**Rangkuman Data Wrangling**

**Pengenalan Data Wrangling**

Data wrangling merupakan sebuah proses atau kumpulan kegiatan yang meliputi pengumpulan data (Gathering data), penilaian data (Assessing data), serta pembersihan data (Cleaning data) sebelum data digunakan dalam proses analisis data.

* **Gathering data**  
  Tahap data wrangling, dimulai dengan proses pengumpulan data. Pada proses ini kita akan mengumpulkan semua data yang dibutuhkan untuk menjawab semua pertanyaan atau masalah bisnis yang ingin kita hadapi.
* **Assessing data**  
  Setelah semua data yang dibutuhkan terkumpul, proses selanjutnya ialah penilaian terhadap data tersebut. Proses ini dilakukan untuk menilai kualitas dan struktur dari sebuah data. Selain itu, proses ini juga bertujuan untuk mengidentifikasi berbagai masalah yang terdapat dalam data, seperti missing value, unstandard value, dll.
* **Cleaning data**  
  Apabila pada proses sebelumnya kita menemukan masalah (missing value, outlier, dll.) yang terdapat di dalam sebuah data, masalah tersebut harus dibersihkan sebelum masuk tahap analisis data. Terdapat beberapa teknik yang dapat kita gunakan untuk membersihkan data. Seluruh teknik tersebut akan kita pelajari pada beberapa materi ke depan.

**Gathering Data**

Proses pengumpulan data merupakan salah satu tahapan paling menantang dalam proyek analisis data. Bergantung pada proyek analisis data yang dikerjakan, data yang Anda butuhkan mungkin terdapat dalam berbagai sumber dan memiliki format yang berbeda-beda.

Salah satu skill yang harus dimiliki oleh seorang praktisi data ialah mengetahui sumber data yang tepat untuk mengumpulkan data yang dibutuhkan. Berikut merupakan beberapa sumber data yang bersifat publik dan sering digunakan oleh para praktisi data.

* **Kaggle**  
  Kaggle merupakan platform andalan bagi praktisi data untuk mencari dataset. Selain menyediakan dataset, kaggle juga menyediakan berbagai kompetisi dan *challenge* terkait data science dan machine learning. Kaggle juga memungkinkan para praktisi data dan machine learning saling berbagai kode dan pengetahuan terkait data serta machine learning.
* **UCI Machine Learning Repository**  
  UCI Machine Learning Repository merupakan sebuah repositori yang menampung berbagai dataset yang bersifat publik.
* **Google Dataset Search**  
  Google Dataset Search merupakan sebuah search engine yang disediakan oleh Google. Ia dibuat untuk mempermudah para praktisi data dan juga *researcher*dalam mencari dataset yang bersifat publik.
* **Satu Data Indonesia**  
  Tahukah Anda bahwa pemerintah Indonesia sebetulnya telah menyediakan sebuah platform bernama Satu Data Indonesia atau sering disingkat SDI.

Selain menggunakan berbagai sumber dataset publik tersebut, para praktisi data juga seringkali menggunakan data internal yang bersifat privat. Data seperti ini umumnya disimpan dalam sebuah sistem pengolahan *database*.

Pandas menyediakan beberapa *function* yang dapat digunakan untuk membaca atau mengakses data menjadi sebuah DataFrame. Saat ini, Pandas mendukung pembacaan atau pengaksesan data dari berbagai format berkas data dan sumber data seperti pada contoh di bawah ini.

* **Format berkas CSV**  
  Berkas CSV (*Comma Separated Values*) merupakan format berkas data tabel yang paling sering digunakan dan telah menjadi standar dalam industri. Pandas menyediakan sebuah function **read\_csv()**untuk membaca berkas CSV.
* **Format berkas XLSX atau XLS**  
  Berkas XLSX atau XLS merupakan format berkas *spreadshee*t yang dibuat menggunakan aplikasi Microsoft Excel. Pandas juga menyediakan function **read\_excel()** untuk membaca berkas data dengan format XLSX atau XLS.
* **Format berkas JSON**  
  Berkas JSON (*JavaScript Object Notation*) merupakan format berkas data yang memiliki struktur data yang mirip seperti *data structure dictionary* dalam Python yang terdiri dari pasangan *keys* dan *values*. Pandas juga menyediakan function **read\_json()** untuk membaca berkas data berformat JSON.
* **Format berkas HTML**  
  HTML atau dikenal juga sebagai *HyperText Markup Language* merupakan sebuah markup language standar yang digunakan untuk merancang tampilan sebuah dokumen/halaman di *web browser*. Untuk membaca berkas ini, pandas menyediakan function **read\_html()**.
* **Format berkas XML**  
  Format data selanjutnya yang akan kita bahas ialah XML yang merupakan singkatan dari Extensible Markup Language. Ia sering digunakan untuk merepresentasikan berbagai struktur informasi, seperti dokumen, data, konfigurasi, dll. Pandas menyediakan function **read\_xml()** untuk membaca format data ini.
* **Akses data dari SQL database**  
  Selain membaca data dari berbagai format, library pandas juga memungkinkan kita untuk mengakses data langsung dari sebuah database, seperti PostgreSQL, MySQL, dll. Tentunya untuk mengakses database tersebut kita membutuhkan library pendukung yaitu SQLAlchemy.  
    
  Untuk berinteraksi dengan database, pandas menyediakan tiga function seperti berikut.
  + **read\_sql\_table()** untuk membaca SQL database table dan mempresentasikannya ke dalam bentuk pandas DataFrame.
  + **read\_sql\_query()** untuk membaca SQL query dan mempresentasikannya ke dalam bentuk pandas DataFrame.
  + **read\_sql()** untuk membaca SQL query atau table dan mempresentasikannya ke dalam bentuk pandas DataFrame.

Seperti yang telah kita bahas sebelumnya, untuk menyelesaikan sebuah permasalahan bisnis, terkadang kita perlu menggunakan lebih dari satu tabel data. Tentunya untuk melakukan analisis kita perlu menggabungkan data tersebut. Salah satu teknik penggabungan data yang paling sering digunakan ialah *merge* atau *join*.

Berdasarkan cara penggabungannya, proses merge atau join dapat dibagi menjadi empat jenis yaitu seperti berikut.

* **Inner**  
  Inner join merupakan proses join yang hanya mengambil nilai yang bersesuaian di kedua tabel.
* **Left**  
  Left join merupakan proses join yang akan mengambil semua nilai dari tabel kiri beserta nilai yang bersesuaian dari tabel kanan.
* **Right**  
  Right join merupakan proses join yang akan mengambil semua nilai dari tabel kanan beserta nilai yang bersesuaian dari tabel kiri. Ia merupakan kebalikan dari left join.
* **Outer**  
  Outer join atau sering juga disebut *full outer join* merupakan proses join yang akan mengambil semua nilai dari kedua tabel. Ia merupakan gabungan dari left dan right join.

Sebagai tool andalan dalam pengolahan dan analisis data, pandas menyediakan sebuah function bernama **merge()**. Ia dapat digunakan untuk menggabungkan dua buah DataFrame.

**Assessing Data**

Pemeriksaan data ini dilakukan dengan menjalankan proses assessing data. Ia merupakan proses yang bertujuan untuk mengidentifikasi masalah yang terdapat dalam data dan memastikan data tersebut berkualitas.

Seperti yang telah kita bahas sebelumnya, kebersihan dan kualitas dari sebuah data merupakan tantangan utama dalam proyek analisis data di industri. Data yang kotor umumnya memiliki masalah dalam kontennya. Berikut beberapa masalah yang umum dijumpai dalam sebuah data.

* **Missing value**  
  Masalah ini muncul karena adanya nilai yang hilang dari sebuah data dan biasanya direpresentasikan sebagai nilai **NaN** dalam library pandas. Hal ini biasanya terjadi karena adanya human error, masalah privasi, proses merging/join, dll.  
  Library pandas menyediakan sebuah method bernama **isnull()** atau**isna()** untuk mengidentifikasi missing value dalam sebuah DataFrame. Keduanya sering dipadukan dengan method **sum()** untuk menghitung total missing value pada setiap kolom dalam sebuah DataFrame.
* **Invalid value**  
  Masalah ini muncul ketika terdapat beberapa nilai yang tidak masuk akal, tidak sesuai dengan ketentuan, dan *background knowledge* dari data tersebut.  
  Untuk mendeteksi masalah seperti ini, kita membutuhkan teknik sedikit *advance*. Salah satu teknik yang paling sering digunakan ialah teknik *filtering data* menggunakan *regex*.
* **Duplicate data**  
  Duplicate data merupakan masalah lain yang umum dijumpai di industri. Ia terjadi ketika terdapat sebuah observasi (semua nilai dalam satu unit baris) yang memiliki nilai yang sama persis pada setiap kolomnya. Pandas menyediakan sebuah method **duplicated()** untuk mengidentifikasi apakah terdapat duplikasi pada sebuah DataFrame.
* **Inaccurate value**  
  Inaccurate value merupakan masalah yang muncul ketika nilai dalam sebuah data tidak sesuai dengan hasil observasi. Masalah ini umumnya muncul karena adanya human error atau sistem error.
* **Inconsistent value**  
  Inconsistent value adalah masalah yang muncul ketika sebuah data memiliki nilai yang tidak konsisten baik dari segi satuan maupun ketentuan penilaian. Inkonsistensi ini umumnya muncul karena adanya perbedaan standar dalam proses pengumpulan nilai.
* **Outlier**  
  Outlier atau dalam bahasa indonesia disebut pencilan merupakan titik data yang berada sangat jauh dari titik data yang lain dalam sebuah dataset. Nilai yang sangat jauh ini tentunya akan berdampak terhadap beberapa parameter statistik yang digunakan untuk menganalisis data, seperti nilai *mean* dan *standard deviation*.  
  Terdapat beberapa metode yang dapat digunakan untuk mengidentifikasi outlier dalam sebuah dataset. Metode yang paling sering digunakan ialah IQR method.  
  IQR method merupakan metode penentuan outlier berdasarkan nilai interquartile range (IQR). Ia mengidentifikasi outlier dengan cara membuat nilai *cut-off*sebagai faktor *k* (Umumnya kita menggunakan nilai 1.5 s/d 3) dari nilai IQR. Cut-off tersebut selanjutnya akan digunakan untuk menghitung nilai ambang batas (*boundary values*).  
  Metode lain yang bisa digunakan ialah box plot. Ia merupakan bentuk visual untuk merepresentasikan nilai IQR beserta ambang batas bawah dan atas dari sebuah data. Hal ini tentunya akan membantu kita mengidentifikasi outlier secara lebih mudah yaitu melalui bentuk visual.

**Cleaning Data**

Data yang kotor juga harus kita bersihkan untuk memastikan ia tidak mempengaruhi hasil analisis yang akan kita lakukan nantinya.

Secara umum, proses pembersihan data dapat dibagi ke dalam tiga tahapan, yaitu *define*, *code*, dan *test*.

* **Define:** pada tahap ini, kita akan membuat rancangan tahapan serta metode pembersihan data berdasarkan masalah yang ditemukan dalam proses assessing data.
* **Code:** setelah membuat rancangan pembersihan data, tahap selanjutnya ialah mengonversi hal tersebut menjadi sebuah kode program yang dapat dijalankan.
* **Test:** setelah menjalankan kode program untuk membersihkan data, kita perlu memeriksa kembali data yang telah dibersihkan tersebut.

**Teknik untuk Mengatasi Missing Value**

Secara garis besar, terdapat tiga metode dalam mengatasi missing value antara lain seperti berikut.

* **Dropping**  
  Metode pertama dan yang paling mudah dalam mengatasi missing value adalah dropping. Pada metode ini, kita akan menghapus seluruh baris atau kolom yang memiliki missing value. Untuk melakukannya, kita bisa menggunakan salah satu method **dropna()** yang disediakan oleh library pandas  
  Metode dropping ini memang terkesan mudah, tetapi perlu Anda ketahui bahwa ada banyak sekali pertimbangan yang perlu diperhatikan sebelum menggunakan metode ini. Salah satu pertimbangan penting ialah penerapan metode ini dapat menyebabkan kita kehilangan banyak informasi.
* **Imputation**  
  Metode ini bekerja dengan cara mengisi (*fill*) missing value dengan nilai tertentu. Hal ini tentunya akan mencegah hilangnya informasi akibat missing value.  
  Pada data kontinu, kita bisa menggunakan nilai mean, median, atau mode sebagai pengganti missing value. Jika bekerja menggunakan data kategoris, kita dapat mengisi missing value dengan kategori yang paling sering muncul. Namun, perlu diingat bahwa pemilihan nilai pengganti ini harus didukung oleh *background knowledge*dari data tersebut.  
  Sebagai library andalan kita, pandas telah menyediakan sebuah method bernama **fillna()** untuk mengganti missing value dengan nilai tertentu.  
  Metode ini masih memiliki banyak kekurangan salah satunya ialah dapat mempengaruhi variance atau sebaran dari sebuah data. Selain itu, metode ini juga masih belum cukup baik untuk diterapkan pada data time series.
* **Interpolation**  
  Metode penanganan missing value terakhir yang akan kita bahas ialah interpolation (interpolasi). Sederhananya, interpolasi merupakan salah satu pendekatan numerik yang digunakan untuk menghitung titik data baru berdasarkan range data yang sudah ada. Perhitungan ini menggunakan sebuah persamaan garis *linear* ataupun *polynomial*. Library pandas juga menyediakan method **interpolate()** yang bisa kita gunakan untuk menerapkan metode interpolasi dalam mengatasi missing value.

**Teknik untuk Mengatasi Outlier**

Pada dasarnya, terdapat dua metode yang umum digunakan untuk mengatasi outlier yaitu drop dan imputation.

* **Drop**  
  Metode pertama yang paling mudah ialah men-*drop* atau menghapus seluruh baris yang mengandung outlier. Metode ini mampu mencegah outlier mempengaruhi hasil analisis yang kita buat.
* **Imputation**  
  Metode lain yang bisa Anda gunakan untuk menangani outlier ialah imputation. Konsepnya mirip seperti sebelumnya yaitu mengganti outlier dengan nilai tertentu. Nilai yang bisa kita gunakan ialah mean, media, dan mode. Selain itu, tidak jarang juga kita mengganti outlier dengan *boundary value*.  
  Untuk menerapkan metode ini, kita bisa menggunakan method **mask()** yang disediakan oleh library pandas. Method tersebut menerima parameter **cond** sebagai kondisi untuk memfilter nilai outlier.

**Teknik untuk Mengatasi Duplicate Data**

Ketika menemukan duplikasi pada data, tentunya kita harus menghilangkan atau menghapus duplikasi tersebut. Library pandas telah menyediakan sebuah method **drop\_duplicates()** untuk menghilangkan duplikasi dalam sebuah DataFrame.