



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA



Machine Learning Operations (MLOps) Clase 5

Leticia Rodríguez

Septiembre 2024 - 2do Cuatrimestre - 4to. Bimestre

Universidad de Buenos Aires - FCEyN - Departamento de Computación

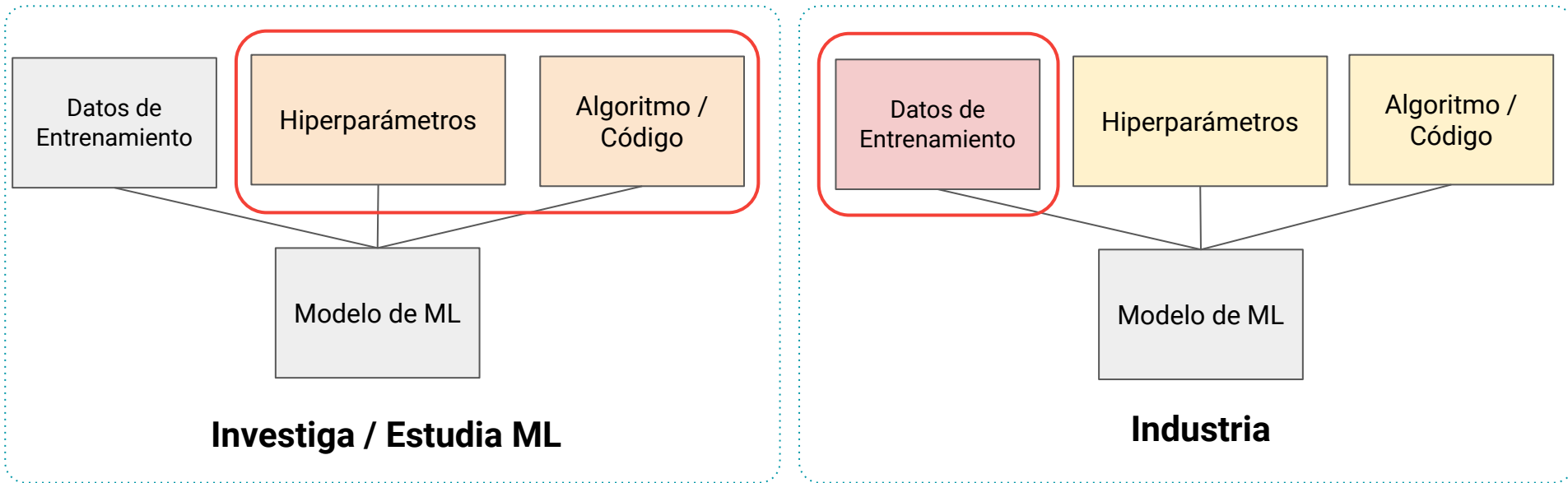
Asistencia

Kahoot de respaso

Aprendiendo de la Experiencia - Post mortem

How ML Breaks:
A Decade of Outages for One Large
ML Pipeline

Armando modelos para problemas de usuarios



Modelos pre-entrenados: ejemplo MTCNN

Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks

Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, *Senior Member, IEEE*, and Yu Qiao, *Senior Member, IEEE*

Abstract—Face detection and alignment in unconstrained environment are challenging due to various poses, illuminations and occlusions. Recent studies show that deep learning approaches can achieve impressive performance on these two tasks. In this paper, we propose a deep cascaded multi-task framework which exploits the inherent correlation between them to boost up their performance. In particular, our framework adopts a cascaded structure with three stages of carefully designed deep convolutional networks that predict face and landmark location in a coarse-to-fine manner. In addition, in the learning process, we propose a new online hard sample mining strategy that can improve the performance automatically without manual sample selection. Our method achieves superior accuracy over the state-of-the-art techniques on the challenging FDDB and WIDER FACE benchmark for face detection, and AFLW benchmark for face alignment, while keeps real time performance.

Index Terms—Face detection, face alignment, cascaded convolutional neural network

I. INTRODUCTION

FACE detection and alignment are essential to many face applications, such as face recognition and facial expression analysis. However, the large visual variations of faces, such as occlusions, large pose variations and extreme lightings, impose great challenges for these tasks in real world applications.

The cascade face detector proposed by Viola and Jones [2] utilizes Haar-Like features and AdaBoost to train cascaded

performance of CNNs in computer vision tasks, some of the CNNs based face detection approaches have been proposed in recent years. Yang *et al.* [11] train deep convolution neural networks for facial attribute recognition to obtain high response in face regions which further yield candidate windows of faces. However, due to its complex CNN structure, this approach is time costly in practice. Li *et al.* [19] use cascaded CNNs for face detection, but it requires bounding box calibration from face detection with extra computational expense and ignores the inherent correlation between facial landmarks localization and bounding box regression.

Face alignment also attracts extensive interests. Regression-based methods [12, 13, 16] and template fitting approaches [14, 15, 7] are two popular categories. Recently, Zhang *et al.* [22] proposed to use facial attribute recognition as an auxiliary task to enhance face alignment performance using deep convolutional neural network.

However, most of the available face detection and face alignment methods ignore the inherent correlation between these two tasks. Though there exist several works attempt to jointly solve them, there are still limitations in these works. For example, Chen *et al.* [18] jointly conduct alignment and detection with random forest using features of pixel value difference. But, the handcraft features used limits its performance. Zhang *et al.* [20] use multi-task CNN to improve the accuracy of multi-view face detection, but the detection accuracy is limited by the initial detection windows produced by a weak face detector

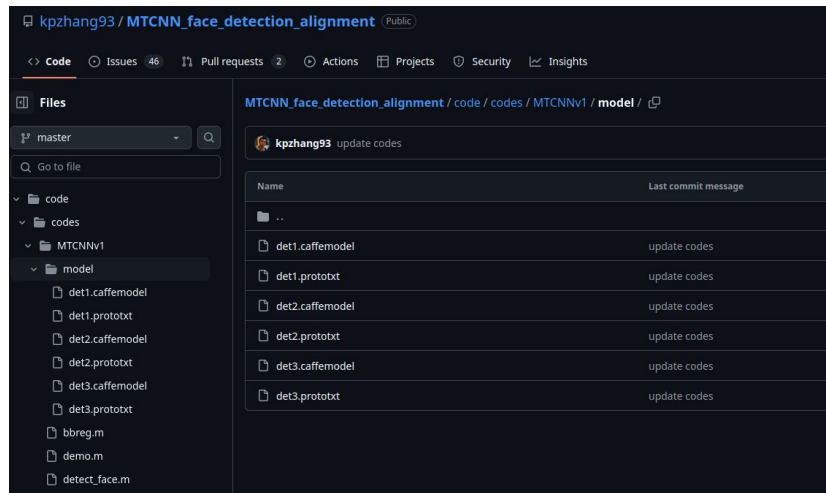


arXiv

<https://arxiv.org> · cs · Traducir esta página

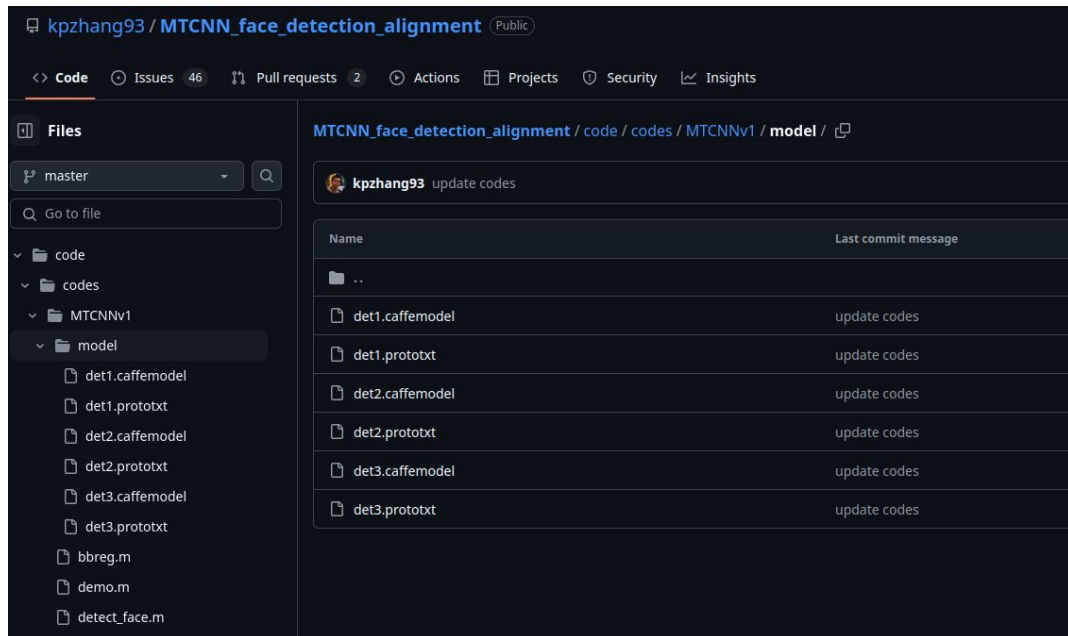
Joint Face Detection and Alignment using Multi-task ...

por K Zhang · 2016 · Mencionado por **6591** — In this **paper**, we propose a deep cascaded multi-task framework which exploits the inherent correlation between them to boost up their...



<https://arxiv.org/pdf/1604.02878>

Modelos pre-entrenados: MTCNN



The screenshot shows the GitHub repository `kpzhang93 / MTCNN_face_detection_alignment`. The file browser on the left shows the directory structure: `code` (containing `master`), `codes` (containing `MTCNNv1`), and `model` (containing `det1.caffemodel`, `det1.prototxt`, `det2.caffemodel`, `det2.prototxt`, `det3.caffemodel`, `det3.prototxt`, `bbreg.m`, `demo.m`, and `detect_face.m`). The main content area shows the commit history for the `model` directory, with a table listing files and their commit messages.

Name	Last commit message
..	
det1.caffemodel	update codes
det1.prototxt	update codes
det2.caffemodel	update codes
det2.prototxt	update codes
det3.caffemodel	update codes
det3.prototxt	update codes

Installation

MTCNN can be installed via pip:

```
pip install mtcnn
```

MTCNN requires Tensorflow >= 2.12. This external dependency can be installed along with MTCNN via:

```
pip install mtcnn[tensorflow]
```

Usage Example

```
from mtcnn import MTCNN
from mtcnn.utils.images import load_image

# Create a detector instance
detector = MTCNN(device="CPU:0")

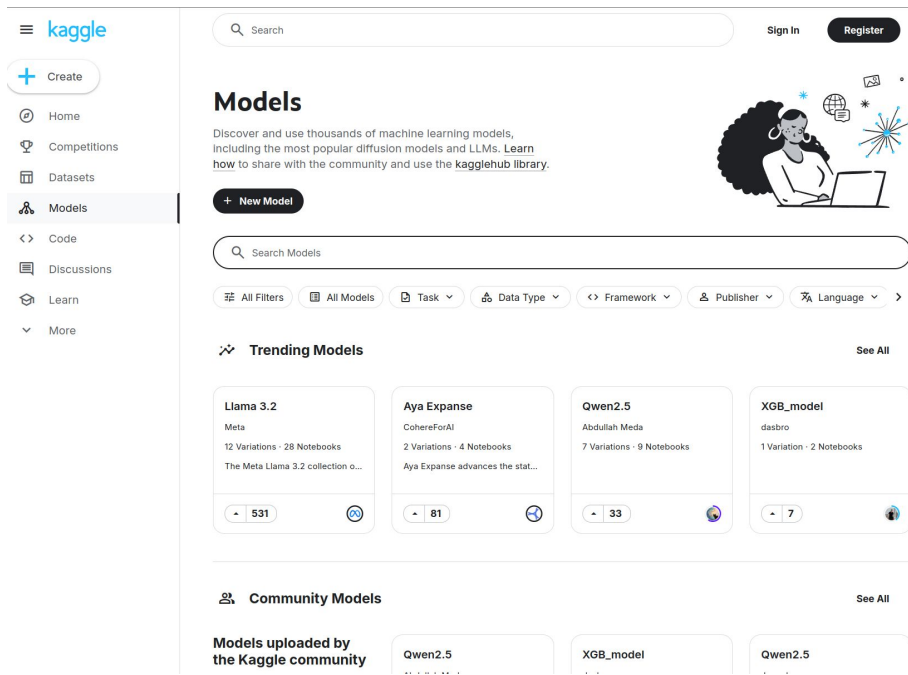
# Load an image
image = load_image("ivan.jpg")

# Detect faces in the image
result = detector.detect_faces(image)

# Display the result
print(result)
```

Modelos pre-entrenados: API

- Se ofrecen distintos modelos pre-entrenados que son el estado del arte de diversas compañías. Podemos nombrar, entre otras:
 - Resnet
 - VGGNet
 - Inception
 - BERT
 - ChatGPT
 - Gemini
 - ROBERTa
 - Gemma
 - Llama
 - Mistral
 - Stable Diffusion
- Los entrenamientos de modelos llevan tiempo de procesamiento, datos y trabajo. Usar un modelo pre-entrenado evita dicha inversión y muchos están disponibles de manera gratuita.
- Incluso algunos como los modelos de lenguaje sería muy caro entrenarlos desde 0 obteniendo resultados similares.



Modelos pre-entrenados: en la Nube

Google Cloud

MLops-letyodri

Buscar (/) recursos, documentos, productos y más

Relaunch to update

Comienza tu prueba gratuita con un crédito de \$300. No te preocupes, no se te cobrará si se acaban los créditos. [Más información](#)

DESCARTAR

COMENZAR GRATIS

Vertex AI

Model Garden

EXPLORAR LA IA GENERATIVA

VER MIS EXTREMOS Y MODELOS

DEPLOY FROM HUGGING FACE

VER NOTAS DE LA VERSIÓN

TOOLS

Panel

Model Garden

Canalizaciones

NOTEBOOKS

Colab Enterprise

Workbench

VERTEX AI STUDIO

Descripción general

Formato libre

Chat

Vision

Traducción

Voz

Galería de instrucciones

Administración de Instruc...

Ajuste

BUILD WITH GEN AI

Extensiones

DATA

Feature Store

Conjuntos de datos

Tareas de etiquetado

MODEL DEVELOPMENT

Entrenamiento

Modalidades

Lenguaje66

Vision88

Tabulares7

Documento8

Voz2

Video6

Multimodal21

Audio1

Tareas

Generación74

Clasificación66

Detección44

Extracción28

Reconocimiento26

Traducción23

Eembedding7

Segmentación12

Recuperación2

Detección de vocabulario abierto2

Segmentación de vocabulario abierto2

Tracking1

Previsión6

Reconocimiento de voz automático1

Buscar modelos

Browse, customize, and deploy machine learning models with Model Garden. Choose from models created by Google and other providers.

Gemini

Imagen 3

Gemma 2

Sort by: [Trending](#) [Newest](#) [Last Update](#)

Modelos de base

Modelos para tareas múltiples previamente entrenados que se pueden ajustar o personalizar aún más para tareas específicas.

Gemini 1.5 Pro

Gemini 1.5 Flash

Gemini 1.0 Pro

Gemini 1.0 Pro V

Featured partners

ANTHROPIC

Meta

Hugging Face

M. A.

Open models on Hugging Face

Deploy some of the most popular open source models from Hugging Face to Vertex AI.

aws.amazon.com/what-is/foundation-models/

Guílines for Eth... ACL, Rolling Review... Complete Guide T... Onboarding Minimalist Green... AI Playlists Onboarding - Task... Conferences Finals RL AI Read Next AI

Acerca de AWS Contacto Soporte Español Mi cuenta Iniciar sesión

Productos Soluciones Precios Documentación Aprender Red de socios AWS Marketplace Habilitación para clientes Eventos Explorar más

¿Qué es la informática en la nube? Centro de conceptos de computación en la nube IA generativa Gen-AI

¿Qué son los modelos fundacionales?

Cree una cuenta de AWS

Explorar servicios de IA generativa

Explorar la IA generativa en AWS

Ver capacitaciones en IA generativa

Leer blogs de IA generativa

¿Qué es un modelo fundacional?

¿Qué tienen de especial los modelos fundacionales?

¿Por qué es importante el modelo fundacional?

¿Cómo funcionan los modelos fundacionales?

¿Qué pueden hacer los modelos fundacionales?

¿Cuáles son algunos ejemplos de modelos fundacionales?

¿Cuáles son los desafíos de los modelos fundacionales?

¿De qué manera AWS puede ayudar?

¿Qué es un modelo fundacional?

Entrenados con conjuntos de datos masivos, los modelos fundacionales (FM) son redes neuronales de aprendizaje profundo que cambiaron la forma en que los científicos abordan el machine learning (ML). En lugar de desarrollar la inteligencia artificial (IA) desde cero, los científicos de datos utilizan un modelo fundacional como punto de partida para desarrollar modelos de ML que impulsen aplicaciones nuevas de manera rápida y rentable. El término modelo fundacional fue acuñado por los investigadores para describir los modelos de ML entrenados en un amplio espectro de datos generalizados y sin etiquetar y capaces de realizar una gran variedad de tareas generales como comprender el lenguaje, generar texto e imágenes y conversar en lenguaje natural.

¿Qué tienen de especial los modelos fundacionales?

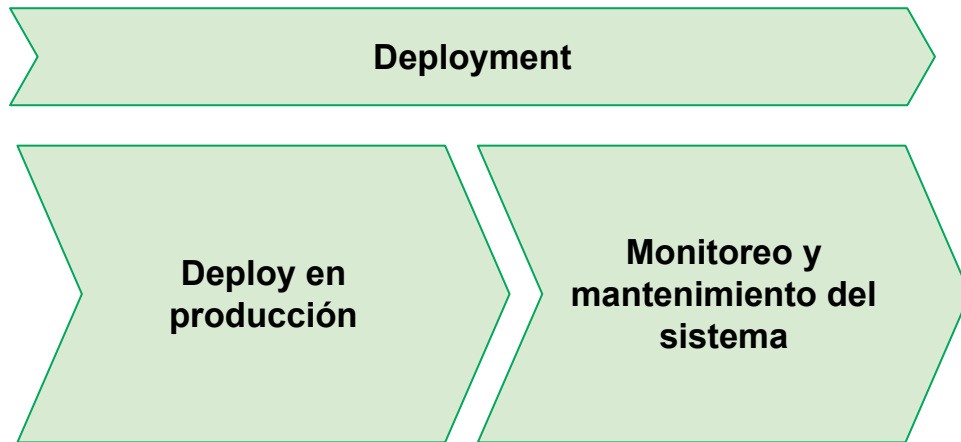
Una característica única de los modelos básicos es su adaptabilidad. Estos modelos pueden realizar una amplia gama de tareas dispares con un alto grado de precisión en función de las indicaciones de entrada. Algunas tareas incluyen el procesamiento de lenguaje natural (NLP), la respuesta a preguntas y la clasificación de imágenes. El tamaño y la naturaleza de uso general de los modelos básicos los diferencian de los modelos de machine learning tradicionales, que suelen realizar tareas específicas, como analizar texto en busca de opiniones, clasificar imágenes y pronosticar tendencias.

Puede utilizar los modelos fundacionales como modelos de base para desarrollar aplicaciones posteriores. Los modelos son la culminación de más de una década de trabajo que los vio aumentar en tamaño y complejidad.

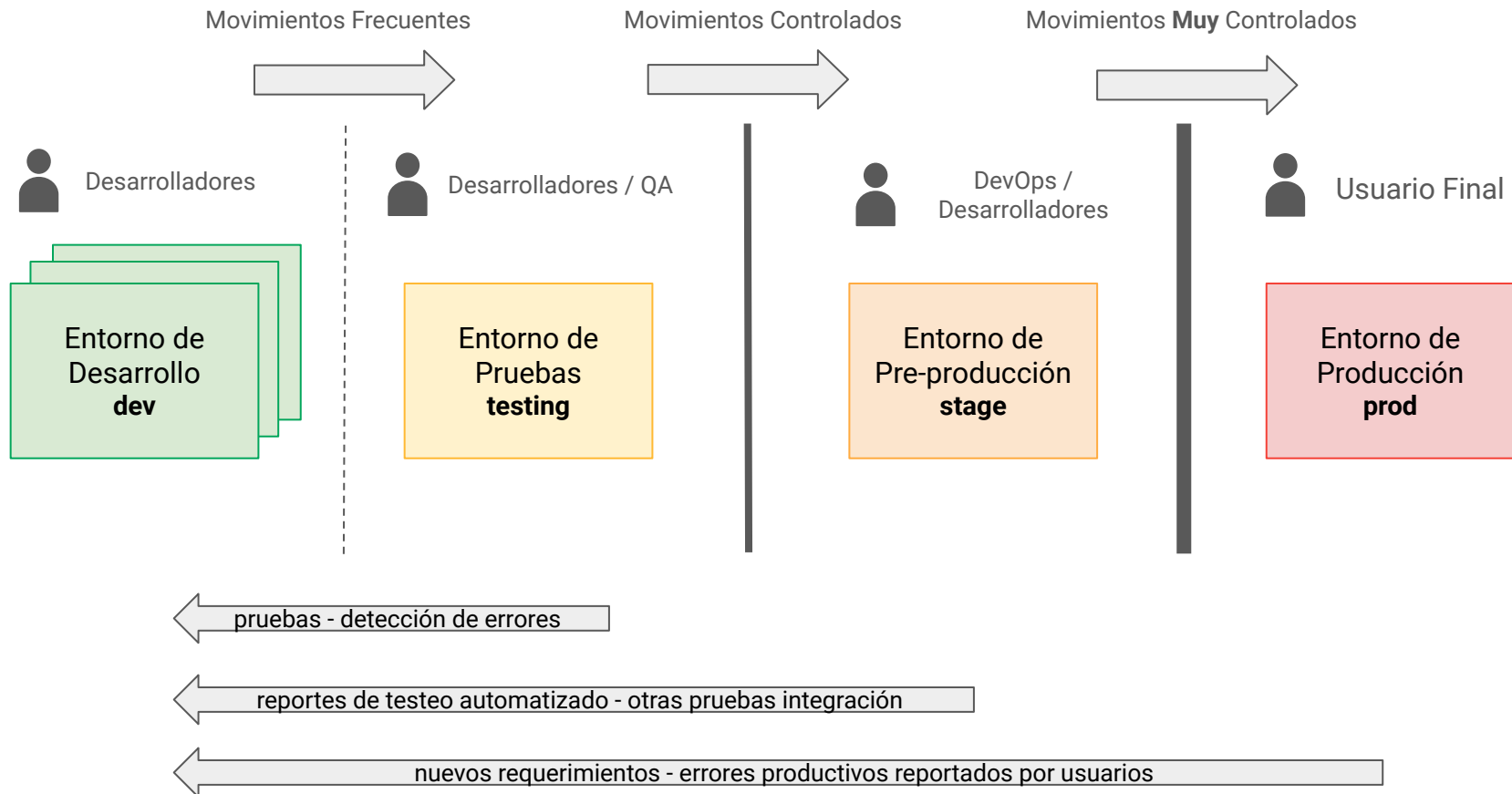
Por ejemplo, BERT, uno de los primeros modelos fundacionales bidireccionales, se lanzó en 2018. Se entrenó con 1.600 millones de parámetros y un conjunto de datos de entrenamiento de 16 GB. En 2023, solo cinco años después, OpenAI entrenó el GPT-4 mediante la utilización de 170 billones de parámetros y un conjunto de datos de entrenamiento de 45 GB. Según OpenAI, la

Puedo ponerlo en contexto de AWS.

Deploy



Los estándares: Ambientes del desarrollo de Software



Métricas de Negocio

- Métricas de ML son necesarias: f1-score, recall, accuracy, precision, otras.
- También, es importante demostrar el impactó que pueden generar en el negocio y esto se hace atraves de distintas métricas de negocio.
- Las Métricas de Negocio están relacionadas al problema a resolver y pueden ser más técnicas o más orientada a los negocios. Muchas incluso llegan a los altos niveles ejecutivos. Algunos ejemplos de páginas web o marketing:
 - Impressions - impresiones
 - CTR -- Click Through Rate - clicks rate: Por ejemplo, en artículos recomendados por una AI, cuantos reciben click
 - Conversion Rate - ratio de conversión: Por ejemplo, en artículos recomendados por una AI, cuantos se transforman en venta
 - ROI - Retorno de la inversión

Actividad: Métricas de ML vs Métricas de Negocio

Del paper: **150 Successful Machine Learning Models: 6 Lessons Learned at Booking.com**

1. Primero vemos como realizan la evaluacion de los modelos en la sección 7. Evaluation.
2. Luego, en grupos, la sección **Sección: 3. Modeling: Offline Model Performance is just a Heath Check** y respondan.

¿Qué conclusión sacaron los investigadores respecto a las performance del modelo en relación al negocio?

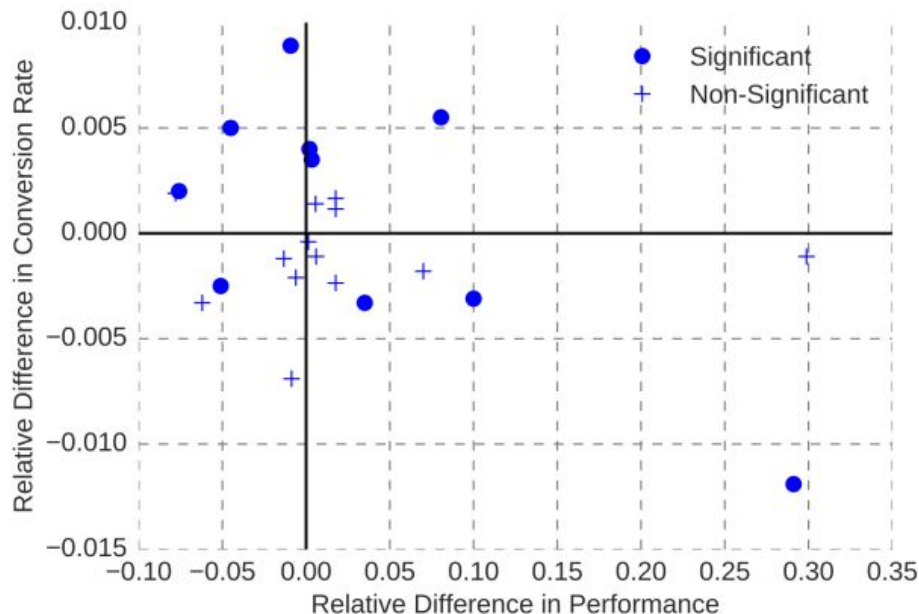
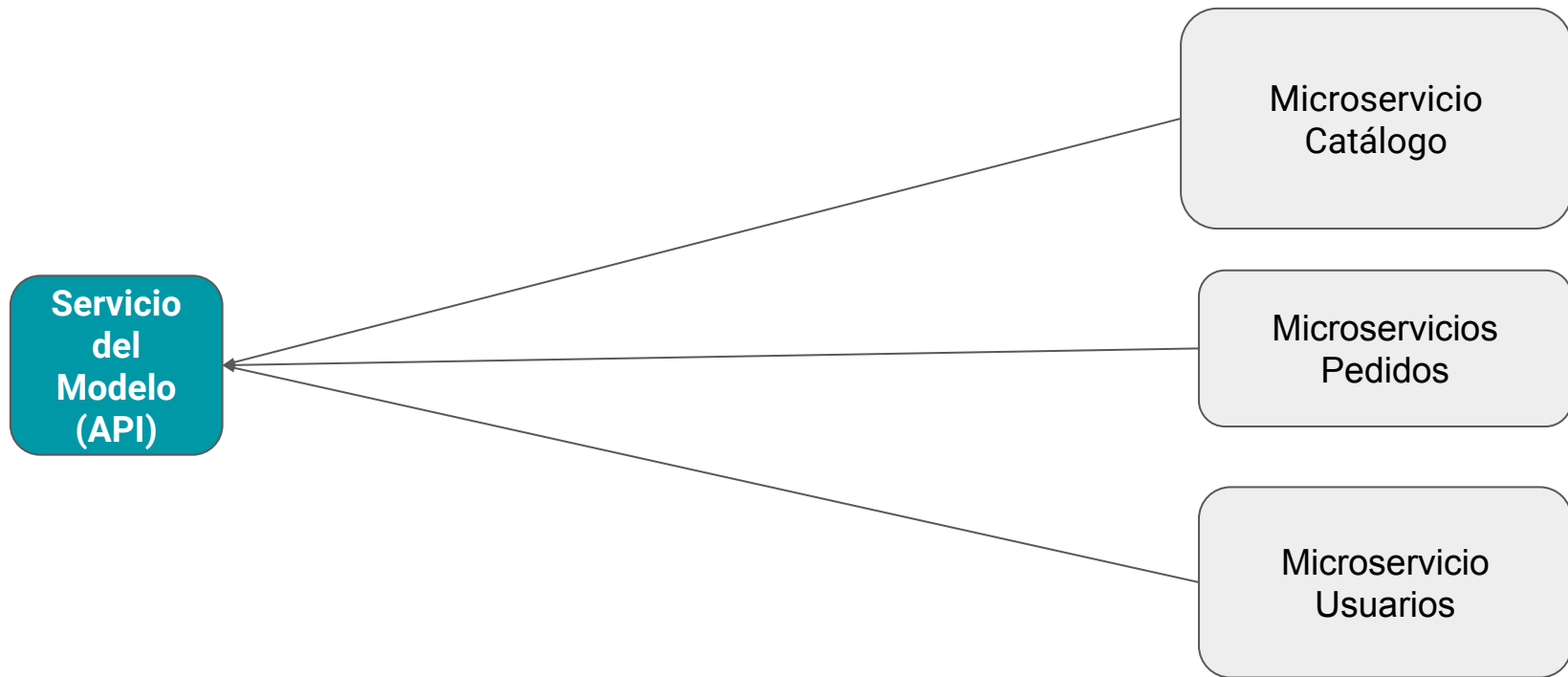


Figure 4: Relative difference in a business metric vs relative performance difference between a baseline model and a new one.

Ejemplo de Integración en el software



Ejemplo de Integración en el software

Hola Agatha



Vendo Auto 90.000 KMs

Vendedor: Cosme Fulanito

Precio: \$ 1.000.000

Comprar

Otros productos que pueden interesarte



*Recomendaciones generadas con
AI consultadas en la API para el
usuario*

Deploy - Subir a producción

Deploy: la tarea de subir / poner nuestro código en uso para los usuarios o consumidores del sistema.

Deployar un sistema es una tarea riesgosa, y desplegar un sistema de ML en producción lo es aún más.

En ML, varias cosas pueden ir mal:

- Fallas, bugs, errores, páginas o APIs que no responden o accesos restringidos
- Servidores sobrecargados de pedidos, respuesta lenta (baja latencia)
- Errores funcionales, no hace lo que se espera que haga
- Errores probabilísticos, para algunos usuarios, las respuestas no son satisfactorias

En la práctica, casi ningún sistema es disponibilizado para la totalidad de sus consumidores o los deploys se hacen de manera controlada. Casualmente, estas maneras controladas de hacer deploys pueden servir para testear en forma online el sistema de AI



Probar los sistemas de ML

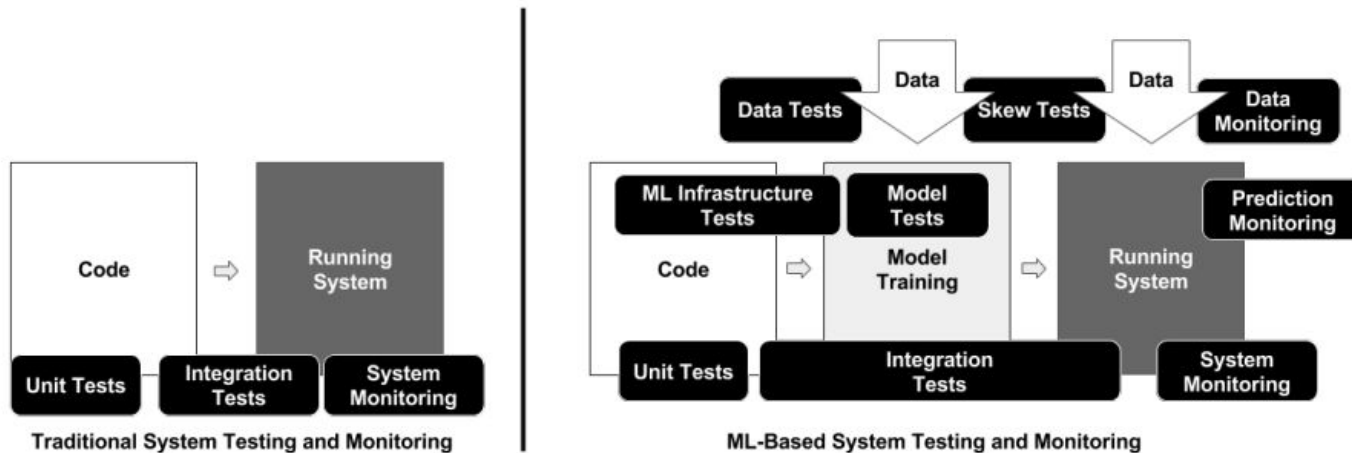
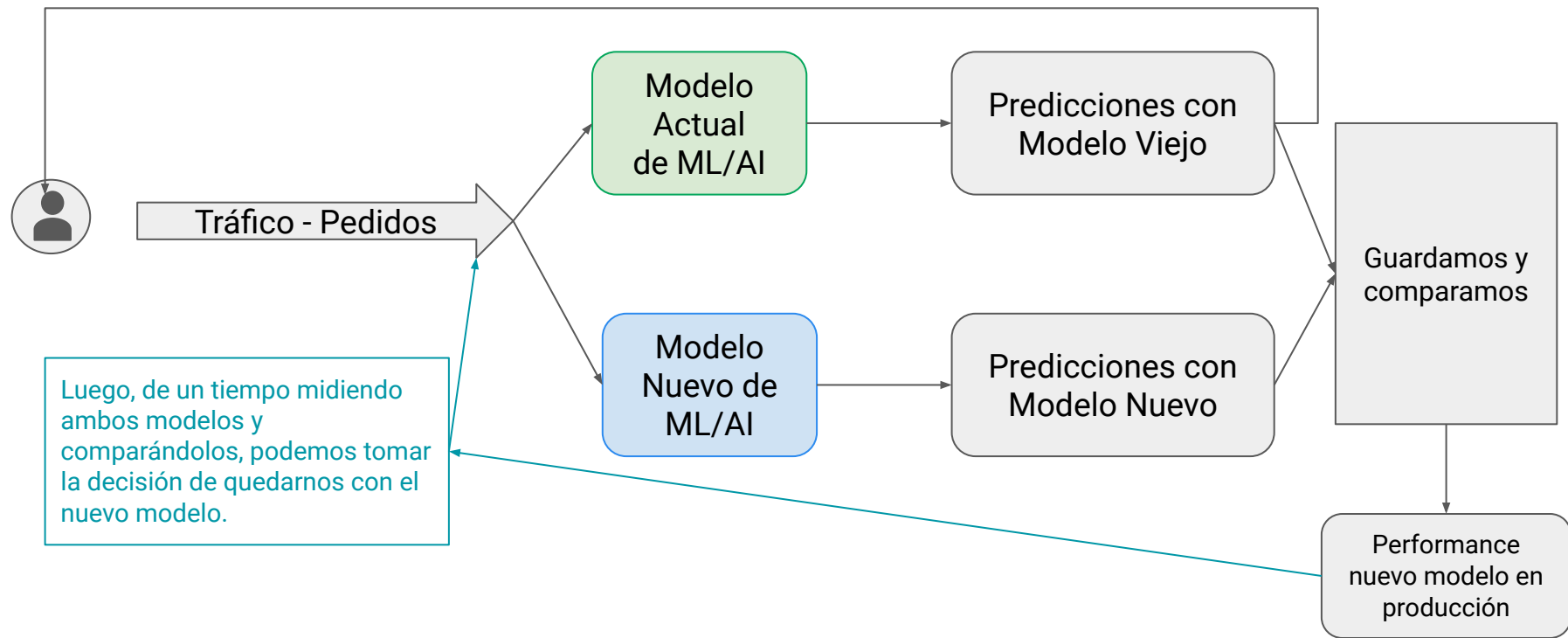


Figure 1. **ML Systems Require Extensive Testing and Monitoring.** The key consideration is that unlike a manually coded system (left), ML-based system behavior is not easily specified in advance. This behavior depends on dynamic qualities of the data, and on various model configuration choices.

Los modelos requieren una combinación de testeo al momento de la creación / desarrollo / implementación del sistema de ML (offline evaluation) y evaluaciones / monitoreos al momento de estar en producción (online evaluation)

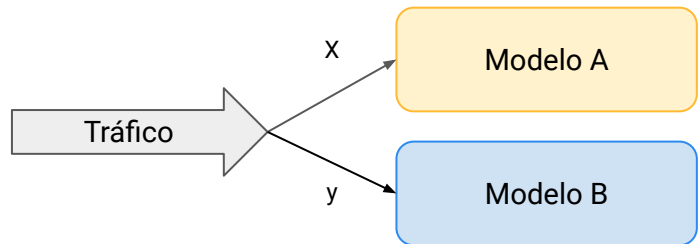
Shadow Deployment para el modelo



Lo mismo aplicaría para el sistema si estamos cambiando el sistema de AI entero

A/B Testing

A/B Testing es una forma de comparar dos variantes de un objeto, generalmente es una porción de software, un email, de alguna forma que nos permita decidir cuál es más efectiva.



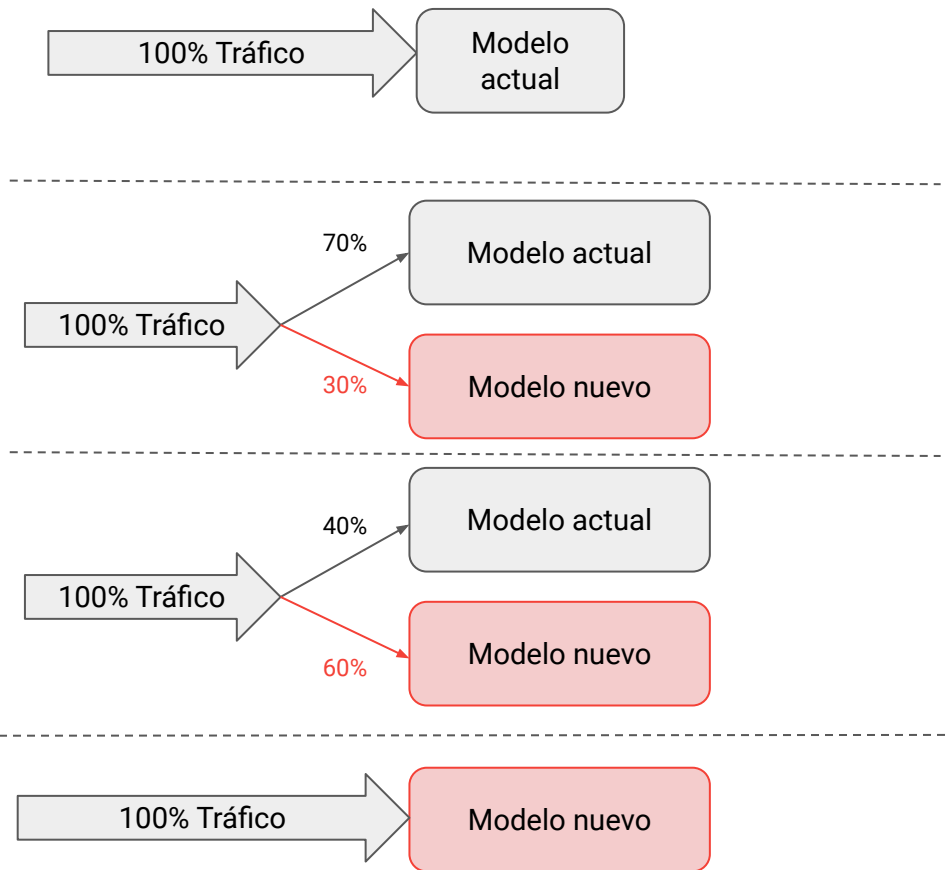
Probamos ambos modelos a ver cuál funciona mejor mandando la misma cantidad de tráfico de manera aleatoria a cada uno y viendo las métricas

x, y son porciones aleatorias del tráfico disjuntas



Analizamos las predicciones de ambos modelos y el feedback del usuario, decidimos cuál vamos a usar.

Canary Deploy



Gradualmente se transfiere el tráfico al nuevo modelo.

Se va redirigiendo el tráfico de a poco y realizando verificaciones al nuevo modelo. Por ejemplo, la primera semana, se manda el 30% del tráfico al nuevo modelo. Una vez que se observa que todo funciona como se espera, a semana siguiente, o más tarde, se manda el 60%. En incrementos secuenciales y validando el funcionamiento en cada paso.

Si los resultados no son los esperados, se vuelve al modelo actual.

El tamaño de los incrementos y el momento depende de las características del problema y la decisión que toma el equipo de ingenieros.

Interleaving Experiments

Hola Agatha



Vendo Auto 90.000 KMs

Vendedor: Cosme Fulanito

Precio: \$ 1.000.000

Comprar

Otros productos que pueden interesarte



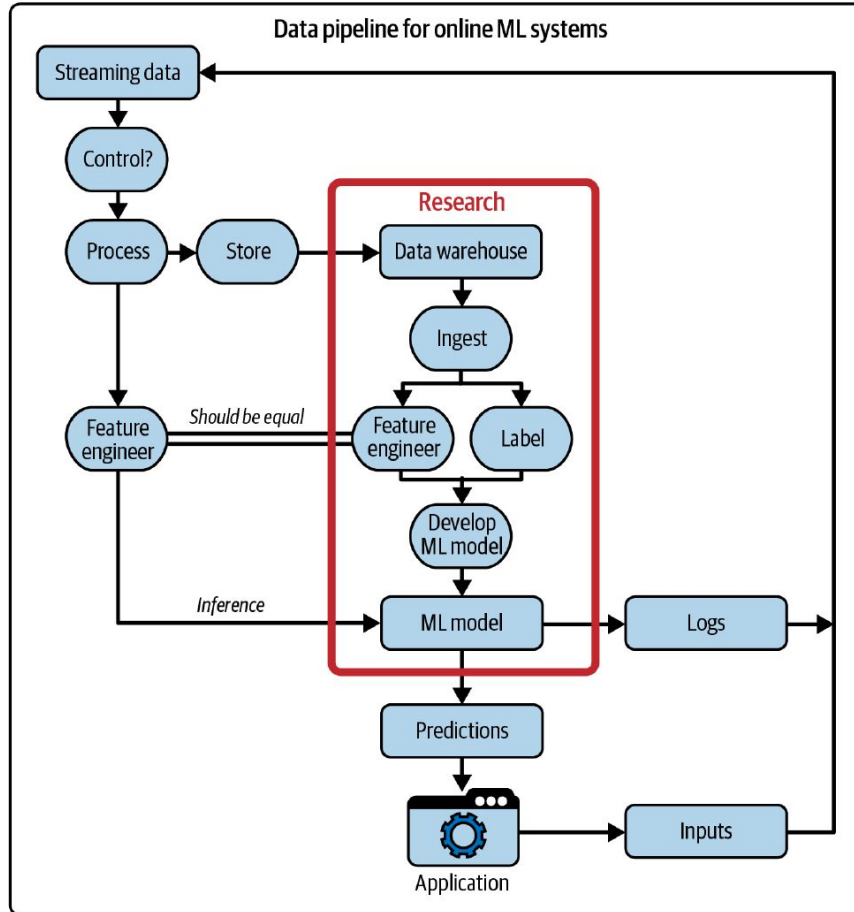
Generado por
Modelo A



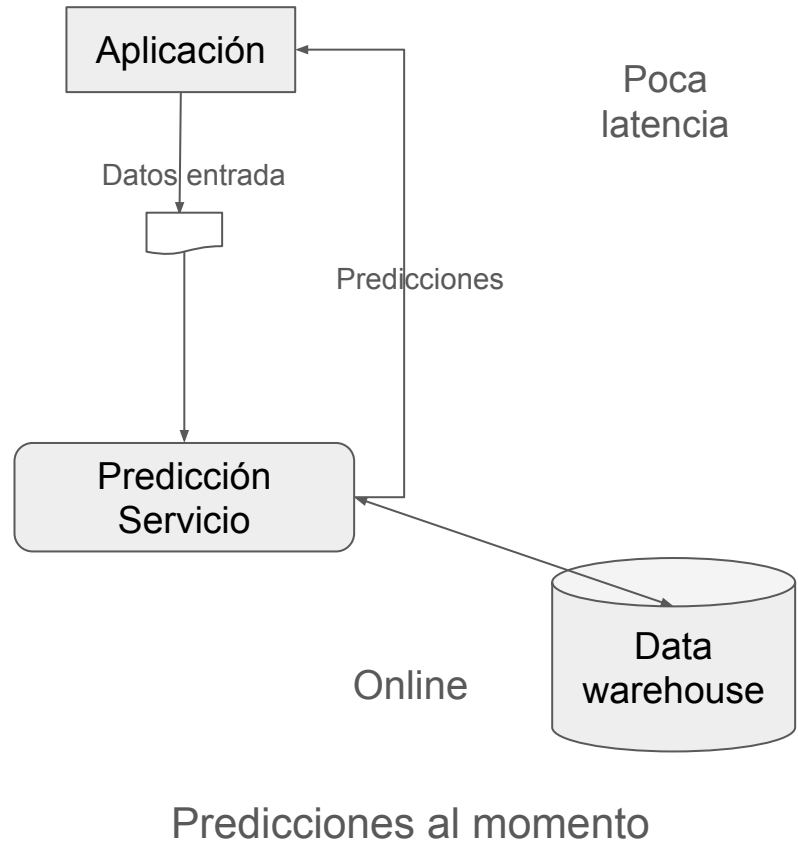
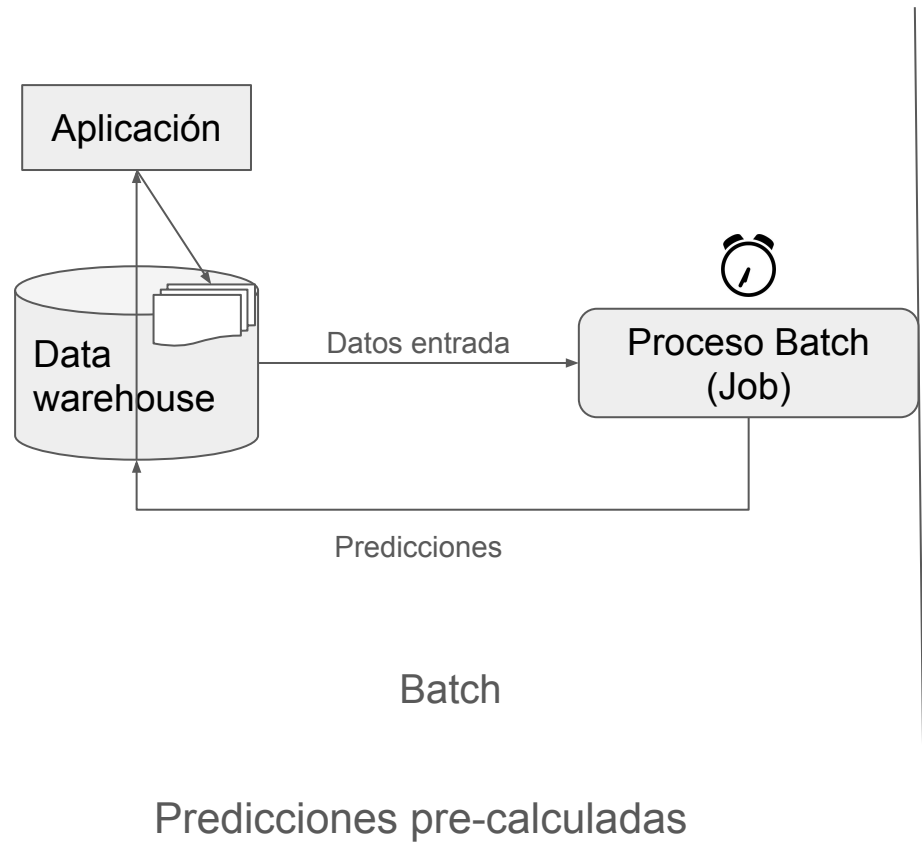
Generado por
Modelo B

En este experimento, se toman recomendaciones de distintos modelos y se observa cuál tiene más aceptación del usuario

Ejemplo de Arquitectura



Predicción Batch vs. Online



Escalabilidad

La infraestructura tiene que ser capaz de agregar más equipos (upscaling) o remover equipos en desuso (downscaling). Esto requiere el monitoreo del equipo y en caso de ser posible, configuraciones de escalamiento automático a partir de métricas. Por ejemplo, cantidad de requests por segundo (solicitudes web http).

Por otro lado, el escalamiento se puede hacer de dos maneras: vertical u horizontal.

- Escalamiento Horizontal: agregar más equipos de las mismas características para contener el incremento de tráfico o recursos requeridos.
- Escalamiento Vertical: aumentar las capacidades de los equipos actuales por ejemplo, agregando más memoria o mejorando los procesadores.



Vertical Scaling
(Scaling up)



Horizontal Scaling
(Scaling out)

Monitoreo

Monitoreo es el acto de traquear, medir y loguear las diferentes métricas que nos pueden ayudar a ver las cosas que pueden salir mal. Generalmente, esa información puede ser usada para investigar que falló o encontrar algún error, a esto se lo denomina observabilidad.

La información de logueo y performance lo de los diferentes niveles son ampliamente usadas en la construcción de software.

Suceden a distintos niveles:

- Equipamiento - CPU - GPU - Memoria - Discos
- Red - Paquetes enviados
- Servidor Web - requests / solicitudes
- Aplicación - Logueo de determinados eventos, funciones

Tenemos que considerar también monitorear métricas de ML

Monitoreo en Machine Learning

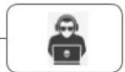
- Entradas de datos en crudo
- Features: validación de schema (test unitarios para datos o table validation o table testing)
 - min, max en rango aceptable
 - valores que satisfagan alguna expresión regular
 - valores pertenezcan a un conjunto determinado
- Predicciones: distrucción, predicciones individuales
- Métricas específicas: Accuracy, F1-score, Precision, Recall
- Cambios en la distribución de los datos entre entranamiento y los usados en producción para predicción
- Recolecciones de feedback

Monitoreo - Herramientas


Logs

```
GNU nano 2.0.9 File: access log
192.168.1.4 - - [09/May/2014:09:34:25 -0400] "GET / HTTP/1.1" 200 100 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:34:25 -0400] "GET /favicon.ico HTTP/1.1" 404 285 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:40:53 -0400] "GET /admin HTTP/1.1" 301 310 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:40:53 -0400] "GET /admin/ HTTP/1.1" 200 103 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:40:53 -0400] "GET /favicon.ico HTTP/1.1" 404 285 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:50:18 -0400] "GET /admin/ HTTP/1.1" 401 478 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:51:05 -0400] "GET /admin/ HTTP/1.1" 401 478 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:09:51:20 -0400] "GET /admin/ HTTP/1.1" 200 103 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:24:19 -0400] "GET /admin HTTP/1.1" 401 478 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:25:34 -0400] "GET /favicon.ico HTTP/1.1" 404 285 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:27:31 -0400] "GET /admin HTTP/1.1" 404 280 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:27:32 -0400] "GET /admin HTTP/1.1" 404 280 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:27:38 -0400] "GET / HTTP/1.1" 200 100 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:30:44 -0400] "GET /admin HTTP/1.1" 301 310 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:30:44 -0400] "GET /admin/ HTTP/1.1" 200 103 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:36:33 -0400] "GET /admin/ HTTP/1.1" 401 478 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:10:37:08 -0400] "GET /admin/ HTTP/1.1" 200 103 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
192.168.1.4 - - [09/May/2014:11:07:18 -0400] "GET /admin HTTP/1.1" 401 478 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 6_8_0; rv:42.0) Gecko/20100101 Firefox/42.0"
^G Get Help ^C WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Alertas



Support/Ops manager



graph

General

Metrics

Axes

Legend

Display

Alert

Time range

Alert Config

Alert Config

Notifications (1)

State history

Delete

Name

Some alert

Evaluate every

300s

Conditions

WHEN

count (1)

OF

query (B, 5m, now)

IS ABOVE

0

+

If no data or all values are null

SET STATE TO

No Data

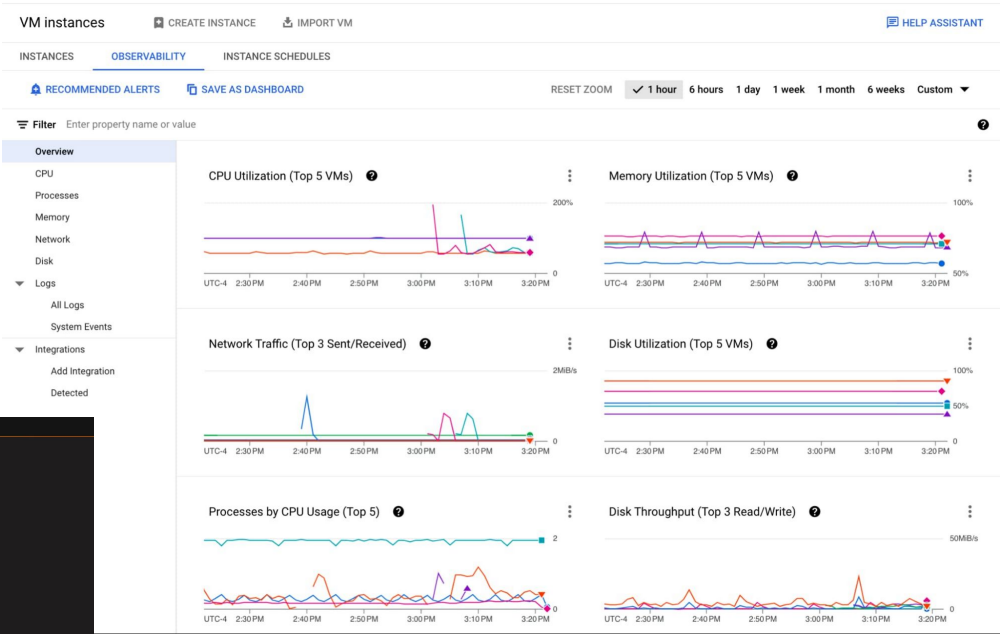
If execution error or timeout

SET STATE TO

Alerting

Test Rule

Dashboards



Degradación de los modelos - Concept drift / Data drift

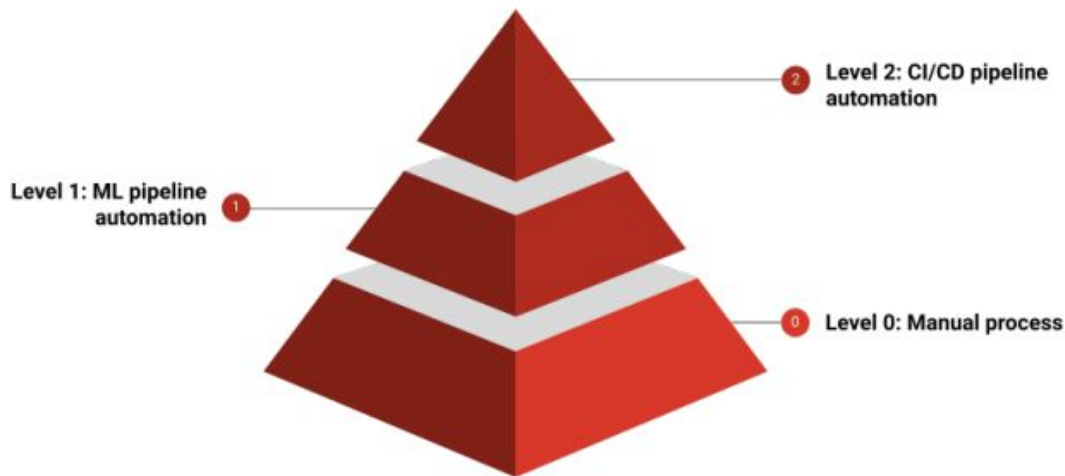
Existen razones para cambiar los modelos o reentrenarlos por el paso del tiempo:

- Los modelos se degradan naturalmente debido a que los datos productivos pueden cambiar en distribución (Data Drift)
- También, los sistemas productivos pueden tener con el correr de los años cambios en los features, ya sea que se agreguen o se eliminen, en incluso que tomen nuevos valores que representen algo nuevo. (Concept Drift)
- La tecnología de AI avanza, surgen nuevos modelos, nuevas optimizaciones, nuevos casos de uso
- Se puede también mejorar el modelo mediante ensemble de modelos
- La llegada de nuevos datos o más información lleva a que sea requerido reentrenar el modelo frecuentemente (ej. en recomendadores)

Niveles de Maduración de MLOps

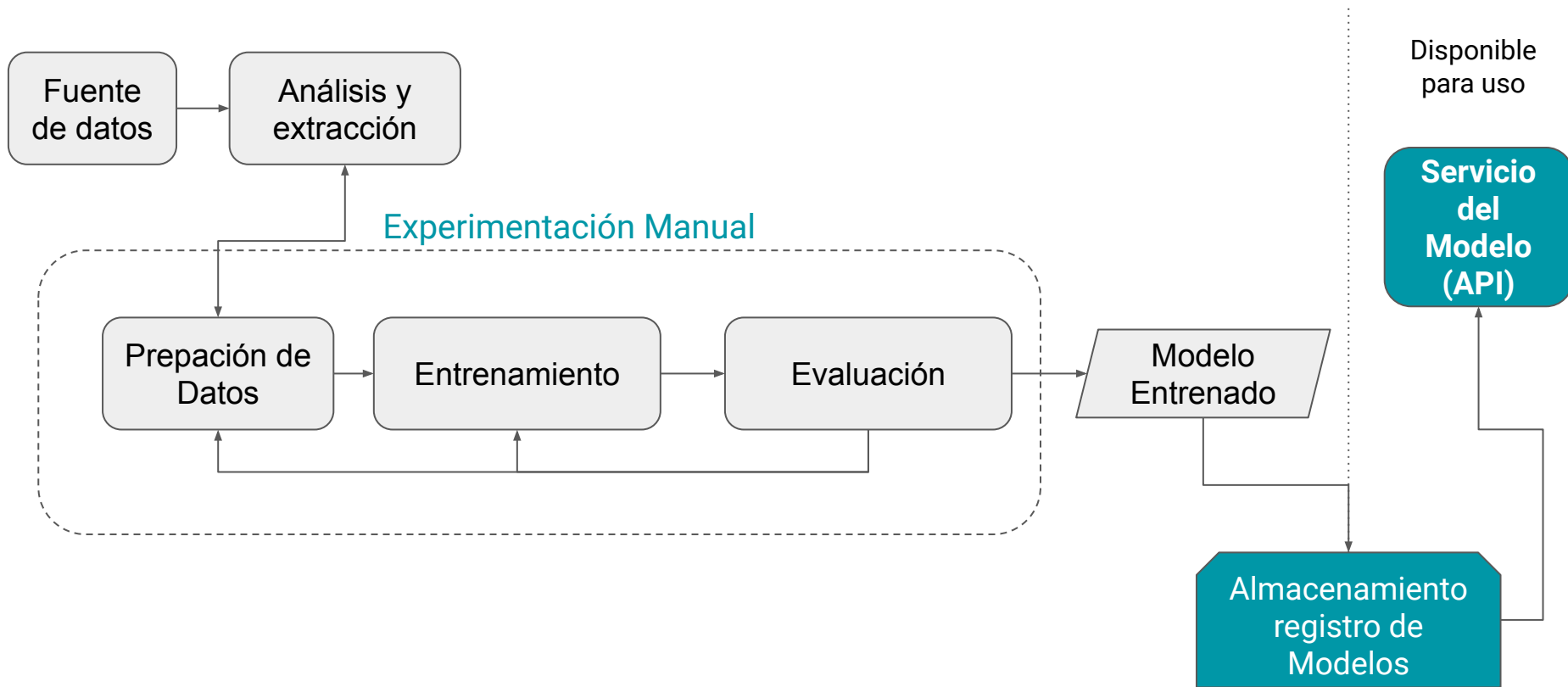
Acorde al grado de automatización / maduración de los distintos pipelines se definen niveles de maduración:

- Nivel 0:
Procesamiento Manual
- Nivel 1:
ML Pipeline automatizado
- Nivel 2:
CI/CD pipeline automatizado



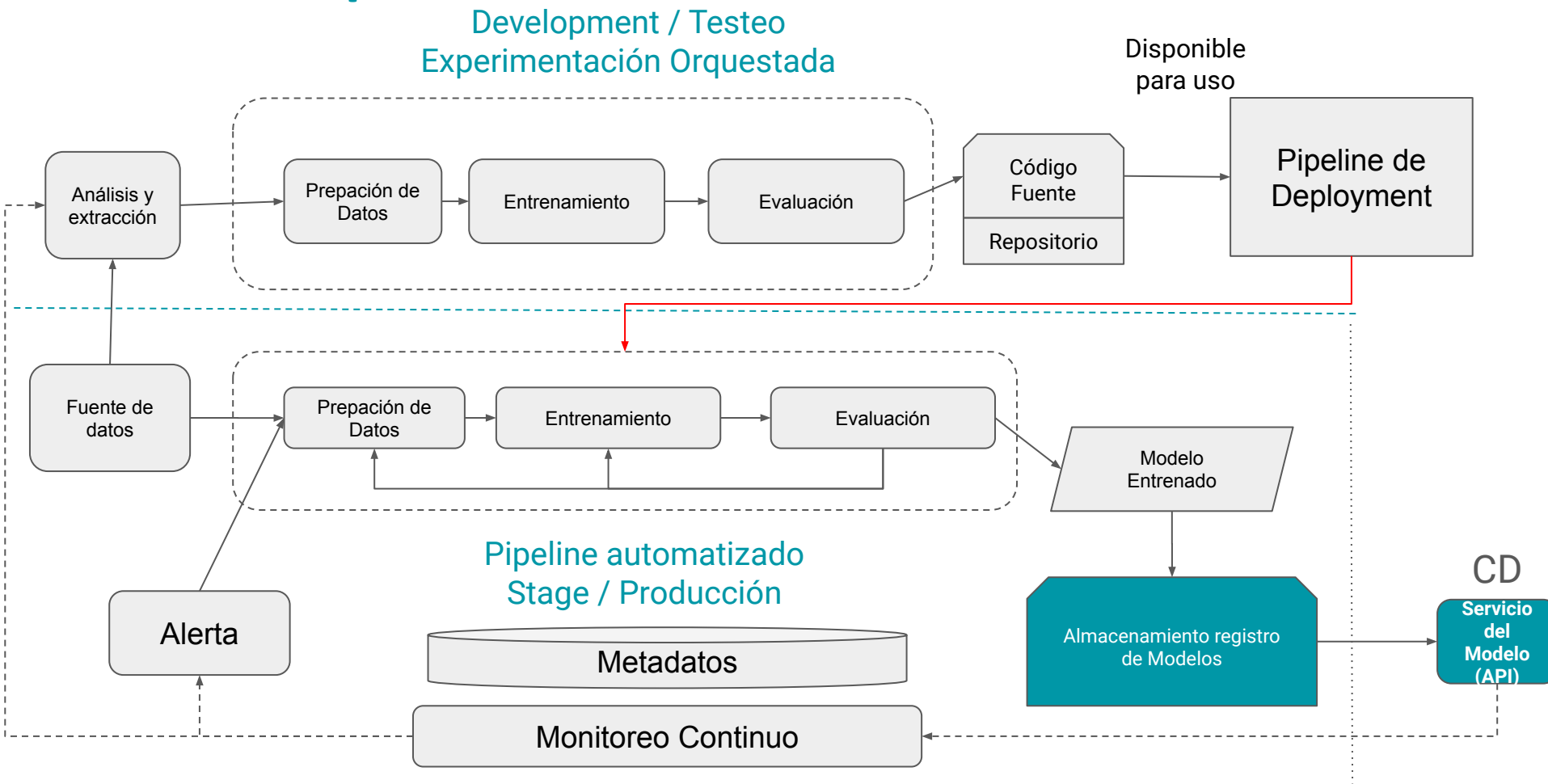
- Los proveedores de cloud pueden definir otros niveles: [Symeonidis et al. MLOps - Definitions, Tools and Challenges, 2022](#)
- Hay trabajos que buscan definirlos: [John et al., Towards MLOps: A Framework and Maturity Model, 2021](#)

Nivel 0 MLOps



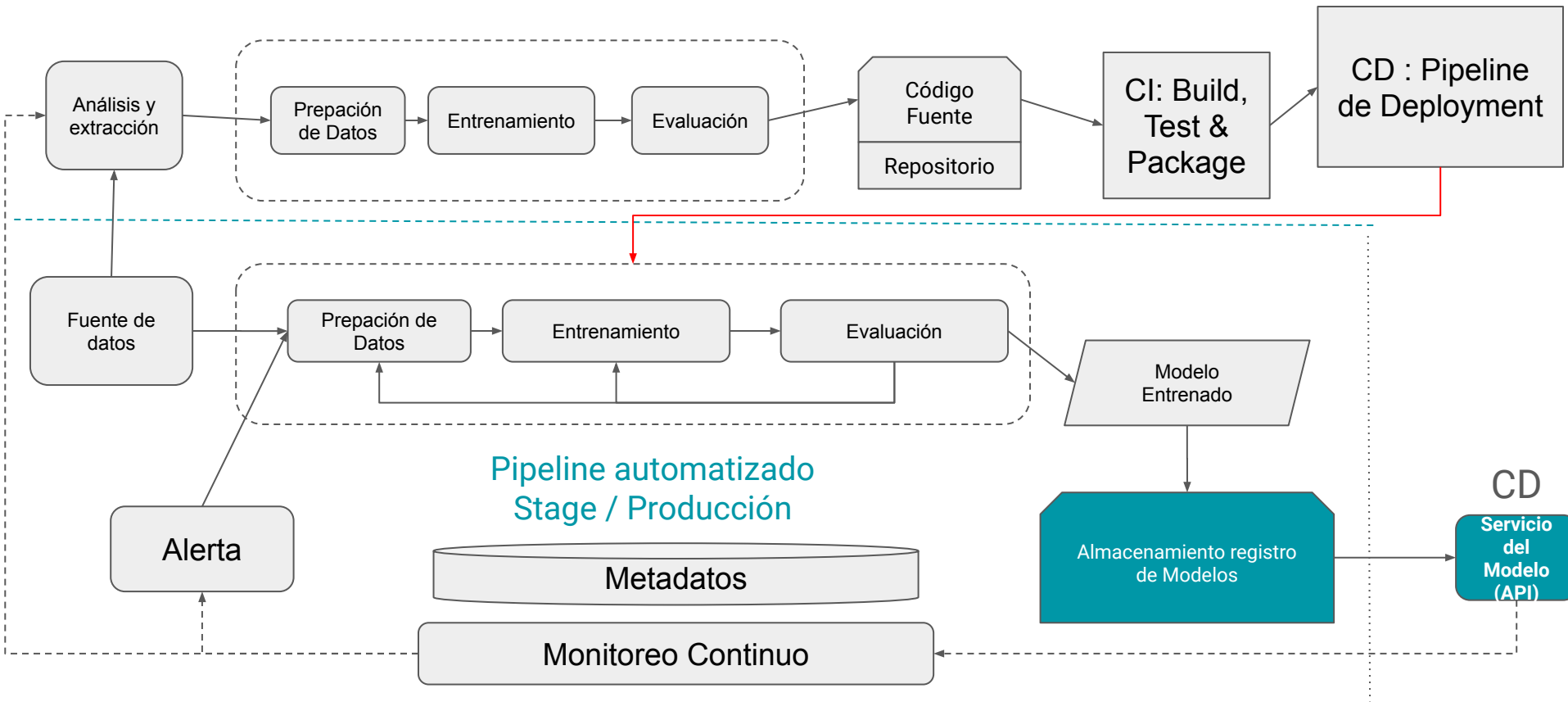
Nivel 1 MLOps

Entrenamiento automatizado y delivery continuo



Nivel 2 MLOps

Development / Testeo
Experimentación Orquestada

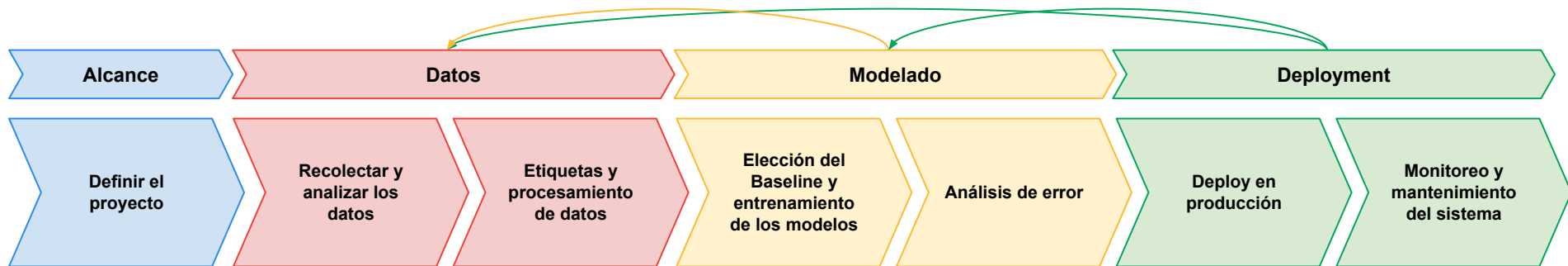


Principios de un buen Sistema de ML por Chip Huyen

1. Resuelve el problema
2. Está testeado
3. Es accesible a los usuarios
4. Es ético
5. Sus componentes son modulares, integrados pero separados
6. Es tan simple como es posible pero no más simple
7. Es transparente
8. Permite el desarrollo iterativo
9. Está versionado
10. Está documentado



El flujo de creación de un modelo de ML



Importante: las diferentes etapas tienen flechas que hacen que se itere entre las etapas anteriores

Material Recomendado de esta semana

[Stanford MLSys Seminar Episode 5: Chip Huyen](#)

[Rules for ML Engineers](#)

[BERT Paper](#)

[KPI in Marketing - Ejemplo metricas de negocio](#)

[The surprising Power of Online Experiments](#)

[Vertex AI Pipelines - Ejemplos](#)

[Grafana](#)