Judul: **Analisis Pola Check-in Pengguna Foursquare di New York dan Tokyo**

Anggota Kelompok

1. Aisyah Tiara Pratiwi (121450074)
2. Adil Aulia Rahma Nurhidayah (122450058)

Deskripsi pembagian pekerjaan

1. Aisyah Tiara Pratiwi (121450074)

* **Pertanyaan analisis visualisasi, ide cerita yang bisa diimplementasikan, dan visualization**
* Memberikan ide atau masukan terkait ide cerita dan pertanyaan yg cocok untuk analisis visualisasi terhadap dataset
* Membuat ide cerita dan pertanyaan visualisasi
* Memperbaiki Preprocessing Data yang sesuai dengan pertanyaan
* Membuat kode untuk tiap pertanyaan visualisasi
* Mendeskripsikan dan melakukan analisis terhadap kode tersebut dan output dari visualisasi tiap pertanyaan
* Mengubah pertanyaan terakhir menjadi pertanyaan yang lebih baik

2. Adil Aulia Rahma Nurhidayah (122450058)

* **Dataset dan Deskripsi,serta Implementation dari Preprocessing Dataset**
* Memberikan ide tentang judul tugas yang relevan dengan dataset yang dipilih
* Menambahkan pendeskripsian dataset yang akan digunakan secara detail
* Menjelaskan secara rinci langkah-langkah yang dilakukan terkait dengan pengolahan dataset sebelum dilakukan visualisasi
* Mencoba membuat kode tiap langkah- langkah Preprocessing
* Mendeskripsikan input dan output dari kode Preprocessing

# Pre-Implementation

* 1. Dataset dan Deskripsi

Dataset dari Kaggle ini berisi data check-in pengguna di Foursquare untuk dua kota besar, New York (NYC) dan Tokyo selama periode sekitar 10 bulan, dari 12 April 2012 hingga 16 Februari 2013. Terdapat total 227.428 check-in di New York City dan 573.703 check-in di Tokyo. Data ini mencakup informasi penting yang dapat digunakan untuk menganalisis perilaku pengguna dan pola aktivitas mereka di lokasi berbasis layanan (LBSNs) yang merujuk pada platform atau aplikasi yang menggabungkan elemen jejaring sosial dengan informasi lokasi.LBSN memungkinkan pengguna untuk berbagi dan menemukan informasi terkait lokasi tertentu, seperti tempat makan, tempat hiburan, atau lokasi lainnya, serta melakukan check-in di tempat-tempat tersebut.Setiap data check-in mencakup beberapa 8 atribut penting:

* **ID pengguna** : Mengidentifikasi pengguna yang melakukan check-in.
* **Venue ID** : ID unik untuk setiap tempat (venue) dimana check-in terjadi
* **Venue Kategori ID** : ID kategori untuk setiap venue
* **Kategori venue**: Tempat check-in dan kategorinya (misalnya, restoran, taman, museum).
* **Latitude** : Koordinat lintang tempat check-in
* **Longitude**: Koordinat bujur tempat check-in, yang dapat digunakan untuk mengidentifikasi lokasi geografis.
* **Time zone offset**: Menyatakan perbedaan waktu check-in terhadap UTC.
* **UTC Timestamp** : Waktu dan tanggal saat check-in dilakukan

Dataset ini bermanfaat untuk menganalisis perilaku pengguna, preferensi tempat, dan pola check-in yang dapat menginspirasi berbagai studi dalam konteks pariwisata, perencanaan kota, dan analisis spasial.

* 1. Pertanyaan analisis visualisasi

Berikut beberapa pertanyaan terkait data dan visualisasi yang akan dibuat :

1. Bagaimana distribusi check-in per kategori tempat di New York dan Tokyo?
2. Kapan waktu-waktu puncak check-in di masing-masing kota?
3. Bagaimana pola check-in berdasarkan waktu dalam sehari di New York vs Tokyo?
4. Area mana saja yang menjadi hotspot atau lokasi dengan konsentrasi check-in tertinggi di Tokyo dan NYC?
   1. Ide cerita

Analisis visualisasi data check-in Foursquare untuk **New York City (NYC)** dan **Tokyo** bertujuan untuk menggali wawasan mendalam mengenai pola aktivitas pengguna, distribusi lokasi yang populer, dan perbedaan preferensi antar kedua kota. Analisis pertama akan melihat **distribusi check-in berdasarkan kategori tempat**, seperti restoran, taman, pusat perbelanjaan, atau tempat wisata. Hal ini bertujuan untuk mengidentifikasi kategori lokasi yang paling diminati pengguna di masing-masing kota dan membandingkan tren tersebut. Apakah pengguna di Tokyo lebih cenderung check-in di lokasi hiburan atau rekreasi, sementara pengguna di NYC lebih fokus pada lokasi komersial dan kuliner?

Selanjutnya, dilakukan analisis **waktu-waktu puncak check-in** untuk memahami kapan aktivitas pengguna mencapai puncaknya dalam sehari dan sepanjang minggu. Visualisasi pola ini akan menunjukkan jam sibuk dan hari-hari favorit check-in di Tokyo dan NYC. Dengan melihat pola ini, perbedaan dalam ritme aktivitas antara kedua kota dapat diketahui, seperti apakah penduduk Tokyo memiliki kebiasaan check-in lebih pagi dibanding penduduk NYC, atau sebaliknya.

Kemudian, visualisasi akan difokuskan pada **pola check-in berdasarkan waktu dalam sehari**. Analisis ini akan membandingkan bagaimana aktivitas check-in tersebar di pagi, siang, sore, dan malam hari. Hal ini akan memberikan gambaran tentang bagaimana pengguna di kedua kota membagi waktu mereka dalam beraktivitas, termasuk waktu-waktu puncak untuk kategori tempat tertentu seperti restoran atau tempat hiburan.

Untuk memahami persebaran aktivitas check-in secara geografis, akan dibuat **visualisasi heatmap** yang menampilkan **area hotspot** atau lokasi dengan konsentrasi check-in tertinggi di Tokyo dan NYC. Heatmap ini akan membantu mengidentifikasi area-area yang paling sering dikunjungi pengguna, apakah lokasi tersebut terkonsentrasi di pusat kota, area wisata, atau tersebar di berbagai wilayah. Dengan demikian, pola distribusi spasial check-in dapat dibandingkan untuk melihat apakah Tokyo memiliki hotspot yang lebih terpusat atau lebih tersebar dibanding NYC.

# Implementation

* 1. Preprocessing Dataset
     1. Remove Data

Sebelum memulai proses analisis, langkah pertama yang dilakukan adalah memastikan tidak adanya nilai yang hilang dalam dataset terpilih, baik untuk dataset NYC maupun Tokyo. Pemeriksaan dilakukan menggunakan metode .isnull().sum() untuk melihat jumlah nilai yang hilang pada setiap kolom.

1. Missing Value

Nilai hilang (missing values) dalam dataset merujuk pada entri yang tidak memiliki data atau informasi pada kolom tertentu. Penanganan nilai hilang adalah langkah penting dalam analisis data karena dapat mempengaruhi hasil dan akurasi analisis dengan menggunakan perintah **isnull().sum()** untuk mengembalikan di mana setiap elemen menunjukkan apakah nilai tersebut adalah NaN (Not a Number) atau tidak.Hasil dari **sum()** memberikan total jumlah nilai hilang untuk setiap kolom.

Output

| Jumlah nilai hilang di NYC : | | Jumlah nilai hilang di Tokyo : | |
| --- | --- | --- | --- |
| userID | 0 | userID | 0 |
| venueID | 0 | venueID | 0 |
| venueCategoryID | 0 | venueCategoryID | 0 |
| venueCategoryID | 0 | venueCategoryID | 0 |
| latitude | 0 | latitude | 0 |
| longitude | 0 | longitude | 0 |
| timezoneOffset | 0 | timezoneOffset | 0 |
| utcTimestamp | 0 | utcTimestamp | 0 |

Hasil pemeriksaan menunjukkan bahwa tidak ada nilai yang hilang di kedua dataset, di mana semua kolom seperti userID, venueID, venueCategoryID, venueCategory, latitude, longitude, timezoneOffset, dan utcTimestamp memiliki nilai utuh. Dengan demikian, dapat dipastikan bahwa dataset telah bersih dari nilai kosong dan siap untuk proses analisis lebih lanjut.

Code program

| # Check for missing values  print("Jumlah nilai hilang di NYC:")  print(nyc\_data.isnull().sum())  print("\nJumlah nilai hilang di Tokyo:")  print(tokyo\_data.isnull().sum()) |
| --- |

1. Duplicates

Setelah memastikan tidak adanya nilai hilang, tahap selanjutnya adalah menangani data duplikat untuk menjaga integritas data. Pada dataset NYC, ditemukan sebanyak 250 baris duplikat sebelum dilakukan penghapusan. Data duplikat dihapus menggunakan fungsi **.drop\_duplicates(**), sehingga jumlah duplikat setelah penghapusan menjadi 0. Hal serupa juga dilakukan pada dataset Tokyo, di mana ditemukan 577 baris duplikat sebelum penghapusan, yang kemudian berhasil dihapus sepenuhnya sehingga tidak ada lagi data duplikat. Proses ini penting untuk memastikan bahwa setiap observasi dalam dataset memiliki nilai unik dan tidak terjadi pengulangan data yang dapat mempengaruhi hasil analisis atau visualisasi.

Output

| Jumlah duplikat di NYC sebelum dihapus: 250 |
| --- |
| Jumlah duplikat di NYC setelah dihapus: 0 |
| Jumlah duplikat di Tokyo sebelum dihapus: 577 |
| Jumlah duplikat di Tokyo setelah dihapus: 0 |

Code program

| # Check for duplicates  print("Jumlah duplikat di NYC sebelum dihapus:", nyc\_data.duplicated().sum())  # Drop duplicates  nyc\_data.drop\_duplicates(inplace=True)  # Verify that duplicates have been removed  print("Jumlah duplikat di NYC setelah dihapus:", nyc\_data.duplicated().sum())  # Check for duplicates  print("Jumlah duplikat di Tokyo sebelum dihapus:", tokyo\_data.duplicated().sum())  # Drop duplicates  tokyo\_data.drop\_duplicates(inplace=True)  # Verify that duplicates have been removed  print("Jumlah duplikat di Tokyo setelah dihapus:", tokyo\_data.duplicated().sum()) |
| --- |

* + 1. Konversi Timestamp dan Ekstraksi Fitur Waktu

Langkah berikutnya adalah mengonversi kolom utcTimestamp ke dalam format datetime untuk memudahkan analisis berbasis waktu. Proses konversi ini dilakukan menggunakan fungsi **pd.to\_datetime()** dengan parameter **errors='coerce'** untuk menangani potensi kesalahan format. Hasil konversi menunjukkan bahwa semua data dalam kolom utcTimestamp berhasil diubah ke dalam format datetime.Kali ini kami juga mengekstraksi fitur waktu tambahan yaitu fitur Year, Month, dan Hour sehingga didapatkan output seperti dibawah ini.

Input data konversi timestamp

| Waktu |
| --- |
| 2020-02-30 10:00:56 |
| 2020-02-30 10:00:56 |
| .. |

Output Konversi

| utcTimestamps | Date | Time |
| --- | --- | --- |
| 2020-02-30 10:00:56 | 2020-02-30 | 10:00:56 |
| 2020-02-30 10:00:56 | 2020-02-30 | 10:00:56 |
| .. | … | … |

Output Ekstraksi Fitur pada dataset NYC

| NYC Data dengan fitur tambahan : | | | | |
| --- | --- | --- | --- | --- |
| utcTimestamp | | Year | Month | Hour |
| 0 | 2012 - 04- 03 18:00:09+00:00 | 2012 | 4 | 18 |
| 1 | 2012 - 04- 03 18:01:11+00:00 | 2012 | 4 | 18 |
| 2 | 2012 - 04- 03 18:01:25+00:00 | 2012 | 4 | 18 |
| 3 | 2012 - 04- 03 18:02:43+00:00 | 2012 | 4 | 18 |
| .. | … | .. | .. | … |

Code program

| # Mengonversi kolom 'utcTimestamp' ke format datetime  nyc\_data['utcTimestamp'] = pd.to\_datetime(nyc\_data['utcTimestamp'], format="%a %b %d %H:%M:%S %z %Y", errors='coerce')  tokyo\_data['utcTimestamp'] = pd.to\_datetime(tokyo\_data['utcTimestamp'], format="%a %b %d %H:%M:%S %z %Y", errors='coerce')  # Ekstraksi fitur waktu  nyc\_data['Year'] = nyc\_data['utcTimestamp'].dt.year  nyc\_data['Month'] = nyc\_data['utcTimestamp'].dt.month  nyc\_data['Hour'] = nyc\_data['utcTimestamp'].dt.hour  tokyo\_data['Year'] = tokyo\_data['utcTimestamp'].dt.year  tokyo\_data['Month'] = tokyo\_data['utcTimestamp'].dt.month  tokyo\_data['Hour'] = tokyo\_data['utcTimestamp'].dt.hour  if nyc\_data['utcTimestamp'].isnull().any():  print("Ada nilai yang tidak dapat dikonversi menjadi datetime.")  # Ekstraksi fitur waktu  nyc\_data['Year'] = nyc\_data['utcTimestamp'].dt.year  nyc\_data['Month'] = nyc\_data['utcTimestamp'].dt.month  nyc\_data['Hour'] = nyc\_data['utcTimestamp'].dt.hour  # Menampilkan beberapa baris untuk melihat fitur baru yang ditambahkan  print("NYC Data dengan fitur waktu tambahan:")  print(nyc\_data[['utcTimestamp', 'Year', 'Month', 'Hour']].head(10)) |
| --- |

* + 1. Normalization

Membersihkan nama kategori dengan menghilangkan spasi ekstra sehingga konsistensi nama dapat dijaga. Kita akan menggunakan MinMaxScaler dari sklearn.preprocessing untuk menormalkan nilai nama kategori. Ini akan membantu dalam analisis yang lebih baik, terutama jika kita ingin memvisualisasikan data di peta.Disini **str.strip()** untuk menghapus karakter spasi di awal dan akhir string dalam kolom venueCategory yang memastikan tidak ada spasi yang tidak sengaja mempengaruhi konsistensi data. Sedangkan **str.replace(' +', ' ')** untuk menggantikan beberapa spasi berturut-turut di dalam string dengan satu spasi yang dapat membantu dalam menstandarkan jarak antar kata dalam nama kategori, memastikan keseragaman di seluruh entri.

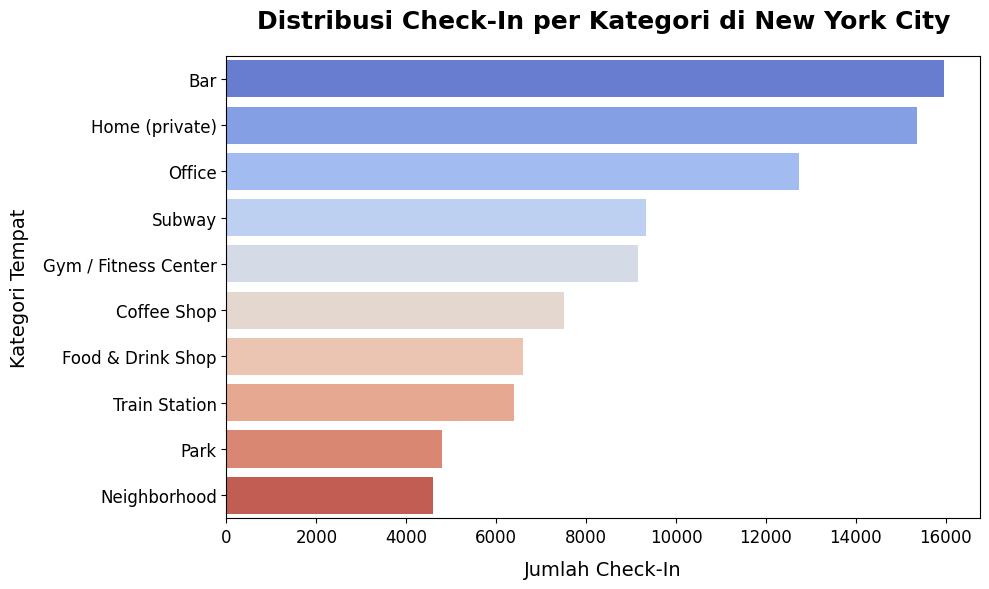
Code program

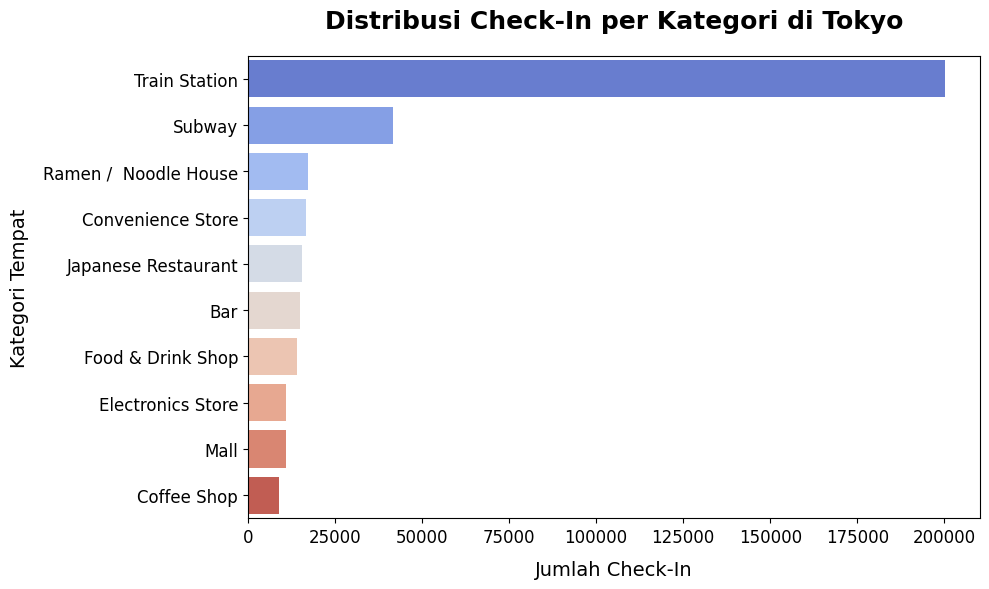
| from sklearn.preprocessing import LabelEncoder, MinMaxScaler  # Normalisasi Nama Kategori  nyc\_data['venueCategory'] = nyc\_data['venueCategory'].str.strip().str.replace(' +', ' ')  tokyo\_data['venueCategory'] = tokyo\_data['venueCategory'].str.strip().str.replace(' +', ' ') |
| --- |

* 1. Visualisasi dan Analisis
     1. Pertanyaan 1: Bagaimana distribusi check-in per kategori tempat di New York dan Tokyo?

Distribusi check-in di New York City menunjukkan bahwa kategori tempat yang paling populer adalah Bar, diikuti oleh Home (private) dan Office. Pola ini mencerminkan dominasi aktivitas sosial dan pekerjaan dalam kehidupan masyarakat New York. Selain itu, tempat umum seperti Subway, Gym/Fitness Center, dan Coffee Shop juga menunjukkan popularitas yang signifikan, mengindikasikan peran penting transportasi dan kesehatan. Sebaliknya, ruang terbuka seperti Park dan lingkungan umum seperti Neighborhood memiliki jumlah check-in yang lebih rendah, menandakan bahwa masyarakat New York cenderung beraktivitas di tempat-tempat spesifik dibandingkan di ruang terbuka.

Di Tokyo, kategori tempat yang paling sering dikunjungi adalah Train Station, diikuti oleh Subway, yang menyoroti ketergantungan masyarakat pada transportasi umum. Tempat makan seperti Ramen/Noodle House dan Japanese Restaurant juga cukup populer, menunjukkan budaya makan di luar yang kuat di kota ini. Selain itu, tempat seperti Electronics Store dan Convenience Store memiliki jumlah check-in yang signifikan, mencerminkan peran teknologi dan kebutuhan sehari-hari dalam kehidupan masyarakat Tokyo. Perbedaan ini menunjukkan bahwa New York lebih berfokus pada aktivitas sosial dan pekerjaan, sedangkan Tokyo lebih menonjol dalam aspek transportasi dan konsumsi harian.





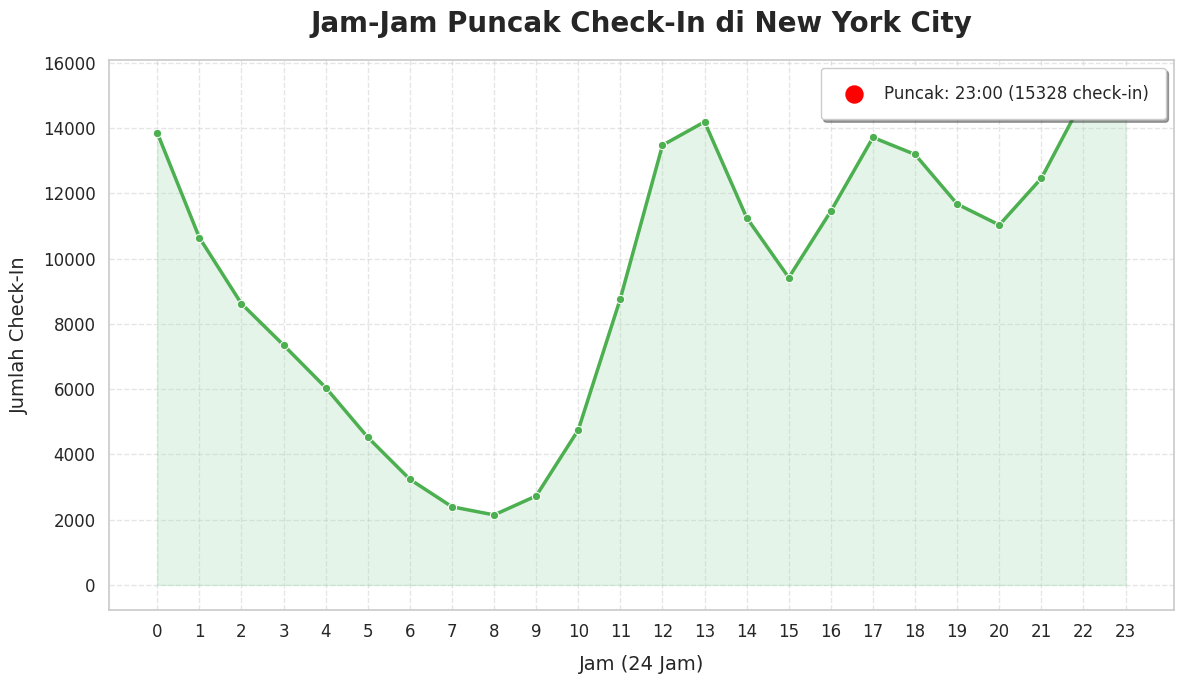
| import matplotlib.pyplot as plt  import seaborn as sns  def plot\_checkin\_distribution(city\_data, city\_name):  plt.figure(figsize=(10, 6))  category\_counts = city\_data['venueCategory'].value\_counts().head(10)  sns.barplot(  x=category\_counts.values,  y=category\_counts.index,  palette="coolwarm"  )  plt.title(f"Distribusi Check-In per Kategori di {city\_name}", fontsize=18, fontweight='bold', pad=20)  plt.xlabel("Jumlah Check-In", fontsize=14, labelpad=10)  plt.ylabel("Kategori Tempat", fontsize=14, labelpad=10)  plt.xticks(fontsize=12)  plt.yticks(fontsize=12)  plt.tight\_layout()  plt.show()  plot\_checkin\_distribution(nyc\_data, "New York City")  plot\_checkin\_distribution(tokyo\_data, "Tokyo") |
| --- |

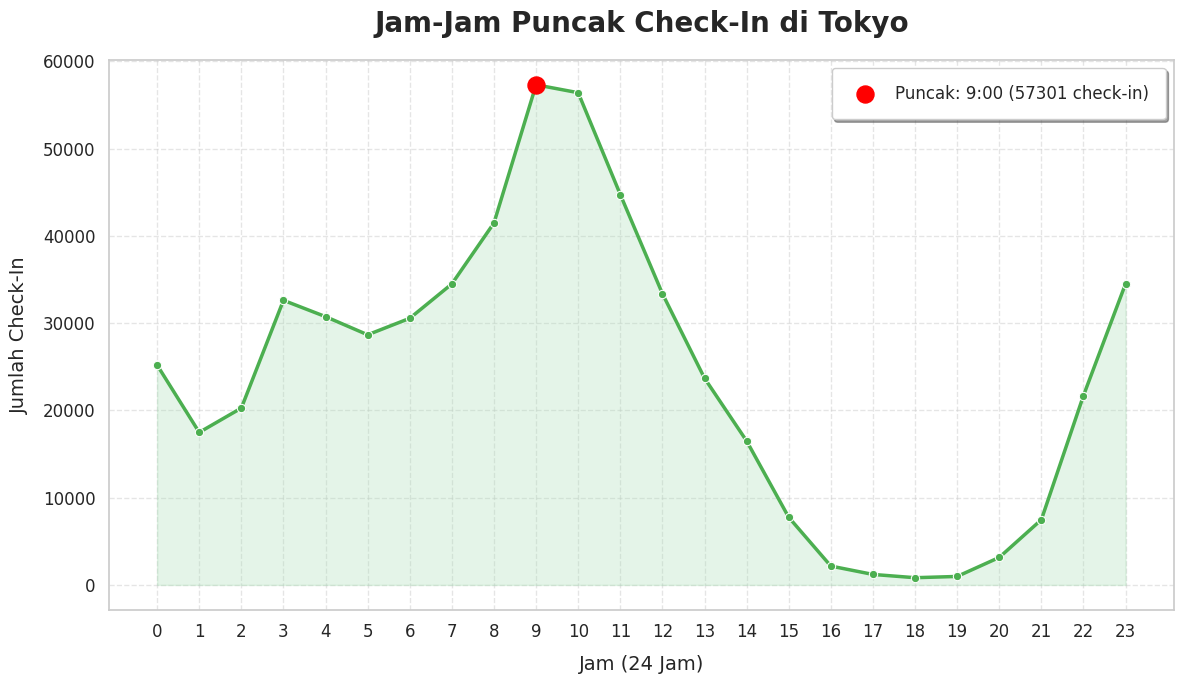
Untuk menganalisis distribusi check-in per kategori tempat di New York City dan Tokyo, dilakukan beberapa langkah pra proses data. Pertama, data dibersihkan dengan menghapus duplikasi menggunakan drop\_duplicates agar setiap baris data unik. Selanjutnya, kolom venueCategory dinormalisasi dengan menghilangkan spasi berlebih menggunakan str.strip dan str.replace untuk memastikan konsistensi data kategori. Terakhir, data difilter dengan memilih sepuluh kategori tempat dengan check-in tertinggi menggunakan value\_counts() untuk memfokuskan analisis pada kategori yang paling relevan.

* + 1. Pertanyaan 2: Kapan waktu-waktu puncak check-in di masing-masing kota?

Hasil visualisasi menunjukkan perbedaan signifikan dalam waktu-waktu puncak check-in di New York City dan Tokyo. Di Tokyo, puncak check-in terjadi pada pukul 09:00, dengan total 57.301 check-in. Hal ini menunjukkan bahwa sebagian besar aktivitas masyarakat Tokyo, seperti memulai hari kerja atau bepergian, terjadi di pagi hari, yang konsisten dengan pola transportasi umum yang dominan di kota tersebut. Sementara itu, di New York City, puncak check-in terjadi pada pukul 23:00, dengan total 15.328 check-in, mengindikasikan dominasi aktivitas malam hari seperti kehidupan sosial dan hiburan.

Praproses data yang dilakukan berperan penting dalam analisis ini. Data waktu di kolom utcTimestamp sebelumnya dikonversi ke format datetime untuk memastikan konsistensi format dan akurasi data waktu. Selanjutnya, fitur Hour diekstraksi dari kolom datetime tersebut untuk menghitung jumlah check-in berdasarkan jam dalam sehari. Praproses ini memungkinkan analisis mendalam tentang pola check-in harian di kedua kota dengan fokus pada waktu-waktu tertentu. Secara keseluruhan, perbedaan ini menggambarkan dinamika aktivitas masyarakat di Tokyo yang lebih aktif pada pagi hari, sedangkan New York lebih cenderung menunjukkan aktivitas intens di malam hari, mencerminkan budaya kota masing-masing.





Code program

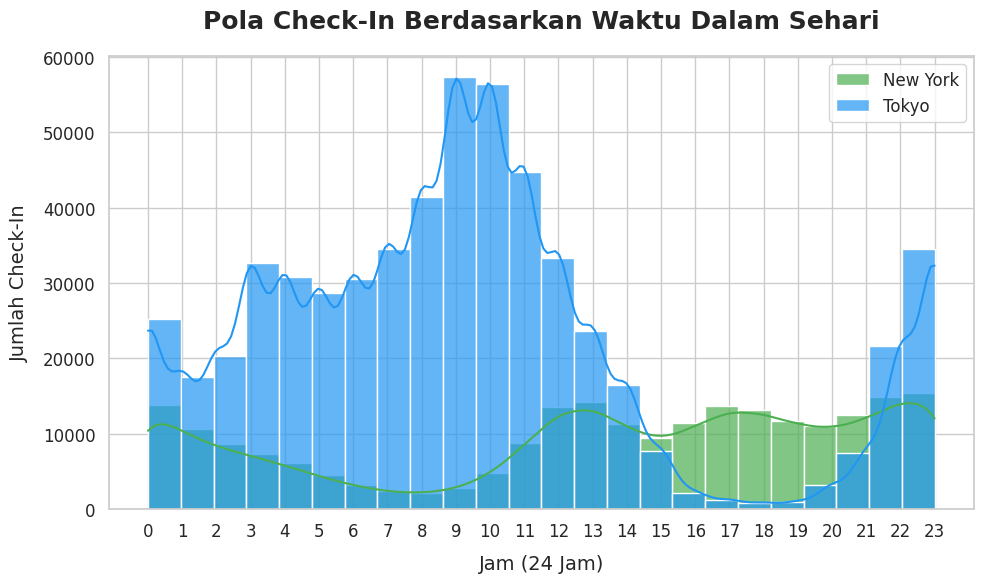
| def plot\_peak\_hours(city\_data, city\_name):  hourly\_counts = city\_data['Hour'].value\_counts().sort\_index()  plt.figure(figsize=(12, 7))  sns.set\_theme(style="whitegrid")  sns.lineplot(x=hourly\_counts.index, y=hourly\_counts.values, marker="o", linewidth=2.5, color="#4CAF50")  plt.fill\_between(hourly\_counts.index, hourly\_counts.values, color="#a8ddb5", alpha=0.3)  peak\_hour = hourly\_counts.idxmax()  plt.scatter(peak\_hour, hourly\_counts.max(), color="red", s=150, label=f"Puncak: {peak\_hour}:00 ({hourly\_counts.max()} check-in)")  plt.title(f"Jam Puncak Check-In di {city\_name}", fontsize=20, fontweight='bold')  plt.xlabel("Jam (24 Jam)", fontsize=14)  plt.ylabel("Jumlah Check-In", fontsize=14)  plt.xticks(range(0, 24), fontsize=12)  plt.legend(fontsize=12, loc='upper right', frameon=True, shadow=True)  plt.grid(axis='both', linestyle='--', alpha=0.5)  plt.tight\_layout()  plt.show()  plot\_peak\_hours(tokyo\_data, "Tokyo")  plot\_peak\_hours(nyc\_data, "New York City") |
| --- |

* + 1. Pertanyaan 3: Bagaimana pola check-in berdasarkan waktu dalam sehari di New York vs Tokyo?

Visualisasi pola check-in berdasarkan waktu dalam sehari di New York dan Tokyo menunjukkan perbedaan pola aktivitas masyarakat di kedua kota tersebut. Di Tokyo, aktivitas check-in memuncak pada pagi hari, terutama antara pukul 07:00 hingga 10:00, yang menunjukkan tingginya aktivitas masyarakat pada awal hari, seperti perjalanan ke tempat kerja atau sekolah. Setelah puncak pagi, aktivitas check-in menurun secara bertahap hingga sore hari dan kembali meningkat sedikit pada malam hari. Hal ini mencerminkan ritme kehidupan yang terstruktur dengan aktivitas utama di pagi hari.

Sebaliknya, di New York, pola check-in menunjukkan aktivitas yang relatif merata sepanjang hari dengan dua puncak utama: satu di siang hingga sore hari sekitar pukul 12:00 hingga 14:00, dan puncak lainnya pada malam hari, terutama sekitar pukul 21:00 hingga 23:00. Pola ini menggambarkan kehidupan kota yang aktif hingga larut malam, mencerminkan budaya sosial dan hiburan yang kuat.

Proses praproses data memainkan peran penting dalam analisis ini. Data waktu di kolom utcTimestamp sebelumnya dikonversi ke format datetime untuk memastikan akurasi informasi waktu. Setelah itu, fitur Hour diekstraksi dari kolom datetime untuk mempermudah analisis pola waktu harian. Dengan praproses ini, data menjadi siap untuk divisualisasikan dan memberikan wawasan yang jelas tentang pola aktivitas masyarakat di kedua kota besar tersebut. Pola-pola ini mencerminkan perbedaan budaya dan dinamika kehidupan yang khas antara Tokyo dan New York.



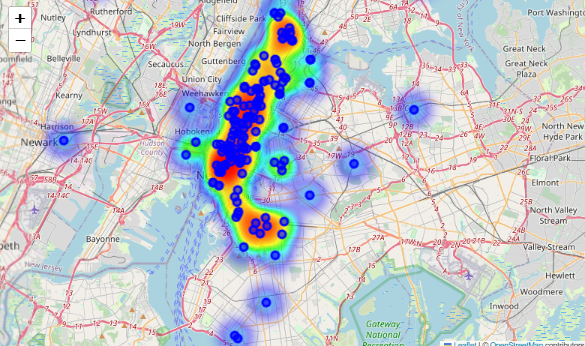
Code program

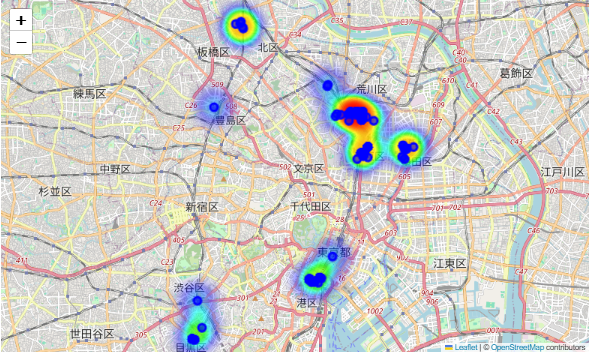
| def plot\_daily\_patterns(nyc\_data, tokyo\_data):  plt.figure(figsize=(10, 6))  sns.histplot(nyc\_data['Hour'], kde=True, label="New York", color="#4caf50", bins=24, alpha=0.7)  sns.histplot(tokyo\_data['Hour'], kde=True, label="Tokyo", color="#2196f3", bins=24, alpha=0.7)  plt.title("Pola Check-In Berdasarkan Waktu Dalam Sehari", fontsize=18, fontweight='bold', pad=20)  plt.xlabel("Jam (24 Jam)", fontsize=14, labelpad=10)  plt.ylabel("Jumlah Check-In", fontsize=14, labelpad=10)  plt.xticks(range(0, 24), fontsize=12)  plt.yticks(fontsize=12)  plt.legend(fontsize=12)  plt.tight\_layout()  plt.show()  plot\_daily\_patterns(nyc\_data, tokyo\_data) |
| --- |

* + 1. Pertanyaan 4: Area mana saja yang menjadi hotspot atau lokasi dengan konsentrasi check-in tertinggi di Tokyo dan NYC?

Berdasarkan kode dan gambar heatmap yang telah dibuat, analisis ini berfokus pada distribusi spasial check-in pengguna di dua kota, yaitu Tokyo dan New York City (NYC). Kode tersebut menggunakan Folium untuk menghasilkan peta interaktif yang menampilkan persebaran lokasi check-in melalui heatmap dan Circle Marker dengan informasi kategori tempat. Data difilter berdasarkan ID pengguna tertentu, lalu dipetakan untuk memperlihatkan pola lokasi yang sering dikunjungi. Heatmap digunakan untuk mengidentifikasi titik-titik dengan intensitas check-in tertinggi, ditandai dengan warna merah, yang menunjukkan lokasi dengan frekuensi aktivitas tertinggi.

Berdasarkan visualisasi heatmap, terlihat bahwa lokasi check-in dengan konsentrasi tertinggi di **NYC** terkumpul di beberapa area spesifik yang menunjukkan aktivitas pengguna yang sangat intens. Warna merah yang dominan pada beberapa titik mengindikasikan lokasi yang sering dikunjungi, sedangkan warna biru menunjukkan frekuensi check-in yang lebih rendah tetapi masih relevan. Pola ini menunjukkan adanya fokus aktivitas pada area-area tertentu yang menjadi pusat pergerakan pengguna. Sementara itu, pada Tokyo, distribusi check-in menunjukkan pola yang lebih tersebar dengan beberapa titik konsentrasi utama. Warna merah pada peta menandakan lokasi dengan jumlah check-in tertinggi, sementara titik-titik biru menunjukkan lokasi dengan intensitas yang lebih rendah. Meskipun lebih menyebar dibandingkan NYC, beberapa hotspot utama masih terlihat jelas sebagai pusat aktivitas pengguna.





Code program

| import folium  from folium.plugins import HeatMap  # Fungsi untuk membuat heatmap dengan marker  def create\_user\_map(data, user\_id):  user\_data = data[data['userId'] == user\_id]  map\_center = [user\_data['latitude'].mean(), user\_data['longitude'].mean()]  user\_map = folium.Map(location=map\_center, zoom\_start=12)  # Tambahkan HeatMap  heat\_data = user\_data[['latitude', 'longitude']].values.tolist()  HeatMap(heat\_data).add\_to(user\_map)  # Tambahkan CircleMarker  for \_, row in user\_data.iterrows():  folium.CircleMarker(  location=[row['latitude'], row['longitude']],  radius=5,  color='blue',  fill=True,  fill\_color='blue',  fill\_opacity=0.6,  popup=row['venueCategory'],  tooltip=row['venueCategory']  ).add\_to(user\_map)  return user\_map  # Peta untuk NYC (userId: 470)  nyc\_map = create\_user\_map(nyc\_data, 470)  # Peta untuk Tokyo (userId: 868)  tokyo\_map = create\_user\_map(tokyo\_data, 868)  nyc\_map  tokyo\_map |
| --- |

* + 1. Dashboard Interaktif

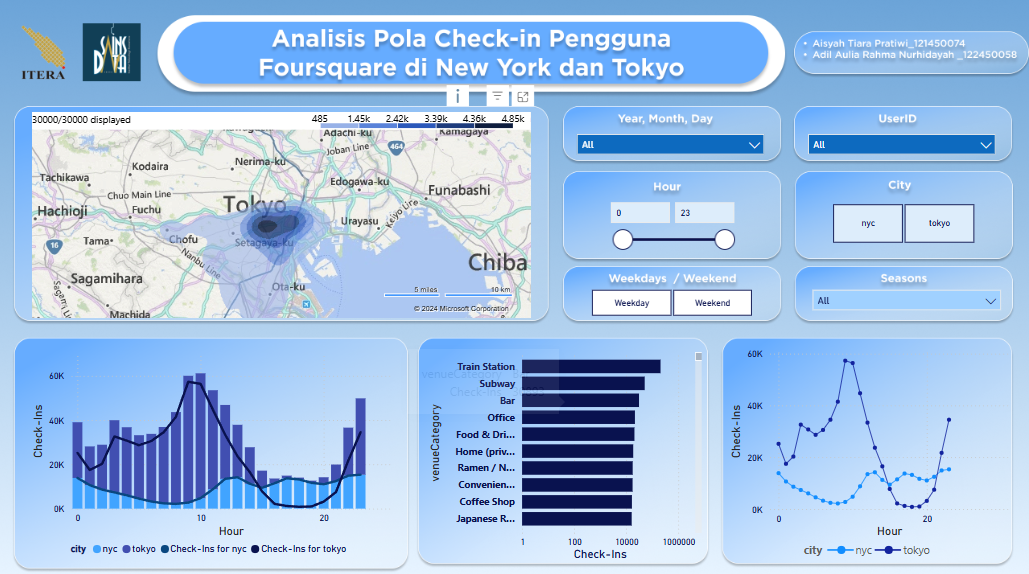
Dashboard interaktif yang dibuat memiliki sejumlah fitur yang dirancang untuk mempermudah pengguna dalam mengeksplorasi data check-in secara mendalam dan fleksibel. Salah satu fitur utamanya adalah kemampuan untuk memfilter data berdasarkan waktu, seperti tahun, bulan, hari, dan rentang jam tertentu. Dropdown untuk memilih periode waktu tertentu memungkinkan pengguna memusatkan analisis pada periode spesifik, misalnya melihat aktivitas check-in pada hari libur nasional atau selama jam sibuk pagi dan malam hari. Slider interaktif untuk memilih rentang jam dalam sehari mempermudah analisis lebih rinci, misalnya memahami kapan puncak aktivitas check-in terjadi.

Fitur lain yang tak kalah penting adalah filter berdasarkan lokasi, yakni antara New York City (NYC) dan Tokyo. Dengan filter ini, pengguna dapat dengan mudah membandingkan pola check-in antara kedua kota tersebut, baik dalam hal gaya hidup, preferensi lokasi, maupun tingkat aktivitas pada waktu tertentu. Selain itu, terdapat opsi untuk membedakan data check-in antara hari kerja (weekday) dan akhir pekan (weekend). Fitur ini membantu pengguna memahami bagaimana perilaku check-in berubah tergantung pada jenis hari, seperti dominasi check-in di kantor pada hari kerja atau peningkatan aktivitas di tempat rekreasi selama akhir pekan.

Dashboard ini juga menyediakan filter untuk menyaring data berdasarkan musim, yang relevan untuk menganalisis pengaruh cuaca terhadap preferensi lokasi check-in. Misalnya, pengguna dapat melihat tren check-in di lokasi indoor seperti kafe selama musim dingin atau di lokasi outdoor seperti taman selama musim panas. Selain itu, tersedia filter untuk memilih User ID tertentu, memungkinkan analisis perilaku individu, seperti lokasi favorit dan waktu check-in yang paling sering dilakukan.

Dari segi visualisasi, dashboard ini menampilkan heatmap interaktif yang memperlihatkan konsentrasi check-in secara geografis. Warna terang pada heatmap menunjukkan area dengan aktivitas check-in tertinggi, yang mempermudah identifikasi hotspot, seperti pusat bisnis atau lokasi wisata. Grafik waktu dalam bentuk garis juga membantu pengguna memahami pola aktivitas check-in berdasarkan jam dalam sehari atau hari dalam seminggu, dengan perbandingan langsung antara NYC dan Tokyo. Bar chart yang menampilkan jumlah check-in berdasarkan kategori tempat semakin memperjelas lokasi atau aktivitas yang paling populer di setiap kota.

Integrasi antar filter menjadikan dashboard ini sangat responsif. Setiap perubahan pada filter, seperti memilih kota Tokyo dan rentang waktu pagi hari, secara otomatis memperbarui semua visualisasi, mulai dari heatmap hingga grafik waktu dan bar chart. Hal ini memungkinkan eksplorasi data secara real-time tanpa hambatan.



# Kesimpulan

Berdasarkan analisis pola check-in pengguna Foursquare di New York dan Tokyo, berikut adalah kesimpulan yang dapat diambil:

1. Distribusi Check-In per Kategori

Di New York City, kategori tempat yang paling populer adalah Bar, diikuti oleh Home dan Office, mencerminkan fokus pada aktivitas sosial dan pekerjaan. Kategori lain seperti Subway dan Gym/Fitness Center juga menunjukkan popularitas yang signifikan, sementara ruang terbuka seperti Park memiliki jumlah check-in yang lebih rendah.Sedangkan di Tokyo, kategori tempat yang paling sering dikunjungi adalah Train Station, diikuti oleh Subway dan tempat makan seperti Ramen/Noodle House. Hal ini menunjukkan ketergantungan masyarakat Tokyo pada transportasi umum dan budaya makan di luar.

1. Waktu Puncak Check-In

Puncak check-in di Tokyo terjadi pada pukul 09:00, dengan 57.301 check-in, mencerminkan aktivitas pagi yang tinggi terkait perjalanan kerja. Sebaliknya, di New York City, puncak check-in terjadi pada pukul 23:00, dengan 15.328 check-in, menunjukkan dominasi aktivitas malam hari.ak Check-In

1. Pola Check-In berdasarkan waktu dalam sehari

Aktivitas check-in di Tokyo memuncak antara pukul 07:00 hingga 10:00, menandakan tingginya aktivitas masyarakat saat memulai hari. Sementara itu, pola check-in di New York lebih aktif di malam hari, mencerminkan perbedaan kebiasaan sosial antara kedua kota.

1. Analisa Geografis

Visualisasi heatmap menunjukkan area hotspot untuk check-in di kedua kota, membantu mengidentifikasi lokasi-lokasi yang paling sering dikunjungi oleh pengguna. Analisis ini memberikan wawasan tentang konsentrasi aktivitas pengguna dalam konteks urban.

Secara keseluruhan, analisis ini memberikan gambaran yang jelas tentang bagaimana perilaku pengguna Foursquare berbeda antara New York dan Tokyo, baik dari segi preferensi kategori tempat maupun waktu aktivitas. Hasil analisis dan visualisasi ini dapat digunakan untuk memahami lebih dalam tentang dinamika sosial dan kebiasaan pengguna di kedua kota besar tersebut.

**Lampiran**

**Link Github**:

[Link Github Kelompok 16](https://github.com/ateps/Analisis-Pola-Check-in-Pengguna-Foursquare-di-New-York-dan-Tokyo-/tree/main)

**Link Dashboard Interaktif**:

<https://bit.ly/dashboard_kelompok16>