

Universidad Nacional de Córdoba
Facultad de Matemática, Astronomía, Física y Computación



Trabajo Especial de la Licenciatura en Ciencias de la Computación

**Aprendizaje automático aplicado a la huerta
familiar agroecológica en Patagonia**

Directora:
Dra. Luciana Benotti

Candidato:
Guido Ivetta

Año Académico 2022



Aprendizaje automático aplicado a la huerta familiar agroecológica en Patagonia
por Guido Ivetta se distribuye bajo una Licencia Creative Commons
Atribución-NoComercial-CompartirIgual 4.0 Internacional.

Agradecimientos

A mi mamá, papá, hermano, amigos y docentes de la facultad.

Especial mención a Luciana y a todo el equipo de este proyecto por hacer esto posible.

Espero poder seguir disfrutando de su compañía, apoyo, y partidas de ping-pong por muchos años más.

Resumen

En las regiones del país donde hay climas extremos, los productos hortícolas no se producen localmente, entonces son caros y de mala calidad. Instituciones nacionales como Prohuerta y programas de ministerios de producción provinciales como PRODA en Neuquén generan mucha información útil y localizada para la autoproducción. Sin embargo, no dan abasto sistematizando y difundiendo esta información para poder asistir a todas las familias que tienen huerta. Este proyecto busca aportar tecnología digital para acompañar los procesos productivos de huertas agroecológicas, que permita dar acceso a información útil y relevante para la zona para el desarrollo de una huerta. Una huerta agroecológica familiar no sólo permite tener acceso a un alimento de mejor calidad sino a generar conciencia de la importancia de mejorar la alimentación. Hoy en día en internet existe mucha información sobre huerta pero de otras partes del mundo lo cual lleva a errores en las prácticas de familias principiantes en el cultivo hortícola. El objetivo técnico de este proyecto es sistematización y disponibilización de que permita obtener información local para el desarrollo de una huerta a partir de bases de datos de conocimiento de huerta adaptados a realidades del norte cordillerano de la patagonia. Esta información ha sido creada y recolectada por técnicos del PRODA y del taller productivo Amulén de Junín de los Andes interactuando con huerteros de la región tanto en las ciudades de la zona como con las zonas más rurales e indígenas. La interfaz de usuario de la información será un chatbot, es decir un sistema interactivo por mensajería instantánea. El sistema es capaz de responder preguntas sobre identificación de plagas, plantas y también de qué residuos se pueden reciclar usando clasificación automática de imágenes y de preguntas y de identificación de objetos, a través de la interfaz de un chatbot en la plataforma Telegram. El sistema desarrollado recupera saberes e información local, y la hace más accesible, para que sirva de apoyo y guía para que las familias puedan producir sus propios alimentos y para que más familias se puedan sumar a la autoproducción a través de esta herramienta para su "autoasistencia". El beneficio social consistiría en mejorar su alimentación y conservar, recuperar y potenciar el conocimiento local que circula y se comparte.

Este proyecto fue financiado parcialmente por el programa Nacional de Tecnologías e Innovación Social (PTIS y PROCODAS) del Ministerio de Ciencia Tecnología e Innovación Productiva de la Nación cuyo objetivo principal es el financiamiento de proyectos de Tecnologías para la Inclusión Social en las áreas de: Economía Social; Agricultura Familiar; Hábitat y Discapacidad. También contó con el apoyo del Ministerio del Producción de la Provincia de Neuquén y con el aval (no financiado) de una beca BITs de la Universidad Nacional de Córdoba. También contó con el apoyo de estudiantes del Compromiso Social Estudiantil de la Universidad de Córdoba. Este proyecto pretende ser un pequeño paso en la propuesta de soberanía tecnológica en el área de inteligencia artificial.

Summary

In regions of Argentina where there are extreme climates, horticultural products are not produced locally, so they are expensive and of poor quality. National institutions such as Prohuerta and programs of provincial production ministries such as PRODA in Neuquén generate a lot of useful and localized information for self-production. However, they are not able to systematize and disseminate this information at scale in order to assist all families with vegetable gardens. This project seeks to provide digital technology to accompany the productive processes of agroecological gardens, providing access to useful and relevant information for the area. An agroecological family garden not only allows access to better quality food but also to raise awareness of the importance of improving our nutrition. Nowadays, there is an abundance of information on the internet about vegetable gardens, but from other parts of the world, which leads to errors in the practices of families who are new to vegetable gardening. The technical objective of this project is to systematize and make available local information for the development of a vegetable garden based on vegetable garden knowledge databases adapted to the realities of the northern mountain range of Patagonia. This information has been created and collected by technicians of PRODA and the productive workshop Amulén of Junín de los Andes, interacting with gardeners of the region, both in the cities of the area and in the more rural and indigenous areas. The user interface for the information will be a chatbot, an interactive instant messaging system. The system is capable of answering questions about pest identification, plants and also about what waste can be recycled using automatic classification of images and questions and identification of objects, through the interface of a chatbot on the Telegram platform. The developed system recovers local knowledge, and makes it more accessible, to serve as a support and guide for families to produce their own food. The social benefit would consist of improving their food and preserving, recovering and enhancing the local knowledge that is shared.

This project was partially funded by the National Program of Technologies and Social Innovation (PTIS and PROCODAS) of the National Ministry of Science, Technology and Productive Innovation whose main objective is to finance projects of Technologies for Social Inclusion in the areas of: Social Economy; Family Agriculture; Habitat and Disability. It also had the support of the Ministry of Production of the Province of Neuquén and the endorsement (not financed) of a BITs grant from the National University of Córdoba. It also had the support of students of the Student Social Commitment of the University of Córdoba. This project aims to be a small step in the proposal of technological sovereignty in the area of artificial intelligence.

Contents

1	Introducción	10
1.1	La inteligencia artificial como herramienta para la comunidad	10
1.2	Acerca del contexto del proyecto y su surgimiento	11
1.3	Miembros del proyecto	12
1.4	El proyecto en sí: Inteligencia artificial aplicada a la huerta	14
1.5	El bot como interfaz	15
1.6	Novedades, relevancia y aspiraciones del proyecto	17
1.7	Reflexión y desafíos de la informática	20
1.8	Mapa de la tesis	21
2	Sistematización de datos	24
2.1	Concurso “Mi Huerta en Casa”	24
2.2	Calendarios de siembra	26
2.3	Fichas de cultivo	27
3	Clasificando imágenes	30
3.1	Aprendizaje por transferencia	30
3.2	Identificación de enfermedades de plantas	30
3.3	Identificación de plantas	34
3.4	Identificación de insectos	34
3.5	Identificación de residuos reciclables y compostables	38
4	Manejo conversacional	41
4.1	Propiedades de la conversación humana	41
4.2	Sistemas de diálogo orientados a tareas	42
4.2.1	Interpretación de la intención y slot filling	42
4.2.2	Generación de lenguaje natural	43
4.2.3	Evaluación de los sistemas de diálogo	43
4.3	Un framework para hacer chatbots: RASA	44
4.4	Diagramas de manejo conversacional	47
5	Generación de respuestas	50
5.1	Modelos de secuencia a secuencia	51
5.2	Modelos preentrenados y ajuste fino	52
5.3	Utilizando modelos de paráfrasis y tareas relacionadas	54
5.3.1	Bert2Bert	54
5.3.2	T5 en inglés para preguntas	54
5.3.3	Question-Answering	55
5.4	Entrenando un modelo de paráfrasis	56
5.4.1	Ejemplo 1	57
5.4.2	Ejemplo 2	57

5.4.3	Ejemplo 3	58
6	Evaluación de paráfrasis	60
6.1	Enfoque manual	60
6.2	Enfoque automático	61
6.3	Resultados de nuestros modelos de paráfrasis	61
7	Conclusiones	67
7.1	Discusión y conclusiones	67
7.2	Trabajo futuro	68

List of Figures

1	Captura de pantalla de la funcionalidad de identificación de insecto y posterior pregunta clasificada con el reconocimiento de intenciones de RASA . .	16
2	Captura de pantalla de la funcionalidad de paráfrasis al volver a preguntar el mismo interrogante.	17
3	Captura de pantalla de la funcionalidad de identificación de planta y pedido de ficha de cultivo.	18
4	Ficha de cultivo de Kale, detalle de la conversación de la Figura 3.	19
5	Imágenes publicadas por los comunicadores del proyecto como ejemplo para la primera (izquierda) y segunda (derecha) categoría.	25
6	Imágenes publicadas por los comunicadores del proyecto como ejemplo para la tercera (izquierda) y cuarta (derecha) categoría.	26
7	Sección del calendario de siembra proporcionado por las ingenieras del proyecto. S: Siembra, P: Plantación, C: Cosecha, SA: Siembra en Almácigo, SM: Siembra en Maceta	27
8	Muestra de algunas categorías del dataset de PlantVillage con sus respectivas etiquetas. Este dataset de 54.309 imágenes abarca 14 especies de cultivos: Manzana, arándano, cereza, uva, naranja, melocotón, pimiento, patata, fresa, calabaza, fresa y tomate. Además contiene imágenes de 17 enfermedades fúngicas, 4 enfermedades bacterianas, 2 enfermedades de moho (oomicetos), 2 enfermedades víricas y 1 enfermedad causada por ácaros. . .	31
9	Estadísticas de entrenamiento a lo largo de cinco epochs.	32
10	Matriz de confusión del modelo entrenado para clasificar plantas y enfermedades.	33
11	Sección de la matriz de confusión para el modelo entrenado para clasificar insectos del dataset IP102.	35
12	Muestra de algunas categorías de nuestro dataset para clasificación de insectos con sus respectivas etiquetas.	36
13	Evolución de la tasa de error para el modelo final de clasificación de insectos.	36
14	Matriz de confusión del modelo entrenado para clasificar insectos.	37

15	Muestra de algunas categorías de nuestro dataset para clasificación de residuos con sus respectivas etiquetas.	38
16	Evolución de la tasa de error para el modelo final de clasificación de residuos.	39
17	Matriz de confusión del modelo entrenado para clasificar residuos.	39
18	Etapas del funcionamiento interno de RASA.	46
19	Diagrama de diálogo de la funcionalidad de obtener más información de una planta luego de ser identificada por el sistema.	47
20	Diagrama de diálogo de la funcionalidad de identificar un insecto. Sólo se muestra en detalle el camino de identificación por foto.	48
21	Diagrama de diálogo de la funcionalidad de pregunta libre en una planta ya identificada. En este caso se destaca el uso de paráfrasis frente a re-preguntas.	49
22	Algunas de las funcionalidades del modelo T5, solicitadas en lenguaje natural al comienzo del prompt.	53
23	Evolución de la función de loss en modelo de paráfrasis	57
24	Histograma de puntaje en evaluación para la métrica de coherencia interna	63
25	Histograma de puntaje en evaluación para la métrica de conservación de semántica	63
26	Histograma de puntaje en evaluación para la métrica de gramaticalidad	64
27	Histograma de puntaje en evaluación para la métrica de no redundancia	64

1 Introducción

Esta tesis se desarrolla en colaboración con un equipo interdisciplinario de ingenieros agrónomos, especialistas en comunicación y en servicio social e investigadores y estudiantes de ciencias de la computación. Propone ser un ejemplo de un proyecto de soberanía tecnológica que busca crear herramientas de inteligencia artificial que sirva a mejorar las condiciones de vida (en este caso, la alimentación y el conocimiento local) de comunidades históricamente marginalizadas. Busca crear tecnología que aprenda de estas comunidades en lugar de que las comunidades se tengan que adaptar a tecnología de inteligencia artificial creada en el norte global con el objetivo de maximizar ganancias económicas.

Este capítulo comienza describiendo qué es la inteligencia artificial y cuáles son sus ventajas y desventajas en términos comprensibles para todo el equipo. La sección 1.2 justifica y propone un cambio de paradigma en la creación de IA proponiendo como caso de estudio este proyecto. La sección 1.3 brevemente describe los miembros de este equipo interdisciplinario. La sección 1.4 describe los objetivos del proyecto. La sección 1.5 explica cómo un chatbot se usa para la interfaz de usuario uniendo los distintos modelos de IA involucrados e ilustra las capacidades del sistema tanto de clasificación de imágenes como en la generación de paráfrasis para evitar la repetitividad que suelen tener los chatbots. También ilustra la puesta en valor de información local en el contenido de las respuestas que incluyen texto y fichas de cultivo creados por el PRODA. La sección 1.6 resalta los aspectos novedosos del proyecto, y su relevancia. Finalmente la sección 1.7 reflexiona sobre los desafíos para la IA si se piensa de forma comunitaria como propone este proyecto. La sección 1.8 cierra este capítulo con una breve descripción de todos los capítulos de esta tesis.

Una versión anterior de este capítulo fue publicada en las *Jornadas Nacionales de Investigadoras en Economías Regionales* organizada por la *Facultad de Agronomía de la Universidad de Buenos Aires*. Dicho artículo fue presentado colaborativamente por el equipo en Buenos Aires el 28 de octubre del 2022 y fue seleccionado para ser publicado en la *Revista de Ciencias Sociales Segunda Época* publicada ininterrumpidamente desde el 1998 por la Universidad Nacional de Quilmes, ISSN: 2347-1050.

1.1 La inteligencia artificial como herramienta para la comunidad

Inteligencia artificial (IA) es el área de computación de aprendizaje automatizado por computadora. La IA permite la construcción de algoritmos que hacen que el comportamiento de un programa no dependa de un programador, sino de los datos que le fueron dados. Las aplicaciones de este tipo son muy variadas. Un ejemplo de ésta índole, es una persona conectándose al homebanking donde se le informa que tiene un crédito preaprobado por un monto determinado; es muy probable que para determinar el monto, haya un algoritmo de aprendizaje automatizado que se encarga de estimar cuál es la probabilidad de que la persona devuelva el crédito. Otro ejemplo de IA, es la búsqueda de personas en fotos que se toman con celular, por reconocimiento facial. Estos algoritmos de aprendizaje profundo comparan rasgos con bases de datos de personas y sus nombres. El procesamiento de lenguaje natural también usa aprendizaje profundo basado en redes neuronales artificiales,

por ejemplo cuando una persona empieza a escribir algo en un buscador y le sugiere opciones de autocompletado, el sistema usa una técnica que se llama “modelos de lenguaje” para hacer las sugerencias. Estos modelos se utilizan también para reconocimiento del habla que sirve para por ejemplo, poder dictar a un celular. Hay algoritmos que simplifican muchas tareas y hacen la vida más cómoda pero que también cometen errores que son muy diferentes a los que cometería una persona. Por ejemplo, con la pandemia, surgieron muchas aplicaciones que trataban de usar datos para hacer predicciones. Hubo un proyecto que se realizó en hospitales de Estados Unidos donde un sistema de IA, cruzando distintos datos del paciente como historia clínica y estudios, trataba de predecir quiénes, de los que daban positivo para COVID, necesitaban ser internados. Resultó que uno de los parámetros que usaba el sistema para predecir era el tipo de letra de la historia clínica (si era times new roman, arial u otra), un criterio que no tiene sentido para una persona. Estos sistemas tienen un problema grave para darse cuenta qué es relevante y qué no, porque no entienden de relaciones causales. Sí entienden de correlaciones: si ven que algo pasa frecuentemente junto con otra cosa, entienden que hay una relación. Estos errores pueden ser peligrosos si al algoritmo se le da poder de decisión sobre temas importantes, como por ejemplo la otorgación de un crédito o la asignación de camas en un hospital. En este proyecto se busca crear un diseño comunitario de IA, como una manera de resistir a las inteligencias artificiales que históricamente colonizan la comunidad. La IA es una herramienta, no es buena ni mala, pero si es desarrollada para maximizar ganancias y no para proteger la vida, no es sorprendente que maximice ganancias y no proteja las vidas. Así, teniendo como premisa hacer de la IA una herramienta que aporte a las necesidades de las comunidades, aparece esta iniciativa desde el territorio patagónico, dentro de la provincia del Neuquén, más precisamente en las localidades de Junín y San Martín de los Andes, y surge bajo una demanda construida: el asesoramiento para el autocultivo de alimentos.

1.2 Acerca del contexto del proyecto y su surgimiento

A pesar de que muchas familias de las localidades de Junín de los Andes, San Martín de los Andes y parajes cercanos hacen huerta en sus casas, la gran mayoría de los productos hortícolas no se producen localmente. Son transportados grandes distancias, por lo que aumenta sus costos y demanda el uso de conservantes o manipulación genética para aumentar su durabilidad, disminuyendo así su calidad. Por lo que no responden a los principios de la soberanía alimentaria, y contribuyen a la creciente emisión de gases de invernadero, uno de los responsables del cambio climático. La soberanía alimentaria “defiende los derechos de las personas a comer alimentos saludables y nutritivos, y fomenta los ciclos productivos agroecológicos, respetando las diversidades climáticas y culturales de cada comunidad” (Vía Campesina, 2021). Según la Red CALISAS (2018), “se puede considerar que se trata de la autogestión de las comunidades sobre la administración de los alimentos, asegurando la previsión y la sanidad de los mismos, mediante modos de producción y comercialización beneficiosos para el conjunto de la sociedad”. Existen en la región programas como INTA ProHuerta (nacional) y PRODA (Programa de Desarrollo Agroalimentario, provincial) que asisten a las familias para la autoproducción. La demanda es muy alta, creció mucho

durante la pandemia y no se logra dar respuesta a la misma. Las personas que tienen acceso a internet suelen hacer consultas sobre cómo producir, pero las recomendaciones que encuentran, no suelen ser acordes a la zona por una cuestión climática, y hay prácticas puntuales que se deben hacer para tener éxito. La capacidad técnica a campo, sobre todo en pandemia, no alcanza a responder todas las dudas o llegar a varios lugares a la vez, observando además que muchas de las dudas y consultas se repiten y/o son frecuentes. Por otra parte, en la búsqueda de crear un diseño comunitario de IA, surge la propuesta de articulación para poder desarrollar una herramienta informática, adaptada localmente, que pueda ser utilizada para consultar sobre cómo producir determinados cultivos, y que pueda dar respuestas/orientaciones adaptadas a la zona desde una perspectiva no sólo climática, sino también cultural, de manejo agroecológico y que contribuya a la soberanía alimentaria: “La discusión sobre la agricultura debe ir más allá de lo que sucede dentro de los límites de la unidad de producción individual. La agroecología se trata de un modelo de producción que se percibe como un sistema mucho más vasto, con muchas partes interactuantes incluyendo componentes ambientales, económicos y sociales” [39]. Así, el objetivo del proyecto es desarrollar modelos que sean capaces de contestar preguntas multimodales usando técnicas de IA frecuentes de personas haciendo huerta en la región, y que contribuyan a los procesos locales de construcción de la Soberanía Alimentaria, Soberanía Tecnológica y Agroecología del lugar. El proyecto se denomina Inteligencia artificial aplicada a la huerta familiar agroecológica y al compostaje, que fue presentado y aprobado para su ejecución al Programa PROCODAS en la línea de Tecnologías para la inclusión - Programa Nacional de Tecnología e Innovación Social - financiado por el Ministerio de Ciencia, Tecnología e Innovación de la Nación (MINCYT). PROCODAS tiene como objetivo general promover la inclusión social mediante el desarrollo e implementación de tecnologías que mejoren la calidad de vida, fomentando las articulaciones entre el sector científico-tecnológico y las organizaciones sociales y territoriales. La Dra. en Informática Luciana Benotti es quien dirige el proyecto, cuyo núcleo asociativo está compuesto por el área de Computación de la Universidad Nacional de Córdoba, la entidad solicitante Asociación Civil Taller Productivo Amulen de Junín de los Andes, y la entidad participante y asociativa el Programa De Desarrollo Agroalimentario (PRODA) del Ministerio de Producción e Industria de la provincia del Neuquén, con la participación además de INTA ProHuerta y el apoyo del Programa Compromiso Social Estudiantil de la Universidad Nacional de Córdoba. Los modelos de aprendizaje automático que se propone integrar son innovadores en el contexto nacional e internacional y siguen en tema a recientes publicaciones de la Dra. Benotti [3] [2].

1.3 Miembros del proyecto

- Luciana Benotti es profesora asociada de la Universidad Nacional de Córdoba (UNC) en Argentina e investigadora de CONICET en el área de procesamiento de diálogo humano automático multimodal. Es miembro del comité editorial de revistas y congresos internacionales de Lingüística Computacional. Actualmente es la presidenta electa del capítulo norteamericano de la Asociación de Lingüística Computacional, donde representa a sus compañeros latinoamericanos y trabaja por la diversidad y

la inclusión reales. Participa del proyecto en el área tecnológica como experta de dominio en sistemas de diálogo.

- Andrea Properzi, Lic. en Servicio Social de la Universidad Nacional del Comahue. Técnica del Programa de Desarrollo Agroalimentario (PRODA) del Ministerio de Producción e Industria del Neuquén. Articula en áreas de procesos socio organizativos de lxs huerterxs e interinstitucionales. Participa en el proyecto aportando a la recopilación y ordenamiento de la información para la integración de las bases de datos, y en algunas actividades vinculadas a la difusión y participación.
- Mariana Benotti es ingeniera agrónoma de la Universidad Nacional del Comahue. Técnica del Programa de Desarrollo Agroalimentario (PRODA) del Ministerio de Producción e Industria del Neuquén. En el programa articula y capacita en espacios comunitarios sobre huerta agroecología y elaboración de cosmética natural en coordinación con otras instituciones, hospitales, escuelas y municipios Participa en el proyecto aportando datos locales de la zona de precordillera de la provincia de Neuquén, dónde trabaja hace más de 15 años.
- Fabiana Silgero es ingeniera agrónoma de la Universidad Nacional de Córdoba y trabaja para la secretaria de desarrollo territorial y ambiente de la provincia de Neuquén.
- Roberto Lacarpia es ingeniero agrónomo de la UBA y trabaja en el INTA prohuerta. Reparten semillas y asesora sobre huerta e invernaderos.
- Guido Ivetta es estudiante del último año de la licenciatura en Ciencias de la Computación de la Universidad Nacional de Córdoba. Trabaja como ayudante alumno y desarrolla su tesis de licenciatura en este proyecto. Realizó el entrenamiento y la recolección de datos de imágenes para los modelos de clasificación de insectos y residuos, desarrolló el manejo conversacional del bot, y entrenó los modelos de paráfrasis utilizados por el mismo.
- Maria Rosa Teti es maestra especial y presidenta del taller productivo Amulen de Junín de los Andes. Trabaja con chicos discapacitados en separación de residuos, reciclaje, compostaje de grandes generadores y huerta.
- Nair Carolina Mazzeo es Lic. en Comunicación Visual de la UNLa, huertera, estudiante apicultura y de Tecnicatura Forestal de la UNCO. Comparte la responsabilidad de la comunicación dentro del proyecto.
- Rodrigo Gómez Insausti es Téc. Superior en Publicidad del ISET 18 de Rosario, estudiante realizando su tesis de Ingeniería Ambiental, huertero, músico y creador de contenido audiovisual. Forma parte del equipo que está llevando a cabo la comunicación del proyecto.
- Lucía Martínez Gavier, estudiante del tercer año de la Licenciatura en Ciencias de la Computación en la UNC. Dentro del proyecto, comparte la responsabilidad de la

redacción de párrafos informativos en el área de insectos, generación de paráfrasis de los mismos para entrenar modelos, y finalmente evaluación de las paráfrasis generadas por dichos modelos. Con este proyecto, Lucía cumple con su responsabilidad del programa Compromiso Social Estudiantil.

- Diego Gimenez es un estudiante del tercer año de la carrera de Licenciatura en Ciencias de la Computación en la UNC. Participó en el área de generar los párrafos para la creación de paráfrasis en el área de información nutricional, entrenar modelos con las mismas y finalmente evaluar los resultados. Con este proyecto, Diego cumple con su responsabilidad del programa Compromiso Social Estudiantil.

1.4 El proyecto en sí: Inteligencia artificial aplicada a la huerta

El proyecto busca aportar tecnología digital brindando información útil para acompañar el proceso productivo de la huerta agroecológica familiar fomentando una buena y mejor calidad en la alimentación. Así el objetivo técnico de este proyecto - basado en casos de estudios para entrenamiento y evaluación de modelos de IA y procesamiento de lenguajes naturales, que clasifiquen automáticamente [27] imágenes e identificación de objetos - es el desarrollo de un chatbot de Telegram donde los huerterxs podrán compartir fotos, recibir recomendaciones y recordatorios, realizar consultas y obtener información útil para el desarrollo de una huerta agroecológica. El chatbot es un sistema interactivo por mensajería instantánea capaz de anexar un calendario de huerta considerando variables de la región particular (como temperatura, lluvias, suelo, etcétera) y dar recomendaciones personalizadas de actividades diarias, semanales y mensuales a realizar en una huerta. Además, y gracias a la información brindada, incorpora habilidades en responder preguntas sobre identificación de plantas, insectos benéficos o dañinos, y permite catalogar qué residuos se pueden reciclar o compostar usando clasificación automática de imágenes e identificación de objetos. Como objetivos específicos, el proyecto busca: Generar recomendaciones de acciones a realizar en la huerta (por ejemplo, regar o no regar) a partir de la combinación de diferentes variables. Para combinar estas variables se comparará el desempeño de distintos algoritmos de aprendizaje automático. Diseñar intenciones frecuentes y estrategias de diálogo en una plataforma de diseño y entrenamiento de chatbots de código libre (como RASA). En esta plataforma los chatbots se entrenan usando aprendizaje automatizado autosupervisado para sistemas de reconocimiento de la intención de las preguntas en español que se le puedan hacer al chatbot sobre actividades de huerta, adaptando las librerías para español disponibles en plataformas de código libre. Entrenar y evaluar el estado actual de modelos de paráfrasis, con los cuales se pueden ampliar nuestra batería de respuestas de forma en que el bot no responda una pregunta de la misma forma. Esto ayudará a la naturalidad de la conversación. Integrar un chatbot a Telegram que implemente y sea acorde a las funcionalidades que se diseñen en los dos objetivos anteriores, y evaluarlo con los huerteros. Este desarrollo tecnológico tiene como particularidad e innovación que considera los saberes locales con una perspectiva participativa para la construcción de las bases de datos que nutrirá ese chatbot. Para ello se están construyendo datos a partir de metodologías participativas que recuperan los saberes

locales (por ejemplo, a través de un concurso de fotos dirigido a lxs huerteros) y también de las experiencias territoriales de las instituciones que forman parte: INTA ProHuerta, PRODA y Amulen. El apoyo del Programa Compromiso Social Estudiantil de la Universidad Nacional de Córdoba, aporta más participación para la construcción y recolección de bases de datos.

1.5 El bot como interfaz

Se realizó una evaluación de datasets internacionales en conjunto con los datos obtenidos del concurso aumentados por webscrapping en área de insectos, plagas, compostaje y reciclaje, que serán utilizados para entrenar modelos de clasificación automática de imágenes e identificación de objetos. Se entrenaron diversos modelos de identificación de insectos y de clasificación de residuos compostables o reciclables. Se concluyó que los modelos entrenados utilizando tecnología proporcionada por la librería `fastai` otorgó los mejores resultados. El primer modelo entrenado utilizó 102 categorías de insectos tomadas del dataset `ip102`, logrando una precisión del 88%. Este dataset solo poseía algunos insectos de la región, por lo que una versión reducida compuesta sólo con ellos será entrenada posteriormente. Para la clasificación de residuos fue utilizado el `Garbage Classification Dataset` de Kaggle. Este dataset contiene imágenes divididas en categorías de vidrio, cartón, metal, papel, plástico y basura genérica. A partir de estos datos se entrenó un modelo de clasificación logrando una precisión del 84

La información sobre insectos de la zona, se re acomodó en formato excel. Se sistematizó información que es habitualmente requerida por interesados en realizar huertas, como la descripción física de los insectos y las formas de controlarlos o mantener el equilibrio biológico. Se decidió trabajar con el framework `RASA` para el desarrollo del chatbot multimodal. Se incorporó la funcionalidad de reconocer la intención identificación de insectos, generando el pedido y análisis de la imagen por el modelo de identificación entrenado en la actividad 3. Se integró el sistema de versionado del código del chatbot en Github, aportando técnicas de integración y entrega continua para corroborar las funcionalidades futuras mediante testings automáticos cada vez que se realice un cambio en el código.

En la Figura 1 se puede observar un extracto de conversación con el bot, en la cual el usuario busca identificar un insecto que está afectando el desarrollo de su huerta. Ya con el contexto de conversación establecida, el bot puede responder preguntas sobre este insecto a través de identificación de preguntas frecuentes en lenguaje natural.

En la Figura 2 se destaca el uso de paráfrasis para los casos en los que se instancia de nuevo una pregunta hecha en el pasado. La respuesta tiene el mismo significado pero al utilizar modelos de paráfrasis podemos generar una conversación más natural para el usuario.

En la Figura 3 el usuario busca identificar una planta a través de una foto, luego de ser identificada el bot ofrece a través de botones interactivos las diferentes funcionalidades con las que puede ayudar al desarrollo de esa planta específica. En este caso se solicita la ficha de cultivo de esta planta (detalle en la Figura 4), donde se señalan numerosos datos útiles para el cultivo tales como meses de siembra y cosecha, asociaciones con otras plantas para

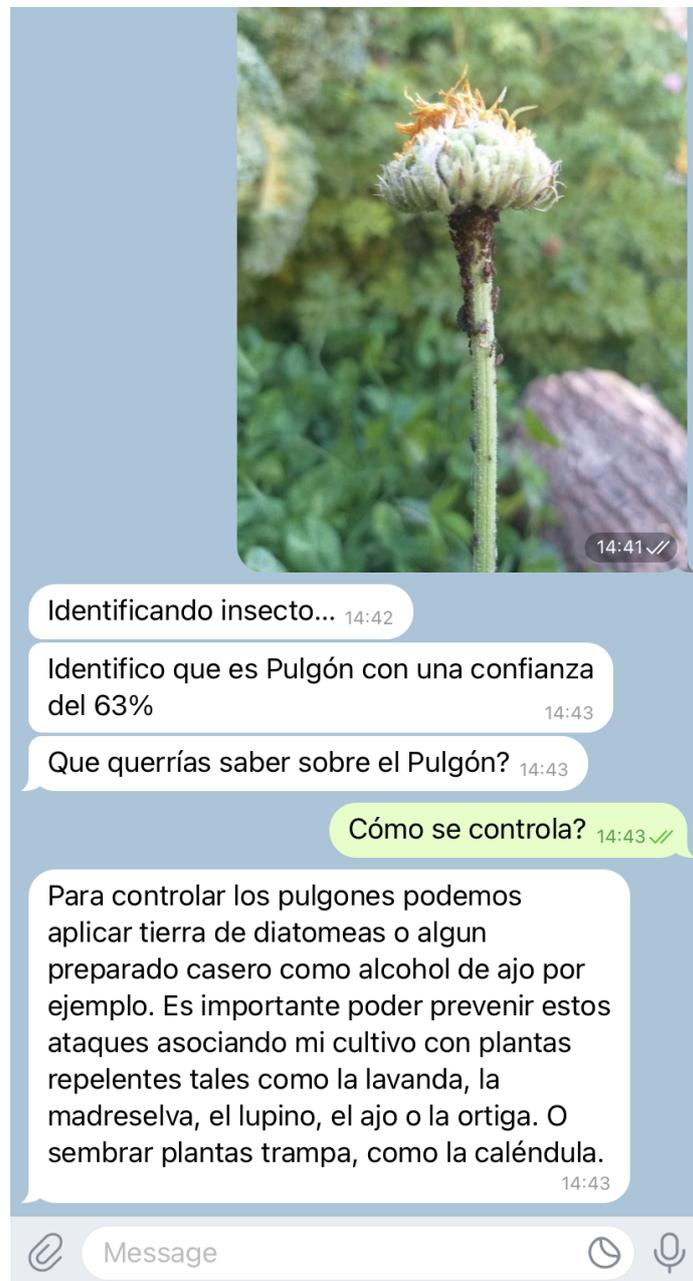


Figura 1: Captura de pantalla de la funcionalidad de identificación de insecto y posterior pregunta clasificada con el reconocimiento de intenciones de RASA

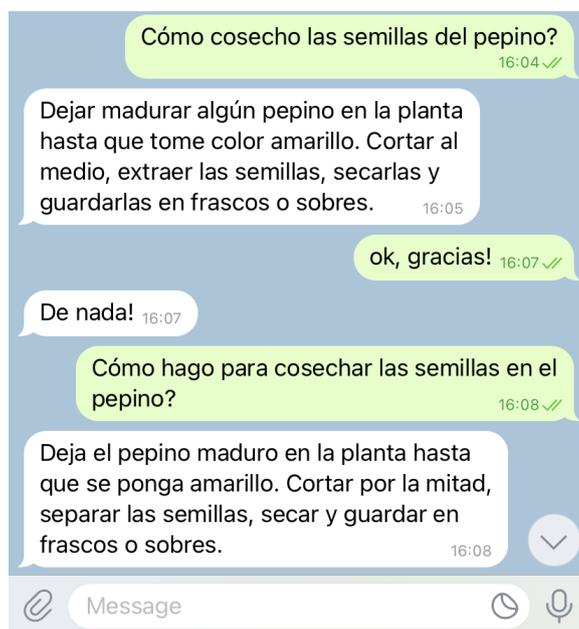


Figura 2: Captura de pantalla de la funcionalidad de paráfrasis al volver a preguntar el mismo interrogante.

incentivar la biodiversidad en la huerta, consejos para la cosecha de semillas, etcétera.

1.6 Novedades, relevancia y aspiraciones del proyecto

Este trabajo, como se mencionó, trata de una de las primeras experiencias de desarrollo de modelos de IA adaptados a la zona, con un fuerte componente de innovación tecnológica sobre procesamiento de lenguaje natural y de imágenes, desde una mirada descolonizadora. Sostiene desde su creación, una tecnología para y desde la comunidad, basada en metodologías participativas para la recolección de datos y la construcción de información. Nutre al chatbot como una herramienta informática accesible, participativa, interactiva, adaptada local, social y culturalmente, colaborando en el asesoramiento para que más personas puedan cultivar sus propios alimentos y mejoren así, el acceso a una buena nutrición. El proyecto se realiza en un proceso global con marcadas diferenciaciones sociales y un entramado de relaciones subordinadas entre países/regiones centro y las periferias, y es en esa diferenciación donde se encuentra la relevancia de este trabajo. La investigación y el desarrollo de tecnologías de IA actualmente está en manos de mega-industrias que tienen el poder económico para financiarlas. El conocimiento tecnológico está sobre todo en el norte global, occidentalizado, hegemónico, en países “del norte”, con un contexto totalmente distinto al “del sur”, argentino y latinoamericano; que está muy lejos de poder alcanzar el nivel de materialidad que tienen aquellas industrias. Pero además, desde dónde surge este proyecto: “un poco más al sur”, en un territorio alejado del epicentro de Buenos Aires,

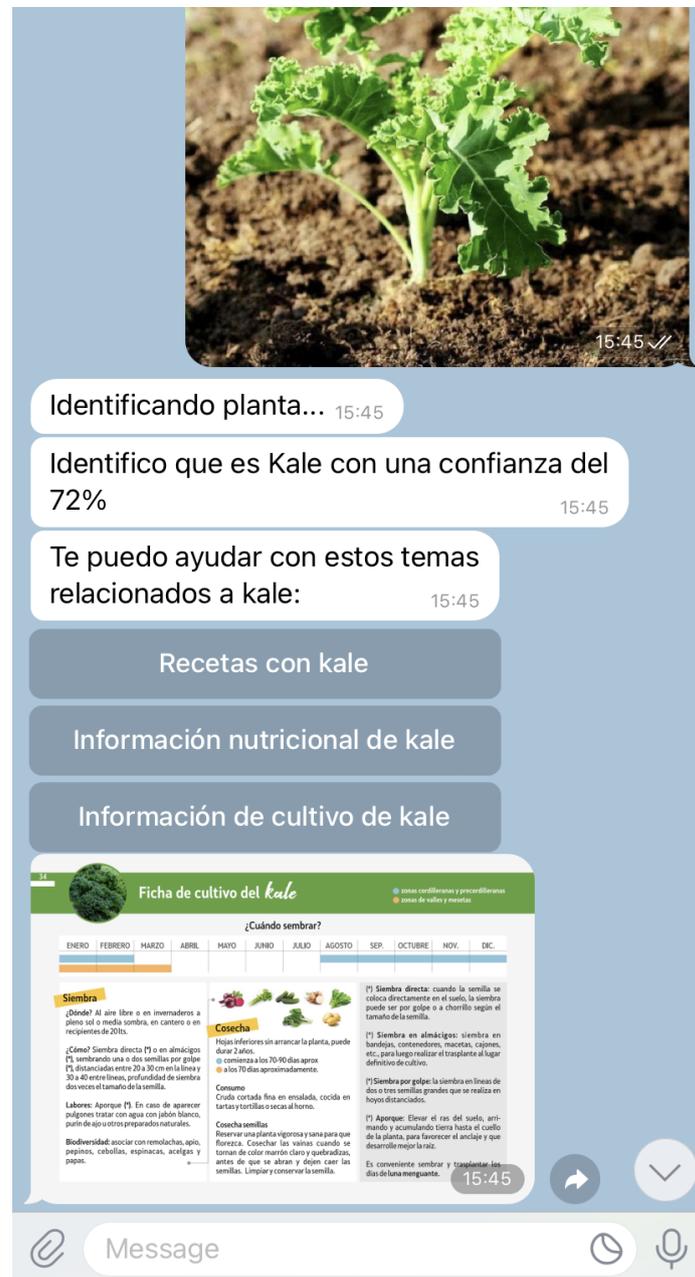


Figura 3: Captura de pantalla de la funcionalidad de identificación de planta y pedido de ficha de cultivo.



Figura 4: Ficha de cultivo de Kale, detalle de la conversación de la Figura 3.

con sus propias implicancias: la Patagonia, con diferentes distancias, realidades, diversidad intercultural, clima austral, dificultades en el acceso a internet y a buenas redes - equipos, entre otros factores que atraviesan directamente al proyecto. Donde además se complejiza sostener procesos de resistencia contrahegemónicos a las megas industrias y empresas, debido a la poca inversión y desfinanciación desde los gobiernos locales tanto en el desarrollo e investigación de modelos basados en IA, como en estrategias de valorización para expertxs locales en la temática. Se hace difícil sostener así comunidades científicas que tengan el perfil de poner la ciencia y tecnología al servicio del mejoramiento y cuidado de las vidas, es importante que “para construir una economía adecuada a los seres humanos, la producción se vincule al mantenimiento de la vida y al bienestar de las personas” [28]. Así, se podría decir que la relevancia de este trabajo fundamentalmente se basa en dos aspectos. Por un lado, que es un proyecto local para investigación y desarrollo de modelos basados en IA, que ha logrado cierto financiamiento y apoyo por parte del estado. Y por otro, en el cómo se inicia y se lleva a cabo: en la metodología y en la perspectiva de trabajo, sustentadas en un marco teórico de mirada descolonizadora y de descentralización de la tecnología. Donde profesionales y especialistas locales desarrollan tecnología adaptada al territorio, organizada desde la comunidad aspirando a una soberanía tecnológica: una transferencia de tecnología de aplicación directa, accesible, a través de un trabajo interdisciplinario y participativo, que busca como resultado también aportar a los procesos locales de construcción de soluciones que mejoren ni más ni menos que la calidad de vida de las poblaciones. Desde el

equipo interdisciplinario del proyecto, se da valor a “estimular formas de racionalidad que favorezcan relaciones de apoyo mutuo entre seres humanos y la tierra” supone entonces pensar “en marcos alternativos centrados en la ética de la reciprocidad, la democracia radical y la cooperación que involucren a todas las personas, tanto en el terreno de los derechos como en el de las obligaciones.” [20]. En este caso, generando una herramienta que pueda aportar a mejorar la alimentación, fortaleciendo las huertas agroecológicas, incentivando el autocultivo de alimentos locales y de estación, de manera segura y soberana.

1.7 Reflexión y desafíos de la informática

Es parte del mismo proceso de llevar adelante un proyecto, el hacer reflexiones sobre las prácticas, hacia dónde se va, las dificultades, los logros, las tensiones y contradicciones, entre otros aspectos. Se observa mucho de esto cuando se emprende la tarea de escribirlo para poder compartirlo. Y sobre todo cuando se hace de manera colectiva. Surgen preguntas vinculadas al campo que atraviesa el proyecto, poniendo en debate el acceso a la inteligencia artificial. Es necesario y urgente traer a las minorías a estas discusiones buscando otras formas de pensar estas tecnologías, fomentando la pluralidad de voces, de géneros, de culturas, de grupos etarios, de discapacidad, de multi versidades, así como también poner en discusión y en valor la diversidad geográfica, territorial y económica. Poner en discusión los modos de producción en diferentes ámbitos, desde los modelos de IA, la forma de alimentarse y de producir los alimentos, las metodologías de trabajo, entre otros, desde una mirada descolonial para fortalecer la tecnología inclusiva, soberana y adaptada a las necesidades de las poblaciones que la usan. Otro desafío para el campo tecnológico de la IA, sería poder focalizar en las inversiones públicas: cuántos proyectos se financian hoy día y bajo qué criterio, de qué temática tratan, quiénes lxs están desarrollando. ¿Por qué hay poca inversión de los gobiernos locales en la investigación en este tipo de tecnologías? ¿Se sigue dejando en manos del norte global? Es importante también reflexionar sobre quién investiga, cómo, y para qué. ¿Quién recolecta los datos y quién los valida? Para poner luz a las relaciones de poder y a las ciencias/paradigmas hegemónicos dominantes. Y en relación a la inclusividad ¿Cómo es el acceso a la tecnología? ¿Hay accesibilidad? ¿Conexión a internet, celulares con capacidad para usar aplicaciones como el Telegram? ¿Quiénes acceden y cómo? ¿Por qué aún, en ciertos ámbitos, hay desconocimiento sobre la utilización del campo tecnológico de la informática y la IA en particular como una herramienta que puede aportar al servicio de las comunidades? ¿Qué ocurre en el trayecto formativo de las carreras vinculadas al campo de la informática? ¿Hay una educación que promueva saberes y un ejercicio profesional para el beneficio de la sociedad toda, con una mirada crítica, reflexiva? ¿Se puede pensar en poner en debate la desmercantilización de las profesiones? Las preguntas pueden transformarse en desafíos. Los desafíos entonces quizás pueden ir oteando hacia el construir comunidad para seguir generando estos debates y discusiones; organización que dispute espacios de poder para ir encontrando nuevas formas de vincularse y de producción social, con nuevas lógicas. Es importante entonces, fortalecer y ampliar los espacios colectivos interdisciplinarios donde se reflexione críticamente sobre las prácticas, los movimientos de diversidad e inclusión y los comités de ética de la utilización/impacto de este tipo de tecnologías, como la asociación

de lingüística computacional. Disputar esos espacios en comunidad, para que otros mundos sean posibles.

1.8 Mapa de la tesis

Capítulo 1: “Introducción” En este capítulo se discute el contexto del proyecto, su surgimiento, y los desafíos analizados al inicio del mismo. Además se detallan los roles de los miembros del proyecto, las decisiones de interfaz de usuario y funcionalidades a proveer. Por último concluimos con algunas reflexiones y desafíos de la informática, en particular el rol que esta puede cumplir en temáticas de soberanía alimentaria y tecnológica.

Capítulo 3: “Sistematización de datos” Este proyecto tuvo su origen debido a la falta de sistematización de datos disponibles para desarrollar huertas orgánicas en las localidades de Junín y San Martín de los Andes. En este capítulo se exponen los diversos conjuntos de datos sistematizados por lxs ingenierxs del proyecto, así también por la comunidad de la región a través de participación voluntaria.

Capítulo 2: “Clasificando imágenes” Una de las principales funcionalidades que se proveen en nuestra herramienta es la clasificación automática de plantas, insectos y residuos. En este capítulo se analizan los procesos de búsqueda de datos, técnicas de entrenamiento de nuestros modelos, así también como la comparación y uso de modelos disponibles en el estado del arte actual.

Capítulo 4: “Manejo conversacional” La fluidez e intuitividad de la conversación con bots es integral para hacer un producto de calidad. En este capítulo se discuten conceptos fundamentales de los sistemas de diálogo automáticos orientados a tareas. Además se finaliza, detallando a través de diagramas, los distintos caminos posibles que pueden tomar los usuarios en la interacción con nuestra herramienta.

Capítulo 5: “Generación de respuestas” Construyendo a partir de los capítulos 2, 3 y 4, el último bloque necesario para completar un sistema de diálogo son las respuestas que produce. En este capítulo comentamos nuestra generación de fichas de cultivo a partir de los datos del capítulo 3 y desarrollamos un modelo de paráfrasis con el cual logramos que el sistema se comunique más naturalmente sin repetir de forma literal al lidiar con re-preguntas.

Capítulo 6: “Evaluación de paráfrasis” Medir la calidad y precisión de modelos de generación de texto no es una tarea fácil, es por esto que en este capítulo discutimos algunas técnicas de evaluación manual de paráfrasis y lo ponemos a prueba con nuestro modelo.

Capítulo 7: “Conclusiones” En este capítulo resumimos el trabajo realizado y volvemos a reflexionar sobre su propuesta con respecto a la soberanía tecnológica y alimentaria. Además, discutimos el largo camino que aún queda por recorrer incluyendo la evaluación de la interfaz chatbot completa con huerteros de la región.

2 Sistematización de datos

La recolección y generación de datos en este proyecto es un aspecto crucial como factor originador del mismo. Una de las principales razones por la cual se dio origen es que en la zona de Junín y San Martín de los Andes, las personas que tienen acceso a internet suelen hacer consultas sobre cómo producir sus huertas. Sin embargo las recomendaciones que encuentran, no suelen ser acordes a la zona por una cuestión climática, y hay prácticas puntuales que se deben hacer para tener éxito. La capacidad técnica a campo, sobre todo en pandemia, no alcanza a responder todas las dudas o llegar a varios lugares a la vez, observando además que muchas de las dudas y consultas se repiten y/o son frecuentes.

Es por esto que a continuación se detallarán todas las formas en la que en el proyecto se trató con la sistematización local de datos, tanto desde las ingenieras del proyecto como a aportes comunitarios y voluntarios de la zona.

2.1 Concurso “Mi Huerta en Casa”

Para extender nuestra recolección de datos, y principalmente para conocer cómo serían las imágenes que efectivamente serían enviadas al bot, se realizó un concurso destinado a la población de la zona de San Martín de los Andes, con distintas categorías y premios.

En todas las categorías se pidió una imagen en primer plano de la planta, identificar qué planta aparece en la imagen de entre 36 opciones (estas opciones fueron tomadas de los calendarios de huerta proporcionados por los ingenieros, más información en la siguiente sección de este Capítulo), la fecha aproximada en la que se tomó la imagen, dónde está ubicada la huerta, y una descripción en lenguaje natural de máximo 280 caracteres de los puntos claves de la imagen.

Las cuatro categorías y qué pregunta particular presentó cada una:

1. ¡Que orgullo me da mi planta! (buenas prácticas a recomendar):

Aquí se preguntó qué considera el usuario que hizo que su planta se desarrolle tan bien. Se podían seleccionar múltiples respuestas como “La sembré siguiendo el calendario de siembra”, “A la planta le llega bien la luz”, “Aplicué caldos nutritivos y preventivos”, etc. En la Figura 5 se puede ver la imagen compartida por los comunicadores del proyecto para informar y dar ejemplo para esta categoría.

2. Plantas que salen sin que yo las siembre:

En esta categoría se preguntó qué planta quería plantar originalmente el usuario. Se reutilizó las opciones para identificar la planta de la imagen común a todas las categorías.

Además, se preguntó qué hizo el usuario para que tu planta crezca igual. Las posibles respuestas fueron: “Removí las plantas no deseadas”, “Transplante la que si deseaba”, “Cubrí el suelo donde crecen las no deseadas con material seco”. En la Figura 5 se puede ver la imagen compartida por los comunicadores del proyecto para informar y dar ejemplo para esta categoría.



Figura 5: Imágenes publicadas por los comunicadores del proyecto como ejemplo para la primera (izquierda) y segunda (derecha) categoría.

3. ¡Mi planta está enferma!:

En este caso se preguntó qué le hizo notar al usuario que su planta se ha debilitado. Algunas de las opciones fueron “Hay manchas en las hojas y/o frutos”, “Cuenta con poco desarrollo”, “Se marchita / está seca. Se le caen las hojas”, etc.

Además, se preguntó qué hizo el usuario para levantarle un poco el ánimo a tu planta. Entre las opciones se encontraban “Aboné / Fertilizé”, “La transplanté a un sitio mejor”, “Puse cobertura en el suelo para reducir la sequedad, mejoré el riego”, etc. En la Figura 6 se puede ver la imagen compartida por los comunicadores del proyecto para informar y dar ejemplo para esta categoría.

4. ¡Los bichos se comen las plantas antes que yo!:

Aquí se preguntó qué bichos son los que están cosechando más rápido que vos? Las opciones fueron los insectos y pestes más comunes de la zona, como los pulgones, cochinillas, chinches, babosas, etc.

Además, se preguntó qué tácticas había realizado el usuario para disminuir el número de estos insectos y plagas. Entre las opciones se encontraban “Planté aromáticas”, “Usé tierra de diátomeas”, Introduje insectos benéficos que tengan la voluntad de comerse a los primeros”, etc. En la Figura 6 se puede ver la imagen compartida por los comunicadores del proyecto para informar y dar ejemplo para esta categoría.



Figura 6: Imágenes publicadas por los comunicadores del proyecto como ejemplo para la tercera (izquierda) y cuarta (derecha) categoría.

2.2 Calendarios de siembra

Al desarrollar cualquier tipo de huerta, es esencial realizar consultas de fechas de siembra, plantación y cosecha de huertas agroecológicas de la zona. Esta fue uno de los primeros puntapiés que revelaron la necesidad de una fuente confiable de información para huertas en la zona específica de San Martín y Junín de los Andes.

Los ingenieros del proyecto nos proporcionaron con varios calendarios de siembra y cosecha agrupados por zona, en la Figura 7 se observan las primeras filas de los datos generados por las ingenieras del proyecto, en el se observan los distintos estadios por los que pasan distintas plantas de la zona. Las siguientes son las referencias para su lectura:

- S: Siembra
- P: Plantación
- C: Cosecha
- SA: Siembra en Almacigo
- SM: Siembra en Maceta

Cultivo	enero	febrero	marzo	abril	mayo	junio	julio	agosto	septiembre	octubre	noviembre	diciembre	cultivo protegido (*)	cultivo al aire libre
1 Acelga	S/C	S/C	C	C	C	C	C	S/C	S/C	S/C	S/C	S/C	si	si
2 Achicoria/ Radicheta	S/C	S/C	S/C	C	C	no se siembra porque el suelo esta muy frio	no se siembra porque el suelo esta muy frio	S/C	S/C	S/C	S/C	S/C	si	si
3 Albahaca	C	C	C	C	C	no se siembra porque el suelo esta muy frio	no se siembra porque el suelo esta muy frio	no se siembra porque el suelo esta muy frio	S	S	S	C	si	no
4 Ajo	C	C	P	P	P	no se siembra porque el suelo esta muy frio	no se siembra porque el suelo esta muy frio	es tarde para sembrar, cultivo en fase de crecimiento	es tarde para sembrar, cultivo en fase de crecimiento	es tarde para sembrar, cultivo en fase de crecimiento	es tarde para sembrar, cultivo en fase de crecimiento	C	no	si

Figura 7: Sección del calendario de siembra proporcionado por las ingenieras del proyecto. S: Siembra, P: Plantación, C: Cosecha, SA: Siembra en Almacigo, SM: Siembra en Maceta

2.3 Fichas de cultivo

Por último, una de las tareas más importantes que realizaron las ingenieras del proyecto fue la generación de información de cultivo para más de 40 plantas de la zona. Esta información es de alta calidad y focalizada en la región para la cual se desarrolla este proyecto. Es necesario valorar estos datos ya que fueron la razón por la cual se originó el proyecto y por los cuales los usuarios de nuestra herramienta serán capaces de recibir recomendaciones acordes a sus huertas y a su clima, tarea hasta ahora no satisfecha por búsquedas convencionales.

En las fichas de cultivo se abarcarán los siguientes ocho interrogantes para el cultivo de plantas:

- **¿Cuándo sembrar?**

Aquí se establece el período de siembra en meses. Además se realizan aclaraciones con respecto a cómo lidiar con heladas.

- **¿Dónde se siembra?**

Aquí se describen los recipientes o ubicaciones de siembra, por ejemplo si es al aire libre, en invernadero, en canteros, macetas, etc.

- **¿Cómo se siembra?**

Aquí se describen algunas características de la siembra, por ejemplo si se hace en siembra directa, por golpe, a chorrillo, si es necesario realizar pozos y la necesidad de transplantes.

- **Labores, Cuidados**

En esta categoría se describen las tareas de desyuyado, frecuencia de riego, aplicación de abono, realización de podas, etc.

- **Biodiversidad / Asociaciones**

Aquí se realizan recomendaciones de otras plantas que, al ser cultivadas a próxima distancia, se consiguen beneficios como ser atrayente o repulsor de ciertos insectos o mejoras en el transcurso de la siembra.

- **¿Cuándo cosechar?**

Aquí se establece el período de cosecha en meses. Además se realizan aclaraciones con respecto a si se cosechan distintas partes de la planta en distintos momentos del año.

- **Consumo**

Aquí se realizan algunas recomendaciones de formas de consumo de la planta, desde si se puede consumir fresca o es necesario cocción hasta qué partes de la planta son aptas para consumo humano.

- **Cosecha de semillas**

Aquí se proveen algunas recomendaciones para conseguir las mejores semillas posibles de nuestras plantas. Desde reservar algunas plantas vigorosas para que produzcan la vara floral hasta dejar madurar frutos hasta que sus semillas se encuentren óptimas para su cosecha.

A partir de estos datos se generó desde el proyecto un infográfico por cada planta explicando de forma gráfica cada uno de estos aspectos. Fue una decisión conjunta con los miembros del área de comunicación del proyecto, donde expresaron la necesidad de mostrar esta información de forma sintética y amigable para captar la atención de los usuarios.

3 Clasificando imágenes

Una de las principales funcionalidades que se proveen en nuestra herramienta es la clasificación automática de plantas, insectos y residuos. En este capítulo se analizan los procesos de búsqueda de datos, técnicas de entrenamiento de nuestros modelos, así también como la comparación y uso de modelos disponibles en el estado del arte actual.

En la primera sección, brevemente introducimos el concepto de aprendizaje por transferencia y discutimos la relevancia para nuestro proyecto. En las siguientes secciones describe el trabajo de clasificación de imágenes según la identificación de plantas, enfermedades, insectos y residuos.

3.1 Aprendizaje por transferencia

La clasificación de imágenes es la tarea de asociar una o más etiquetas a una imagen determinada. Utilizamos una técnica llamada Transfer Learning en inglés y aprendizaje por transferencia en español. En ella se utiliza un modelo entrenado en un conjunto de datos grande (como ImageNet [13]) y luego se adapta a nuestro propio conjunto de datos. Este modelo ha aprendido a reconocer características en todos estos datos, y nos beneficiamos de este conocimiento, especialmente si nuestro conjunto de datos es pequeño, en comparación con comenzar de un modelo con pesos inicializados al azar. Para lograrlo, nos apoyamos en funciones y documentación de la librería `fastai` [21]. `Fastai` es un grupo de investigación sin ánimo de lucro centrado en Deep Learning y la inteligencia artificial.

Para crear el modelo de Transfer Learning, utilizamos la función `cnn_learner` que toma los datos de entrenamiento para el dominio de destino, la red de base entrenada en el dominio de origen y las métricas. Las métricas sólo se utilizan para imprimir el rendimiento del entrenamiento. Decidimos usar la red preentrenada `resnet34`, esta es una red neuronal convolucional de 34 capas que puede utilizarse como modelo de clasificación de imágenes de última generación. Se trata de un modelo preentrenado en el conjunto de datos ImageNet, que contiene más de 100.000 imágenes de 200 clases diferentes. Decidimos usarla ya que `fastai` la recomienda especialmente para tareas de clasificación de imágenes.

3.2 Identificación de enfermedades de plantas

Se trabajó con el dataset de `PlantVillage`, que recopila imágenes de distintas plantas de huerta con enfermedades en sus hojas. En la Figura 8 se ven algunos ejemplos de las diferentes clases de recomendación del dataset de `PlantVillage`.

Este dataset de 54.309 imágenes abarca 14 especies de cultivos: Manzana, arándano, cereza, uva, naranja, melocotón, pimiento, patata, frambuesa, soja, calabaza, fresa y tomate. Además contiene imágenes de 17 enfermedades fúngicas, 4 enfermedades bacterianas, 2 enfermedades de moho (oomicetos), 2 enfermedades víricas y 1 enfermedad causada por ácaros. 12 especies de cultivos también tienen imágenes de hojas sanas que no están visiblemente afectadas por una enfermedad. Se intentará clasificar las imágenes en una de las clases objetivo, es decir, alguna combinación posible del producto cartesiano entre el

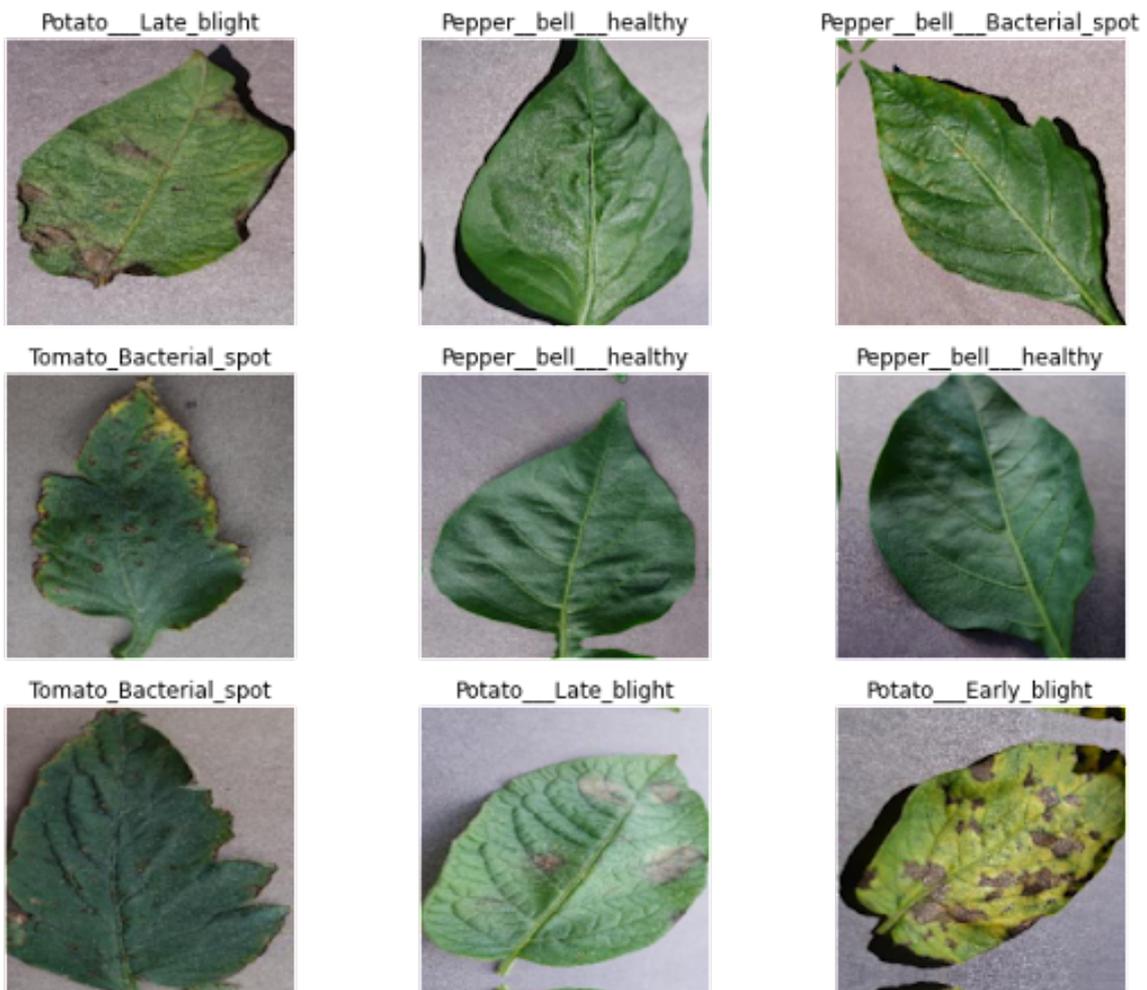


Figura 8: Muestra de algunas categorías del dataset de PlantVillage con sus respectivas etiquetas. Este dataset de 54.309 imágenes abarca 14 especies de cultivos: Manzana, arándano, cereza, uva, naranja, melocotón, pimiento, patata, frambuesa, soja, calabaza, fresa y tomate. Además contiene imágenes de 17 enfermedades fúngicas, 4 enfermedades bacterianas, 2 enfermedades de mohos (oomicetos), 2 enfermedades víricas y 1 enfermedad causada por ácaros.

```
model.fit_one_cycle(5)
```

epoch	train_loss	valid_loss	accuracy	error_rate	time
0	0.572705	0.063141	0.979568	0.020432	19:21
1	0.215012	0.034071	0.987711	0.012289	01:50
2	0.114522	0.028422	0.990228	0.009772	01:49
3	0.065896	0.014471	0.995410	0.004590	01:48
4	0.046472	0.013936	0.995706	0.004294	01:49

Figura 9: Estadísticas de entrenamiento a lo largo de cinco epochs.

conjunto de plantas y el conjunto de enfermedades, es decir cada categoría es de la forma nombreDePlanta_NombreDeEnfermedad.

Esta forma de entrenamiento logró resultados favorables. Como podemos ver en la Figura 9, ejecutando sólo cinco epochs, nuestra precisión para esta tarea de clasificación es de alrededor de 99,57%.

En la Figura 10, la matriz de confusión se pueden observar pocos errores, con la mayoría de las predicciones en la diagonal de la matriz. Notar que de 997 imágenes de planta de pimiento con Bacterial Spot se clasificaron correctamente 993, mientras que las cuatro restantes fueron clasificadas correctamente como planta de pimiento, pero incorrectamente como saludables. Resultados similares se pueden observar con los otros grupos de clasificación.

Luego de debatir en el grupo de investigación, llegamos a la conclusión de que sería difícil realizar identificación de enfermedades a partir de imágenes ya que los propios ingenieros no podrían hacerlo ellos mismos. Es decir el acuerdo entre expertos es bajo si sólo se cuenta con la imagen de una hoja. Es por esto que el problema se fue de a poco redefiniendo hasta llegar a algo tan útil en la práctica como posible, lo cual en este caso nos alejó de cómo la tarea se define en los proyectos de investigación.

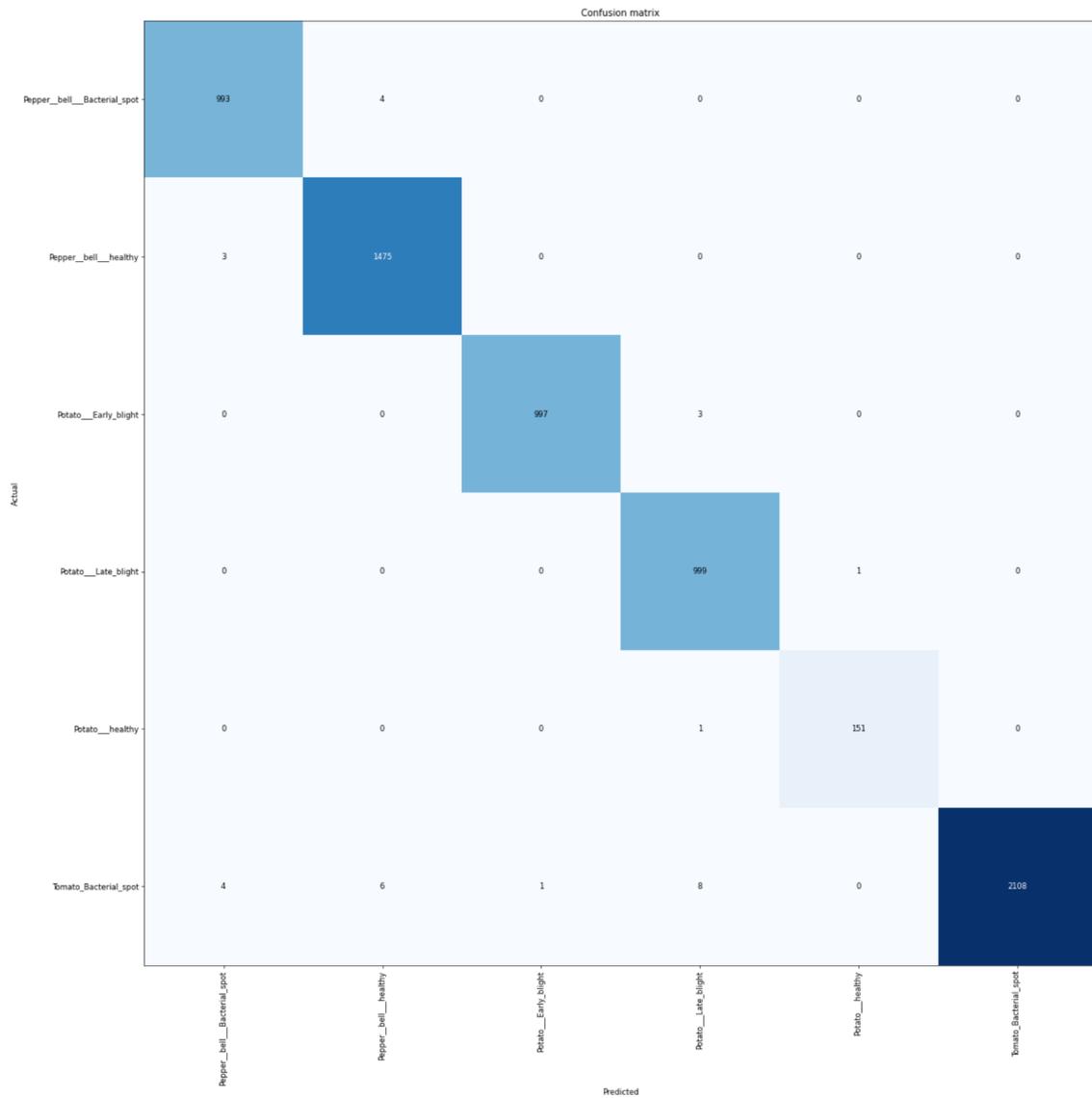


Figura 10: Matriz de confusión del modelo entrenado para clasificar plantas y enfermedades.

3.3 Identificación de plantas

La identificación automática de la especie de una planta es una tarea que últimamente ha tenido éxito por su precisión y utilidad a través de aplicaciones como PictureThis [17]. Luego de una extensa búsqueda de datasets nacionales e internacionales, no logramos conseguir suficientes datos para entrenar un modelo competitivo frente al estado del arte actual. Es por esto que decidimos utilizar los recursos proporcionados por el proyecto Pl@ntNet [19]. Pl@ntNet es un sistema de identificación que ayuda a identificar las plantas a través de imágenes. Se trata de un proyecto de investigación y ciencia ciudadana, apoyado inicialmente por la Fundación Agropolis, y desarrollado desde 2009 en el marco de un consorcio que reúne al Cirad, el INRA, el Inria y el IRD. Es de código abierto sin fines de lucro con años de trayectoria y financiamiento de EU.

Las herramientas proporcionadas por Pl@ntNet están disponibles de forma gratuita de libre acceso, sin embargo hay un límite de 500 accesos diarios. Nos pusimos en contacto con esta organización y debido a los objetivos y el modo de financiamiento de nuestro proyecto, se ofrecieron a elevar este límite para nuestros accesos para ayudar al desarrollo de nuestro asistente.

3.4 Identificación de insectos

El problema de la identificación de insectos dentro de una huerta es de gran utilidad para los usuarios, ya que nuestro sistema podría no solo ayudar en un mejor entendimiento del estado de la huerta sino también dando métodos para combatirlos. Se comenzó la búsqueda de datos con el dataset [42] IP102, que provee más de 75000 imágenes de insectos y plagas de huertas, clasificadas en 102 categorías de insectos que afectan el sudeste asiático. A partir de estos datos, se realizó el entrenamiento del modelo de clasificación con fastai (ver primer enfoque de los antecedentes), arrojando prometedores resultados a juzgar por el siguiente confusion matrix. Al ser 102 categorías, para lograr una correcta visualización sólo se muestra el comportamiento de las primeras 10. Sin embargo, el resto de la matriz de confusión se comporta de una manera similar.

En la Figura 11 se observa que la mayoría de las predicciones se encuentran en la diagonal de la matriz, indicando buenos resultados. Sin embargo, existen casos como la *Cicadella viridis* donde se predijo correctamente en 90 casos de los 127 casos de test a los que fue sometida. Indicando una precisión de alrededor del 70

Luego de mostrar avances en el grupo de investigación, se concluyó que lamentablemente solo unos pocos de los insectos presentes en el dataset efectivamente habitan en la zona de Junín de los Andes. De esta forma, sería necesario conseguir el resto de los datos de otro lado. Se encontraron diversos datasets específicos de insectos que sí están presentes en Neuquén, como es el caso de la mosca blanca [40], pulgones [30], etc. Además, se utilizaron técnicas de web scraping para obtener más imágenes de insectos para los que no se encontraron datasets. De esta forma logramos encontrar imágenes de entrenamiento para los diez siguientes insectos frecuentemente encontrados en huertas orgánicas de la zona de Junín y San Martín de los Andes:

Adrityrannus	22	0	2	0	0	0	0	0	0	0	0	0	0	0	0
Aleurocanthus_spiniferus	0	56	0	0	0	0	0	0	0	0	1	0	1	0	0
Ampelophaga	0	0	51	0	0	0	1	0	0	0	0	0	0	2	0
Aphis_citricola_Vander_Goot	0	0	0	13	0	0	0	0	0	0	0	0	1	0	0
Apolygus_lucorum	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0
Bactrocera_tsuneonis	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
Beet_spot_flies	0	2	0	0	0	0	14	0	0	0	0	0	0	0	0
Brevipalpus_lewisi_McGregor	0	0	0	0	1	0	0	2	0	0	0	0	0	0	0
Ceroplastes_rubens	0	0	0	0	0	0	0	0	19	0	1	0	0	0	0
Chlumetia_transversa	2	0	1	0	0	0	0	0	0	7	0	0	0	0	0
Chrysomphalus_aonidum	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
Cicadella_viridis	0	0	0	0	0	0	0	0	0	0	90	22	0	0	0
Cicadellidae	0	0	0	0	0	0	0	0	0	0	17	49	0	0	0
Colomerus_vitis	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6

Figura 11: Sección de la matriz de confusión para el modelo entrenado para clasificar insectos del dataset IP102.

- Babosa
- Babosita del Peral
- Caracol
- Chinche Verde
- Cochinilla
- Hormiga
- Mosca Blanca
- Oruga
- Pulgón
- Tijereta

En la Figura 12 se muestran algunos ejemplos de las imágenes de nuestro conjunto de datos de entrenamiento con sus respectivas etiquetas.

Utilizando las técnicas de entrenamiento mencionadas al inicio de este capítulo, entrenamos nuestro modelo por 8 épocas ya que la tasa de error comenzó a estancarse. La evolución de esta puede ser observada en la Figura 13, logrando una precisión del alrededor del 85% al evaluar el modelo con casos de prueba que nunca había visto con anterioridad.



Figura 12: Muestra de algunas categorías de nuestro dataset para clasificación de insectos con sus respectivas etiquetas.

Evolución de la tasa de error en modelo de clasificación de insectos

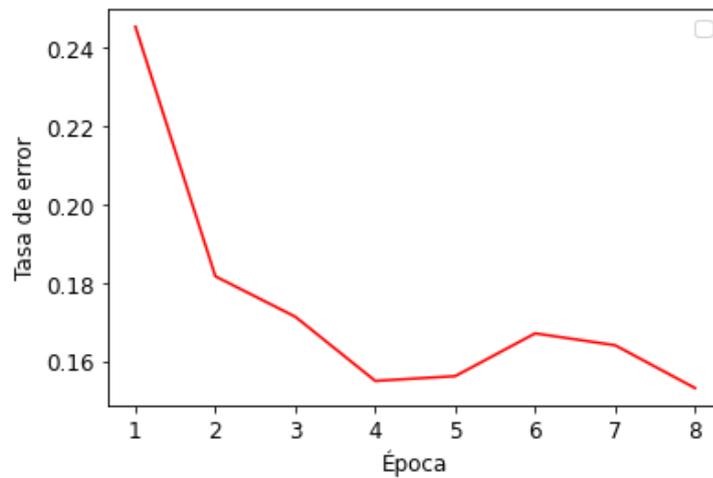


Figura 13: Evolución de la tasa de error para el modelo final de clasificación de insectos.

En la Figura 14 se muestra la matriz de confusión para nuestro modelo. Aquí hay algunos detalles a mencionar. Principalmente notar la mayor pronunciación de resultados sobre la diagonal de la matriz, implicando un buen funcionamiento del modelo. Por otro lado, evaluar el tipo de errores que produce el modelo. Por ejemplo en el caso del caracol se observa que la mayoría de confusiones son al clasificar erróneamente como babosa y viceversa, ya que son dos insectos con textura, tamaño y viscosidad similar. También es el caso de la hormiga y la tijereta al tener estructuras de cuerpo parecidas.

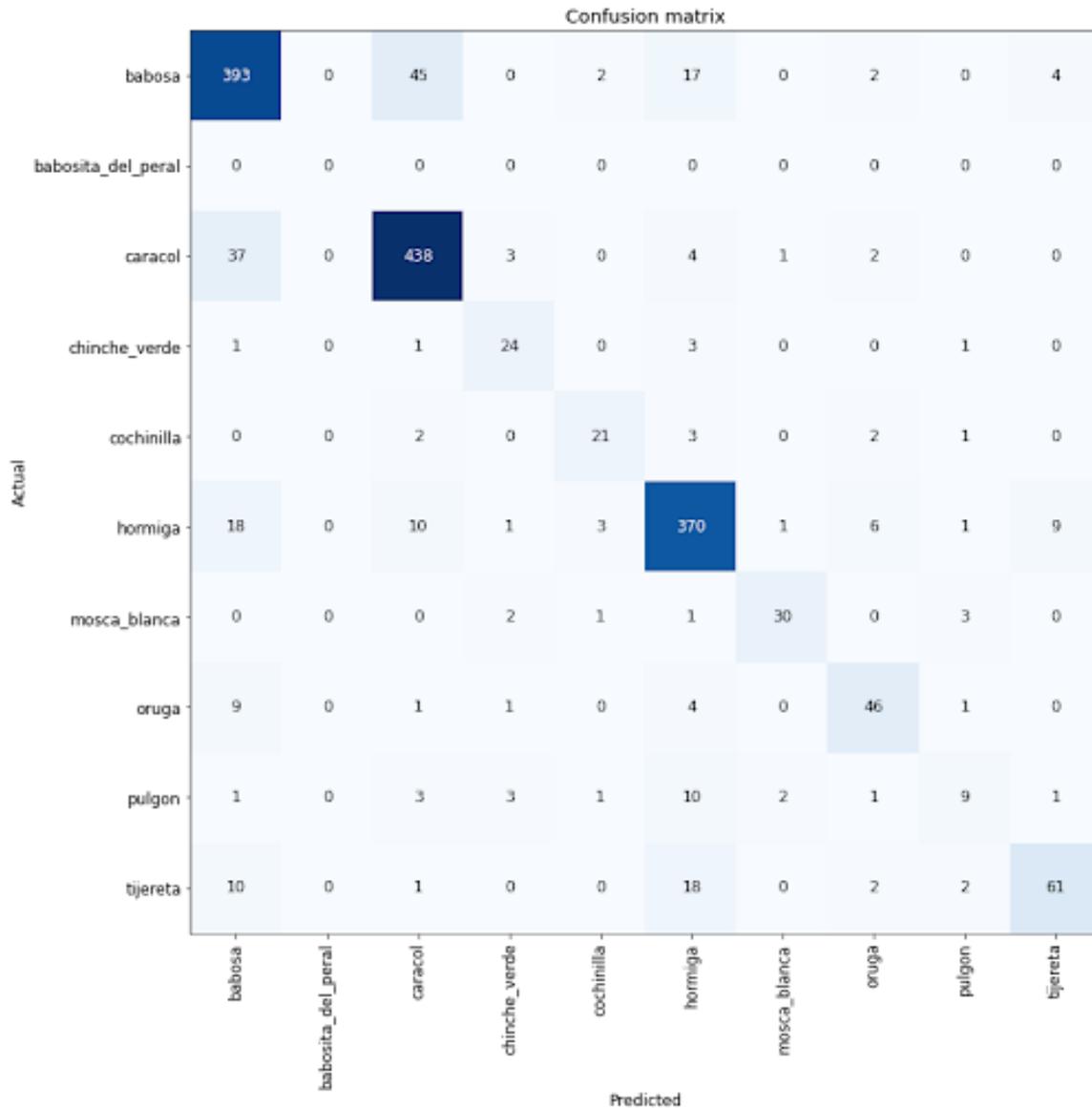


Figura 14: Matriz de confusión del modelo entrenado para clasificar insectos.

3.5 Identificación de residuos reciclables y compostables

El problema de la clasificación de residuos dentro de una huerta es de utilidad para los usuarios, ya que nuestro sistema podría no solo ayudar en un mejor entendimiento de generar compost sino también de reducir la cantidad de residuos generados. Luego de una evaluación de datasets disponibles públicamente, decidimos combinar todos los que conseguimos. No todos tenían las mismas clases de clasificación, pero decidimos seguir el modelo de clasificación del Garbage Classification Dataset [8], que divide a sus imágenes en las categorías de Cartón, Vidrio, Metal, Papel, plástico, y no reciclable. Esta decisión fue resultado de consultas a los ingenieros del proyecto con su experiencia de las preguntas más frecuentes que se realizan en el Taller Amulén.

Algunos conjuntos de datos como RecycleEye [16] eran muy específicos en su clasificación, así que agrupamos esas clases en las superclases más generales del Garbage Classification Dataset.

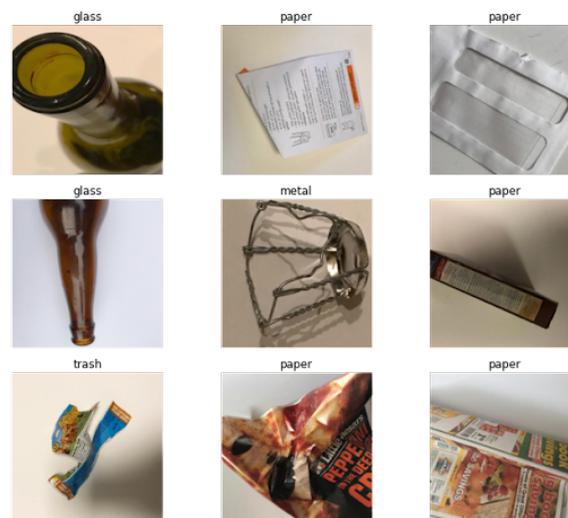


Figura 15: Muestra de algunas categorías de nuestro dataset para clasificación de residuos con sus respectivas etiquetas.

En la Figura 15 se muestran algunos ejemplos de las imágenes de nuestro conjunto de datos de entrenamiento con sus respectivas etiquetas. Utilizando las técnicas de entrenamiento mencionadas anteriormente entrenamos nuestro modelo por 8 épocas ya que la tasa de error comenzó a estancarse. La evolución de esta puede ser observada en la Figura 16, logrando una precisión del alrededor del 82% al evaluar el modelo con casos de prueba que nunca había visto con anterioridad. En la Figura 17 se muestra la matriz de confusión para nuestro modelo. Además de notar la correcta clasificación al estar marcada la diagonal de la matriz, podemos observar que confunde frecuentemente el vidrio por metal o plástico y comete gran cantidad de errores al determinar que en una imagen hay residuos en general no reciclables denominada la clase "trash".

Evolución de la tasa de error en modelo de clasificación de residuos

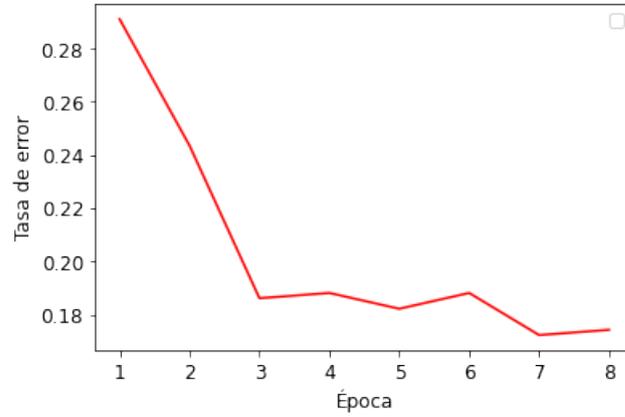


Figura 16: Evolución de la tasa de error para el modelo final de clasificación de residuos.

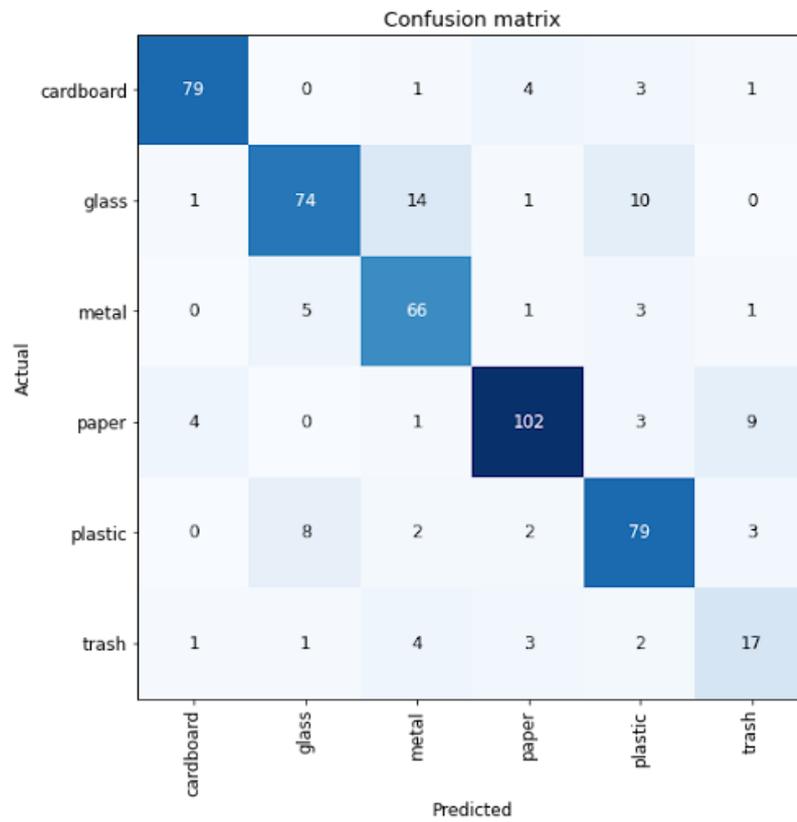


Figura 17: Matriz de confusión del modelo entrenado para clasificar residuos.

4 Manejo conversacional

En el capítulo anterior se describieron en detalle los modelos de clasificación de imágenes que forman parte de las funcionalidades del asistente automático. El siguiente paso en el desarrollo del mismo es la interfaz de usuario que permita la utilización de estos modelos, posibilitando al usuario a seguir los hilos conversacionales que espera, además de permitirle recordar datos previamente mencionados para dar soluciones a partir del contexto. El área de investigación sobre chatbots se conoce como sistemas de diálogo y es una de las áreas más antiguas pero también más desafiantes de la inteligencia artificial. Aún hoy en día el área cuenta con muchos problemas abiertos.

El lenguaje es esencial para el ser humano. La conversación y el diálogo es el tipo de lenguaje que utilizamos habitualmente para comprar comida, participar en reuniones de negocios, hablar con nuestras familias, reservar vuelos, e informarnos sobre el estado del tiempo. Lamentablemente, no es tarea fácil incorporar esta natural manera de compartir información a sistemas automáticos. En este capítulo se introducen los algoritmos fundamentales de sistemas de diálogo, agentes conversacionales, y agentes de diálogo orientados a tareas. Así también como técnicas de evaluación y arquitectura de diseño de los mismos. En particular, para el desarrollo de nuestro proyecto nos focalizaremos en agentes de diálogo orientados a tareas. Estos no buscan imitar las conversaciones desestructuradas entre humanos, sino ayudar a realizar tareas específicas con fines prácticos de forma conversacional.

A continuación se hará una breve introducción a los conceptos fundamentales de la conversación humana que los sistemas de diálogo intentan emular mediante una interpretación de [23] en el contexto de esta tesis. Luego se describirán las características y desafíos del estado del arte de los sistemas de diálogo orientados a tareas. Posteriormente se presentará el framework en el cual implementamos nuestro sistema de diálogo orientado a tareas llamado RASA. Finalmente se expondrán funcionalidades y diagramas de gestión de diálogo de nuestro chatbot.

4.1 Propiedades de la conversación humana

Jurafsky y Martin [23] sostienen que antes de diseñar un agente conversacional que converse con los seres humanos, es fundamental entender cómo conversan los humanos entre sí. Nos pareció particularmente interesante el concepto de *grounding*. Un diálogo no es sólo una serie de actos de habla independientes, sino un acto colectivo realizado por el hablante y el oyente. Como todo acto colectivo, es importante que los participantes establezcan en lo que ambos acuerdan, lo que se llama el terreno común. Para ello, los hablantes se apoyan mutuamente en sus intervenciones. El *grounding* significa reconocer que el oyente ha entendido al hablante. Por otra parte, el concepto de *inferencia conversacional* será de suma importancia para el desarrollo de nuestro sistema. El hablante parece esperar que el oyente haga ciertas inferencias; en otras palabras, el hablante comunica más información de la que parece estar presente en las palabras pronunciadas. Por ejemplo, si nuestro sistema recibe la oración “Hay insectos en mi huerta”, debemos inferir que el usuario desea informarse sobre estos insectos, y en el caso de no ser deseados proveer métodos de erradicación.

4.2 Sistemas de diálogo orientados a tareas

En esta sección se presenta la arquitectura GUS [4], diseñada para el diálogo orientados a tareas. Todos los sistemas de diálogo modernos orientados a tareas, ya sea la sencilla arquitectura GUS, o aquellas más sofisticadas que describiremos luego, se basan en frames. Un frame es un tipo de estructura de conocimiento que representa los tipos de intenciones que el sistema puede extraer de las frases del usuario, y consiste en una colección de slots, cada uno de los cuales puede tomar un conjunto de posibles valores posibles. Este conjunto de frames se denomina a veces ontología del dominio. El objetivo del sistema es llenar los espacios del frame con los slots que el usuario desea, y luego realizar la acción pertinente para el usuario (responder a una pregunta, reservar un vuelo, etc). Para ello, el sistema hace preguntas al usuario utilizando plantillas de preguntas preestablecidas a cada slot de cada frame. En primer lugar, será necesario clasificar el dominio del problema a resolver: ¿este usuario está hablando de vuelos, programando un despertador, o consultando su calendario? Luego debemos determinar la intención: ¿qué tarea u objetivo general está tratando de lograr el usuario?. Por último debemos llenar los slots asociados: necesitamos extraer los slots particulares que el usuario pretende que el sistema entienda de su enunciado con respecto a su intención.

De estas tareas, nos centraremos en la parte de llenar los slots asociados. Decidimos tomar esta decisión debido a que usaremos la plataforma de Rasa [5] para desarrollar el chatbot. En esta plataforma, se deben especificar los slots apropiados de forma precisa y delicada para incrementar al máximo la capacidad de clasificación y llenado de los mismos. Comprender en mayor profundidad su funcionamiento interno nos será de ayuda para usarlos correctamente.

4.2.1 Interpretación de la intención y slot filling

La tarea de llenar slots y clasificación de dominios e intenciones, son casos especiales de la tarea de análisis semántico supervisado, en el que en el que tenemos un conjunto de datos de entrenamiento que asocia cada frase con el conjunto correcto de slots, dominio e intención. Un método sencillo consiste en entrenar un modelo de secuencia para asignar la representación de las palabras de entrada a los rellenos de slots, el dominio y la intención. Debemos calcular una representación de una frase, pasando la frase por un contextual embedding network como BERT [14]. La representación de la frase resultante puede pasar por una capa feedforward y luego por un simple clasificador 1-de-N para determinar su dominio e intención. Una vez que el etiquetador de secuencias ha etiquetado la expresión del usuario, se puede extraer una cadena de relleno para cada slot de las etiquetas y estas cadenas de palabras pueden ser normalizadas a la forma correcta en la ontología. En contextos industriales, los sistemas basados en el aprendizaje automático para el llenado de slots suelen partir de sistemas basados en reglas del tipo GUS de una manera de aprendizaje semisupervisado. En primer lugar, se crea un sistema basado en reglas para el dominio y se etiqueta cuidadosamente un conjunto de pruebas. A medida que llegan nuevas expresiones de los usuarios, se emparejan con el etiquetado proporcionado de las reglas para crear

tuplas de entrenamiento. A continuación, se puede entrenar un clasificador de prueba para comprobar el rendimiento del clasificador frente al sistema basado en reglas.

4.2.2 Generación de lenguaje natural

Una vez decidido un acto de diálogo, hay que generar el texto de la respuesta al usuario. La tarea de generación de lenguaje natural en esta arquitectura se suele modelar en dos etapas: la planificación del contenido (qué decir) y la realización de la frase (cómo decirlo). Aquí supondremos que se ha elegido el acto de diálogo que va a generar, y se han clasificado y llenado algunos atributos (slots y valores) que el planificador quiere decir al usuario (ya sea para darle la respuesta, o como parte de una estrategia de confirmación). Los datos de entrenamiento son difíciles de conseguir; es poco probable que veamos todas las combinaciones posibles de atributos, valores y redacciones de oraciones. Por ello, en la realización de frases es habitual aumentar la generalidad de los ejemplos de entrenamiento mediante la deslexicalización. La deslexicalización es el proceso de reemplazar palabras específicas en el conjunto de entrenamiento que representan valores de ranura con un token genérico de marcador de posición que representa el slot. El mapeo de los frames a las frases deslexicalizadas se realiza generalmente mediante modelos de codificación-decodificación.

La entrada al codificador es una secuencia de tokens que representan el acto de diálogo y sus argumentos. El codificador lee todas las representaciones de slot/valor, y el decodificador emite una frase deslexicalizada en lenguaje natural. Por último, podemos utilizar el frame de entrada del planificador de contenidos para relexicalizar. Por ejemplo una frase deslexicalizada puede ser “Los nombre_de_insecto suelen atacar a las plantas de tomate”. Esta frase luego será relexicalizada para obtener “Los pulgones suelen atacar a las plantas de tomate”.

4.2.3 Evaluación de los sistemas de diálogo

En esta tesis no se expondrán los resultados de la evaluación de nuestro sistema de diálogo ya que esta tarea no se realizó todavía. Sin embargo daremos una introducción al tema ya que al final del proyecto de investigación dentro del cual se enmarca esta tesis haremos una evaluación presencial del sistema con usuarios reales de nuestro bot.

En el caso del diálogo orientados a tareas, si la tarea no es ambigua, podemos medir simplemente el éxito absoluto de la tarea. Para tener una idea más precisa, podemos calcular un índice de satisfacción del usuario, haciendo que interactúen con un sistema de diálogo para realizar una tarea y luego haciéndoles rellenar un cuestionario. Basándonos en [41], algunas preguntas posibles para realizar en el cuestionario son:

- ¿Fue fácil de entender el sistema?
- ¿El sistema entendió lo que has dicho?
- ¿Fue fácil encontrar el mensaje/vuelo/tren que quería?
- ¿El ritmo de interacción con el sistema fue el adecuado?

- ¿Supo qué se puede decir en cada momento?
- ¿Con qué frecuencia el sistema fue lento y tardó en responderle?
- ¿Funcionó el sistema como usted esperaba?
- ¿Cree que utilizará el sistema en el futuro?

A menudo es económicamente inviable realizar estudios completos de satisfacción de los usuarios después de cada cambio en un sistema. Por esta razón, es útil disponer de una evaluación del rendimiento que se correlaciona bien con la satisfacción humana. Se han estudiado varios de estos factores y heurísticos, a menudo agrupados en dos tipos de criterios: lo bien que el sistema permite a los usuarios alcanzar sus objetivos (maximizando el éxito de la tarea) con la menor cantidad de problemas (minimizando los costes). Algunas métricas también utilizadas para evaluar sistemas de diálogo orientados a tareas son:

- **Éxito en la finalización de la tarea:**

El éxito de la tarea puede medirse evaluando la corrección de la solución total. En el caso de una arquitectura basada en frames, podría ser la tasa de error de los slots (el porcentaje de slots que se llenaron con los valores correctos).

- **Coste de eficiencia:**

Los costes de eficiencia son medidas de la eficacia del sistema para ayudar a los usuarios. Puede medirse por el tiempo total transcurrido para el diálogo en segundos, el número de turnos totales o del sistema, o el número total de consultas.

- **Coste de la calidad:**

El coste de la calidad mide otros aspectos de las interacciones que afectan la percepción que el usuario tiene del sistema. Una de estas medidas es el número de veces que el sistema ASR (reconocimiento automático del habla) no devuelve ninguna frase, o el número de avisos de rechazo del ASR. Otras métricas similares son el número de veces que el usuario tuvo que interrumpir el sistema, o el número de avisos de tiempo de espera que se reproducen cuando el usuario no responde con la suficiente rapidez.

4.3 Un framework para hacer chatbots: RASA

Rasa [5] es un framework para la creación de asistentes y chatbots, escrito en Python y de código abierto. Basa su funcionamiento en dos componentes principales, NLU y Core. NLU (intérprete) es la parte encargada de tomar el texto, analizarlo y descomponerlo de tal manera que el bot comprenda el contenido del mensaje. Core es la parte encargada de tomar decisiones y dar seguimiento a la conversación según lo que NLU provee.

Para entender el funcionamiento de RASA es necesario familiarizarse con los siguientes términos específicos:

- **Intents:**

Representa las intenciones que el usuario quiere expresar cuando envía su mensaje al Chatbot. Por ejemplo el mensaje “hola” podemos asociarlo a la intención de saludo. Para especificar qué mensajes se asocian a qué intent, debemos proporcionar algunas frases de ejemplo que el usuario puede realizar al estar refiriéndose a ese intent.

- **Entities:**

Las entidades son palabras clave que representan algunos datos específicos que el Chatbot puede utilizar para realizar la discusión con el usuario. Las entidades se utilizan para extraer algunos valores dentro de la entrada del mensaje. En nuestro caso utilizamos una entidad por cada planta e insecto del cual tenemos información.

- **Responses:**

Luego de interpretar los intents, el asistente necesita respuestas que pueda enviar al usuario.

Nuestro asistente es capaz de enviar texto, imágenes, enlaces a páginas relacionadas y también ubicaciones de ser necesario. Si una respuesta tiene varias opciones de texto, se elegirá una de ellas al azar cada vez que se prediga esa respuesta. Esto nos será de utilidad en el futuro cuando utilicemos paráfrasis para generar numerosas respuestas con la misma semántica pero distinta sintaxis para aumentar la naturalidad de la conversación.

- **Stories** Las stories son ejemplos de conversaciones que son usados para entrenar al asistente para responder correctamente en función de lo que el usuario haya dicho previamente en la conversación. El formato de las historias muestra la intención del mensaje del usuario seguido de la acción o respuesta del asistente.

- **Rules** Las reglas describen partes de las conversaciones que deben seguir siempre el mismo camino, independientemente de lo que se haya dicho anteriormente en la conversación.

Si queremos que nuestro asistente responda siempre a una determinada intención con una acción específica utilizamos una regla para asignar esa acción a la intención.

- **Tracker** Componente de Rasa que mantiene el estado del diálogo, que se representa como un objeto JSON que lista los eventos de la sesión actual.

- **Policy** Los componentes de Rasa que predicen la siguiente acción del sistema de diálogo. Las Políticas toman decisiones sobre cómo debe proceder el flujo de la conversación. Una configuración típica incluye varias políticas, y la policy con mayor confianza decide la siguiente acción que se debe realizar en la conversación.

- **Action** Un solo paso que un bot realiza en una conversación (por ejemplo, llamar a una API o enviar una respuesta al usuario).



Figura 18: Etapas del funcionamiento interno de RASA.

El proceso de funcionamiento básico de rasa puede ser descrita por la Figura 18 y con las siguientes seis etapas:

1. El usuario envía la solicitud (se envía un mensaje al intérprete)
2. En la sección del Intérprete, tenemos a Rasa NLU que captura la solicitud, y extrae las intenciones para entender lo que el usuario quiere, y al mismo tiempo extrae algunas palabras clave específicas (entidades) que representan los datos importantes. A continuación, envía el resultado al Tracker. El resto de los elementos a continuación está contenido dentro de Rasa Core.
3. El Tracker aquí, se utiliza para almacenar la historia de la conversación en la memoria. Por este hecho, el Chatbot debe saber en qué nivel está la conversación, mantiene el estado de la conversación.
4. Después de eso, viene la Policy; elige la acción que se ejecutará en cada paso en el diálogo, con respecto a la historia de la conversación.
5. La Action aquí, busca dentro de la historia para consultar el estado de la conversación. Este paso puede desencadenar otra Action.
6. La Action se ejecuta y se genera un output (mensaje del Chatbot) al usuario.

4.4 Diagramas de manejo conversacional

En esta parte del capítulo mostraremos algunas de las funcionalidades de nuestro sistema de diálogo a través de diagramas de diálogo. Exponiendo en detalle las distintas acciones que puede tomar el usuario, los slots que recuerda rasa, y los tipos de respuestas que se ofrecen.

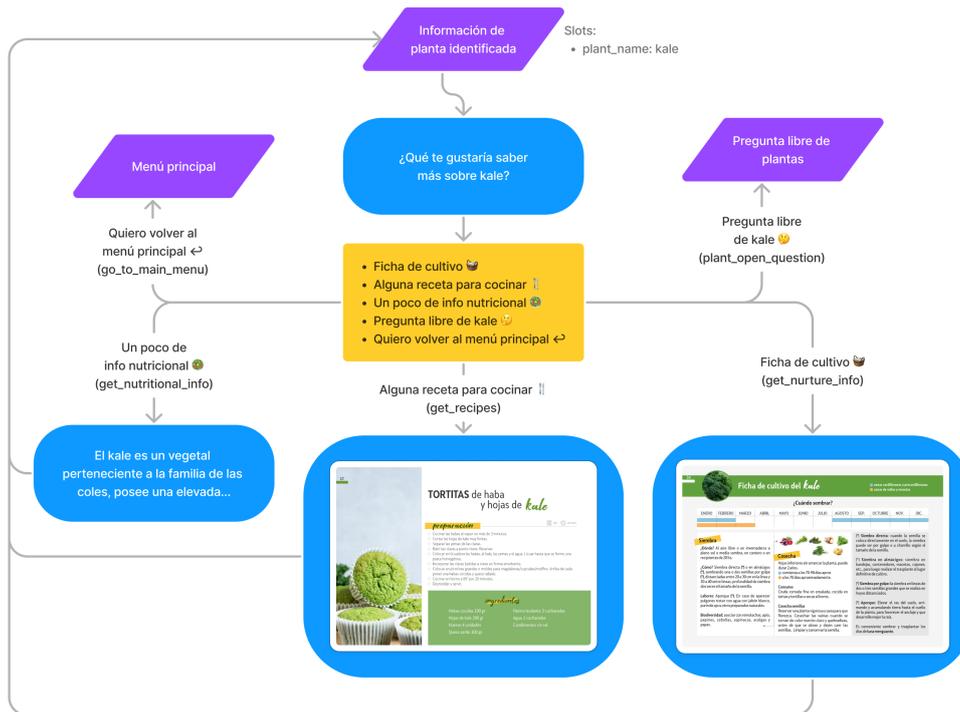


Figura 19: Diagrama de diálogo de la funcionalidad de obtener más información de una planta luego de ser identificada por el sistema.

En color lila se destacan los nombres del inicio de las stories. En gris se expone que el contexto de la conversación está ocurriendo luego de ser identificada la planta de kale, y por lo tanto fue guardada en el slot `plant_name`. En azul y amarillo se muestran los mensajes que fueron enviados por el bot al usuario. En azul aquellos que solo contienen información de texto o imagen y en amarillo cuando contiene botones interactivos. Por último, en las flechas con nombre se observa cuando el usuario manda un mensaje y entre paréntesis el intent que produce para el funcionamiento interno de RASA.

En la Figura 19 se observa el diagrama de diálogo de la funcionalidad de obtener más información de una planta luego de ser identificada por el sistema. En este caso se muestra con la planta de Kale. Notar que en el caso de la story de Menú Principal continuaría

con otras posibilidades para el usuario no graficadas en el diagrama. Las dos imágenes de recetas y ficha de cultivo fueron generadas a partir de la información recolectada por los ingenieros del proyecto. Para ver más detalle de los datos de la ficha de cultivo, ver el Capítulo 2.

En la Figura 20 se expone el diagrama de diálogo de la funcionalidad de identificar un insecto. En este caso sólo se muestra en detalle el camino de identificación por foto. Notar como en la burbuja azul más extrema a la derecha se llena el slot `insect_name` al lograr identificación gracias a nuestro modelo de clasificación. El caso de la story de Identificar insecto por texto se permite al usuario escribir el nombre del insecto, y en el caso de no poder asociarlo con ninguno de los cuales poseemos información, se le ofrece una lista de botones con aquellos con los que sí podemos ayudarle.

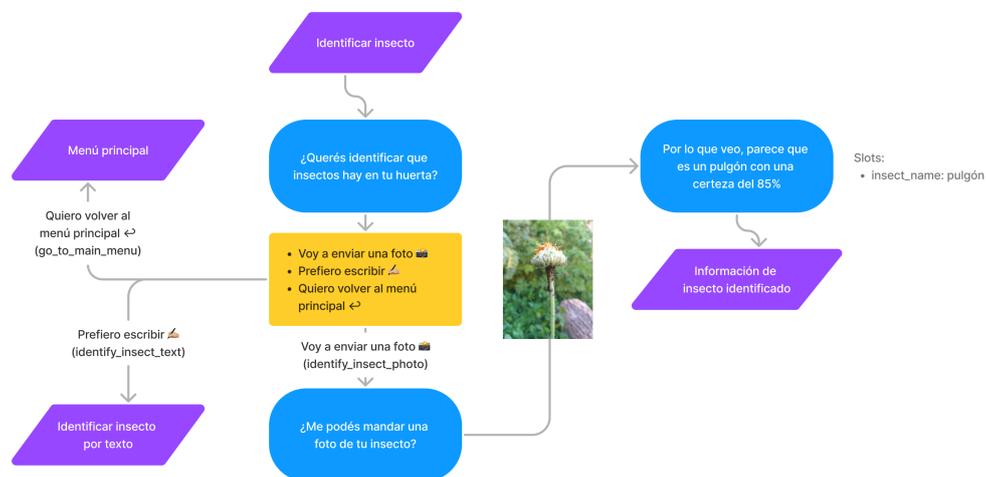


Figura 20: Diagrama de diálogo de la funcionalidad de identificar un insecto. Sólo se muestra en detalle el camino de identificación por foto.

En la Figura 21 se ve el diagrama de diálogo de la funcionalidad de pregunta libre en una planta ya identificada. En este caso se destaca el uso de paráfrasis frente a re-preguntas. Esta paráfrasis es generada automáticamente por nuestro modelo detallado en el capítulo 5 y luego curado y corregido por miembros del proyecto en el desarrollo del sistema de diálogo.

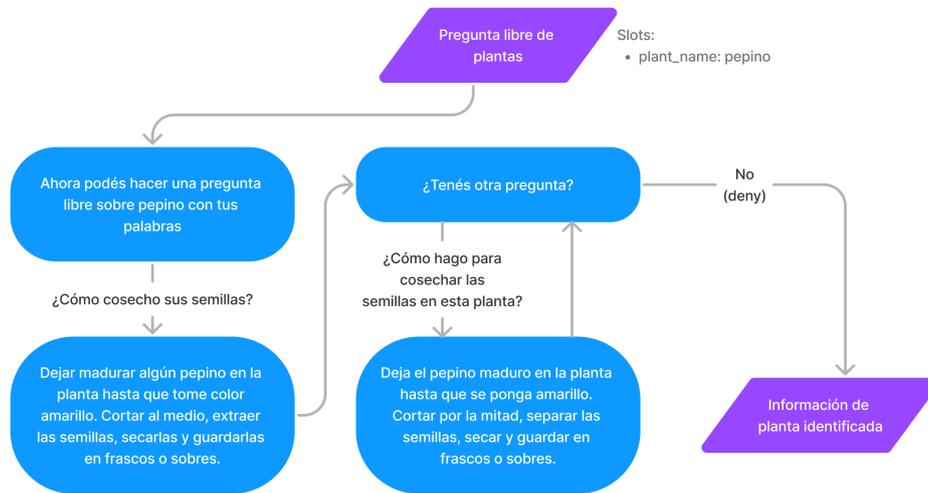


Figura 21: Diagrama de diálogo de la funcionalidad de pregunta libre en una planta ya identificada. En este caso se destaca el uso de paráfrasis frente a re-preguntas.

5 Generación de respuestas

La generación de lenguaje en español, al igual que la mayoría de tareas en machine learning, sufre de no poseer tanto trabajo actual ni datos de gran calidad y tamaño. Sin embargo, una de las mejoras posibles a la naturalidad de la conversación con un asistente automático es la paráfrasis.

La paráfrasis o parafraseo en lingüística computacional es la tarea de procesamiento del lenguaje natural que consiste en detectar y generar paráfrasis. Las aplicaciones de la paráfrasis son variadas, entre ellas la recuperación de información, la respuesta a preguntas, el resumen de textos y la detección de plagios. Para parafrasear eficazmente, desde una palabra hasta una reformulación completa, no basta con cambiar la estructura de una frase, ya que no abarca todo el alcance de la paráfrasis. Una idea puede expresarse de forma diferente cambiando algunas palabras (por ejemplo, buscando sinónimos) o reordenando sus elementos. Además, no es necesario utilizar ninguna de las palabras contenidas en la frase original, por lo que podemos parafrasear una idea siempre que mantengamos el concepto principal. En cambio, podemos utilizar las mismas palabras en dos textos que no tienen ningún tipo de relación. Por tanto, parafrasear no es sólo sustituir elementos o su estructura.

Nuestra aplicación de paráfrasis será sobre un conjunto de párrafos descriptivos de insectos en los cuales se detallan características físicas, métodos para combatirlos y daños que producen a las huertas, datos provenientes de manuales proporcionados por los ingenieros del proyecto. Estos datos son obtenibles a través de la herramienta desarrollada en este proyecto. Nosotros utilizaremos la paráfrasis como método de lidiar con re-preguntas, en lugar de enviar nuevamente el mismo párrafo descriptivo de manera literal, enviaremos una paráfrasis del mismo. De esta forma el usuario podrá obtener la misma semántica pero con distinta sintaxis, todo en lenguaje natural. Las paráfrasis generadas por el modelo desarrollado en esta tesis serán posteriormente curadas por expertos de dominio para garantizar que la información proporcionada sigue manteniendo el mismo estándar de calidad.

El desafío más grande de esta tarea fue el de encontrar modelos entrenados con corpus en español, se explorará debido a esto la posibilidad de realizar traducciones intermedias y de esa forma utilizar modelos entrenados en otros idiomas.

Uno de los trabajos más recientes en resumen automático es [11], donde se realiza la tarea de resumen automático de textos mediáticos. Esta tarea tiene relación con la generación de paráfrasis automática ya que en un sentido resumir es hacer paráfrasis y además reducir el tamaño del texto. En nuestro caso esta segunda condición no es requerida. Algunas descripciones de conceptos de las secciones 5.1 y 5.2 fueron tomadas de este trabajo.

Este capítulo comienza describiendo qué son y para qué se utilizan los modelos secuencia a secuencia. Luego, la sección 5.2 expone el concepto de modelo preentrenado y algunas técnicas utilizadas para construirlos. Además se resaltan BERT y T5 como exponentes de los mismos, explicando en alto nivel su arquitectura y usos. En la sección 5.3 brevemente mostramos y ponemos a prueba algunos modelos de generación de texto en español e inglés relacionados a la paráfrasis. Finalmente en la sección 5.4 describimos el proceso de desarrollo

del modelo de generación de paráfrasis, producto en este trabajo. Aquí detallamos los datasets utilizados como conjunto de entrenamiento, el modelo preentrenado que se utilizó y vemos la evolución de la función de error a lo largo de su entrenamiento.

5.1 Modelos de secuencia a secuencia

En el campo del procesamiento del lenguaje natural (PLN) existen distintas formas de estructurar la información para modelos de aprendizaje automático. En este trabajo nos interesan especialmente los llamados modelos de secuencia a secuencia (abreviados Seq2Seq en inglés). Éstos se caracterizan por recibir como entrada una secuencia de texto y producir como salida una nueva secuencia de texto. Estos modelos son capaces de resolver diversos problemas, algunos ejemplos son:

- **Modelado del lenguaje:**

Esta es la tarea más conocida, consistente en predecir el próximo token (palabra, parte de palabra u otro tipo de símbolo) en un documento. Los modelos de lenguaje en ocasiones son entrenados para generar texto de cualquier dominio y, en otros casos, estos modelos son especializados en alguna área en particular.

- **Traducción:**

En esta tarea el modelo recibe como entrada una frase y genera la misma frase en otro idioma.

- **Resumen (summarization):**

Puede ser abstracto, consistente en generar un texto nuevo que resuma un texto más largo, o resumen extractivo, cuando se resume un texto a partir de una porción de ese mismo texto.

- **Simplificación:**

En este caso el texto original es simplificado, por ejemplo, omitiendo términos poco importantes, aclarando o ejemplificando términos inusuales, cambiando estructuras sintácticas, etc.

Si bien han habido distintas maneras de modelar lenguaje natural, en la actualidad los métodos con resultados más útiles en las aplicaciones prácticas se enmarcan en una familia de modelos llamados transformers. Recientemente este campo de investigación ha suscitado gran interés, en particular por los avances logrados para generar texto libre a partir de GPT-2.

Por lo general, existen dos instancias en la aplicación de modelos tan grandes como éstos. En una primera instancia el modelo es preentrenado, en esta etapa el modelo ajusta sus parámetros de modo de ubicarse en algún lugar del espacio de parámetros que se estima cercano al objetivo que tiene después y, por lo tanto, es la tarea más costosa. Para poder aplicar el modelo en una tarea en particular es necesario continuar con el entrenamiento pero adaptando la salida del modelo para el caso, el llamado ajuste fino del modelo.

La mayoría de estos desarrollos suelen publicarse acompañados con el código para reproducir los experimentos y con los modelos ya entrenados. Sin embargo, gran parte estos esfuerzos están orientados al inglés y, en menor número, a otros idiomas.

5.2 Modelos preentrenados y ajuste fino

Modelos como BERT o GPT-2 dividen el entrenamiento en dos etapas. A la tarea de entrenar un modelo en un conjunto de datos extremadamente grande de manera no supervisada se la conoce como preentrenamiento. En esta etapa se espera que el modelo aprenda representaciones generales del lenguaje que sean de utilidad para más de una tarea específica. Luego, este modelo preentrenado puede ser adaptado en tareas específicas (a veces llamadas *downstream tasks* en inglés) que, generalmente, requiere un cambio menor en la arquitectura del modelo, a esto se lo conoce como ajuste fino (o *fine-tuning* en inglés). Si bien existen distintas técnicas para realizar un ajuste fino, la idea principal es reutilizar el conocimiento existente en los parámetros aprendidos previamente. El surgimiento y extensión de este tipo de modelos genera un cambio de paradigma en el mundo de la Inteligencia Artificial. Por este motivo, se los conoce como modelos fundacionales. Allí se resume la relevancia de estos modelos en dos palabras: emergencia y homogeneización. La emergencia implica que el aprendizaje de estos modelos surge implícitamente de los patrones de datos, antes que explícitamente. Esto hace referencia a que durante el preentrenamiento no se emplea un objetivo supervisado sino semi-supervisado, por ejemplo, predecir la siguiente palabra en un modelo de lenguaje. En cuanto a la homogeneización, en este caso el término capta la consolidación de metodologías para construir modelos de Machine Learning en una gran variedad de aplicaciones.

GPT-2 toma el decoder de Transformers, con lo cual usa el método de pre-entrenamiento “natural” para esta arquitectura, es decir, tratar de predecir la siguiente palabra. En BERT se implementa un modelo basado prácticamente sólo en bloques de encoder de Transformers, difiere levemente ya que en vez de usar activaciones ReLU usa activaciones GELU. Para poder lograr un entrenamiento bidireccional BERT se preentrena con nuevas tareas no supervisadas:

- **Masked Language Model (MLM):**

en vez de predecir la siguiente palabra, en esta tarea se enmascaran (se ocultan) al azar una serie de tokens (el 15% en BERT), reemplazándolos por el token [MASK]. A este tipo de tarea también se le llama *cloze task*. Ahora bien, en la tarea de ajuste fino el token [MASK] no aparece, con lo cual se está introduciendo un sesgo. Para mitigar este sesgo, al 80% de los tokens elegidos se los reemplaza por [MASK], a un 10% por un token al azar, y a otro 10% se lo deja sin intercambiar. Finalmente, la tarea consiste en predecir ese token. Por ejemplo, dada la frase ‘El gato está [MASK] la caja’, el modelo debería predecir un vector con la probabilidad de corresponder a la máscara de cada palabra.

- **Next Sentence Prediction (NSP):**

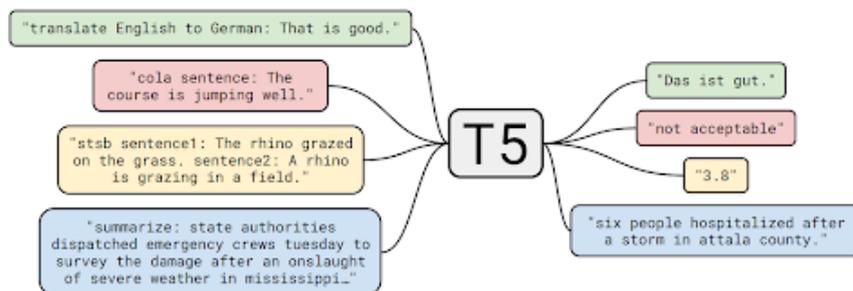


Figura 22: Algunas de las funcionalidades del modelo T5, solicitadas en lenguaje natural al comienzo del prompt.

Ciertas tareas se basan en entender relaciones entre dos frases, lo cual no está capturado en una tarea de modelado de lenguaje, por ejemplo, en las tareas de QA y Natural Language Inference (NLI). Por este motivo, los autores crean la tarea NSP que consiste en tomar al azar dos frases, en la mitad de los casos la segunda frase es la frase que le sigue en el texto original. En la otra mitad de los casos la segunda frase es una frase de otro lugar del texto. El objetivo de la tarea es predecir si la segunda frase le sigue o no le sigue a la primera.

En lugar de BERT [15] (sólo codificador) o GPT [6] (sólo decodificador), utilizamos un modelo seq2seq con codificador y decodificador, como T5 [32], BART [24] o Pegasus [44].

Los modelos vistos anteriormente cuentan con un modelo preentrenado que, para usarlo en una tarea específica, requiere que se realice un ajuste fino. Ahora bien, si se quiere resolver una segunda tarea se debe volver a aplicar un ajuste fino sobre el modelo original generando así un nuevo modelo. De este modo contamos con un modelo ajustado a cada tarea específica (dos en total, en este ejemplo). Por supuesto, cada modelo puede tener distintos hiperparámetros. En [32] se plantea un enfoque en el cual se preentrena un modelo que luego es ajustado en múltiples tareas pero siempre utilizando el mismo modelo, hiperparámetros y función de pérdida. Este modelo es llamado Text-to-Text Transfer Transformer o T5. El modelo emplea la arquitectura original de Transformer con cambios menores pero con mejoras en la representación de la información y en el entrenamiento. En cuanto a cómo se representa la información, todos los problemas son tratados como problemas de secuencia a secuencia (o texto a texto), lo que justamente permite unificar los problemas. Para hacerlo se introduce un prefijo (prefix token, en inglés) en la entrada que identifica qué tipo de problema es llamado token de control. Por ejemplo: "translate Spanish to English: El gato está en la caja.". Puede verse otros ejemplos en la Figura 22.

Para poder entrenar un modelo que efectivamente resuelva múltiples tareas los autores construyen el conjunto de datos Colossal Clean Crawled Corpus (C4), tomando y limpiando 750 GB de información en inglés de Common Crawl. El modelo luego es ajustado en sucesivas tareas de clasificación como Glue y SuperGlue, en total sumando 17.

En particular nos enfocamos en el modelo mT5 [43]. mT5 es una variante multilingüe del modelo T5 de Google que fue preentrenado sobre un conjunto de datos de más de 101 idiomas y que contiene entre 300 millones y 13.000 millones de parámetros. La arquitectura y el entrenamiento del modelo seguidos para mT5 emulan estrechamente los de T5. Este modelo de IA aporta varios tipos de información en medio de lenguas idénticas que favorecen a las lenguas de bajos recursos y concede para el procesamiento zero-shot del lenguaje. El aprendizaje de zero-shot (ZSL) es una configuración de problemas en el aprendizaje automático, donde en el momento de la prueba, el modelo observa muestras de clases que no fueron observadas durante el entrenamiento, y necesita predecir la clase a la que pertenecen. Los métodos de "zero-shot" generalmente funcionan asociando las clases observadas y no observadas a través de algún tipo de información auxiliar, que codifica las propiedades distintivas observables de los objetos.

5.3 Utilizando modelos de paráfrasis y tareas relacionadas

En esta sección expondremos algunos modelos del estado del arte en tareas relacionadas a la nuestra para obtener una visión general de lo que se logra y se trabaja en la actualidad. Realizamos una breve descripción de cada uno y lo ponemos a prueba.

Uno de los modelos que está en auge pero lamentablemente no pudimos mostrar debido a que no fue posible utilizarlo de manera gratuita fue GPT-3 [7]. Este es un modelo que utiliza prompting para acceder a todas sus funcionalidades de forma similar a T5.

5.3.1 Bert2Bert

El primer intento fue utilizando el modelo "Spanish Bert2Bert (shared) fine-tuned on PAUS-X es for paraphrasing" [33]. Dio prometedores resultados al probarse con esta frase: **Original:** "Los pulgones son organismos de pequeño tamaño, de colores variados, principalmente verdes, amarillos o negros."

- **Paráfrasis 1:** "Los pulgones son organismos de pequeño tamaño, de colores variados, predominantemente verdes, amarillos o negros."
- **Paráfrasis 2:** "Los pulgones son organismos de pequeño tamaño, de colores variados, principalmente de color negro, amarillos o negros."
- **Paráfrasis 3:** "Los pulgones son organismos de pequeño tamaño, de colores variados, principalmente de colores negros, amarillos o negros."

5.3.2 T5 en inglés para preguntas

El segundo intento fue utilizando el modelo "T5-base fine-tuned on Quora question pair dataset for Question Paraphrasing" [35]. Este modelo realiza paráfrasis de preguntas y fue entrenado con el corpus de Quora [38] en inglés.

Para utilizarlo es necesario escribir la palabra paraphrase seguida de la pregunta a parafrasear. Este modelo está basado en el modelo T5 [31] el cual realiza numerosas tareas

que deben ser solicitadas en lenguaje natural al principio del prompt como puede verse en la Figura 22.

Prompt: “paraphrase: What is the best framework for dealing with a huge text dataset?”

- **Paráfrasis 1:** “What is the best framework for dealing with a huge text dataset?”
- **Paráfrasis 2:** “What is the best framework for dealing with a large text dataset?”
- **Paráfrasis 3:** “What is the best framework to deal with a huge text dataset?”
- **Paráfrasis 4:** “What are the best frameworks for dealing with a huge text dataset?”
- **Paráfrasis 5:** “What is the best framework for dealing with huge text datasets?”

5.3.3 Question-Answering

La respuesta a preguntas (QA) es una rama de la inteligencia artificial dentro de los campos del procesamiento del lenguaje natural y la recuperación de la información; la construcción de sistemas que responden a preguntas planteadas en un lenguaje natural por los humanos.

Los programas de respuesta a preguntas pueden construir respuestas a través de la consulta de una base de conocimientos (una base de datos estructurada de conocimientos) o de una colección no estructurada de documentos en un lenguaje natural. Es por esto que resulta de interés explorar este tipo de modelos al desarrollar un sistema de diálogo automático. Lamentablemente no entrenamos un modelo de este tipo porque los ingenieros del proyecto lograron producir una lista de preguntas frecuentes que ellos mismos respondieron en numerosas ocasiones, y las que de más utilidad serían para el sistema. De cualquier forma, aquí damos una introducción muy breve a este tema y ponemos a prueba un modelo en español.

Los sistemas de respuesta a preguntas son de dominio cerrado (responden a preguntas de un dominio específico) o de dominio abierto (se basan en ontologías generales y conocimientos generalizados). Watson [18] de IBM es un ejemplo de este último tipo de sistemas de respuesta a preguntas.

Los sistemas de respuesta de dominio abierto toman las preguntas en lenguaje natural y las transforman en una consulta estructurada. La extracción de palabras clave se utiliza para determinar el tipo de pregunta (quién, dónde, cuántos). Para determinar el tipo de respuesta (persona, lugar, número) se utilizan técnicas de etiquetado de partes del discurso y de análisis sintáctico. A continuación, se utiliza un sistema de recuperación de información para encontrar datos que contengan las palabras clave. A continuación, se utiliza el análisis sintáctico para traducir la respuesta en un texto con sentido.

Este intento se llevó a cabo a través del modelo “Spanish T5 (small) fine-tuned on SQAC for Spanish QA” [34]. Este modelo recibe contexto informativo en lenguaje natural y es capaz de responder preguntas en lenguaje natural sobre dicho contexto. Fue entrenado principalmente en artículos enciclopédicos, noticias, y contenido del dataset Ancora [22], constituidos fundamentalmente por textos periodísticos.

Contexto: “La célula (del latín *cellula*, diminutivo de *cella*, ‘celda’) es la unidad morfológica y funcional de todo ser vivo. De hecho, la célula es el elemento de menor tamaño que puede considerarse vivo. De este modo, puede clasificarse a los organismos vivos según el número de células que posean: si solo tienen una, se les denomina unicelulares (como pueden ser los protozoos o las bacterias, organismos microscópicos); si poseen más, se les llama pluricelulares. En estos últimos el número de células es variable: de unos pocos cientos, como en algunos nematodos, a cientos de billones (10¹⁴), como en el caso del ser humano. Las células suelen poseer un tamaño de 10 μm y una masa de 1 ng, si bien existen células mucho mayores.”

Pregunta: “¿Cuál es el nombre que se le da a la unidad morfológica y funcional de los seres vivos?”

Respuesta: “la célula”

5.4 Entrenando un modelo de paráfrasis

Tratamos la paráfrasis como una tarea supervisada, así que utilizamos un corpus de paráfrasis llamado Tapaco [37] extraído de la base de datos Tatoeba. Tatoeba es un proyecto de crowdsourcing dirigido principalmente a los estudiantes de idiomas. Su objetivo es proporcionar frases de ejemplo y traducciones para determinadas construcciones lingüísticas y palabras. El corpus de paráfrasis se crea rellenando un grafo con frases de Tatoeba y enlaces de equivalencia entre frases que “significan lo mismo”. A continuación, se recorre este grafo para extraer conjuntos de paráfrasis. Se aplican varios filtros y pasos de poda independientes del idioma para eliminar las frases poco interesantes. Una evaluación manual realizada en tres idiomas muestra que entre la mitad y las tres cuartas partes de las paráfrasis inferidas son correctas y que la mayoría de las restantes son correctas pero triviales, o casi paráfrasis que neutralizan una distinción morfológica. El corpus contiene un total de 1,9 millones de frases, con entre 200 y 250 000 frases por lengua. En nuestro caso sólo tomamos la sección en español.

Además, utilizamos el corpus del desafío PARMEX 2022 que se encuentra dentro del programa de IberLEF 2022. Este dataset tiene originalmente como objetivo la identificación de paráfrasis en lugar de la generación de los mismos, sin embargo podemos filtrar estos datos por aquellos que son efectivamente paráfrasis entre sí para aumentar nuestro dataset.

Por último añadimos datos de dominio específico de plantas e insectos utilizando una técnica llamada backtranslation. Backtranslation o traducción inversa, es el proceso de volver a traducir el contenido de la lengua de destino a su lengua de origen en términos literales. Por ejemplo, si se traduce un contenido del inglés al sueco, el traductor también escribirá una traducción inversa al inglés para que se entienda fácilmente la intención de la opción traducida. Las traducciones inversas no afectan a la memoria de traducción ni a otros recursos, como los glosarios, utilizados por el traductor.

Dentro del entrenamiento se utilizó una función `get_batch` que toma pares de paráfrasis de los distintos datasets mencionados anteriormente. Para nuestro modelo utilizamos la misma proporción de todos los datos: 20% del dataset Tapaco. 20% del dataset PARMEX. 20% del dataset de backtranslation de recomendaciones de cultivo de plantas. 20% del

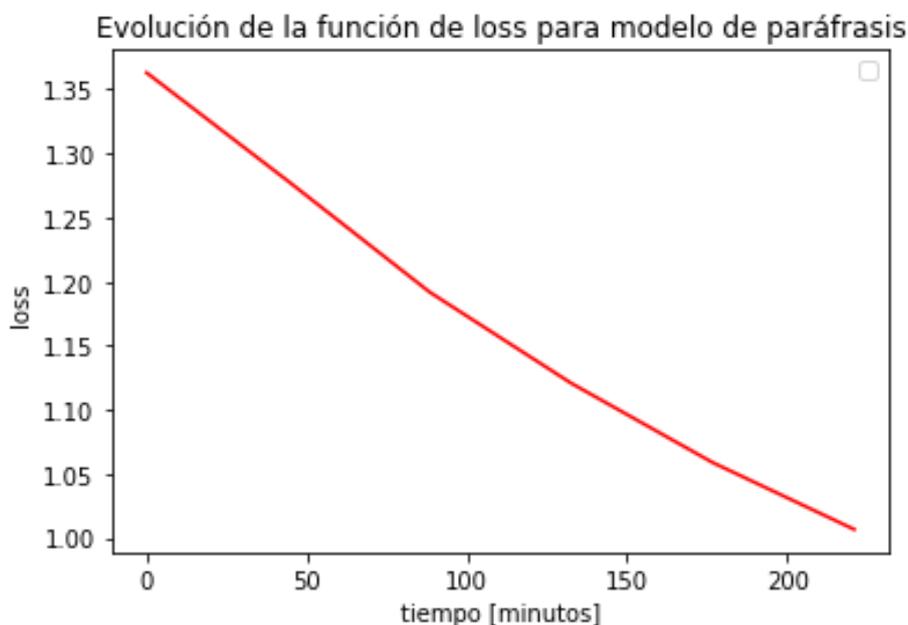


Figura 23: Evolución de la función de loss en modelo de paráfrasis

dataset de backtranslation de descripciones de insectos. 20% del dataset de backtranslation de información nutricional de plantas.

Para el entrenamiento del modelo de paráfrasis se utilizó como base el modelo mt5-small-mlsum [12]. Este modelo tiene como objetivo el resumen automático de noticias, similar a la paráfrasis pero con una reducción en el volumen de texto esperado como output.

Como podemos ver en la Figura 23 entrenando por 3horas 41min, la función de loss se disminuyó a 1.007. Debimos detener el entrenamiento en ese momento por limitaciones en el tiempo de cómputo alocado.

A continuación se expone el modelo en funcionamiento con algunos ejemplos que no fueron utilizados en el entrenamiento:

5.4.1 Ejemplo 1

- text1: El bus estaba totalmente repleto.
- text2: El autobús estaba lleno hasta su capacidad máxima.
- output modelo: Los buses estaban repletos.

5.4.2 Ejemplo 2

- text1: La función tuvo una buena aceptación.

- text2: La función fue bien acogida.
- output modelo: La función fue buena.

5.4.3 Ejemplo 3

- text1: Yo soy tan fuerte como vos.
- text2: Soy tan fuerte como tú.
- output modelo: Yo soy muy fuerte como yo.

En el siguiente capítulo se realizará una introducción a la evaluación de sistemas de diálogo y se pondrá a prueba este modelo a través de una evaluación manual bajo distintos criterios de calidad.

6 Evaluación de paráfrasis

La evaluación de la paráfrasis es una tarea complicada porque es una tarea subjetiva y distintas personas pueden tener distintas opiniones. Una de las razones es que las lenguas naturales son ambiguas y que hay muchas maneras diferentes de decir lo mismo pero esa interpretación depende del receptor. Gran parte de su complejidad se debe a que cada persona interpreta el lenguaje de forma diferente. Con tantas posibilidades, es un reto desde el punto de vista computacional llegar a una puntuación de evaluación robusta.

En la evaluación de la paráfrasis automática, lo usual es comparar la frase objetivo con la frase "gold standard". Pero es difícil definir una única frase gold standard. Una frase puede parafrasearse de numerosas formas posibles que sean igualmente buenas. Esto es problemático tanto para los humanos como para las computadoras. Cuando un humano realiza paráfrasis de un texto, es probable que las opiniones sobre la calidad de la paráfrasis difieran de un lector a otro.

A la hora de evaluar, se pueden utilizar enfoques de evaluación tanto manuales como automáticos. Veamos con más detalle cada uno de ellos. Luego presentaremos los resultados de la evaluación humana de los modelos de paráfrasis usados en nuestro sistema y presentados en el capítulo anterior.

6.1 Enfoque manual

Los expertos de dominio proporcionan los resultados más confiables y consistentes en cuanto a la medición de la calidad y el análisis de los errores. Permite evaluar métricas clave como las puntuaciones de adecuación y fluidez, las medidas de post-edición, la clasificación humana de las paráfrasis a nivel de frase y las evaluaciones basadas en tareas.

Los humanos evalúan la paráfrasis automática de varias maneras. La primera consiste en asignar una calificación a la calidad general de la paráfrasis de destino. Esto suele hacerse en una escala de 1 a 10 (o un porcentaje), que va desde "muy mala calidad" hasta "calidad impecable".

Otra forma de evaluar la paráfrasis es por su adecuación, es decir, cuánto del significado del texto de origen se ha conservado en el texto de destino. Normalmente se califica en una escala que va desde "no se conserva el significado" hasta "se conserva todo el significado".

La fluidez es otra métrica útil para que los expertos de dominio juzguen la calidad de una traducción. Las escalas suelen ir de "incomprensible" a "fluido".

Los evaluadores humanos también pueden utilizar el análisis de errores para identificar y clasificar los errores en el texto traducido por la máquina. El proceso exacto depende de la lengua, pero, en general, los evaluadores buscarán tipos de errores como "palabras que faltan", "orden incorrecto de las palabras", "palabras añadidas" o "parte de la oración incorrecta".

Uno de los principales problemas del enfoque manual se debe a la naturaleza subjetiva del juicio humano. Por ello, es importante medir el nivel de acuerdo inter-evaluador (consistencia entre múltiples evaluadores). Un acuerdo medio o alto entre anotadores a la evaluación tiene mayor robustez que si se hiciera con un sólo anotador. Las anotaciones

con acuerdo denotan mayor seguridad en el resultado. Otra desventaja es que la evaluación humana es costosa y requiere mucho tiempo. Pero hasta el momento no existen métricas automáticas confiables para la evaluación de generación de lenguaje natural.

6.2 Enfoque automático

Para evitar estos problemas comunes con el enfoque manual de la evaluación de la paráfrasis automática, los investigadores han desarrollado una serie de enfoques automáticos, cada uno con sus propias ventajas y desventajas. En este trabajo no los usaremos debido a que lo-gramos realizar evaluación manual gracias a voluntarios participantes del proyecto. Sin embargo mencionaremos algunos. Los investigadores de este campo proponen constantemente las métricas de evaluación de la paráfrasis automática, además de crear otras nuevas. Algunas de las más utilizadas son BLEU [29], METEOR [1] y ROUGE [26].

6.3 Resultados de nuestros modelos de paráfrasis

En nuestro caso, se realizó una evaluación con enfoque manual gracias a Lucía Martínez Gavier y Diego Gimenez, miembros del proyecto enmarcados en horas de trabajo voluntaria por el programa Compromiso Social Estudiantil. El Programa Compromiso Social Estudiantil está destinado a incorporar en la formación de todos los estudiantes de la UNC acciones, tareas o proyectos vinculados con la extensión universitaria que consistan en acciones socialmente relevantes.

La evaluación consistió en calificar paráfrasis generadas por el modelo entrenado en nuestro proyecto con respecto a cuatro criterios de calidad:

1. Coherencia interna

Todas las referencias intra-ortográficas o cruzadas en el resumen deben ser inequívocas y dentro del ámbito de la paráfrasis. Por ejemplo, si se utiliza un pronombre, el sustantivo correspondiente al que se refiere también debe estar presente en algún punto antes de él en el resumen. Tampoco debe haber ambigüedades respecto a la entidad o información exacta (como un punto anterior) a la que se hace referencia.

2. Conservación de semántica

El texto debe verbalizar adecuadamente la información presente en los datos. Se desea cubrir todos los detalles o los más significativos.

3. Gramaticalidad

En lingüística, la gramaticalidad se refiere a la propiedad de una construcción gramatical de estar bien formada. Una asunción común es que los hablantes nativos de una lengua tienen una intuición sobre qué frases están correctamente formadas y cuales no sin necesidad de haber sido instruidos en dicho conocimiento.

4. No redundancia

La paráfrasis no debe repetir ningún punto, e idealmente debe tener la máxima cobertura de información dentro de la longitud limitada del texto.

Estos cuatro criterios fueron tomados a partir de [36], en este paper se describen distintos criterios de evaluación para resumen automático. La tarea de paráfrasis es similar al resumen automático ya que también busca mantener la semántica alterando la sintaxis del texto original, con la distinción que en el resumen además se tiene como objetivo reducir el tamaño del texto. Todos los criterios de calidad utilizados para nuestra evaluación aplican para ambas tareas.

Desde el dataset de paráfrasis descrito en el Capítulo 5, se seleccionaron aleatoriamente 50 textos de un grupo con los que nuestro modelo no fue entrenado y se les generaron sus respectivas paráfrasis automáticas.

A cada uno de los evaluadores se les entregó 30 pares de textos, el original con su paráfrasis automática, y se les pidió evaluar los previos cuatro criterios en una escala del 1 al 5. Donde en todas las categorías un número más alto significa mejor rendimiento. Este modo de evaluación es conocido como Escala de Likert [25].

De estos 30 pares de textos, 20 son únicos para cada evaluador, mientras que 10 fueron entregados a ambos para realizar análisis del acuerdo inter-evaluador presentado. Este es el grado de acuerdo entre observadores independientes que califican, codifican o evalúan el mismo fenómeno. Particularmente en tareas de generación de texto, por muchas de las razones descritas al inicio de este capítulo, es complejo asignar un puntaje a criterios de calidad de redacción y lenguaje. Es por esto que decidimos computar la métrica de Coeficiente kappa de Cohen [10] para cada uno de los criterios de calidad evaluados. Esta métrica es una medida estadística que ajusta el efecto del azar en la proporción de la concordancia observada. En el paper original Cohen sugirió que el resultado de Kappa se interpretara de la siguiente manera: los valores menores o iguales 0 indican que no hay acuerdo, los valores entre 0.01 y 0.20 indican que no hay acuerdo o que es leve, los valores entre 0.21 y 0.40 indican que es regular, los valores entre 0.41 y 0.60 indican que es moderado, los valores entre 0.61 y 0.80 indican que es sustancial y los valores entre 0.81 y 1.00 indican que es casi perfecto.

En el caso de la métrica de coherencia interna obtuvimos un coeficiente kappa de 0.552, con 7 de 10 observaciones coincidiendo perfectamente.

Para la métrica de conservación de semántica obtuvimos un coeficiente kappa de 0.267, el peor resultado de los cuatro criterios. Con sólo 2 de 10 observaciones coincidiendo perfectamente, aunque 6 de los 8 fallos fueron a sólo un punto de puntaje de distancia.

En la métrica de gramaticalidad obtuvimos un coeficiente kappa de 0.545, con 6 de 10 observaciones coincidiendo perfectamente.

Por último, la métrica de no redundancia obtuvo un coeficiente kappa de 0.615, con 7 de 10 observaciones coincidiendo perfectamente. A diferencia de la métrica de coherencia interna, aquí las no coincidencias se dieron de manera más cercana, aumentando el coeficiente.

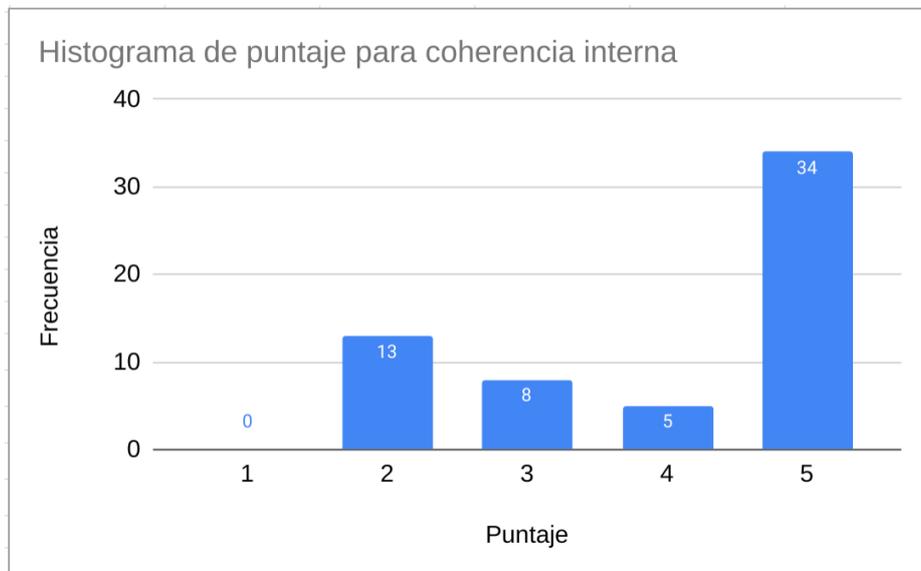


Figura 24: Histograma de puntaje en evaluación para la métrica de coherencia interna

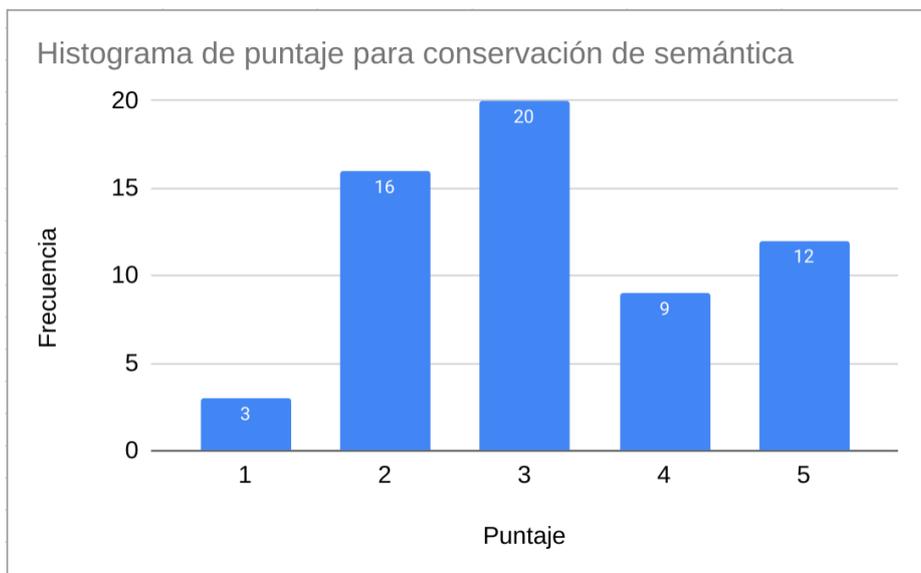


Figura 25: Histograma de puntaje en evaluación para la métrica de conservación de semántica

En la Figura 24 tenemos el histograma de puntaje en evaluación para la métrica de coherencia interna. Aquí se observa una alta frecuencia del puntaje más alto, donde 34 de los 60 casos de evaluación totales fueron clasificados de esta manera. El promedio en esta métrica a través de toda la evaluación fue de 4.

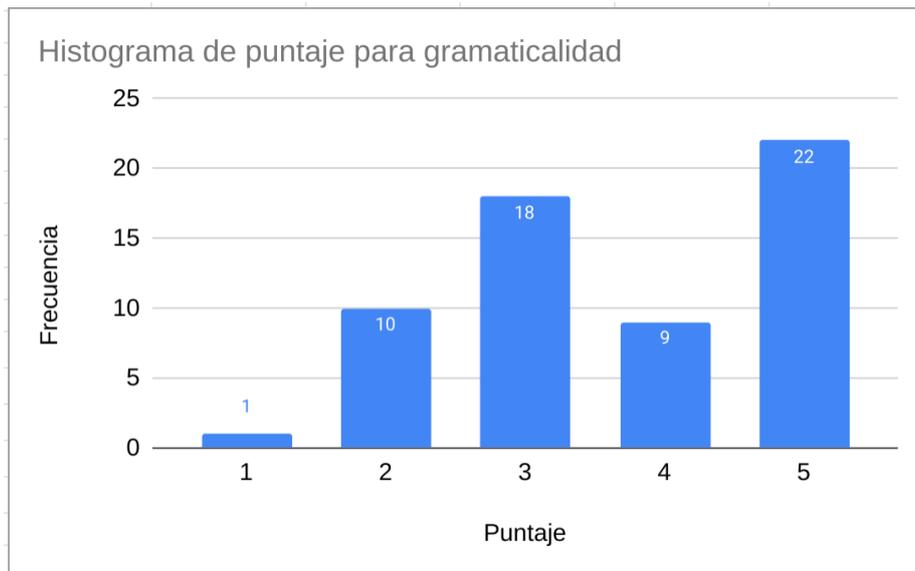


Figura 26: Histograma de puntaje en evaluación para la métrica de gramaticalidad

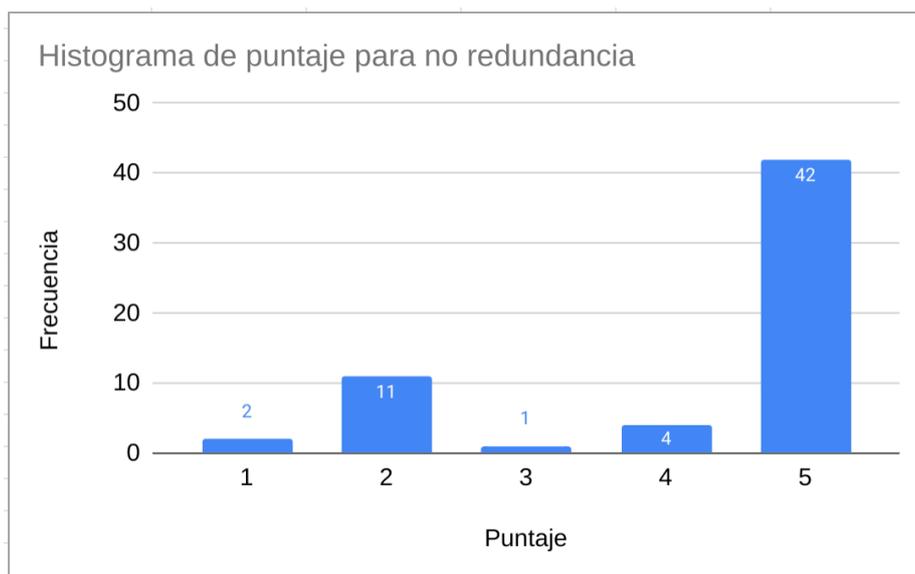


Figura 27: Histograma de puntaje en evaluación para la métrica de no redundancia

En la Figura 25 tenemos el histograma de puntaje en evaluación para la métrica de conservación de semántica. Aquí se observa el rendimiento más pobre de los cuatro criterios de calidad evaluados, donde sólo 21 de los 60 casos de evaluación totales fueron clasificados con puntaje 4 o mejor. El promedio en esta métrica a través de toda la evaluación fue de 3.18.

En la Figura 26 tenemos el histograma de puntaje en evaluación para la métrica de gramaticalidad. Aquí se observa un mejor rendimiento que la métrica anterior, donde sólo 11 de los 60 casos de evaluación totales fueron clasificados con puntaje 2 o peor. El promedio en esta métrica a través de toda la evaluación fue de 3.68.

En la Figura 27 tenemos el histograma de puntaje en evaluación para la métrica de no redundancia. Aquí se observa el mejor rendimiento de todas las métricas, donde 42 de los 60 casos de evaluación totales fueron clasificados con el puntaje más alto. El promedio en esta métrica a través de toda la evaluación fue de 4.22.

7 Conclusiones

7.1 Discusión y conclusiones

La localidad de Junín de los Andes se encuentra en la precordillera de la provincia de Neuquén. Es un centro urbano de aproximadamente 17.000 habitantes, rodeado por estancias ganaderas de gran extensión con poca ocupación de mano de obra. En ésta localidad, al igual que en la mayoría de los pueblos o ciudades de la provincia, los productos hortícolas que se consumen no son producidos en ella, sino que provienen de otras regiones del país, principalmente de Cuyo, a más de 800 km, lo que provoca precios elevados por costos de transporte, y productos de baja calidad.

Este proyecto busca aportar tecnología digital para acompañar los procesos productivos de huertas agroecológicas, que permita dar acceso a información útil para el desarrollo de una huerta. Una huerta agroecológica familiar no sólo permite tener acceso a un alimento de mejor calidad sino a generar conciencia de la importancia de mejorar la alimentación. Este desarrollo tecnológico tiene en cuenta los saberes locales recolectados durante décadas por instituciones muy conocedoras del territorio como la asociación civil Amulén y el PRODA del Ministerio de Producción e Industria de la Provincia de Neuquén. En sus bases de datos y su personal técnico Amulén y PRODA cuentan con información productiva adaptada a lo local, no solo geográficamente sino cultural y socialmente.

Lograr que continúe el crecimiento del desarrollo de huertas familiares agroecológicas es de enorme interés para la salud del suelo y las de los que las producen. El desarrollo de esta tarea apoyado en la tecnología hace que más individuos estén acompañados al comenzar su huerta propia, además de aliviar los desafíos ya expresados por los miembros de la Asociación Civil Taller Productivo Amulén y del PRODA para asesorar a 800 familias.

A pesar de que muchas familias de las localidades de Junín de los Andes, San Martín de los Andes y parajes cercanos hacen huerta en sus casas, la gran mayoría de los productos hortícolas no se producen localmente. Son transportados grandes distancias, por lo que aumenta sus costos y demanda el uso de conservantes o manipulación genética para aumentar su durabilidad, disminuyendo así su calidad. Por lo que no responden a los principios de la soberanía alimentaria, y contribuyen a la creciente emisión de gases de invernadero, uno de los responsables del cambio climático.

Entrenamos modelos de identificación de insectos locales, clasificación de residuos y facilitamos el acceso a modelos del estado del arte en identificación de plantas. Además, se proporcionan fichas de cultivo, información nutricional, recetas y estrategia de prevenir daños por insectos, información esencial para el desarrollo de una huerta agroecológica del hogar. Se desarrolló un sistema de diálogo orientado a tareas, logrando que recuerde el contexto previo de la conversación para que ofrezca la información necesaria del tópico conversacional del momento. Por último se entrenó un modelo de generación de paráfrasis automático para aportar a la naturalidad de la conversación al lidiar con re-preguntas.

Se realizó una ponencia en las "XV Jornadas Nacionales de Investigadores en Economías Regionales" de la Facultad de Agronomía en la Universidad de Buenos Aires, en esta se puso en foco el concepto de soberanía tecnológica y alimentaria. Discutiendo el rol de la

inteligencia artificial para el servicio de las comunidades, cuestionando la inversión de los gobiernos locales en la investigación en este tipo de tecnologías, y la accesibilidad a los datos y tecnología desarrollada de manera local.

Es necesario subrayar el enorme aporte que realizaron los ingenieros del proyecto al curar y generar la información con la cual se nutren las fichas de cultivo y los calendarios de cosecha. Uno de los principales originadores del proyecto fue la falta de accesibilidad a información local en la zona de Junín y San Martín de Los Andes, es por esto que debemos valorar y tratar con cuidado y respeto los datos que proporcionamos en nuestra herramienta.

7.2 Trabajo futuro

El proyecto en el cual se enmarca esta tesis todavía no ha concluido, todavía quedan por realizar numerosas tareas y esperamos continuar con otras nuevas en futuros proyectos similares. En particular se harán evaluaciones de las decisiones elegidas para el manejo conversacional con usuarios reales. También se realizará el despliegue del bot en producción, para lo cual será necesario analizar el poder de cómputo y red requerido para lidiar con la cantidad de accesos al mismo. Además se propondrá aumentar la cantidad de plantas, insectos y tipos de residuos a clasificar. Para ello será necesario recolectar no sólo nuevas imágenes, sino todos los datos que nuestro sistema ofrece luego de su identificación, tales como recetas, fichas de cultivo, estrategia de prevención de daño por insectos, entre otros.

Por otro lado, es interesante continuar el trabajo realizado con paráfrasis y extenderlo a la extracción y generación de paráfrasis basadas en descripciones de imágenes. Este concepto, titulado como *iParaphrasing* por [9] propone extraer y posteriormente generar paráfrasis de distintos objetos en imágenes a partir de datasets de image-captioning.

References

- [1] Satanjeev Banerjee and Alon Lavie. “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, June 2005, pp. 65–72. URL: <https://aclanthology.org/W05-0909>.
- [2] Luciana Benotti and Patrick Blackburn. “A recipe for annotating grounded clarifications”. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021, pp. 4065–4077. DOI: 10.18653/v1/2021.naacl-main.320. URL: <https://aclanthology.org/2021.naacl-main.320>.
- [3] Luciana Benotti and Patrick Blackburn. “Grounding as a Collaborative Process”. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, Apr. 2021, pp. 515–531. DOI: 10.18653/v1/2021.eacl-main.41. URL: <https://aclanthology.org/2021.eacl-main.41>.
- [4] Daniel G. Bobrow et al. “GUS, a frame-driven dialog system”. In: *Artificial Intelligence* 8.2 (1977), pp. 155–173. ISSN: 0004-3702. DOI: [https://doi.org/10.1016/0004-3702\(77\)90018-2](https://doi.org/10.1016/0004-3702(77)90018-2). URL: <https://www.sciencedirect.com/science/article/pii/0004370277900182>.
- [5] Tom Bocklisch et al. “Rasa: Open Source Language Understanding and Dialogue Management”. In: *CoRR* abs/1712.05181 (2017). arXiv: 1712.05181. URL: <http://arxiv.org/abs/1712.05181>.
- [6] Tom Brown et al. “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf>.
- [7] Tom B. Brown et al. “Language Models are Few-Shot Learners”. In: *CoRR* abs/2005.14165 (2020). arXiv: 2005.14165. URL: <https://arxiv.org/abs/2005.14165>.
- [8] CCHANG. *Garbage Classification*. 2018. DOI: 10.34740/KAGGLE/DS/81794. URL: <https://www.kaggle.com/ds/81794>.
- [9] Chenhui Chu, Mayu Otani, and Yuta Nakashima. “iParaphrasing: Extracting Visually Grounded Paraphrases via an Image”. In: *CoRR* abs/1806.04284 (2018). arXiv: 1806.04284. URL: <http://arxiv.org/abs/1806.04284>.
- [10] Jacob Cohen. “A Coefficient of Agreement for Nominal Scales”. In: *Educational and Psychological Measurement* 20.1 (1960), pp. 37–46. DOI: 10.1177/001316446002000104. eprint: <https://doi.org/10.1177/001316446002000104>. URL: <https://doi.org/10.1177/001316446002000104>.

- [11] Leonardo Ignacio Córdoba. “Generación de resúmenes de texto en español”. PhD thesis. 2021.
- [12] Leonardo Ignacio Córdoba. *mt5-small-mlsum*. 2022. URL: <https://huggingface.co/LeoCordoba/mt5-small-mlsum> (visited on 09/06/2022).
- [13] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [14] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805. URL: <http://arxiv.org/abs/1810.04805>.
- [15] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: <https://aclanthology.org/N19-1423>.
- [16] Victor Dewulf. *Reinventing the economics of Recycling*. Nov. 2022. URL: <https://recycleye.com/>.
- [17] Victoria Evigan. *Plant Identifier app: Plant Identification Online*. 2014. URL: <https://www.picturethisai.com/>.
- [18] D. A. Ferrucci. “Introduction to “This is Watson””. In: *IBM Journal of Research and Development* 56.3.4 (2012), 1:1–1:15. DOI: 10.1147/JRD.2012.2184356.
- [19] Hugo Gresse. *Home*. Aug. 2019. URL: <https://plantnet.org/en/>.
- [20] Y. Herrero. “Sujetos arraigados en la tierra y en los cuerpos. Hacia una antropología que reconozca los límites y la vulnerabilidad”. In: 2018.
- [21] Jeremy Howard et al. *fastai*. <https://github.com/fastai/fastai>. 2018.
- [22] Oriol Borrega Cepa Joan Aparicio Mena. *Ancora*. 2021. URL: <http://clic.ub.edu/corpus/en> (visited on 07/01/2022).
- [23] D. Jurafsky and J.H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall series in artificial intelligence. Pearson Prentice Hall, 2009. ISBN: 9780131873216. URL: <https://books.google.com.ar/books?id=fZmj5UNK8AQC>.
- [24] Mike Lewis et al. “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”. In: *ACL*. 2020.
- [25] R. LIKERT. “A technique for the measurement of attitudes”. In: *Arch Psych* 140 (1932), p. 55. URL: <https://cir.nii.ac.jp/crid/1573105975534220928>.
- [26] Chin-Yew Lin. “ROUGE: A Package for Automatic Evaluation of Summaries”. In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. URL: <https://aclanthology.org/W04-1013>.

- [27] Jiasen Lu et al. “12-in-1: Multi-Task Vision and Language Representation Learning”. In: *CoRR* abs/1912.02315 (2019). arXiv: 1912.02315. URL: <http://arxiv.org/abs/1912.02315>.
- [28] A. Pérez Orozco. “Perspectivas feministas en torno a la economía: el caso de los cuidados”. In: 2006.
- [29] Kishore Papineni et al. “Bleu: a Method for Automatic Evaluation of Machine Translation”. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, July 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: <https://aclanthology.org/P02-1040>.
- [30] Jorge Parraga-Alava. *LeLePhid: An Images Dataset for Aphids Detection and Infestation Severity on Lemons Leaf V1*. 2021.
- [31] Colin Raffel et al. “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer”. In: *CoRR* abs/1910.10683 (2019). arXiv: 1910.10683. URL: <http://arxiv.org/abs/1910.10683>.
- [32] Colin Raffel et al. “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer”. In: *Journal of Machine Learning Research* 21.140 (2020), pp. 1–67. URL: <http://jmlr.org/papers/v21/20-074.html>.
- [33] Manuel Romero. *Spanish Bert2Bert (shared) fine-tuned on PAUS-X es for paraphrasing*. 2021. URL: https://huggingface.co/mrm8488/bert2bert_shared-spanish-finetuned-paus-x-paraphrasing (visited on 07/01/2022).
- [34] Manuel Romero. *Spanish T5 (small) fine-tuned on SQAC for Spanish QA*. 2021. URL: <https://huggingface.co/mrm8488/spanish-t5-small-sqac-for-qa> (visited on 07/01/2022).
- [35] Manuel Romero. *T5-base fine-tuned on Quora question pair dataset for Question Paraphrasing*. 2020. URL: <https://huggingface.co/mrm8488/t5-small-finetuned-quora-for-paraphrasing> (visited on 07/01/2022).
- [36] Ananya B. Sai, Akash Kumar Mohankumar, and Mitesh M. Khapra. “A Survey of Evaluation Metrics Used for NLG Systems”. In: *CoRR* abs/2008.12009 (2020). arXiv: 2008.12009. URL: <https://arxiv.org/abs/2008.12009>.
- [37] Yves Scherrer. “TaPaCo: A Corpus of Sentential Paraphrases for 73 Languages”. English. In: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 6868–6873. ISBN: 979-10-95546-34-4. URL: <https://aclanthology.org/2020.lrec-1.848>.
- [38] Lakshay Sharma et al. “Natural Language Understanding with the Quora Question Pairs Dataset”. In: *CoRR* abs/1907.01041 (2019). arXiv: 1907.01041. URL: <http://arxiv.org/abs/1907.01041>.
- [39] Carlos Guadarrama-Zugasti Stephen R. Gliessman. “Agroecología un enfoque sustentable de la Agricultura Ecológica”. In:

-
- [40] Jeremy Tusubira. *Cassava Whitefly Dataset V2*. 2022.
- [41] Marilyn A. Walker, Jeanne C. Fromer, and Shrikanth Narayanan. “Learning Optimal Dialogue Strategies: A Case Study of a Spoken Dialogue Agent for Email”. In: *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*. 1998. URL: <https://aclanthology.org/C98-2214>.
- [42] Xiaoping Wu et al. “IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition”. In: *IEEE CVPR*. 2019, pp. 8787–8796.
- [43] Linting Xue et al. “mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer”. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021, pp. 483–498. DOI: 10.18653/v1/2021.naacl-main.41. URL: <https://aclanthology.org/2021.naacl-main.41>.
- [44] Jingqing Zhang et al. *PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization*. 2019. arXiv: 1912.08777 [cs.CL].

Los abajo firmantes, miembros del Tribunal de evaluación de tesis, damos fe que el presente ejemplar impreso se corresponde con el aprobado por este Tribunal.

