizados. Al despedir el día, se aprecia agotamiento en los y las participante. Así, es posible indicar que aquellos comentarios sobre el “largo” del programa estén condicionados por el cansancio, luego de esta sesión destinada —principalmente— a abordar conceptos matemáticos que además se dictan en niveles superiores y universitarios.

Respecto a los aprendizajes al realizar Kahoot, una de las complicaciones fue con el modo de juego evidenciado principalmente en la primera pregunta. Hubo confusión con las alternativas por no poder leer directamente las preguntas en su botonera, en particular dificultad o desconocimiento de cómo seleccionar la alternativa. Por ende, es posible declarar que algunas/os no pudieron o supieron cómo responder, pero quizás sí tenían noción de la respuesta correcta. Una vez explicada nuevamente la modalidad logrando comprensión en los y las participantes, al preguntar si es verdadero o falsa la afirmación de que un algoritmo y programar es lo mismo, 18 participantes de forma correcta indicaron que es falso, pudiendo decir que durante el taller logran captar la diferencia práctica entre estas dos actividades.

Luego, al preguntar sobre las características de algoritmos en AI, abordado previamente en el cuestionario inicial preguntando a qué le suena AI, esta vez hubo 8 participantes que escogieron la alternativa correcta. Aquí, se evidencia avance en comparación con el cuestionario del primer día, y cabe destacar que en Kahoot las respuestas disponibles fueron similares aumentando más la complejidad para seleccionar la correcta. Respecto al proceso de entrenamiento o elección de *datasets* —dado un escenario planteado— se hicieron 4 preguntas, y en 3 de estas la mayoría escogió la alternativa correcta. En la interrogante donde no se mantuvo esta tendencia, al preguntar sobre cómo se entrena un detector de correo no deseado, se podría deber a que probablemente la mayoría no da un uso frecuente a su correo (como también se apreció en las complicaciones de uso de Google Drive al inicio del taller). Y, en la pregunta “*P7 Clasificación se refiere a*” hubo un incremento de respuestas correctas —15 participantes— considerando una pregunta similar en el cuestionario post CI-ML4kids, pudiendo decir que niñas y niños logran reconocer la utilidad de clasificación, sin embargo describir el concepto con sus palabras es más complejo para ellas y ellos.

Sesión 4

Al realizar la comparación de los distintos tipos de clasificadores vistos en el taller —al comienzo de la última sesión— y además de nuevamente exponer sobre ExtBi y el algoritmo Bayes ingenuo, a diferencia de la primera explicación sobre estos conceptos el día anterior. Esta vez hubo mayor cantidad de participantes que asientan comprender sobre la implementación del clasificador de texto con ExtBi. Luego en el CI, algunos participantes percatan también algunas ventajas de aplicar SL con bloques ExtBi versus ML4Kids, como por ejemplo tener —o acceder— a todo el clasificador en una vista, no necesitando de entrenar o configurar el modelo en un sitio para luego programar en Scratch (otro sitio o página externa). Incluso, en un grupo pudieron apreciar esta ventaja, ya que tuvieron que cambiar el nombre al clasificador solo actualizando un parámetro de un bloque, no necesitando hacer tareas adicionales en otro sitio. Por otro lado, igualmente compararon la fase de entrenamiento entre ML4Kids y ExtBi, notando que en la primera es necesario indicar cada ejemplo de forma individual, mientras que en la segunda es posible importar un archivo. Además, lo

consideran más rápido, pudiendo deberse a que ExtBi no necesita de servicios externos evitando latencia de comunicación. Con todo esto, es posible indicar que las y los participantes muestran preferencia por ExtBi sobre ML4Kids al programar el clasificador de texto con el método SL.

Luego, al preguntar sobre la posibilidad de incorporar una tercera categoría, hubo participantes que indican las tareas de crear una lista de ejemplos, entrenar la categoría y luego calcular Bayes para esta. Lo cual implica, comprensión de aquellas/os participantes, sobre el orden de pasos en Bayes ingenuo y que la clasificación no está restringida solo entre dos categorías. Continuando con el manejo de los bloques, algunos participantes identificaron cuáles utilizar pero no los llaman por su nombre sino como “el cuadrado”, “el redondo”, etc. En general, esto podría deberse a la poca práctica con Scratch. Por otro lado, algunos niños y niñas presentaron dudas en los bloques **entrenar** y **bayes** al requerir —como parámetro— el nombre de la categoría, indicando distintos *ids* para referirse a la misma. Esto puede sugerir que, el diseño del bloque ofrece confusión al declarar o requerir identificadores de tal forma; o a este grupo de niñas y niños falta reforzar sus habilidades en programas usando estos identificadores.

También, en el CI del último día se observó mayor fluidez en los participantes al verbalizar lo que van realizando y aumentó la resolución de problemas directamente entre pares. Destaca la sinergia en los grupos, ya que cuando un participante no sabía cómo realizar una tarea, una compañera/o estaba dispuesta a guiarle. También, participantes atrasados en las tareas para construir su clasificador, lograron realizar lo pendiente de forma autónoma (los tutores no le indican cómo hacerlo). Finalmente, no se podría mencionar que hubo —durante el CI— diferencia de habilidades entre niños y niñas; en algunos grupos una niña evidencia mayor habilidad mientras, que en otro grupo es un niño.

En relación al cuestionario post CI, al preguntar sobre el propósito del conjunto de entrenamiento, una respuesta a destacar fue “*porque así establece una probabilidad independiente*”, evidenciando que adquirió —también— nociones sobre los tópicos matemáticos, logrando asociarlo a la fase de entrenamiento y el algoritmo Bayes ingenuo. Luego, 16 participantes reconocen que es posible hacer clasificación no solo entre dos categorías, confirmando así lo observado durante el CI. Por otro lado, al preguntar sobre la secuencia de bloques que ellos escogerían para implementar el clasificador de texto, 22 participantes seleccionan correctamente el orden de bloques. Dada esta evidencia, se permite concluir que las y los participantes pueden indicar correctamente el orden de pasos, para construir un clasificador de texto con los bloques ExtBi. Además, al solicitar que describieran las fases del algoritmo Bayes ingenuo, 6 participantes lograron explicarlo correctamente, aunque también 8 participantes que en la pregunta previa escogieron de manera correcta la secuencia de bloque, esta vez no lo describieron. Con esto último, se podría indicar que a pesar de poseer una correcta idea de la secuencia con bloques ExtBi, se requiere reforzar el propósito y en qué consiste cada uno, junto a cómo se asocia las secciones del código con las etapas de un algoritmo en AI (entrenamiento, definición del modelo y predicción).

Dado lo anterior, tal progreso en la definición de Bayes ingenuo se ajusta con lo que entienden por algoritmo en general, ya que, al solicitar que describieran —algoritmo— hubo 10 participantes con respuestas precisas. También, al comparar con un cuestionario previo

—qué entienden por algoritmo— es posible indicar que se mantuvo la cantidad aciertos. Sin embargo el resto, 14 participantes, esta vez tuvo mayor precisión en su respuesta.

Continuando con las preguntas sobre clasificación con Bayes ingenuo, al consultar sobre cómo agregarían más ejemplos para mejorar el clasificador, 8 participantes responden correctamente ampliando la lista y no haciendo referencia a crear otra estructura o ir a algún sitio. Esto sugiere también comprensión sobre relevancia del conjunto de entrenamiento en SL y cómo concretarlo con la ExtBi. Además hubo una respuesta precisa al indicar los pasos del algoritmo Bayes ingenuo, evidenciando una noción acertada de su definición.

Luego, al preguntarles sobre cuál alternativa —ExtBi o ML4Kids— prefieren para construir un clasificador, 14 participantes señalan ExtBi dada su rapidez, sencillez, por ser más específico, entendible y valorando que esté todo en Scratch. Así, se confirma lo observado en el CI sobre la predilección por ExtBi. También, algunos destacaron particularmente la ventaja durante la fase de entrenamiento y “*ser menos tediosa*”. Aunque una respuesta que indica preferir ML4kids es porque en ExtBi es “*todo manual*”. Sobre esta respuesta, puede deberse a que el clasificador con ExtBi requiere realizar el entrenamiento y pasos del algoritmo mediante bloques, mientras que en ML4Kids se logra directamente en el sitio —distinto a Scratch— que dispone.

Finalmente, hubo preguntas que permitieron reflexionar sobre la comprensión de los y las participantes, respecto a características del método SL. Por ejemplo, al preguntar —literalmente— sobre una ventaja de SL, la mayoría escogió la alternativa referente a mejorar el desempeño usando ejemplos, pudiendo así declarar que, tal grupo de participantes logró identificar un atributo característico de SL. Por otro lado, al solicitar elegir la alternativa para alcanzar el mejor sistema entre dos estrategias de entrenamiento: (1) con mayor cantidad de ejemplos para una categoría que la otra, y (2) con la misma cantidad en ambas categorías. Hubo 18 participantes que escogieron la segunda, con lo cual es posible decir que este grupo de participantes logró consciencia de que un entrenamiento uniforme contribuye a una predicción superior. También, es posible decir que la mayoría de las y los participantes identificó como característica de SL, el poder construir un programa entrenándolo con datos versus especificando explícitamente sus reglas, al preguntar sobre una afirmación de SL entre cuatro alternativas. Para cerrar, otra utilidad que identificó la mayoría de los y las participantes sobre SL, es que permite reconocer nuevos valores según el conjunto de entrenamiento usado. Esto, al preguntar sobre el proceso de SL para hacer un sistema de reconocimiento de texto, solicitando indicar la alternativa correcta entre cuatro opciones sutilmente diferente, destacándose así aún más el logro.

En relación a los programas construidos con ExtBi, hubo 22 archivos fuente subidos a la nube por las y los participantes. Dado que ExtBi es una aplicación que no depende de servicios externos, todos los códigos se pudieron volver a ejecutar o revisar, el doble de programas que con ML4Kids. En 18 programas implementaron el algoritmo Bayes ingenuo, mientras que en 4 resultaron incompletos o les faltó algún paso. Aquí, la mayoría de los programas utilizaron las listas correctamente inicializadas en el entrenamiento, pero también hubieron dos casos que al nombrar a estas estructuras fue usando una frase, por ejemplo “*Clasificador_Noticias*”, lo que sugiere reforzar el propósito la lista en el entrenamiento. Otro hecho que se repitió, fue que el *id* de una categoría como parámetro, tuvo un nombre en el bloque `entrenar` y otro —

similar— en el bloque `bayes categoria`. Con esto, se podría confirmar que se debe reforzar la definición de identificadores y su uso como parámetro en los bloques ExtBi. Finalmente, en los programas se aprecia creatividad para mostrar el resultado final, ya sea utilizando una mejora del mensaje o modificando el escenario y disfraz del personaje. Lo interesante es que a pesar de las modificaciones igualmente lograron el clasificador y no introdujeron errores manteniendo la correcta secuencia del algoritmo.

6.2. Implicancias de los Aprendizajes

Dada la síntesis de resultados, indicada en la sección anterior, a continuación se presentan las implicancias teóricas y prácticas que conllevó realizar el caso de estudio.

6.2.1. Teóricas

Para describir lo que implicó el caso de estudio, desde el espacio teórico, se realizó una triangulación con el estado del arte de acuerdo a lo discutido en el Capítulo 2. En concreto, la solución propuesta en este trabajo de tesis se compara con herramientas para facilitar la programación y currículos educativos sobre AI. En síntesis, el taller y la ExtBi se enmarcan con iniciativas educativas para los niveles K-12, presentes en la literatura actual.

Herramientas de programación

La primera alternativa a analizar, que facilita la aplicación de SL y programación a niños y niñas, es ML4kids. Dado que también se utilizó en el caso de estudio, el contraste preponderante entre ExtBi y ML4Kids es que con la primera sí es posible implementar algún algoritmo de clasificación, mientras que en la segunda el resultado o predicción es la respuesta del bloque `recognise`. En ML4Kids el algoritmo detrás de la clasificación es caja negra. La razón de estas diferencias se debe al diseño de cada solución. Por una parte, ML4Kids utiliza un sitio para definir el entrenamiento; y por otra, internamente usa la API Watson de IBM para configurar un modelo de ML, y finalmente utilizarlo o realizar clasificación para un nuevo valor —mediante el uso del bloque `recognise`—. En cambio, ExtBi no depende de ninguna plataforma externa al entorno Scratch. Con esto, el entrenamiento y configuración del modelo con ExtBi, se estructura mediante bloques y se concreta al ejecutar el programa. Así, el conjunto de entrenamiento y pasos que definen el modelo de del algoritmo Bayes ingenuo, son visibles en el código en todo momento. Esto último, implica finalmente que ExtBi sea más rápido en la ejecución del programa —respecto a ML4Kids— ya que no existe latencia de comunicación entre plataformas.

Análogo a ML4Kids, en finalidad y modo de uso, se encuentra LearningML [19]. Esta plataforma también intenta apoyar la enseñanza de ML a niños y niñas, mediante la creación de proyectos prácticos con Scratch permitiendo clasificar texto o imágenes. Al igual que ML4Kids, es una plataforma web con dos componentes principales, un sitio para editar el conjunto de entrenamiento y configurar el modelo; y por otro lado una versión web de Scratch 3, adaptada para utilizar el modelo ML —definido previamente— mediante bloques. A pesar de esto, igualmente hay diferencias entre ambas herramientas. La primera, es que LearningML no depende de otros servicios externos como sí lo hace ML4Kids con IBM Watson; y la segunda es que con LearningML es posible compartir con otros usuarios de la plataforma, o

descargar en el computador, un determinado proyecto. De hecho, los autores de LearningML destacan que esta última característica de su solución, quizás sea la más interesante permitiendo la creación de un repositorio a gran escala de la actividad. Finalmente, es posible implicar que dadas las similitudes entre ML4Kids y LearningML, y según preferencia de la mayoría de las y los participantes de ExtBi sobre ML4Kids, podría considerarse que niñas y niños chilenos también preferirían ExtBi sobre LearningML para construir un clasificador de texto.

Aquí, conviene enfatizar que LearningML al compartir —o descargar— no considera el código Scratch (archivo `.sb3`) posiblemente asociado a ese proyecto, sino que solo exporta el conjunto de entrenamiento utilizado para definir el modelo. Respecto a esto último, y dado que ExtBi tiene como características que el conjunto de entrenamiento y configuración del modelo se definen usando bloques, al compartir código implícitamente también se comparte los ejemplos utilizados para entrenar y la secuencia de bloques que definen el modelo según Bayes ingenuo. En otras palabras, con ExtBi el archivo `.sb3` contiene el modelo y entrenamiento utilizado, más la animación o video —código— en Scratch. Finalmente, los autores de LearningML justifican no ahondar en algoritmos relacionados con SL, dado que consideran los conocimientos previos no son abordables por niñas y niños. Sin embargo, los resultados obtenidos por el caso de estudio propuesto en este trabajo de tesis permiten declarar lo opuesto, a saber, que escolares pueden adquirir nociones de un algoritmo usado en ML como Bayes ingenuo.

SnAIp [11] es otra herramienta, también mencionada en el Capítulo 2, que permite introducirse a ML. Su principal característica es que intenta abordar el método de RL para ser comprensible por niños y niñas y utiliza el entorno de programación Snap!, lenguaje basado en bloques similar a Scratch. A pesar de las diferencias con ExtBi, tanto en el fin como en los lenguajes en cuales se basan, igualmente hay una similitud de paradigma en cada solución. En concreto, ambas buscan que niños y niñas puedan adquirir nociones sobre un algoritmo de ML. Por un lado, SnAIp posibilita la opción de comprender la secuencia de pasos o definición del algoritmo Q-Learning, aplicando el método RL; mientras que, y como se ha mencionado previamente, ExtBi posibilita la comprensión del algoritmo Bayes ingenuo, aplicando el método SL. Una implicancia de esta similitud, es que se podría considerar una solución que incorpore ambas soluciones; así por ejemplo, la ExtBi sea implementada para Snap!, y dadas las características de este lenguaje, sea posible seguir profundizando en la implementación interna de cada bloque. O tal vez, también es posible implementar el algoritmo Q-Learning en Scratch, tomando como referencia el de SnAIp.

Otro trabajo similar, en cuanto diseño de investigación, es el presentado por Dasgupta y Hill [3] proponiendo una extensión para Scratch —Scratch Community Blocks— con el fin de que niñas y niños puedan realizar actividades de DS con datos desde la comunidad de Scratch. Si bien, la extensión al entorno fue diseñada para permitir a niños y niñas hacer tareas visualización y aprendizaje, el propósito principal fue posibilitar a que las y los participantes respondan a propias preguntas usando datos reales. Así como Logo fue diseñado para dar foco al aprendizaje de las y los niños, más que solo intentar convertirlos en programadores, Scratch Community Blocks posibilita plantear y responder preguntas sobre la comunidad Scratch más que hacer visualización. En esa línea, se puede decir que la ExtBi facilita la adquisición de ideas relacionadas al algoritmo Bayes ingenuo, más que solo permitirles programar un

clasificador.

Programas educativos

Por otro lado, respecto a currículos educativos sobre AI para los niveles K-12, el trabajo propuesto por Lee et. al. [14] considera actividades prácticas mediante computador y también desconectadas o *unplugged*. Un ejemplo de este último tipo, es un ejercicio de clasificación con algoritmo árbol de decisión, realizable con lápiz y papel sin recurrir a código. No obstante, otra actividad consideró la codificación en Scratch de un clasificador de imágenes, utilizando un modelo de ML generado con Teachable Machine de Google. Cabe destacar que, la herramienta Teachable Machine permite clasificar imágenes (incluyendo posturas) y audio, mas no texto.

Lee et. al. en su propuesta, para hacer uso de dicho modelo y clasificar, utilizan un entorno Scratch extendido disponiendo del bloque “use model”, el cual requiere como parámetro la URL del modelo Teachable Machine configurado. Una vez asociado el modelo, hay otro bloque que permite clasificar nuevos ejemplos, pero es caja negra igual que el bloque `recognise` de ML4Kids. En consecuencia, se desprende que esta propuesta práctica para clasificación con Scratch no permite conocer la secuencia del algoritmo que determina el resultado de la predicción. Finalmente, este programa educativo también aborda conceptos de Redes Neuronales, mediante la técnica Generative Adversarial Networks (GANs) para la creación videos o imágenes. Esto se aborda mediante el uso herramientas web que buscan presentar la utilidad GANs; sin embargo, no ahondan en el funcionamiento interno para lograr el resultado de crear una foto, texto o musica. Así, es posible que declarar que el programa de Lee et. al. no aborda el algoritmo involucrado en GANs, sino que se busca presentar sus posibles usos.

Por otro lado, respecto al programa educativo de la Fundación Telefónica el concepto de clasificación se presenta como caja negra. Esta opción educativa, concluye con la construcción de un clasificador para imágenes con ML4Kids, mediante el uso del bloque `recognise`. Dado esto, se podría decir que tal programa considera hasta la mitad de la sesión 2, según la solución presentada en esta tesis. Finalmente, a la fecha no hay evidencia de que hayan realizado un taller, como el caso de estudio ejecutado en esta tesis, a saber, con una sistematización de actividades donde tutores acompañan a niñas y niños en el proceso de aprendizaje y práctica de los conceptos entregados.

En otro ámbito, dado el diseño metodológico del caso de uso, se puede enmarcar esta solución en una Trayectoria de Aprendizaje (LT, por sus siglas en inglés) basada en piezas de conocimientos, según Rich et. al. [18]. A saber, durante el taller se procuró justificar la conexión de los tópicos presentados, por ejemplo al comenzar cada sesión el expositor resumió lo realizado en las sesiones anteriores y explicando el motivo de los próximos pasos. También, esto implica que es posible formalizar y acoplar el proceso de enseñanza en el taller, con las LTs para secuencia y condicional, propuestas por Rich et. al.

Iniciativa de aprendizajes

Respecto a marcos de aprendizaje, es posible asociar la solución presentada en esta tesis con las “*Grandes Ideas en AI*” que niños y niñas deberían conocer, definidas por el grupo AI4K12 [26]. En concreto, con el taller para concientizarse sobre Bayes ingenuo usando la ExtBi, se abordan dos de las cinco ideas planteadas. Por un lado, la idea de representación

y razonamiento —*Gran Idea 2*—, se consideró al permitir que las y los participantes sepan cómo agentes o programas usando datos construyen su representación, y las limitaciones de este modelo en su razonamiento. Y también, la idea de aprendizaje —*Gran Idea 3*— se abordó al posibilitar a las y los participantes conocer que ML es un tipo de inferencia que detecta patrones en los datos. Dado que, la ExtBi y el taller son alternativas en sintonía con los lineamientos de la organización AI4K12, es posible considerarlas como opción válida para educar en materia de IA para los niveles K-12.

6.2.2. Prácticas

Antes de ahondar en posibles próximos pasos con la solución, se describen implicancias y limitaciones en sus componentes, principalmente sobre la ExtBi y el proceso de generación del material educativo. En esta línea, luego se hace referencia también a posibles amenazas a la validez del caso de estudio. Cabe precisar que trabajos futuros son indicados en el siguiente capítulo.

Sobre la ExtBi

Respecto a la ExtBi —y lo que conlleva su empleo— es que el resultado de la clasificación no es resultado de un bloque, a diferencia de soluciones como ML4Kids o LearningML. En cambio, programar un clasificador con ExtBi requiere implementar Bayes ingenuo, permitiendo así tener mayor visibilidad de este algoritmo y cómo se logra la clasificación de un nuevo elemento. De este modo, otra característica posible de declarar es que permite aplicar el método SL solo con bloques en Scratch, no siendo necesario el uso de otro sitio web. También, los bloques en ExtBi no hacen uso de alguna API para obtener algún servicio web. Sino que toda la lógica de negocio está contenida en los bloques en la extensión, no existiendo latencia de comunicación y mayor rapidez al ejecutar el programa. Para esto, internamente en los bloques se abstraen procesos para entrenar el modelo, calcular la probabilidad de que un nuevo valor pertenezca a una categoría, y finalmente seleccionar el *id* de la categoría con mayor probabilidad. Con esto, se mantiene igualmente un grado de ocultar procesamiento con los nuevos bloques.

De todas formas, es posible continuar profundizando en los conceptos que considera el algoritmo y que por ahora se delega al funcionamiento de los nuevos bloques. Sin embargo, requiere ejercitar aún más tópicos como bucles y probabilidad condicional. Por otro lado, uno de los requisitos que contribuyen a una mejor comprensión de los bloques y el algoritmo, son tener nociones sobre listas y definir de manera precisa su identificador. Y precisamente, respecto al empleo de listas, las y los participantes del taller estimaron que el entrenamiento de las categorías en el clasificador, fuese posible mediante esta estructura al importar un archivo `.txt` y evitando así que indicarlos individualmente.

Respecto a limitaciones de los bloques en la solución, se señala que están solamente disponible en idioma español. También, se menciona que —por ahora— es posible solo un clasificador por programa. Además, cuando el porcentaje de predicción tiende a cero, los decimales se presentan con notación exponencial. Finalmente, si se implementara un algoritmo como árbol de decisión, para que niñas y niños adquieran nociones sobre este, dada las limitaciones de Scratch —a saber, no es un lenguaje de primera clase— Snap! podría resultar adecuado.

Sobre el Material Educativo

Respecto a la componente sobre el material teórico y práctico, presentaciones y guías, además de ser útil para introducir conceptos sobre clasificación y algoritmo Bayes ingenuo, es posible incorporarlo en cursos de Pensamiento Computacional. Por ejemplo, las sesiones de Scratch y ML4Kids diseñadas, resultan apropiadas para introducir al entorno y programación con bloques básicos. En esta línea, el material matemático que se construyó podría ser parte de una solución para introducir al Pensamiento Probabilístico, por ejemplo utilizando la herramienta *Let's Chance* de Dariwhal [4].

En relación a las limitaciones del material educativo, es posible indicar que está destinado al rango etario 10 a 12 años, a quienes está dirigida esta propuesta. Dados los conceptos computacionales y matemáticos que se abordan, se mantiene como público objetivo a estudiantes de 10 años de edad, o que hayan cursado el quinto nivel escolar. Por otro lado, en relación a la planificación, se considera mantener la cantidad de horas pero distribuirla en 5 días. Respecto a la modalidad del taller, se mantiene como requisito disponer de un dispositivo —computador personal o móvil— con la suficiente capacidad de poder utilizar herramientas simultáneamente como Scratch, Zoom y el navegador.

6.2.3. Sobre Amenazas a la Validez

Tal como lo indica Shull et. al. en el libro *Guide to Advanced Empirical Software Engineering* [21], es necesario discutir las amenazas que puedan tener un impacto en la validez de los resultados. Por esto, en los siguientes apartados se presentan las acciones realizadas con tal de brindar validez de constructo, interna, externa y ecológica a las conclusiones del caso de estudio; además de mencionar tratamientos o mitigaciones a posibles amenazas.

Constructo. Con tal de abordar posibles amenazas al constructo, u operacionalización de las medidas del caso de estudio, durante el diseño se consideraron múltiples fuentes de evidencia, y así medir el desempeño de las y los participantes frente a la solución propuesta. También, cabe precisar que los distintos instrumentos y métodos para recolectar evidencia, fueron validados por expertos y expertas en dominios como ciencias de la computación o matemáticas. Finalmente, se realizó una triangulación de los resultados obtenidos, entre las distintas fuentes de datos producto de aplicar los instrumentos o métodos. Por todo esto, es posible indicar que las amenazas de constructo fueron mitigadas.

Interna. Luego, para afrontar amenazas a la validez interna de la solución, el diseño metodológico del taller consideró que la realización de las actividades educativas, tuviesen relación con el comportamiento de las y los participantes frente a la solución. De esta forma, el diseño del estudio consideró el modo de ejecución del caso de estudio, protocolo, recolección de evidencia y la elección de la técnica analítica *pattern matching*, por mencionar algunas medidas. Sin embargo, una posible amenaza interna pueden ser los cuestionarios, ya que si bien fueron diseñados ad-hoc —y calibrados luego la primera ejecución del taller—, resta realizar una validación psicométrica de estos instrumentos con usuarios finales.

Externa. Posibles amenazas externas, tienen relación con inconvenientes para generalizar los resultados —del caso de estudio realizado— a otros contextos. Para esto, es necesario

reconocer que los resultados del caso de estudio son válidos en el grupo objetivo, escolares chilenos entre 10 a 12 años de edad. Esto, es consistente con la elección metodológica de ejecutar un caso de estudio, según se indica en [29], [22]. Sin embargo, es posible replicar el caso de estudio con otro grupo, escolares de otros países, o mediante otra modalidad —por ejemplo presencial.

Ecológica. Respecto a amenazas ecológicas, podrían producirse dado el medio —remoto *online*— en el cual se desarrolló el caso de estudio. Dado lo anterior, si se realiza el taller de forma presencial pueden haber diferencias de resultados. Incluso, si se mantiene remoto pero *offline*, podría presentarse disparidad en las conclusiones del caso de estudio.

Próximos Pasos y Mejoras

Producto de las implicancias teóricas y prácticas mencionadas, es posible considerar la solución propuesta en esta tesis como una opción de programa educativo para niñas y niños según STEM. Por otro lado, dadas las características metodológicas del taller, además de continuar como un campamento educativo en vacaciones, podría adaptarse para complementar jornadas escolares actuales, por ejemplo en clases de tecnología o ciencia. Si bien, el taller está diseñado para ser en modalidad online remota, resulta viable hacerlo de forma presencial. Es válido analizar la opción de replicar el caso de uso con participantes de otras ciudades del país, e incluso en otros países de la región, inicialmente de forma *online* y así confirmar —o no— los resultados obtenidos en el caso de estudio.

Al intentar concientizar sobre conceptos de SL en Scratch con Bayes ingenuo, es válido pensar en la opción de abordar este método con otros algoritmos como Redes Neuronales (NN, por sus siglas en inglés). Por otro lado, también es posible reflexionar sobre la opción de considerar otros métodos en ML como Aprendizaje No Supervisado (UL, por sus siglas en inglés) o RL para ser introducido mediante este lenguaje. Sin embargo, como se mencionó, se debe considerar que Scratch no es lenguaje de primera clase, por ende abstraer determinados módulos para un algoritmo como NN o Clustering, quizás no sea posible con Scratch y en ese caso lo recomendable sea a través Snap!. Y respecto al material educativo, destinado para que niños y niñas capten las ideas relacionadas a estos nuevos tópicos, se recomienda seguir la planificación usada en este trabajo de tesis. A saber, investigar y determinar que conceptos en particular son posibles de abstraer, validar con expertos y/o usuarios finales, y finalmente implementar la extensión para el lenguaje de bloques. Finalmente, dado que los conceptos de AI están fuertemente ligados a conceptos de probabilidad y estadística, se hace imperativo contar con profesionales o investigadores pertenecientes a esa área, que junto a profesionales de CSE, permitan incrementar la robustez del producto.

Por otro lado, es posible continuar refinando la ejecución del caso de estudio, con el propósito de contribuir a que la mayoría de los participantes logre una comprensión del algoritmo Bayes ingenuo. Una de las medidas es, por ejemplo, realizar el taller en 5 días de 3.5 horas. Esto, permitiría distender la exposición de los tópicos posibilitando mayor opción de reflexión a los y las participantes. Contar con un rango horario más amplio, permitiría también incorporar otros ejercicios que permita reforzar el aprendizaje. Por ejemplo, una actividad podría consistir en entregar el código de un clasificador con errores y que los y las participantes lo modifiquen hasta lograr el resultado esperado. También, se podría adoptar

la práctica de realizar ejercicios de forma guiada, donde las/los participantes van dirigiendo al tutor o profesor y este ejecuta lo indicado promoviendo la reflexión. Un ejemplo de esto último fue realizado durante el taller de julio, cuando un grupo de participantes guió al tutor para realizar su clasificador de texto, lo cual tuvo resultados positivos según las observaciones del tutor.

En particular a SL, es posible presentar otras aplicaciones o utilidades del método. Por ejemplo, una posible opción de ejercicio es utilizar párrafos —no solo oraciones— y así revelarles la utilidad de SL para construir un reconocedor de libros e historietas. También, podría considerarse un ejercicio para construir un chatbot, dado el interés de algunos plasmado en sus programas. Para este caso, ML4Kids ofrece una guía de actividad donde un clasificador es entrenado con preguntas, prediciendo entonces la categoría a la cual corresponde una determinada pregunta, e indicando finalmente una respuesta sobre esa categoría. O también, se podría incorporar una actividad para desarrollar un proyecto personal o grupal, con tal que niños y niñas implementen un clasificador según sus intereses; por ejemplo sobre videojuegos, animé u otra temática que les entusiasme y que resulte viable construir un clasificador. En esta línea, se podría considerar una actividad previa sobre manejo de variables para múltiples personajes, con el objetivo posterior de abordar clasificación con algoritmo Bayes ingenuo entre escenarios o personajes de Scratch.

Asimismo, se podría fortalecer la idea del algoritmo Bayes ingenuo y su implementación con la ExtBi. Por ejemplo, es posible consolidar el porqué de cada bloque en el código del clasificador, promoviendo comprender qué hace, cuáles parámetros requiere, mientras se refuerza la relación con la etapa correspondiente del algoritmo AI: entrenar, aprender o predecir. Además, es recomendable fortalecer la necesidad de consistencia en los identificadores en un programa, en particular el nombramiento de *ids* para categorías con los bloques ExtBi. Esto último, dado los errores detectados en algunos programas de las y los participantes, al indicar distinto nombre —id— en el código para referirse a una misma categoría. Luego de haber implementado al menos dos clasificadores distintos, un ejercicio práctico podría tratarse de prescindir del bloque `maxima`, e implementar la lógica de esta función directamente usando bloques condicionales y determinar el *id* de la categoría con mayor probabilidad entre las disponibles. Por otro lado, dado que Scratch —en concreto Javascript— los valores decimales cercanos a cero son presentados con notación exponencial, se podría adaptar la solución para mantener en cualquier caso una notación decimal.

Igualmente, es posible mejorar la ejecución de los métodos e instrumentos y así aumentar el volumen de evidencia. En términos generales, no está de más recordar a las y los participantes que si bien no hay calificaciones, es importante que hagan entrega del material que generan, como programas o responder a cuestionarios. Por otro lado, definir que las y los participantes tengan permiso de editor en la carpeta donde cargan programas, mientras que solo asignarles permiso de lector en la(s) carpeta(s) con el material del taller. Esto último, dado que algunos participantes duplicaron los archivos *dataset* de ejemplos, generando confusión en los otros/as participantes.

Respecto al cuestionario Kahoot, es posible refinarlo incorporando preguntas iniciales de prueba, intentando así asegurar que todos dominan la modalidad de la actividad. Además, dado que Kahoot permite asignar puntaje a cada alternativa, se recomienda establecer el

puntaje mayor a la respuesta correcta y uno menor a las otras alternativas. Esto, con el propósito de mantener la atención en las y los participantes hasta la última pregunta del quiz. Por otro lado, se sugiere que los y las participantes utilicen su dispositivo móvil como botonera, mientras que su computador lo usen para leer las preguntas y alternativas. Esto último, con tal facilitar la lectura y comprensión de la pregunta, y ayudar la forma de responder. En base a esto, igualmente se recomienda incorporar una pregunta para que las y los participantes indiquen el dispositivo (móvil y/o de escritorio) ocupado durante la trivía, y así también considerar este dato en el análisis de evidencia.

En el CI, dado que algunos participantes presentaron perplejidad al mencionarles que la actividad se denomina *constructive interaction*, se sugiere indicar que se hará sencillamente un ejercicio colaborativo. También, a pesar de que se procuró clarificarles a los y las participantes la modalidad del CI (verbalizar lo que van haciendo y compartir pantalla), se podría enfatizar principalmente en el CI con ML4Kids, ya que es primera vez que realizan la actividad en el taller. Asimismo, dado que la ortografía de las y los participantes aún está en desarrollo, en la fase de entrenamiento al construir un clasificador se podría considerar solo ejemplos de textos neutrales, simples en su escritura y sin tildes. Finalmente, para analizar la retención de conocimiento en las y los participantes en el tiempo, es posible realizar un focus group en una fecha posterior a la realización del taller. Esto, con el objetivo de medir la retención de los conceptos luego de 3 meses, sobre las características de algoritmos en AI, respecto a la definición de Bayes ingenuo y también concepciones de SL.

6.3. Revisitando las Preguntas de Investigación

Según los aprendizajes del caso de estudio, y definidas las implicancias teóricas y prácticas, es posible dar respuestas a las preguntas de investigación planteadas en la Sección 1.3. Cabe señalar que el objetivo general de esta tesis, a saber concientizar a niños y niñas chilenos en el algoritmo Bayes ingenuo mediante Scratch, fue definido con tal de intentar responder tales preguntas investigativas.

De este modo, la primera pregunta “¿Cómo abordar el tópico de clasificación con algoritmo Bayes ingenuo, para ser entendible por niñas y niños de 10 a 12 años?” resulta válido contestarla según el diseño del taller propuesto en esta tesis. Es decir, considerando la exposición de los conceptos sobre introducción a AI, introducción a SL y clasificación, teorema de Bayes y finalmente el algoritmo Bayes ingenuo. También, considerando los ejercicios prácticos de cada sesión, más la presentación y empleo de la ExtBi. Respecto a la modalidad, si bien este taller solo ha sido impartido de manera remota *online*, hasta el momento no se han encontrado justificaciones para sostener que resultados similares en las y los participantes, no sean posibles con un taller presencial. Finalmente, si en algún escenario no es primordial obtener evidencia del comportamiento de las y los participantes del taller, y por ejemplo se utiliza como una actividad solo con fines de difusión, es posible prescindir de aplicar los cuestionarios como el inicial o verbal probing.

Respecto a la segunda pregunta, “¿Cómo diseñar experiencias prácticas, a desarrollar por escolares, que faciliten la comprensión del algoritmo Bayes ingenuo y programarlo con la extensión para Scratch?”, es posible responderla según lo realizado en la Espira 1 de la metodología de investigación llevada a cabo en este trabajo de tesis. En concreto, seleccionados los

tópicos presentados en el taller —desde introducción a AI hasta algoritmo Bayes ingenuo— se llevó a cabo una abstracción de tales conceptos, luego se construyeron artefactos —material expositivo, práctico y ExtBi— y posteriormente se validó con expertos. Este proceso, descrito en Subsección 1.6.1, se iteró hasta después de haber hecho el taller piloto en Enero del 2021.

Y en la última pregunta de investigación, “*¿Cuáles son las principales limitaciones de la extensión para Scratch al implementar el algoritmo Bayes ingenuo?*”, es posible indicar que por ahora solo es posible hacer clasificación con Bayes ingenuo para datos de tipo texto sin alguna estructura o tabulado. Además, solo es posible construir un clasificador por programa, o archivo fuente `sb3`. Finalmente, los bloques o el *layout* de estos, están solo en idioma español.

Capítulo 7

Conclusión

Países como China, EEUU, Reino Unido y España, están comenzando a desarrollar iniciativas que permiten educar sobre AI en los niveles K-12. Por ejemplo, en el país asiático ya han incorporado al currículo educacional obligatorio, educación en AI para niveles primarios y secundarios. En esta línea, la comunidad científica de computación en educación —y pensamiento computacional— en los últimos años consideran sesiones en conferencias, destinadas a la investigaciones de educación en AI para niñas y niños. También, desde el ámbito empresarial se han desarrollado herramientas, como ML4Kids de IBM o Teachable Machine de Google, que permiten introducirse en conceptos relacionados a AI.

Dado este escenario y considerando el estado del arte —trabajos publicados en los últimos 5 años— se identificó como problema la escasez de conocimiento científico sobre alternativas educativas que permitan a niñas y niños adquirir nociones sobre clasificación de texto, considerando el algoritmo Bayes ingenuo y Scratch para su codificación. Tal problemática resulta de interés a la comunidad de investigadores, ya que principalmente contribuye a la comprensión de AI desde CS, y además aporta al desarrollo de programas educativos y herramientas útiles para la formación en AI o ML desde edades tempranas.

Por esa razón, en esta tesis se propuso como solución una serie de actividades educativas secuenciales que buscan concientizar a niñas y niños en etapa escolar, sobre AI y en específico acerca del algoritmo Bayes ingenuo al clasificar texto. Así, los componentes en la solución son (1) el material educativo teórico y práctico, (2) una extensión para Scratch —ExtBi— con bloques que posibilitan la implementación del algoritmo Bayes ingenuo, y (3) el taller como unidad de análisis o caso para el estudio del comportamiento de participantes frente a la solución. Este caso de estudio, se realizó a dos grupos —distintos— de escolares de 10 a 12 años, en enero y julio del año 2021.

De este modo, producto de la validación formal de la solución, se aprecian resultados desde la primera sesión del taller. Por ejemplo, a pesar de que la mayoría de las y los participantes no tenía experiencia previa en programación y mucho menos en ML, al finalizar el primer día algunos participantes responden correctamente las preguntas sobre conceptos generales de algoritmos en AI, y lograron construir correctamente sus primeros programas en Scratch. Luego, el segundo día también algunos participantes captaron, las complicaciones

de programar un clasificador basado en reglas, seguido de cómo es posible poder abordar estas dificultades con SL, y posteriormente cómo programarlo con ML4Kids. Si bien, hubo complicaciones de algunos participantes con ML4Kids —dificultad y tedio al realizar el entrenamiento, o confusión sobre el uso del bloque `recognise` para predecir la clase—, y además que algunos participantes en el inicio no tuvieron claridad del propósito de un clasificador —predecir la clase para valores que no conoce previamente—, igualmente se aprecia evidencia de progreso —respecto al día anterior— dado el cuestionario *verbal probing*. Por ejemplo, al solicitar que describieran “clasificación” se refirieron de forma más precisa como separación o agrupación de objetos, y utilizaron palabras como categoría, clases y etiquetas.

En la tercera sesión, y luego de exponer los conceptos matemáticos previos, se pudo presentar por primera vez el algoritmo Bayes ingenuo y su utilidad para predecir el diagnóstico de una enfermedad. Luego, se introdujo a los bloques ExtBi con el fin de mostrar cómo es un clasificador de texto programado con estos bloques. Las primeras impresiones de la mayoría de las y los participantes fue de asombro, y algunos consideraron el programa como muy extenso, pudiendo deberse a la cantidad de bloques utilizados o a la longitud del texto en algunas piezas o bloques. Frente a esto, se debe considerar el evidente agotamiento en las y los participantes al final de la sesión —porque se presentaron principalmente conceptos matemáticos—, y por otro lado que la mayoría de las y los participantes no poseían experiencia previa en Scratch.

En la última sesión, al presentar por segunda vez el algoritmo Bayes ingenuo y ExtBi, hubo mayor cantidad de participantes —respecto al día anterior— que asientan comprender estos tópicos. Luego, al construir el clasificador de texto, algunos participantes se percatan de determinadas ventajas de aplicar SL con bloques ExtBi versus ML4Kids. Por ejemplo, rapidez en la ejecución del programa, y que no requiere de sitios extra para configurar el entrenamiento teniendo así visibilidad completa del clasificador y el proceso de SL. También, la mayoría de las y los participantes logró reconocer el orden de pasos del algoritmo Bayes ingenuo, y que la clasificación no está restringida solo a dos categorías. Por otro lado, algunos niños y niñas presentaron dudas en los bloques `entrenar` y `bayes` al requerir —como parámetro— el nombre o identificador de la categoría.

También, con el cuestionario *verbal probing* final, fue posible confirmar que la mayoría de las y los participantes a pesar de poseer una correcta idea de la secuencia con bloques ExtBi, es necesario reforzarles en qué consiste cada uno y su propósito, además de cómo se asocia las secciones del programa con las etapas de un algoritmo en AI (entrenamiento, definición del modelo y predicción). En otras palabras, se debe robustecer en las y los participantes la descripción de cada fase del algoritmo Bayes ingenuo y cómo se logra con ExtBi. En esta línea, en la revisión de programas creados por las y los participantes, se confirmó que se requiere reforzar el propósito de la lista de ejemplos y el proceso de asignar identificadores en el programa. No obstante, igualmente lograron reconocer ventajas de SL como mejorar el desempeño usando ejemplos, que un entrenamiento uniforme contribuye a una predicción superior y que permite reconocer nuevos valores según el conjunto de entrenamiento usado.

Declarado esto, en síntesis es posible indicar que esta solución es una alternativa adecuada para concientizar en clasificación con Bayes ingenuo, incluso para niñas y niños sin conocimientos previos en el área y programación. También, no se podría mencionar que hu-

bo diferencia de habilidades entre niños y niñas, destacando que participantes valoran la solución propuesta versus otras herramientas como ML4Kids. Por consiguiente, es posible determinar que se logró el objetivo general de este trabajo de tesis. Finalmente, luego de responder las preguntas de investigación en la Sección 6.3, es posible declarar que las dos hipótesis proclamadas —Sección 1.2— son válidas.

En consecuencia, es posible considerar este trabajo como una alternativa en la línea investigativa sobre educación en computación sobre AI, o CSE-AI. En concreto, es posible considerar las siguientes implicancias de la solución:

- La ExtBi resulta ser una herramienta que facilita la aplicación del método SL, permitiendo programar un clasificador mediante bloques y transparentando el funcionamiento del algoritmo Bayes ingenuo, a un nivel que resulta comprensible por niñas y niños de 10 a 12 años. Además, esta solución, al no depender de servicios u otros sitios web para su uso, resulta ser más rápida y útil para para visualizar el proceso de SL. Finalmente, se debe considerar que la ExtBi está restringida a un clasificador por programa, solo procesa datos de tipo texto cadena o *strings*, y que los bloques están solo en español.
- Respecto a la serie de actividades diseñadas, estas van en la línea de soluciones promovidas por la comunidad de CSE sobre educación de ML y SL en los niveles K-12, con la característica de poder lograr concientización sobre Bayes ingenuo al programar un clasificador siguiendo este algoritmo. También, respecto a las Grandes Ideas propuestas por el grupo AI4K12 [26], esta solución aborda de manera teórica y práctica los conceptos de representación y razonamiento (Gran Idea #2), como también de aprendizaje (Gran Idea #3).

Para finalizar, es posible continuar desarrollando los componentes de la solución propuesta como trabajo futuro. Respecto a la ExtBi, es posible incorporar nuevos tipos de datos a clasificar, como imágenes o audio. Incluso, es posible desarrollar un micromundo en Scratch considerando otros tipo de clasificadores con SL, mediante NN o árbol de decisión, o bien algoritmos de AI empleados en otros métodos de ML como RL o UL. En este sentido, quizás sea posible utilizar el micromundo *Let's Chance* [4]. Por otro lado, en caso de ser necesario profundizar en cada bloque, podría emplearse Snap! con tal visibilizar su funcionamiento interno.

Respecto al material educativo, como posible idea que permite complementar lo realizado, se puede considerar incorporar una ejercicio práctico que consiste en entregar a los y las participantes el código de un clasificador con errores, y que lo modifiquen hasta lograr el resultado esperado. También, otra actividad posible sería hacer entrega de un programa con un clasificador basado en reglas y otro clasificador basado en datos, con tal de que cada participante pueda contrastar las alternativas y así reflexionar sobre los pros y contras de cada opción. Respecto a la metodología del taller es posible adaptarlo para que sea de forma presencial, y que implica principalmente adaptar las actividades en subconjuntos de participantes. Finalmente, es posible incorporar la realización de un focus group con niñas y niños que ya participaron en el taller, con tal de analizar la retención de conocimientos luego de un tiempo determinado.

Bibliografía

- [1] AI4ALL: *AI4ALL - Our Vision for AI*, may 2021. <https://ai-4-all.org/our-vision-for-ai/>.
- [2] Bell, Tim: *CS Unplugged or Coding Classes?* Commun. ACM, 64(5):25–27, Abril 2021, ISSN 0001-0782. <https://doi.org/10.1145/3457195>.
- [3] Dasgupta, Sayamindu y Benjamin Mako Hill: *Scratch Community Blocks: Supporting Children As Data Scientists*. En *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, páginas 3620–3631, New York, NY, USA, 2017. ACM, ISBN 978-1-4503-4655-9. <http://doi.acm.org/10.1145/3025453.3025847>.
- [4] Dhariwal, Manuj: *Let's Chance: Playful Probabilistic Programming for Children*. M.S. Thesis, Massachusetts Institute of Technology, Cambridge, MA, Agosto 2020. <https://www.media.mit.edu/publications/lets-chance-playful-probabilistic-programming-for-children-thesis/>.
- [5] DiPaola, Daniella, Blakeley H. Payne y Cynthia Breazeal: *Decoding Design Agendas: An Ethical Design Activity for Middle School Students*. En *Proceedings of the Interaction Design and Children Conference*, IDC '20, página 1–10, New York, NY, USA, 2020. Association for Computing Machinery, ISBN 9781450379816. <https://doi.org/10.1145/3392063.3394396>.
- [6] Gong, Xiaoyan, Ying Tang, Xiwei Liu, Sifeng Jing, Wei Cui, Joleen Liang y Fei Yue Wang: *K-9 Artificial Intelligence Education in Qingdao: Issues, Challenges and Suggestions*. páginas 1–6, Octubre 2020.
- [7] Han, Xiao, Fang Hu, Gang Xiong, Xiwei Liu, Xiaoyan Gong, Xiaojie Niu, Wanruo Shi y Xinzhu Wang: *Design of AI + Curriculum for Primary and Secondary Schools in Qingdao*. páginas 4135–4140, Noviembre 2018.
- [8] Hayes, Gillian R.: *The Relationship of Action Research to Human-Computer Interaction*. ACM Trans. Comput.-Hum. Interact., 18(3), Agosto 2011, ISSN 1073-0516. <https://doi.org/10.1145/1993060.1993065>.
- [9] Hitron, Tom, Iddo Wald, Hadas Erel y Oren Zuckerman: *Introducing Children to Machine Learning Concepts Through Hands-on Experience*. En *Proceedings of the 17th ACM Conference on Interaction Design and Children*, IDC '18, páginas 563–568, New

- York, NY, USA, 2018. ACM, ISBN 978-1-4503-5152-2. <http://doi.acm.org/10.1145/3202185.3210776>.
- [10] Inc, Gartner: *Artificial Intelligence Will Create More Jobs Than It Eliminates*, dec 2017. <https://gtnr.it/2vR2096>.
- [11] Jatzlau, S., T. Michaeli, S. Seegerer y R. Romeike: *It's not Magic After All – Machine Learning in Snap! using Reinforcement Learning*. En *2019 IEEE Blocks and Beyond Workshop (B B)*, páginas 37–41, 2019.
- [12] Kahler, Helge, Finn Kensing y Michael Muller: *Methods and Tools: Constructive Interaction and Collaborative Work: Introducing a Method for Testing Collaborative Systems*. *Interactions*, 7(3):27–34, Junio 2000, ISSN 1072-5520. <https://doi.org/10.1145/334216.334226>.
- [13] Kandlhofer, Martin, Gerald Steinbauer, Sabine Hirschmugl-Gaisch y Petra Huber: *Artificial intelligence and computer science in education: From kindergarten to university*. En *2016 IEEE Frontiers in Education Conference, FIE 2015, Eire, PA, USA, October 12-15, 2016*, páginas 1–9, 2016. <https://doi.org/10.1109/FIE.2016.7757570>.
- [14] Lee, Irene, Safinah Ali, Helen Zhang, Daniella DiPaola y Cynthia Breazeal: *Developing Middle School Students' AI Literacy*. En *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education, SIGCSE '21*, página 191–197, New York, NY, USA, 2021. Association for Computing Machinery, ISBN 9781450380621. <https://doi.org/10.1145/3408877.3432513>.
- [15] Mitchell, T.M.: *Machine Learning*. McGraw-Hill international editions - computer science series. McGraw-Hill Education, 1997, ISBN 9780070428072. <https://books.google.cl/books?id=xOGAngEACAAJ>.
- [16] Papert, Seymour: *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, Inc., New York, NY, USA, 1980, ISBN 0-465-04627-4.
- [17] Resnick, Mitchel, Fred Martin, Robert Berg, Rick Borovoy, Vanessa Colella, Kwin Kramer y Brian Silverman: *Digital Manipulatives: New Toys to Think with*. En *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '98*, páginas 281–287, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co., ISBN 0-201-30987-4. <http://dx.doi.org/10.1145/274644.274684>.
- [18] Rich, Kathryn M., Carla Strickland, T. Andrew Binkowski, Cheryl Moran y Diana Franklin: *K-8 Learning Trajectories Derived from Research Literature: Sequence, Repetition, Conditionals*. ICER '17, página 182–190, New York, NY, USA, 2017. Association for Computing Machinery, ISBN 9781450349680. <https://doi.org/10.1145/3105726.3106166>.
- [19] Rodríguez-García, Juan David, Jesús Moreno-León, Marcos Román-González y Gregorio Robles: *Evaluation of an Online Intervention to Teach Artificial Intelligence with LearningML to 10-16-Year-Old Students*. En *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education, SIGCSE '21*, página 177–183, New

York, NY, USA, 2021. Association for Computing Machinery, ISBN 9781450380621. <https://doi.org/10.1145/3408877.3432393>.

- [20] Russell, Stuart y Peter Norvig: *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, USA, 3rd edición, 2009, ISBN 0136042597.
- [21] Shull, Forrest, Janice Singer y Sjøberg Dag I K.: *Guide to advanced empirical software engineering*. Springer, 2008.
- [22] Stol, Klaas Jan y Brian Fitzgerald: *The ABC of Software Engineering Research*. ACM Trans. Softw. Eng. Methodol., 27(3), sep 2018, ISSN 1049-331X. <https://doi.org/10.1145/3241743>.
- [23] Sun, Yunjia: *Novice-Centric Visualizations for Machine Learning*, 2016. <http://hdl.handle.net/10012/10396>.
- [24] Tan, Pang Ning, Michael Steinbach y Vipin Kumar: *Introduction to Data Mining*, volumen 1. Pearson Education, 2006, ISBN 0-321-42052-7.
- [25] Theodoridis, Sergios y Konstantinos Koutroumbas: *Chapter 2 - Classifiers Based on Bayes Decision Theory*. En Theodoridis, Sergios y Konstantinos Koutroumbas (editores): *Pattern Recognition (Fourth Edition)*, páginas 13–89. Academic Press, Boston, fourth edition edición, 2009, ISBN 978-1-59749-272-0. <https://www.sciencedirect.com/science/article/pii/B9781597492720500049>.
- [26] Touretzky, David, Christina Gardner-McCune, Fred Martin y Deborah Seehorn: *Envisioning AI for K-12: What Should Every Child Know about AI?* Proceedings of the AAAI Conference on Artificial Intelligence, 33(01):9795–9799, Jul. 2019. <https://ojs.aaai.org/index.php/AAAI/article/view/5053>.
- [27] Touretzky, David, Fred Martin, Deborah Seehorn, Cynthia Breazeal y Tess Posner: *Special Session: AI for K-12 Guidelines Initiative*. En *Proceedings of the 50th ACM Technical Symposium on Computer Science Education, SIGCSE '19*, páginas 492–493, New York, NY, USA, 2019. ACM, ISBN 978-1-4503-5890-3. <http://doi.acm.org/10.1145/3287324.3287525>.
- [28] Vachovsky, Marie E., Grace Wu, Sorathan Chaturapruek, Olga Russakovsky, Richard Sommer y Li Fei-Fei: *Toward More Gender Diversity in CS Through an Artificial Intelligence Summer Program for High School Girls*. En *Proceedings of the 47th ACM Technical Symposium on Computing Science Education, SIGCSE '16*, páginas 303–308, New York, NY, USA, 2016. ACM, ISBN 978-1-4503-3685-7. <http://doi.acm.org/10.1145/2839509.2844620>.
- [29] Yin, R.K.: *Case Study Research: Design and Methods*. Applied Social Research Methods. SAGE Publications, 2013, ISBN 9781452242569.

Anexos

Anexo A

Formulario de Consentimiento

Consentimiento Informado

Buenas tardes.

El siguiente formulario debe ser completado por el/la apoderado/a del/a niño o niña que participará en el taller.

*Obligatorio

Estimada/o apoderada/o

Como investigadores de la Universidad de Chile, estamos comprometidos con apoyar la educación de nuestro país. Para ello, dentro del marco de este taller realizamos investigación activa acerca de cómo los/as niños/as en edad escolar pueden adquirir y desarrollar habilidades de pensamiento computacional y programación de aplicaciones de software.

Para nosotros es muy importante entender cómo los/as estudiantes perciben y entienden las actividades que les proponemos. Es por eso que continuamente realizamos revisiones a las actividades propuestas y sintetizamos de manera agregada y anonimizada los resultados obtenidos.

A través de este documento, solicitamos a usted colaborar con el proyecto "Concientización a niñas y niños chilenos de 10 a 12 años sobre clasificación y algoritmo Bayes ingenuo utilizando una extensión al lenguaje Scratch". El autor de este trabajo es Patricio Quiroz Vargas junto al profesor guía, el académico Francisco J. Gutiérrez, Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile. Las únicas personas autorizadas a recopilar y procesar la información recogida en este proyecto son las que conforman el equipo de investigación del académico responsable.

Al completar este formulario manifiesto que voluntariamente deseo participar en este estudio y autorizo a mi pupilo/a a participar en este proyecto. Asimismo, declaro conocer los términos de participación y el alcance de los datos recogidos en este proyecto de investigación.

1. Nombre del/a apoderado/a: *

2. Con fecha: *

Ejemplo: 7 de enero del 2019

Autorizo al Departamento de Ciencias de la Computación de la Universidad de Chile a:

Figura A.1: Formulario Consentimiento - Primera Parte

3. 1 - Utilizar los resultados de aprendizaje (tales como actividades parciales, avances de proyectos, proyecto final) de mi pupilo/a como insumos de investigación en la mejora continua del taller realizado u otros fines que el equipo estime pertinente. Esta información será procesada de manera individual o agregada, siempre manteniendo en reserva la identidad del/a alumno/a, respetando su integridad y autoría, pudiendo utilizarse estos datos en reportes técnicos, artículos de investigación (por ejemplo, en conferencias y revistas especializadas sometidas a referato). *

Marca solo un óvalo.

Sí

No

4. 2. Solicitar la participación de mi persona o la de mi pupilo/a en entrevistas o grupos focales, con la finalidad de mejorar continuamente el taller realizado y otros fines estrictamente académicos y de investigación que el equipo considere pertinente. Entiendo que dar mi autorización no implica necesariamente mi participación activa en estas instancias, y que tengo el derecho de desistir de participar en el momento que lo considere pertinente sin tener que dar necesariamente explicaciones para ello: *

Marca solo un óvalo.

Sí

No

Para confirmar, indicar nuevamente la siguiente información

5. Nombre del/a apoderado/a: *

6. Correo del/a apoderado/a: *

Figura A.2: Formulario Consentimiento - Segunda Parte

7. Teléfono de contacto:

En caso de preguntas y/o comentarios:

Por favor, contacte al académico responsable de este proyecto si tiene alguna pregunta sobre el estudio, sobre sus derechos, o sobre el uso de la información recopilada durante la investigación. Los datos de contacto se indican más abajo. El Prof. Francisco J. Gutiérrez suscribe el compromiso de respetar cabalmente las condiciones detalladas más arriba.

Francisco J. Gutiérrez, Profesor Asistente
Departamento de Ciencias de la Computación, Universidad de Chile
frgutier@dcc.uchile.cl | +56 2 2978 0886

Figura A.3: Formulario Consentimiento - Tercera Parte

Anexo B

Formulario de Asentimiento

Asentimiento Informado

Estimada/o participante del taller,

Como investigadores de la Universidad de Chile, estamos comprometidos con apoyar la educación de nuestro país. Para ello, dentro del marco de este taller realizamos investigación activa acerca de cómo los/as niños/as en edad escolar pueden adquirir y desarrollar habilidades de pensamiento computacional y programación de aplicaciones de software.

Para nosotros es muy importante entender cómo los/as estudiantes perciben y entienden las actividades que les proponemos. Es por eso que continuamente realizamos revisiones a las actividades propuestas y sintetizamos de manera agregada y anonimizada los resultados obtenidos.

A través de este documento, solicitamos a usted colaborar con el proyecto "Concientización a niñas y niños chilenos de 10 a 12 años sobre clasificación y algoritmo Bayes ingenuo utilizando una extensión al lenguaje Scratch". El autor de este trabajo es Patricio Quiroz Vargas junto al profesor guía, el académico Francisco J. Gutiérrez, Profesor Asistente del Departamento de Ciencias de la Computación de la Universidad de Chile. Las únicas personas autorizadas a recopilar y procesar la información recogida en este proyecto son las que conforman el equipo de investigación del académico responsable.

Al completar este formulario manifiesto que voluntariamente deseo participar en este estudio y declaro conocer los términos de participación y el alcance de los datos recogidos en este proyecto de investigación.

***Obligatorio**

1. Nombre del/a participante: *

2. Correo del/a participante *

3. Con fecha: *

Ejemplo: 7 de enero del 2019

Figura B.1: Formulario Asentimiento - Primera Parte

Autorizo al Departamento de Ciencias de la Computación de la Universidad de Chile a:

4. Utilizar mis resultados de aprendizaje (tales como actividades parciales, avances de proyectos, proyecto final) como insumos de investigación en la mejora continua del taller realizado u otros fines que el equipo estime pertinente. Esta información será procesada de manera individual o agregada, siempre manteniendo en reserva la identidad del/a alumno/a, respetando su integridad y autoría, pudiendo utilizarse estos datos en reportes técnicos, artículos de investigación (por ejemplo, en conferencias y revistas especializadas sometidas a referato). *

Marca solo un óvalo.

Sí

No

Figura B.2: Formulario Asentimiento - Segunda Parte

Anexo C

Cuestionario Inicial

Preguntas Iniciales

Las siguientes preguntas tienen como fin solo conocer un poco más sobre ti y cuáles conceptos podrías ya poseer. No tiene una calificación.

Selecciona una opción como respuesta.

***Obligatorio**

1. ¿Cuál es tu nombre y apellidos? *

2. ¿Cuál es tu correo? *

3. ¿Cuál dispositivo usarás principalmente en el taller? *

Selecciona todos los que correspondan.

Computador

Tablet

Smartphone

4. Describe qué entiendes por Algoritmo: *

Figura C.1: Cuestionario Inicial - Primera Parte

5. ¿A qué te suena Inteligencia Artificial (IA)? *

6. ¿Haz oído sobre qué es un "conjunto de entrenamiento"? De ser así, por favor comenta *

7. Define con tus palabras qué es "clasificación" *

8. ¿Has programado previamente con Scratch?. De ser así comenta brevemente o indica el link a algún proyecto que hayas creado *

Figura C.2: Cuestionario Inicial - Segunda Parte

9. ¿Has oído del algoritmo Bayes Ingenuo? De ser así, comenta con tus propias palabras de qué se trata *

Figura C.3: Cuestionario Inicial - Tercera Parte

Anexo D

Trivia con Kahoot

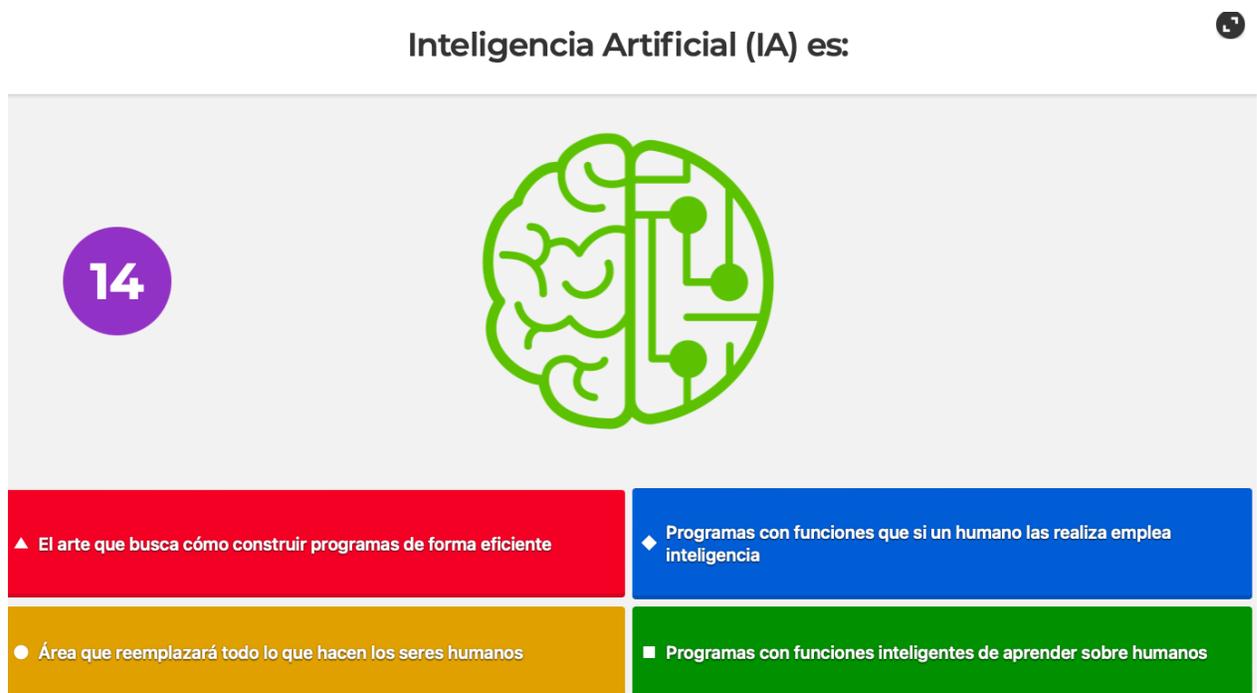


Figura D.1: Kahoot - Primera Pregunta

El reconocimiento facial ¿con cuál conjunto de entrenamiento se logra?



14



▲ imágenes de animales

◆ videos de personas

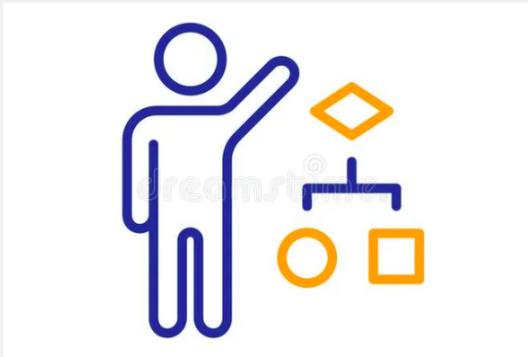
● rostros de personas

Figura D.2: Kahoot - Segunda Pregunta

Hacer un algoritmo es lo mismo que programar



14



◆ True

▲ False

Figura D.3: Kahoot - Tercera Pregunta

Google Maps te recomienda la ruta a tu destino ¿cómo es la ruta recomendada?

14

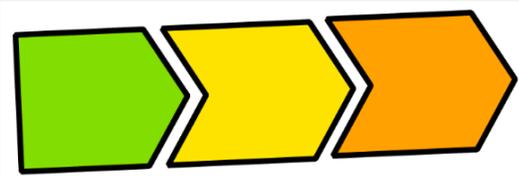


▲ la ruta con más sol	◆ la ruta más corta
● la ruta con mejor paisaje al destino	■ la ruta con menos policías

Figura D.4: Kahoot - Cuarta Pregunta

Algoritmos de IA tienen las siguientes características

14



▲ Conjunto de entrenamiento, acciones y resultado	◆ Un dato, acciones que aprenden y predicción
● Conjunto de entrenamiento, acciones y predicción	■ Conjunto de entrenamiento, acciones aprenden y predicción

Figura D.5: Kahoot - Quinta Pregunta

Si tuvieses que construir un traductor de voz a texto ¿cuál conjunto de entrenamiento usarías?



12



▲ películas subtitradas o dobladas

◆ canciones en mp3

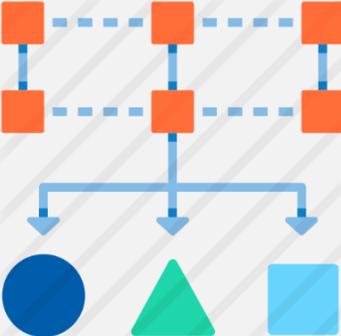
● libros en formato pdf

Figura D.6: Kahoot - Sexta Pregunta

Clasificación se refiere a:



15



▲ Asignar al azar una categoría (o clase) a datos

◆ Separar datos en grupos

● Asignar una categoría (o clase) a datos en base a sus características

■ Ninguna de las anteriores

Figura D.7: Kahoot - Séptima Pregunta

El detector de correo no deseado ¿con cuál(es) ejemplos se entrena para predecir?

15



▲ correos recibidos no deseados

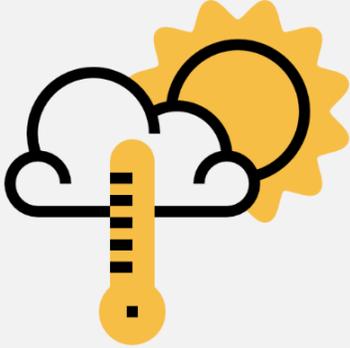
◆ correos recibidos deseados

● todas las anteriores

Figura D.8: Kahoot - Octava Pregunta

¿Cuál conjunto de entrenamiento es más importante para predecir el clima?

16



▲ reporte temperatura

◆ reporte humedad

● reporte precipitaciones

■ todas las anteriores

Figura D.9: Kahoot - Novena Pregunta

Anexo E

Instructivo de Realización - Método CI con ML4Kids

Instructivo Ejecución Constructive Interaction

Se realizará un taller de programación a niñas y niños chilenos entre 10 a 12 años, con el objetivo de acercarlos al área de inteligencia artificial y en específico a clasificación.

En esta ocasión las actividades prácticas contemplan el uso de la herramienta ML4Kids, y para evaluar se llevará a cabo el método Constructive Interaction (CI). Este método tiene como característica que quienes realizan las tareas, los y las participantes, van verbalizando en voz alta lo que hacen. El método CI, está orientado para evaluar tareas colaborativas. Por esto, se definirán grupos de participantes que construirán en forma conjunta un clasificador de texto, turnándose para realizar una tarea y compartiendo su pantalla con el resto mientras verbaliza lo que va haciendo. Cada grupo será guiado por un facilitador quién, además de listar las tareas a ejecutar y velar por su cumplimiento, será el/la encargado de fomentar la participación entre participantes, en los tiempos correspondientes..

1 - Aspectos Generales de la Aplicación

Se espera que al aplicar el CI, los y las participantes ya hayan tenido experiencia con ML4Kids, construyendo un clasificador de texto para identificar si un mensaje expresa un sentimiento de alegría o tristeza. Con esto, la principal tarea ahora será hacer un clasificador de texto para identificar si una noticia (título) está relacionada con videojuegos o animales. Para el uso de la herramienta, lo mínimo es contar con conexión a internet y que cada participante pueda acceder al sitio. Las principales funcionalidades a utilizar son crear un proyecto, entrenar, generar modelo, crear bloques y programar en Scratch el clasificador. Para esto, se considera como tiempo estimado 60 mins.

Se realizará la evaluación con CI de forma online, es decir el facilitador coordinará la actividad a través de una reunión por videollamada, breakout room de Zoom, la cual **debe grabar**. Antes de la aplicación del CI se llevará a cabo una **reunión previa** con el equipo de evaluadores, con el propósito de una última coordinación antes de ejecutar la actividad.

2 - Equipo de Facilitadores

El equipo que realizará las evaluaciones será de seis personas. La principal característica de cada uno es haber cumplido con la inducción del taller. Uno de aquellos será el coordinador del grupo que, además de realizar la evaluación a un grupo, tiene como responsabilidad recopilar todo el material generado en los otros grupos y evaluadores. El coordinador además, guiará las reuniones pre y post evaluación con el grupo de facilitadores, y así gestionar posibles comentarios o resolver dudas del equipo.

Figura E.1: Instructivo Ejecución CI ML4Kids - Primera Parte

3 - Insumos para Aplicación de CI

Los facilitadores, deben considerar las siguientes herramientas para llevar a cabo la evaluación:

- A. Acceso a <https://machinelearningforkids.co.uk/>
- B. Software para videollamadas Zoom, contemplando su correcta configuración en audio y video.
 - a. Link de reunión. Por cada grupo habrá un breakout room Zoom asignado y debiendo grabar la actividad. El facilitador debe validar con participantes que puedan compartir pantalla.
- C. Listado de tareas a ejecutar por participantes (punto 4.1)

4 - Procedimientos de Aplicación

El facilitador comienza explicando la modalidad de la actividad: una actividad colaborativa para construir un clasificador de noticias, en base al título (texto). Para esto hay una serie de tareas definidas que los y las participantes realizan. Al hacer estas tareas la idea es que todos avancen al mismo tiempo, mientras se asigna un turno a una/o para que indique al grupo como realizar la tarea. Para esto, la/el participante con el turno deberá compartir su pantalla y verbalizar lo que hace. Recordar aquellos/as que no están asignados con el turno, deben ir haciendo también la tarea pero sin compartir ni verbalizar; a todos/as en algún serán asignados y deben estar preparados.

En el método CI se valora la interacción entre los participantes, por ende se solicita que los facilitadores además de ir guiando la realización de tareas, conduzca la actividad para que se produzca la participación y colaboración del grupo entero, evitando el [problema del polizón](#). Para esto, el/la facilitador/a **escoge directamente** el/la participante para realizar la tarea en particular, pero solo permitir voluntarios como último recurso (cuando ningún asignado pudo completar la tarea).

IMPORTANTE: El/la facilitador/a **no debe intervenir** en cómo un participante lleva a cabo la tarea, sino sólo limitarse a indicar qué hacer, y la/el participante será responsable de cómo lograrlo. Si al asignar un participante a una tarea, esta/e responde que no sabe o siente inseguridad, el/la facilitador/a debe guiarle (por ejemplo resumir, muy brevemente, lo enseñado y realizado hasta ese momento). Si continúa sin saber qué hacer, hay que preguntar al resto de participantes alguna pista o sugerencia que le pueda ayudar a su compañera/o de grupo. Solo cuando el/la participante indique que ha finalizado, el/la facilitador debe revisar que lo realizado esté correcto o no para avanzar a la siguiente tarea. Finalmente, si no logra hacerlo correctamente o si se equivoca, entonces el facilitador asigna a otro participante para que realice o corrija la tarea. En caso de que ningún participante asignado realice la tarea, como último recurso escoger un voluntario.

Figura E.2: Instructivo Ejecución CI ML4Kids - Segunda Parte

4.1 - Tareas:

A continuación se indican las tareas a realizar por los/as participantes, cada una numerada. Los párrafos no numerados, son comentarios o para los/as facilitadores y no se indican a las y los participantes.

- 1- Cambiar lenguaje del sitio a 'Español Latinoamericano'
- 2- Crear proyecto ML4Kids con nombre significativo, por ejemplo 'clasificador de noticias'

Conjunto de datos. Etapa donde se entrena al clasificador. Se deben crear las categorías (o clases) de noticias y agregar ejemplos. En base al tiempo limitado, se disponen ejemplos para copiar y pegar, sin embargo se recomienda que los participantes sugieran ejemplos de noticias para compartir con el grupo. En caso de que se incorporen otros ejemplos, tutores deben anotar estos nuevos ejemplos para luego revisar programas finales.

- 3- Entrenar clasificador sobre categoría animales
- 4- Entrenar clasificador sobre categoría videojuego

Acciones que aprenden. Etapa donde se configuran las acciones del algoritmo para ajustar el clasificador y luego poder predecir

- 5- Aprender y Probar. Ejecutar configuración del modelo o aprendizaje.

Predicción con Scratch. Etapa necesaria en ML4Kids para poder acceder a los bloques Scratch

- 6- Crear. Acceder a Scratch 3 (redirigido desde ML4Kids)
- 7- Nombrar programa Scratch: ml4kids-noticias-**Nombre_Apellido**
- 8- Construir programa que:
 - a. Permita ingresar un texto o noticia
 - b. Luego hacer una clasificación del mensaje ingresado. **Si** es una noticia de animales, que el personaje indique "esta noticia es sobre animales", **SINO** que el personaje diga "esta noticia es sobre videojuegos". No es necesario construir disfraces para el personaje.
- 9- Tomar una foto (screenshot) del programa. Sino puede tomar foto, descargar programa.
- 10- Subir foto del programa en carpeta **ml4kids** (carpeta compartida)

Los facilitadores deben verificar upload o recibir el programa por cada participante

- 11- Completar cuestionario final

Figura E.3: Instructivo Ejecución CI ML4Kids - Tercera Parte

Después de Aplicación

Una vez concluida cada CI, se considerará una **reunión posterior** con los evaluadores donde se pueda conversar y conocer las apreciaciones o sensaciones de ellos/as. Cada reunión posterior será fijada una vez finalizada las sesiones de ese día, según la planificación. También, se solicitará la entrega del material de cada evaluación: videos y notas de cada evaluador.

Figura E.4: Instructivo Ejecución CI ML4Kids - Cuarta Parte

Anexo F

Instructivo de Realización - Método CI con Extensión Bayes Ingenuo

Instructivo Ejecución Constructive Interaction

Se realizará un taller de programación a niñas y niños chilenos entre 10 y 12 años, con el propósito de acercarlos al área machine learning, aprendizaje supervisado, clasificación, y en específico al algoritmo Bayes ingenuo.

En esta ocasión, las principales actividades prácticas, consideran el uso de una extensión Scratch (ExtBi) útil para implementar el algoritmo Bayes ingenuo. Para validar dicha extensión, se llevará a cabo el método Constructive Interaction (CI). Este método tiene como característica que quienes realizan las tareas, los y las participantes, van verbalizando en voz alta lo que hacen. El método CI, está orientado para evaluar tareas colaborativas. Por esto, se definirán grupos de participantes que construirán en forma conjunta un clasificador de texto, turnándose para realizar una tarea y compartiendo su pantalla con el resto mientras verbaliza lo que va haciendo. Cada grupo será guiado por un facilitador quién, además de listar las tareas a ejecutar y velar por su cumplimiento, será el/la encargado de fomentar la participación entre participantes, en los tiempos correspondientes.

1 - Aspectos Generales Aplicación CI para ExtBi

Se espera que al aplicar el CI, los y las participantes ya hayan tenido experiencia con ExtBi, construyendo un clasificador de texto para identificar si un mensaje expresa un sentimiento de alegría o tristeza. Con esto, la principal tarea ahora será hacer un clasificador de texto para identificar si una noticia (título) está relacionada con videojuegos o animales. Para el uso de la herramienta, lo mínimo es contar con conexión a internet y que cada participante pueda acceder al sitio. Las principales funcionalidades a utilizar son crear un proyecto, entrenar, generar modelo, crear bloques y programar en Scratch el clasificador. Para esto, se considera como tiempo estimado 60 mins.

Se realizará la evaluación con CI de forma online, es decir el facilitador coordinará la actividad a través de una videollamada, breakout room de Zoom, la cual **será grabada**. Antes de la aplicación del CI se llevará a cabo una **reunión previa** con el equipo de facilitadores, con el propósito de asignar las reuniones para cada equipo y llevar a cabo una última coordinación antes de ejecutar la actividad.

2 - Equipo de Facilitadores

El equipo que realizará las evaluaciones será de seis personas. La principal característica de cada uno es haber cumplido con la inducción del taller. Uno de aquellos será el coordinador del grupo que, además de realizar la evaluación a un grupo, tiene como responsabilidad recopilar todo el

Figura F.1: Instructivo Ejecución CI ExtBi - Primera Parte

material generado en los otros grupos y evaluadores. El coordinador además, guiará las reuniones pre y post evaluación con el grupo de facilitadores, y así gestionar posibles comentarios o resolver dudas del equipo.

3 - Insumos para Aplicación de CI

Los facilitadores, deben considerar las siguientes herramientas para llevar a cabo la evaluación:

- A. Acceso a <https://scratch-naive-bayes.dcc.uchile.cl/>
- B. Software para videollamadas Zoom, contemplando su correcta configuración en audio y video.
 - a. Link de reunión. Por cada grupo habrá un breakout room Zoom asignado y debiendo grabar la actividad. El facilitador debe validar con participantes que puedan compartir pantalla.
 - b. Grabar sesión.
- C. Listado de tareas a ejecutar por participantes (punto 4.1)

4 - Procedimientos de Aplicación

El facilitador comienza explicando la modalidad de la actividad: una actividad colaborativa para construir un clasificador de noticias, en base al título (texto). Para esto hay una serie de tareas definidas que los y las participantes realizan. Al hacer estas tareas la idea es que todos avancen al mismo tiempo, mientras se asigna un turno a una/o para que indique al grupo como realizar la tarea. Para esto, la/el participante con el turno deberá compartir su pantalla y verbalizar lo que hace. Recordar aquellos/as que no están asignados con el turno, deben ir haciendo también la tarea pero sin compartir ni verbalizar; a todos/as en algún serán asignados y deben estar preparados.

En el método CI se valora la interacción entre los participantes, por ende se solicita que los facilitadores además de ir guiando la realización de tareas, conduzca la actividad para que se produzca la participación y colaboración del grupo entero, evitando el [problema del polizón](#). Para esto, el/la facilitador/a **escoge directamente** el/la participante para realizar la tarea en particular, pero solo permitir voluntarios como último recurso (cuando ningún asignado pudo completar la tarea).

IMPORTANTE: El/la facilitador/a **no debe intervenir** en cómo un participante lleva a cabo la tarea, sino sólo limitarse a indicar qué hacer, y la/el participante será responsable de cómo lograrlo. Si al asignar un participante a una tarea, esta/e responde que no sabe o siente inseguridad, el/la facilitador/a debe guiarle (por ejemplo resumir, muy brevemente, lo enseñado y realizado hasta ese momento). Si continúa sin saber qué hacer, hay que preguntar al resto de participantes alguna pista o sugerencia que le pueda ayudar a su compañera/o de grupo. Solo cuando el/la participante indique que ha finalizado, el/la facilitador debe revisar que lo realizado esté correcto o no para

Figura F.2: Instructivo Ejecución CI ExtBi - Segunda Parte

avanzar a la siguiente tarea. Finalmente, si no logra hacerlo correctamente o si se equivoca, entonces el facilitador asigna a otro participante para que realice o corrija la tarea. En caso de que ningún participante asignado realice la tarea, como último recurso escoger un voluntario.

4.1 - Tareas

A continuación se indican las tareas a realizar por los/as participantes, cada una numerada. Los párrafos no numerados, son comentarios o para los/as facilitadores y no se indican a las y los participantes.

- 1 - Definir idioma del entorno a 'español latinoamericano'
- 2 - Nombrar programa: **ExtBi-classNoticia-Nombre_Apellido**
- 3 - Agregar extensión de bloques Bayes Ingenuo

Codificación, con las siguientes tareas se comienza con la implementación del clasificador

- 4 - Definir clasificador, indicar nombre para este

Entrenamiento.

- 5 - Entrenar clasificador, categoría noticias animales
- 6 - Entrenar clasificador, categoría noticias videojuegos

Comportamiento, reconociendo patrones

- 7 - Permitir ingresar una nueva noticia

Recordar que según el algoritmo Bayes ingenuo debemos calcular la probabilidad de una categoría dada la nueva noticia (nuevo valor), y debemos almacenar (guardar) estas probabilidades, ya que la idea es determinar la categoría con mayor probabilidad. Para esto, usaremos una lista guardará (almacenará) las probabilidades calculadas con el bloque Bayes.

- 8 - Definir lista donde se almacenarán resultados de fórmula (teorema) de Bayes. Inicializarla vacía.

Por cada categoría calcular (fórmula) Bayes dada la nueva noticia (variable respuesta).

- 9 - Calcular Bayes para categoría animales dado la nueva noticia, y almacenar su resultado en la lista.

- 10 - Calcular Bayes para categoría videojuegos dado la nueva noticia, y almacenar su resultado en la lista.

Figura F.3: Instructivo Ejecución CI ExtBi - Tercera Parte

11 - Determinar la categoría con mayor probabilidad, entre los valores en la lista de resultados bayes

12 - Presentar resultado de la tarea o paso 11.

Probar

13 - Guardar programa, en computador y luego subir a carpeta Drive asignada.

Verificar upload

14 - Finalizar actividad CI indicando link de cuestionario.

15 - Luego volver a la sesión principal para cerrar el Taller.

Después de Aplicación

Una vez concluida cada CI, se considerará una **reunión posterior** con los evaluadores donde se pueda conversar y conocer las apreciaciones o sensaciones de ellos/as. Cada reunión posterior será fijada una vez finalizada las sesiones de ese día, según la planificación. También, se solicitará la entrega del material de cada evaluación: videos y notas de cada evaluador.

Figura F.4: Instructivo Ejecución CI ExtBi - Cuarta Parte

Anexo G

Cuestionario para Facilitadores

Cuestionario – Tutores

Antes de finalizar la inducción, por favor responder las siguientes preguntas

***Obligatorio**

1. ¿Cuál es tu nombre y apellido? *

2. ¿Cómo le describirías "Inteligencia Artificial" a un niño o niña? *

3. ¿Cuáles ventajas ofrece la técnica Aprendizaje Supervisado? *

Marca solo un óvalo.

- aprender en base a errores
- mejorar el desempeño usando ejemplos
- enseñar y aprender al mismo tiempo
- ninguna de las anteriores

Figura G.1: Cuestionario Post Inducción - Primera Parte

4. ¿Cuál de las siguientes afirmaciones es correcta sobre el Aprendizaje Supervisado? *

Marca solo un óvalo.

- Es una técnica que permite entrenar a programas con datos de muestra para que realicen determinadas tareas, en lugar de tener que especificar un conjunto explícito de pasos y reglas a seguir.
- Esta técnica implica que las personas ya no tendrán que pensar en cómo resolver los problemas, ya que los computadores pensarán por nosotros (y mejor que nosotros).
- Esta técnica implica que las personas ya no necesitan aprender a programar.
- Esta técnica sólo puede implementarse en super-computadores con una enorme potencia de cálculo.

5. Tanto Alicia como Roberto quieren entrenar un sistema para reconocer si un determinado texto es "alegre" o "triste". Alicia y Roberto siguen dos estrategias de entrenamiento diferentes. ¿Quién de los dos conseguirá el mejor sistema? *

Marca solo un óvalo.

- Alicia ha recopilado 10 ejemplos de textos alegres y otros 10 ejemplos de textos tristes
- Robert ha recogido 1.000 ejemplos de textos alegres y otros 10 ejemplos de textos tristes

6. En un clasificador ¿por qué es necesaria la etapa de entrenamiento? *

Figura G.2: Cuestionario Post Inducción - Segunda Parte

7. La afirmación "Un clasificador Bayes Ingenuo permite solo dos posibles categorías". ¿es verdadera o falsa? *

Marca solo un óvalo.

- Verdadera
 Falsa

8. Selecciona la alternativa con la secuencia de pasos (o instrucciones) que usarías para programar un clasificador Bayes ingenuo *

The image shows three options, A, B, and C, each represented by a red rounded rectangle containing text and small white ovals. Option A contains the text "máxima categoría entre valores" followed by a white oval containing "lista_bayes". Option B contains the text "bayes categoría" followed by white ovals containing "opc_cat", "dato", and "ej_nuevo". Option C contains the text "entrenar categoría opción" followed by white ovals containing "opc_cat", "con datos", "ejemplos", and "cantidad".

Marca solo un óvalo.

- A, B y C
 B, C y A
 C, B y A
 B, A y C

9. Describe las principales etapas que contempla el algoritmo Bayes ingenuo: *

Figura G.3: Cuestionario Post Inducción - Tercera Parte

10. Imagina que deseas mejorar tu clasificador de noticias agregándole nuevos ejemplos de noticias de videojuegos, ¿cuáles serían los pasos para lograrlo? *

Comparando Machine Learning 4 Kids versus bloques Bayes Ingenuo

11. ¿Cuál de las dos opciones consideras mejor para construir un clasificador y por qué? *

Sobre el método Constructive Interaction - CI

12. Describe con tus palabras en que consiste el método CI *

Figura G.4: Cuestionario Post Inducción - Cuarta Parte

13. Indica alguna sugerencia para aplicar el método CI con niñas y niños *

Figura G.5: Cuestionario Post Inducción - Quinta Parte

Anexo H

Cuestionario Verbal Probing

H.1. Post CI - ML4Kids

Cuestionario - ML4Kids

Por favor, responde las siguientes preguntas basadas en tu experiencia luego de utilizar Machine Learning for Kids (ML4Kids) para construir un clasificador de noticias (animales y videojuegos)

***Obligatorio**

1. Indica tu nombre y dos apellidos *

2. ¿Qué dispositivo usaste para la experiencia con ML4Kids? *

Selecciona todos los que correspondan.

- Computador
- Tablet
- Smartphone

3. ¿Cómo describirías el concepto de "Clasificación" a algún compañero del colegio? *

4. El bloque "recognise text" determina la categoría del valor ingresado. Para lograr esto ¿cómo crees que trabaja internamente? *



5. Imagina que deseas mejorar tu clasificador agregándole nuevos ejemplos sobre noticias de videojuegos, ¿cuáles serían los pasos para lograrlo con ML4Kids? *

6. Sin considerar que los bloques de ML4Kids están en inglés, ¿qué otra(s) dificultad(es) consideras que tiene esta herramienta? *

Figura H.2: Verbal Probing ML4Kids - Segunda Parte

H.2. Post CI - ExtBI

Cuestionario - Bayes Ingenuo

Por favor, responde las siguientes preguntas basadas en tu experiencia con el algoritmo Bayes Ingenuo y usando Scratch para construir tu clasificador de noticias (animales y videojuegos)

***Obligatorio**

1. ¿Cuál es tu nombre y apellido? *

2. ¿Cómo le describirías "Clasificación" a algún compañera/o del colegio? *

3. En un clasificador ¿para qué es necesaria la etapa de entrenamiento? *

4. La afirmación "Un clasificador Bayes Ingenuo permite reconocer solo dos posibles categorías", ¿es verdadera o falsa? *

Marca solo un óvalo.

Verdadera

Falsa

Figura H.3: Verbal Probing ExtBi - Primera Parte

5. Selecciona la alternativa con la secuencia de pasos (o instrucciones) que usarías para programar un clasificador Bayes ingenuo *

The image shows three sequences of steps for programming a Naive Bayes classifier, each in a red rounded rectangle on a grid background:

- A**: máxima categoría entre valores lista_bayes
- B**: bayes categoría opc_cat dado ej_nuevo
- C**: entrenar categoría opcion opc_cat con datos ejemplos cantidad

Marca solo un óvalo.

- A, B y C
- B, C y A
- B, A y C
- C, B y A

6. Describe las principales fases que componen al algoritmo Bayes ingenuo: *

7. Imagina que deseas mejorar tu clasificador agregándole nuevos ejemplos de noticias videojuegos, ¿cuáles serían los pasos para lograrlo? *

Figura H.4: Verbal Probing ExtBi - Segunda Parte

Comparando Machine Learning 4 Kids versus bloques Bayes Ingenuo

8. ¿Cuál de las dos opciones consideras mejor para construir un clasificador y por qué? *

Preguntas Finales

A continuación te pedimos que respondas las últimas preguntas del taller

9. ¿Qué dispositivo usaste para la experiencia con los bloques Bayes ingenuo? *

Selecciona todos los que correspondan.

- Computador/Notebook
- Tablet
- Smartphone

10. Describe qué entiendes por algoritmo: *

Figura H.5: Verbal Probing ExtBi - Tercera Parte

11. ¿A qué te suena Inteligencia Artificial (IA)? *

12. ¿Cuáles ventajas ofrece la técnica Aprendizaje Supervisado? *

Marca solo un óvalo.

- aprender en base a errores
- mejorar el desempeño usando ejemplos
- enseñar y aprender al mismo tiempo
- ninguna de las anteriores

13. Tanto Alicia como Roberto quieren entrenar un sistema para reconocer si un determinado texto es "alegre" o "triste". Alicia y Roberto siguen dos estrategias de entrenamiento diferentes. ¿Quién de los dos conseguirá el mejor sistema? *

Marca solo un óvalo.

- Alicia ha recopilado 10 ejemplos de textos alegres y otros 10 ejemplos de textos tristes
- Robert ha recogido 1.000 ejemplos de textos alegres y otros 10 ejemplos de textos tristes

Figura H.6: Verbal Probing ExtBi - Cuarta Parte

14. ¿Cuál de las siguientes afirmaciones es correcta sobre el Aprendizaje Supervisado? *

Marca solo un óvalo.

- Es una técnica que permite entrenar a programas con datos de muestra para que realicen determinadas tareas, en lugar de tener que especificar un conjunto explícito de pasos y reglas a seguir.
- Esta técnica implica que las personas ya no tendrán que pensar en cómo resolver los problemas, ya que los computadores pensarán por nosotros (y mejor que nosotros).
- Esta técnica implica que las personas ya no necesitan aprender a programar.
- Esta técnica sólo puede implementarse en super-computadores con una enorme potencia de cálculo.

15. Al implementar Aprendizaje Supervisado en un sistema de reconocimiento de texto: *

Marca solo un óvalo.

- Presentamos al programa un conjunto de textos de muestra y este, tras procesarlos es capaz de reconocer sólo los textos que coinciden exactamente con esos ejemplos
- Presentamos un conjunto de textos de muestra al programa y este, tras procesarlos es capaz de reconocer textos similares a esos ejemplos (es decir, de reconocer textos nuevos que no ha visto antes).
- Presentamos un conjunto de textos de muestra al programa y este, tras procesarlos es capaz de reconocer cualquier texto que le presentemos.
- Presentamos un conjunto de textos de muestra al programa y este, tras procesarlos es capaz de reconocer cualquier texto, imagen o sonido que le presentemos.

Y antes de decir adiós

Sobre tu experiencia con el taller.

Figura H.7: Verbal Probing ExtBi - Quinta Parte

Anexo I

Conjunto de Entrenamiento Clasificador - V1

```
estas muy grande  
haces todo bien  
eres increíble  
muy bien hecho  
muchas gracias  
eres magnifica  
eres bueno  
eres una buena persona  
te quiero  
lo hiciste bien  
quiero estar contigo  
eres una ganadora
```

Listing I.1: Conjunto de Entrenamiento - Categoría Sentimiento Alegría

```
no sirves para nada  
lo hiciste mal  
eres una mala persona  
eres malvado  
mal hecho  
no hay nada que agradecer  
eres malo  
no te quiero  
haces todo mal  
ya no quiero estar contigo  
no lo hiciste bien  
has perdido
```

Listing I.2: Conjunto de Entrenamiento - Categoría Sentimiento Tristeza

```
han encontrado una nueva especie de felino
los tiburones son peces
el perro es el mejor amigo del hombre
los gatos vivian con los faraones
el perro desciende del lobo
las tortugas pueden llegar a cien anos
los osos polares viven en clima frio
las jirafas son animales muy altos
hay animales que dan a luz y otros ponen huevos
se han extinguido mas de mil especies de animales
```

Listing I.3: Conjunto de Entrenamiento - Categoría Noticias de Animales

```
el minecraft tiene nueva version
la nueva consola se estrena en noviembre
el juego amoung us es para smartphone
con scratch puedes crear tus propios juegos
la nintendo switch lite es otra version
los juegos ahora se compran en linea
hacer partidas con jugadores de otros paises
el nuevo juego pokemon ya esta disponible
hay nuevos disfraces para fortnite
hay un nuevo juego de sonic
```

Listing I.4: Conjunto de Entrenamiento - Categoría Noticias de Videojuegos

Anexo J

Carta Aprobación Estudio - Comité de Ética en FCFM

CERTIFICACION N°025

CERTIFICACION COMITÉ DE ÉTICA Y BIOSEGURIDAD PARA LA INVESTIGACIÓN FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS UNIVERSIDAD DE CHILE

El Comité de Ética y Bioseguridad para la Investigación de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile certifica haber analizado en detalle el proyecto FONDECYT de Iniciación 2018, N° 11190248, titulado “**A Reference Framework of Software Development Practices Calibrated for Very Novice Programmers**” cuyo investigador responsable es Francisco Gutiérrez Figueroa, Profesor Asistente del Departamento de Ciencias de la Computación de la Facultad.

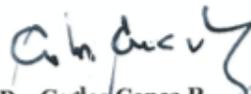
La metodología del proyecto incluye talleres de computación, encuestas, entrevistas y grupos focales. Se pedirá Consentimiento Informado de los apoderados y el Asentimiento informado de los participantes en las actividades descritas. Los datos serán almacenados en repositorios del proyecto y manejados en un servidor, terminales y disco duro, que serán propiedad del proyecto de investigación. Estos antecedentes permiten certificar que:

- i) El proyecto cumple con los estándares nacionales e internacionales de ética de la investigación, de acuerdo a la Declaración Universal de los Derechos Humanos, el Pacto de Derechos Civiles y Políticos, el Pacto de Derecho Económicos Sociales y Culturales, las leyes chilenas y el Documento oficial de ética para la investigación de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile.
- ii) El Comité de Ética considera que la investigación no vulnera la dignidad de los sujetos, no constituye una amenaza bajo ninguna circunstancia ni causa daño.
- iii) Asimismo, el Comité que suscribe auditará, al menos una vez durante la ejecución del proyecto, el cumplimiento de los estándares arriba enunciados propios de la disciplina

Figura J.1: Carta Aprobación Estudio - Primera Página

involucrada y la correcta aplicación del modelo de consentimiento informado, si es aprobado, para asegurarse del cumplimiento. Esta auditoría, además, tiene el propósito de asegurar la garantía del derecho a la privacidad, confidencialidad y el anonimato de los sujetos involucrados en la investigación.

iv) Dejamos constancia que el profesor Gutiérrez será responsable por eventuales daños causado a las personas por errores que puedan cometerse durante la investigación.


Dr. Carlos Conca R.
Profesor Titular
Departamento de Ingeniería
Matemática




Dra. Marcela Munizaga M.
Profesora Titular
Directora Académica y de Investigación


Dr. Patricio Jorquera E.
Secretario

Santiago, 11 de octubre de 2019

MMM CUR-PJE-rub

Figura J.2: Carta Aprobación Estudio - Segunda Página

Anexo K

Cuestionario - Piloto de Exploración

Cuestionario - Taller IA & Extensión Bayes Ingenuo

Por favor, responde las siguientes preguntas basadas en tu experiencia luego de utilizar la extensión Scratch Bayes Ingenuo (ExtNB) construyendo el clasificador

***Obligatorio**

1. Correo *

2. Luego del taller, si tu madre pregunta qué es Inteligencia Artificial ¿Qué le responderías? *

3. Menciona un ejemplo de aplicación clasificador que conozcas *

Figura K.1: Cuestionario Estudiantes de 1er Año - Primera Parte

4. Verdadero o Falso: "Solo es posible considerar dos hipótesis (posibles valores) para un clasificador Bayes Ingenuo" *

Marca solo un óvalo.

- Verdadero
 Falso

5. En base a tu respuesta anterior indica el por qué de tu elección, de la misma forma como si se estuvieses explicando a un amigo/a *

6. Verdadero o Falso: "Los datos tabulados son aquellos datos que son posibles de graficar en una tabla. Generalmente vienen en formato de archivos separados por coma" *

Marca solo un óvalo.

- Verdadero
 Falso

Figura K.2: Cuestionario Estudiantes de 1er Año - Segunda Parte

7. Menciona a continuación cuáles son las dos principales etapas que contempla el algoritmo Naive Bayes: *

8. Para las fases de Aprendizaje Supervisado (entrenamiento, definición del modelo, clasificación) ¿cómo asocias los bloques (Entrenar, Bayes, Max Hipótesis)? *



Figura K.3: Cuestionario Estudiantes de 1er Año - Tercera Parte

9. Una vez ya generados los bloques, etapa 3 Crear, ¿cuál sería el proceso para agregar una nueva posible categoría? (Por ejemplo agregar noticias de películas) *

10. ¿Qué mejora incorporarías a los bloques Bayes Ingenuo? *

Figura K.4: Cuestionario Estudiantes de 1er Año - Cuarta Parte

Anexo L

Formulario de Inscripción

Inscripción :: Taller de Inteligencia Artificial para Escolares

El presente formulario debe completarse con los datos de él o la participante del taller, y también con dato de la/él apoderado o tutor. Recuerda que los cupos son limitados.

***Obligatorio**

Datos de la o él participante

Necesitamos la siguiente información de quién asistirá al taller

1. ¿Cuál es su nombre? *

2. ¿Correo electrónico? *

3. ¿Cuántos años tienes? *

Consentimiento Infomado

Como investigadores de la Universidad de Chile, estamos comprometidos con apoyar la educación de nuestro país. Para ello, dentro del marco de este taller realizamos investigación activa acerca de cómo los/as niños/as en edad escolar pueden adquirir y desarrollar habilidades de pensamiento computacional y programación de aplicaciones de software.

Para nosotros es muy importante entender cómo los/as escolares perciben y entienden las actividades que les proponemos. Es por eso que continuamente realizamos revisiones a las actividades propuestas y sintetizamos de manera agregada y anonimizada los resultados obtenidos.

Al llenar este formulario para inscribirme en este taller manifiesto que voluntariamente deseo participar en este estudio. Asimismo, declaro conocer los términos de participación y el alcance de los datos recogidos en este proyecto de investigación.

Figura L.1: Formulario de Inscripción - Primera Parte

4. Autorizo que el DCC utilice los resultados de aprendizaje (tales como actividades parciales, avances de proyectos, proyecto final) como insumos de investigación en la mejora continua del taller realizado u otros fines que el equipo estime pertinente. Esta información será procesada de manera individual o agregada, siempre manteniendo en reser la identidad del/a alumno/a, respetando su integridad y autoría, pudiendo utilizarse estos datos en reportes técnicos artículos de investigación (por ejemplo, en conferencias y revistas especializadas sometidas a referato). *

Marca solo un óvalo.

SÍ

NO

Datos del apoderado/a

Necesitamos también algunos datos la persona adulta a cargo de él o la participante

5. Lazo o relación (ejemplo madre, padre, tutor, tía u otro) *

6. Nombre y Apellidos *

7. Correo electrónico *

Figura L.2: Formulario de Inscripción - Segunda Parte