Evaluación y Validación de Sistemas de Recomendación

Cristian Cardellino - Luis Biedma

Contenido

- Evaluación en Sistemas de Recomendación
- Comportamiento del Usuario
- Medición
- Verificación
- Evaluación Offline
- Evaluación Online
- Evaluación Continua

Evaluación en Sistemas de Recomendación

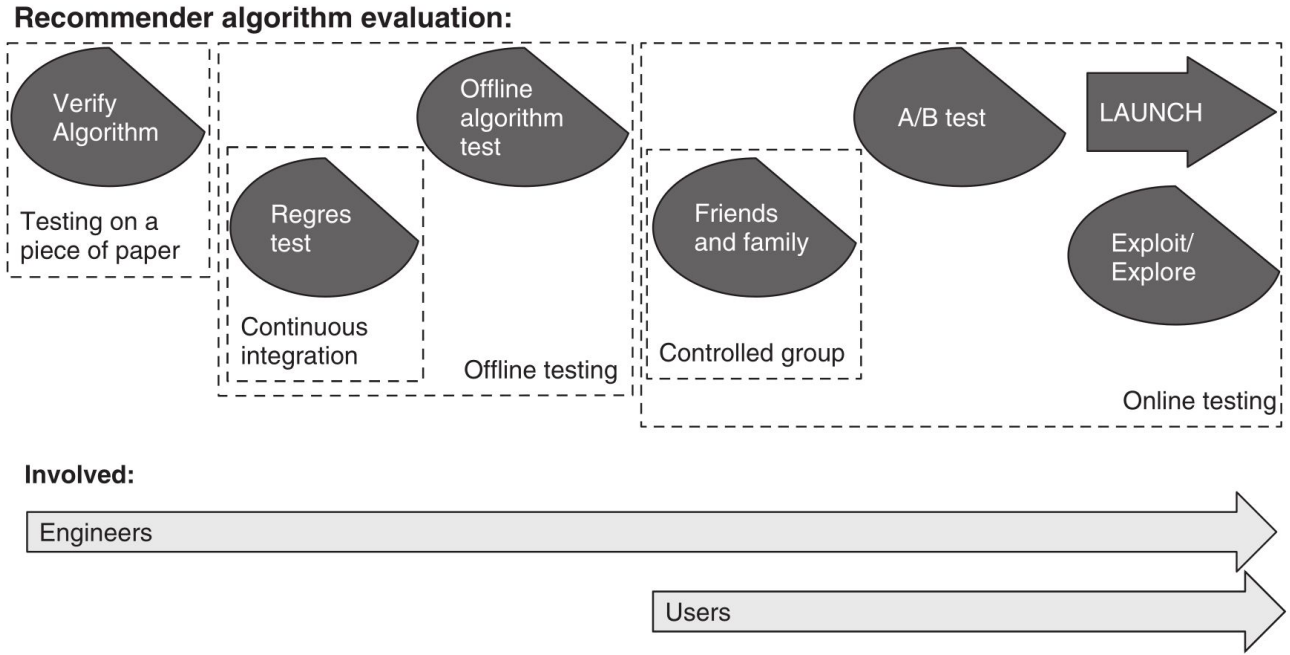
Objetivo de nuestro sistema de recomendación

- Para evaluar un sistema debemos entender **¿Por qué creamos el sistema?** y **¿Qué buscamos ganar?**
    - Varias son las opciones posibles: mayor ganancia monetaria, mayor tráfico en el sitio, prueba de nuevas tecnologías.
- De estas respuestas surgen **distintas técnicas para evaluar** si estamos mejorando.
- El consenso general es que **es difícil evaluar un sistema de recomendación sin ponerlo a prueba en vivo**.

El Ciclo de Evaluación

- Se define un **ciclo de evaluación** para seguir. Arrancando desde lo más simple y avanzando.
- Es importante definir un **Key Performance Indicator (KPI)** sobre el cual mejorar.
  - Muchas veces el **KPI** está dado por el negocio/cliente.
- **No podemos simular tráfico/visitas para evaluar los sistemas**.
- El **sistema cambia de acuerdo a los datos**.
  - Es necesario mantenerlo actualizado.
- Es recomendable **empezar por el algoritmo más sencillo**.
  - Primero ver si **funciona en un conjunto de datos chico**.
  - **Desarrollarlo** sobre todo el conjunto de datos y **evaluarlo offline**.
- Probarlo sobre un **conjunto controlado de usuarios**. Si el KPI mejora, llevarlo al resto.

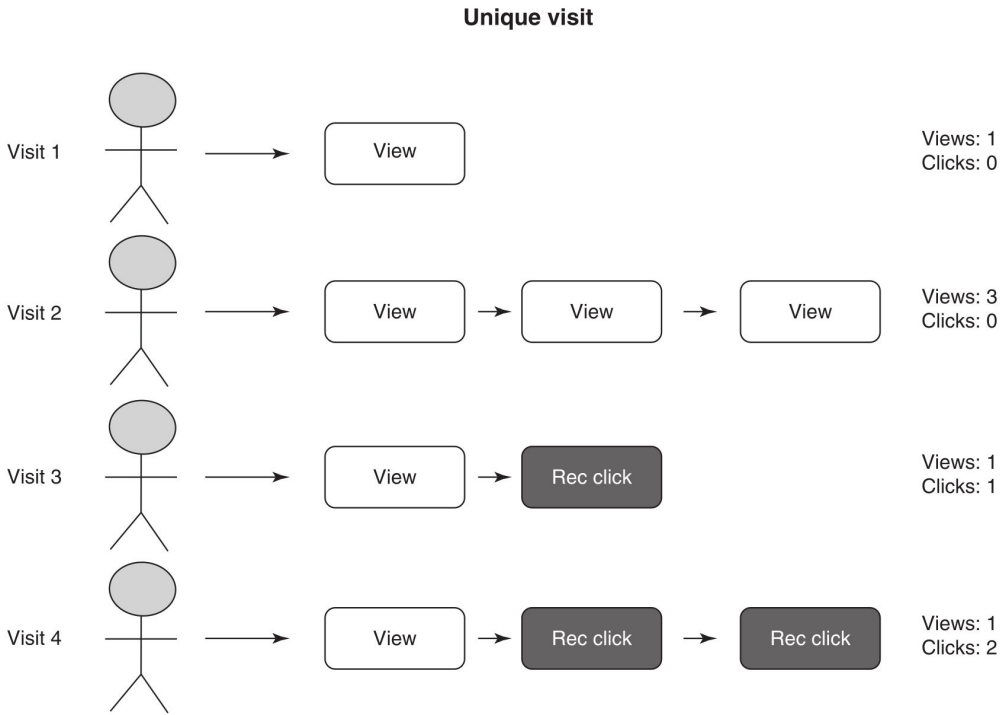
Visualización del ciclo de evaluación



Comportamiento de Usuario

Comportamiento del Usuario

- Vista 1: ¿Vio el usuario la recomendación?
- Vista 2: Asumimos que vio la recomendación y no le interesó.
- Vista 3: Ideal. Vio e interactuó.
- Vista 4: Depende del dominio. ¿Es útil tener múltiples interacciones?



Medición

¿Qué medir?

- El sistema tiene por objetivo **aumentar las ganancias haciendo más felices a los usuarios**.
  - Puede ser en distintos niveles.
- Hay 4 cosas fundamentales que medir:
  - El sistema **no hace recomendaciones erróneas**.
  - El sistema **es diverso al hacer recomendaciones**.
  - El sistema **es capaz de sorprender**.
  - El sistema **cubre todo el catálogo**.

Error de predicción

- Sirve para evaluar la capacidad del sistema de **predecir cosas que le gusten al usuario**.
- Se mide a través del **conjunto de datos**.
- Utilizamos **items ya marcados por los usuarios**.
- Se mide la **cantidad de veces que un sistema recomienda correctamente un item a un usuario**.
- Es la medida más básica para trabajar de manera offline.

Diversidad

- Predecir siempre contenido popular puede generar una burbuja.
  - Se produce un sesgo.
  - No se recomiendan nuevos elementos.
- Es algo **difícil de medir**.
- Una opción es **medir la media de la distancia entre pares de items recomendados**.

Cobertura

- La idea de un sistema de recomendación es **brindar a los usuarios la posibilidad de explorar todo el catálogo**.
    - Mayor diversidad = Mayor cobertura
- Se puede calcular con fuerza bruta, recorriendo todos los usuarios (o items) y viendo que se está recomendando.

Cálculo de cobertura

$$\text{coverage}_{user} = \frac{\sum_{u \in U} \mathbf{1}(|\mathrm{R}(u)| > 0)}{|U|}$$

$$\text{coverage}_{catalogue} = \frac{|\{i \in I : \exists u \in U \wedge i \in \mathrm{R}(u)\}|}{|I|}$$

R(*u*) = Conjunto de recomendaciones del usuario *u*

U = Conjunto de usuarios

I = Conjunto de items

Serendipia (Serendipity)

- Es un **descubrimiento afortunado**.
- Un buen sistema de recomendación busca **sorprender al usuario**.
    - Visitas recurrentes no deberían volverse fáciles de predecir.
- Es **subjetiva y difícil de medir**.
- Dependiendo el sistema, **algunas restricciones pueden funcionar o no**.
    - En general más restricciones equivalen a menos serendipia.

Verificación

Antes de implementar

- Existen tres pasos para verificar antes de implementar un sistema.
  - Verificación del **algoritmo**.
  - Verificación del **conjunto de datos**.
  - Verificar el sistema con **pruebas de regresión** (***regression testing***).
- Es necesario avanzar primero sobre estos.

Verificación del algoritmo

- **Nunca implementar algoritmos complejos** (al menos en el primer intento).
  - Tratar de **adaptar el problema a algoritmos sencillos**.
  - Mantenerse alejado del “estado del arte” la mayor parte del tiempo posible.
- Considerar **costo computacional y de memoria**.
- Probar el algoritmo sobre un **escenario simple** (e.g. un subconjunto del dataset).
  - Seguir los pasos establecidos y probarlo “manualmente”.
- **Estar de acuerdo con el cliente/negocio** acerca de la salida del algoritmo.

Verificación de los datos

- Verificar que los **datos necesarios están disponibles**.
  - ¿Es persistente?, ¿Cambia constantemente?
  - ¿Tenemos datos suficientes?
  - ¿El negocio/cliente nos permite hacer uso de los datos?
- Verificar que los **datos sean diversos**.
  - No sirven datos sobre uno o dos usuarios o uno o dos items.

Prueba de regresión

- Técnica clásica de **ingeniería del software**.
- Se ejecutan **pruebas sencillas con un resultado esperado**.
- Útiles para **capturar fallos** (*bugs*).
- Si bien pueden no ser útiles para evaluar modelos si lo son para **evaluar *pipelines***.
    - Se pueden aplicar a cada paso del *pipeline*.
    - E.g. verificar funciones de distancia sobre vectores sencillos (iguales, ortogonales, etc.).

Evaluación Offline

Evaluación Offline

- Se basa en el **conjunto de datos** (que considera verdadero).
- Se suelen usar **técnicas (y métricas) clásicas de evaluación de aprendizaje automático**.
- Se utiliza el esquema clásico de entrenamiento/validación/evaluación.
    - Es bueno utilizar el baseline de recomendar lo más popular como punto de comparación.
- Hay que tener **cuidado al dividir los datos** para entrenamiento/validación/evaluación.
    - No queremos achicar el conjunto de usuarios/items. ¿O sí?
- Es muy limitado como técnica de evaluación pero es sencillo/barato.

Métricas de Error

- Los sistemas de recomendación pueden pensarse en **términos de regresión**.
- Vemos la **diferencia entre predicción y valor real de un rating**.
- Hay tres métricas clásicas:
  - MAE: Media del Error Absoluto.
  - MSE: Media del Error Cuadrático.
  - RMSE: Raíz cuadrada de la Media del Error Cuadrático.
- MSE/RMSE castigan más a errores más grandes.
  - La diferencia está en las unidades.

Métricas de Decisión

- Buscamos ver **que tan errado está el sistema sobre una decisión**.
- Están pensadas para casos donde **el sistema se piensa como un clasificador**.
    - E.g. un sistema binario donde el indicador sea “escuchó” o no determinada canción.
- Se basan en la **matriz de confusión**.
    - Métricas clásicas son Precisión y Exhaustividad.

Evaluación Online

Evaluación Online

- Basada en el **comportamiento de los usuarios**.
- Requiere poner a funcionar el **sistema en producción**.
- Suele ser **más acertado, pero más arriesgado**.
- Requiere que el **sistema se haya probado offline**.

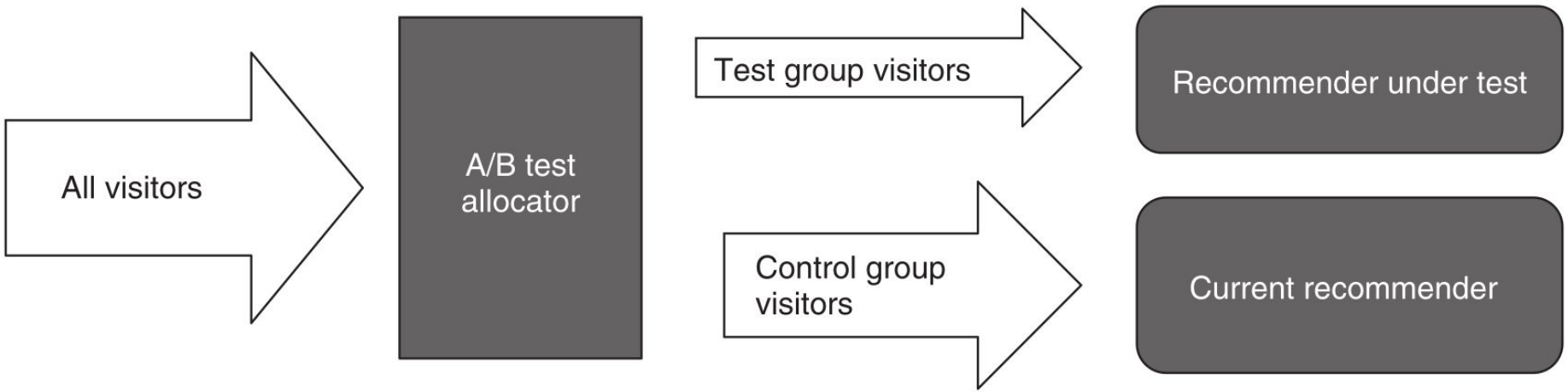
Experimentos Controlados

- Se **invita a algunos usuarios** a formar parte del sistema.
  - E.g. utilizando servicios “freemium” o “early access”.
- **Compara los sistemas** (e.g. nuevo vs. viejo) y las interacciones del usuario.
- Útil para **monitorear comportamiento** del usuario.
  - E.g. se les pregunta a los usuarios que piensan.
- Difícil saber si el **comportamiento del usuario es o no forzado**.
- Puede llevar **tiempo de preparación y obtención de resultados**.

Evaluación A/B (A/B Testing)

- Se dirige a una **porción aleatoria de usuarios al nuevo sistema**.
- Es crucial que los **usuarios no sepan que grupo forman**.
    - Para el usuario la integración debe ser “seamless”.
- Los mismos usuarios, al **interactuar con el nuevo sistema, lo ponen a prueba**.
    - Se pueden comparar fácilmente los resultados del nuevo sistema contra el viejo.
- Un riesgo importante es **pérdida de usuarios ante un sistema malo**.
    - Precio a pagar para innovar.
- Debe **integrarse al proceso de “despliegue”** (*deployment*).

Flujo para Evaluación A/B



Evaluación Continua

Explotación/Exploración

- Es una **metodología reciente**.
- Se puede pensar como una **Evaluación A/B continua**.
- La idea es en ciertas ocasiones **explotar el conocimiento adquirido**.
    - E.g. utilizar un algoritmo de recomendación altamente probado.
- En otras ocasiones se busca **explorar nuevas características**.
    - E.g. utilizar un nuevo algoritmo.
- Es **aplicable a la hora de elegir contenido nuevo** para que no quede frío.