Projeto Final - Disciplina Extração, Tranformação e Gestão de Dados

Eduardo C. Silva[1], **Vanderson Vieira**[2]

[1]Centro Universitário 7 de Setembro (UNI7)
Fortaleza – CE – Brazil

[2]Especialização em Ciência de Dados

Abstract. *This article will describe a practical application applied to the discipline of Data Extraction, Transformation and Management. This practice, popularly called ETL, is the process of collecting data from multiple sources and bringing it together to support discovery, reporting, analysis, and decision making. This data can be very diverse in type, format, volume, and reliability, so the data needs to be processed to be useful when gathered. Target data stores can be databases, data warehouses, or data lakes, depending on goals and technical implementation.*

Resumo. *Este artigo irá descrever uma aplicação prática aplicada a disciplina de Extração, Transformação e Gestão de Dados. Essa prática, popularmente chamada de ETL, é o processo de coletar dados de várias fontes e reuni-los para dar suporte à descoberta, à geração de relatórios, à análise e à tomada de decisões. Esses dados podem ser muito diversas em tipo, formato, volume e confiabilidade, de modo que os dados precisam ser processados para serem úteis quando reunidos. Os armazenamentos de dados de destino podem ser bancos de dados, data warehouses ou data lakes, dependendo das metas e da implementação técnica.*

1. Introdução

As práticas de extração, transformaçao e carregamento de dados, são hoje de extrema importância para as corporações, tendo em vista a enorme quantidade de dados que essas instituições possuem, e a necessidade de se gerar informações que possam auxilia-las nas tomadas de decisão, que diáriamente são cruciais para o seu desenvolvimento.

1.1. Etapas do ETL

Esse processo é dividido em três etapas, que são elas:

Extrair:

Durante a extração, identificamos os dados e os copiamos de suas bases de dados de origem, de forma que se possa transportar os dados para o armazenamento de dados de destino. Os dados podem vir de fontes estruturadas e não estruturadas, incluindo documentos, emails, aplicações de negócios, bancos de dados, equipamentos, sensores e etc.

Transformar:

Como os dados extraídos são brutos em sua forma original, eles precisam ser mapeados e transformados para prepará-los para o armazenamento de dados eventual. No

processo de transformação, validamos, autenticamos, desduplicamos e/ou agregamos os dados de formas que tornam os dados resultantes confiáveis e consultáveis.

Carregar:

No processo de carregamento, movemos os dados transformados para o armazenamento de dados de destino. Esta etapa pode implicar o carregamento inicial de todos os dados de origem ou pode ser o carregamento de alterações incrementais nos dados de origem. Você pode carregar os dados em tempo real ou em lotes programados.

2. Arquitetura de dados

A arquitetura de dados é a tendência em que muitas empresas estão tentando organizar seus dados, ativos digitais e relacionamentos entre eles. Assim, uma arquitetura eficiente garante acesso seguro, oportuno e amigável aos dados. Disponibilidade é o termo que melhor descreve a importância da arquitetura da informação em um contexto de negócios.

Isso significa que as funções dos sistemas e software estão disponíveis quando necessário, ou, na pior das hipóteses, podem ser acessados em tempo hábil, garantindo processos decisórios menos aleatórios. Em algumas empresas, a colaboração e o trabalho online exigem o compartilhamento de dados, formulários, planilhas e arquivos regulamentados. Nesse sentido, a estrutura de dados da empresa deve ser de fácil acesso e segurança, a ponto de impedir que pessoas não autorizadas tenham privilégios indevidos.

Todo software depende do acesso ao respectivo banco de dados para funcionar correctamente. Quando esta função falha Todo o sistema entrará em colapso e se tornará inútil. Desta forma, a arquitetura de dados é a melhor garantia de que seu sistema está sempre pronto para responder quando necessário.

2.1. Arquitetura do projeto

Tendo em vista a importância de uma arquitetura de dados, o referente projeto foi pensado com base em um arquitetura desenvolvida com o pensamento de otimizar o processo de transformação de dados, com um pipeline que facilite o entendimento do que foi desenvolvido.

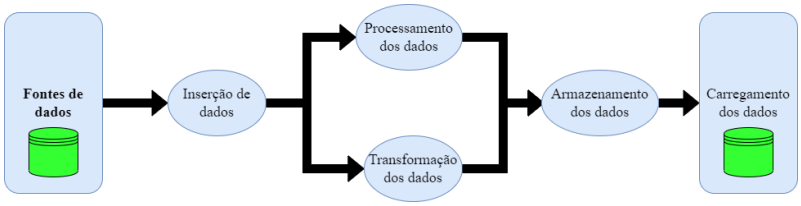

Figure 1. Arquitetura do Sistema

Nesse ponto, é possivel observar todo o ciclo do proceddo, que se inicia na extração dos dados das fontes de dados, passando pelo processo de processamento dos dados em Dataframes, e o posterior tratamento desses dados, afim de obter o resultado previsto no contexto do projeto.

Após isso, os dados tratados são novamente armazenados em um DataFrame, e então são carregados dentro de uma nova fonte de dados que será responsavél por fornecer esses dados que foram limpos para o objetivo final que é a tomada de decisão.

3. Contexto

Para esse projeto foi utilizado como fonte a base de dados de um site de avaliação de filmes, ao qual é responsável por fornecer as notas e críticas do publico sobre filmes de gêneros diversos.

Foi colocado como objetivo, obter os filmes melhores avaliados pelo sistema imdb, que trata-se de um dos maiores sites de avaliação e crítica de cinema do mundo.

3.1. Fontes de dados

Foram utilizadas três fontes de dados. Ambas foram importadas em arquivos com o formato .csv, pela facilidade de importação obtida no sistema.

3.2. Ferramentas utilizadas

Jupyter Notebook:

O Jupyter Notebook é um aplicativo web open-source, ou seja, seu código é aberto e é possível que ele seja utilizado e alterado para diversos fins.

Por ser um software open-source, o Jupyter também é gratuito, já que não é necessário adquirir uma licença para utilizá-lo. Isso permite que a aplicação seja amplamente divulgada e melhorada com o esforço da própria comunidade.

Pode ser definido como um ambiente onde é possível a experimentação de ideias. Você se lembra dos cadernos de nota, onde a qualquer momento poderíamos escrever e guardar memórias.

De forma prática, é possível compilar trechos de códigos de diversas linguagens de programação. Isso transforma as linhas de texto sem vida em gráficos de alta qualidade.

As informações geradas nessa compilação são de fundamental importância durante o desenvolvimento. É através dessas informações que os códigos podem ser corrigidos de maneira mais rápida, sem ter um impacto tão grande dentro de um projeto.

Biblioteca Pandas

A biblioteca pandas pode ser considerada a mais importante dentro do mundo da análise de dados para o Python. É a ferramenta principal para construção de estrutura, manipulação e limpeza de dados, sendo também utilizada com bibliotecas de processamento numérico e construção de gráficos. No post de hoje, iremos realizar uma breve introdução a esta biblioteca.

O ponto chave do pandas está na estrutura de dados no qual a biblioteca permite criar e manipular, são elas: Series e DataFrame.

Series é um objeto array unidimensional (tal qual o array criado com o Numpy) possuindo um índice (rótulos das observações). O DataFrame é considerado como um conjunto de dados retangulares ou dados tabulares, no qual cada coluna tem um tipo de dado, representados por um índice de colunas e um índice para cada observação (linha).

4. Aplicação

A aplicação foi toda realizada no Jupyter Notebook. Para isso, foi necessário a utilização da aplicação Anaconda, que forneceu o ambiente de desenvolvimento, e a instalação dos pacotes e bibliotecas necessárias para a realização do projeto.

As bases de dados foram dividas em 3 arquivos. O arquivo movies.csv, ratings.csv e links.csv. Na base movies, continham os dados dos filmes referênciados para avaliação. O arquivo ratings contém a base de dados com as avaliações de filmes, a qual deveria ser priorizada as realizadas pela imdb.

Importadas as bases, cada uma foi armazenada dentro de um Dataframe. Feito isso, foi dado um merge, na base movies e ratings, e armazenadas dentro de um unico dataframe, afim de analizar os dados em conjunto.

Após foram dropadas as tabelas que não seriam utilizadas para avaliação e os dados em formatos diferentes (object), que no caso foram 2 colunas, foram tratatos e transformados para string. Com a base normalizada, foi utilizado novamente um merge, do dataframe resultante com o dataframe de links.

O penultimo processo, foi o de filtrar os filmes com avaliação acima de 4.0. Esse filtro foi aplicado no dataframe movies-ratings-links', que continha os dados referentes aos três bancos, e armazenado dentro do dataframe 'better-movies-ratings', que referência os filmes melhores avaliados.

O último processo foi o de exportação dos dados, a qual foi utilizado também o formato .csv, como padrão das bases de entrada para a base de saída, assim gerando a base de dados tratados.

5. Conclusão

Com isso foi gerado o repositório no github, ao qual constam o código fonte do projeto, o arquivo com a construição da arquitetura feita no aplicativo Draw.io, além das bases de dados utilizadas no projeto. Esse processo, acabou sendo de grande influência para a aplicação dos conceitos obtidos nas aulas teóricas. Através dele foi possível observar a complexidade de uma aplicação de ETL, e um bom inicio de contato com ferramentas como python e pandas.

Deixamos nosso agradecimento ao professor orientador da disciplina, que com grande maestria nos guiou nesse processo de aprendizado.

6. Links dos repositórios.

https://github.com/eduprt123/Projeto-Final-ETL