

1. Online Extraction vs Offline Extraction. Example?

- Online Extraction: là phương pháp giúp phân tích và khai thác thông tin từ dữ liệu văn bản trực tiếp. Điều này có nghĩa là chúng ta thực hiện rút từ khóa ngay tại thời điểm dữ liệu được truy cập và tạo ra (chưa qua xử lý). Các thuật toán và công cụ online extraction thường được tích hợp vào hệ thống tự động để tìm ra các từ khóa quan trọng của bài viết, tin tức, hoặc nội dung khác.
- Offline Extraction: là phương pháp tìm ra các từ khóa hoặc cụm từ quan trọng từ dữ liệu văn bản sau khi đã thu thập. Điều này có nghĩa là chúng ta xử lý dữ liệu đã có, không phải trực tiếp từ nguồn gốc. Các thuật toán và công cụ offline extraction thường được sử dụng để phân tích dữ liệu đã lưu trữ, tìm kiếm thông tin từ các tài liệu, hoặc bài viết đã được thu thập trước đó.

➔ Tóm lại, Online Extraction tập trung vào việc phân tích từ khóa ngay tại thời điểm truy cập dữ liệu. trong khi Offline extraction thực hiện sau khi dữ liệu đã được thu thập và lưu trữ. Cả hai phương pháp đều hữu ích trong việc hiểu và xử lý thông tin từ văn bản.

- Example:

+ Online Extraction: Một ứng dụng chatbot trên trang web của một công ty du lịch. Khi người dùng nhập câu hỏi về “điểm đến du lịch hấp dẫn”, chatbot sẽ tự động trích rút từ khóa từ câu hỏi và gợi ý các địa điểm du lịch phù hợp.

+ Offline Extraction: Một công ty muốn nắm bắt xu hướng thị trường trong lĩnh vực công nghệ. Họ thu thập các bài viết, báo cáo và tài liệu liên quan, sau đó sử dụng phương pháp trích rút từ khóa ngoại tuyến để tìm ra các từ khóa quan trọng như “AI”, “blockchain”, “IoT”, v.v.

2. Motivation of ELT model. What's the difference between ETL and ELT?

- **ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform)**
- Motivation in ETL:
- Cả ETL và ELT đều là các quy trình tích hợp dữ liệu di chuyển dữ liệu thô từ hệ thống nguồn sang cơ sở dữ liệu đích (chẳng hạn như hồ dữ liệu hoặc kho dữ liệu).
 - + ETL (Extract, Transform, Load):
 - >Extraction: Dữ liệu phi cấu trúc được trích xuất từ hệ thống nguồn.
 - >Transformation: Dữ liệu trải qua quá trình chuyển đổi (làm sạch, sắp xếp và cấu trúc) trong khu vực tổ chức.
 - >Loading: Dữ liệu có cấu trúc được tải vào hệ thống đích.

>Bối cảnh lịch sử: ETL đã xuất hiện từ những năm 1970 và được thiết kế cho cơ sở dữ liệu quan hệ.

>Quy trình tinh chỉnh: Các kỹ sư dữ liệu đã cải tiến đáng kể các quy trình ETL theo thời gian.

>Ưu điểm: Dữ liệu sạch hơn, phù hợp với kho lưu trữ mục tiêu nhỏ hơn.

+ ELT (Extract, Load, Transform):

>Extraction: Dữ liệu phi cấu trúc được trích xuất từ hệ thống nguồn và tải lên hệ thống đích.

>Transformation: Chuyển đổi xảy ra sau khi tải, cho phép cập nhật theo thời gian thực.

> Không có khu vực tổ chức: ELT tránh việc dàn dựng dữ liệu, tận dụng trực tiếp việc lưu trữ dữ liệu.

>Ưu điểm: Triển khai nhanh hơn, khả năng mở rộng và khả năng tương thích với nhiều loại dữ liệu khác nhau (có cấu trúc, không cấu trúc, bán cấu trúc và thô).

>Cân nhắc: ELT mới hơn và dựa trên các tập lệnh SQL, có thể có lỗi mã hóa.

-> Tóm lại, ETL tập trung vào việc chuyển đổi dữ liệu trước khi tải, trong khi ELT tải dữ liệu trước và chuyển đổi dữ liệu sau.

3.

- Nghiên cứu kiến trúc tổng thể của Docker:

+ Tổng quan và kiến trúc Docker Docker là một nền tảng mở để phát triển, vận chuyển và chạy các ứng dụng. Nó cho phép bạn tách các ứng dụng khỏi cơ sở hạ tầng, cho phép phân phối phần mềm nhanh hơn. Dưới đây là các thành phần chính:

> Docker Daemon: Dịch vụ nền quản lý các vùng chứa và hình ảnh Docker.

> Docker Client: Công cụ dòng lệnh tương tác với daemon Docker.

> Hình ảnh Docker: Các tệp bất biến chứa mã nguồn, thư viện, phần phụ thuộc và các công cụ cần thiết để ứng dụng chạy.

> Docker Container: Các phiên bản thời gian chạy của hình ảnh Docker, được tách biệt khỏi hệ thống máy chủ.

> Docker Hub: Sở đăng ký hình ảnh Docker công khai và riêng tư.

+ Docker Images với Docker Containers:

➤ Docker Images:

? Ảnh chụp nhanh bất biến đại diện cho một ứng dụng và môi trường của nó.

? Mẫu chỉ đọc.

? Được sử dụng làm cơ sở để tạo các thùng chứa.

? Các lớp cho phép chia sẻ và tạo phiên bản hiệu quả.

? Được tạo bằng Dockerfile.

➤ Docker Containers:

? Chạy các phiên bản của hình ảnh Docker.

? Có một lớp có thể ghi ở trên cùng của hình ảnh.

? Có thể được sửa đổi trong thời gian chạy.

? Môi trường thời gian chạy bị cô lập.

? Có thể chạy nhiều container từ cùng một image.