Traffic Prediction using IOT

Devesh Rathi1, Deepanshu Paliwal2, Aman Srivastava3, Garima Singh4

1,2,3,4(Department of CSE, ABES Institute of Technology. Ghaziabad (U.P). India)

ABSTRACT: Thanks to IOT, technology nowadays has become an integral part of our lives. Real time traffic information using advanced APIs and IOT based mobile phone sensors to send information required like speed, average road density and differentiation using colors and graphs from low, moderate, and high traffic. I have developed android application by targeting user’s need to see real-time traffic using IOT, Machine Learning and GPS based advance APIs. Here APIs are collecting data using crowdsourcing and real time user’s location accumulation on server by microservices. This application consists of industry ready MVVM architecture, JSON parsing, real-time data rendering, data manipulation, and IOT based cloud computing.

Keywords: Internet of things (IoT), Real time traffic prediction, jsp, kotlin

I. INTRODUCTION

Reducing human efforts has always been the key idea behind any invention. In traffic engineering, collisions and traffic congestion are among the most challenging problems. Road capacities and road accidents have a huge impact on congestion in traffic. An accurate forecast of traffic flow is one of the key measures in Intelligent Transport Systems Android Application based on Real-time traffic data engine using advanced cloud APIs, IOT, Machine learning, and cloud computing. It includes crowdsourcing data concept also to collect the user’s location to enhance the information accuracy from time to time. In back APIs we have used tensorflow advance methods to train our model for giving results effectively. The crowdsensing model is based on the ‘collaboration’ concept. In general, people who use a specific service (e.g. mobile app) provide data and provide a “value” in exchange; for example, information on how to avoid heavy-duty highways, or information on where to shop cheaper.

Technology Used

Here we use a combination of various new technology to tackle the real-time traffic information problem. In order to understand the whole operating model, we should first understand how we quantify traffic at sites.

IOT (Internet of things)

The Internet of Things, or IoT, is a system of interlinked computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to move data over a channel without having human-- or human-- interaction.

GPS - Global Positioning System

This is a collection of 24 satellites for locations and other informations to be received. It provides details about the positions we see on google map, but in the case of traffic, the issues is that there are not enough satellites to track every vehicles in this API that we use crowdsourcing, where we take real-time location from the user’s phone and speed from both the IOT and ML phone sensors.

TensorFlow

For any TensorFlow model, TensorFlow plans to supply best on-contract variation. In either case, rapidly, the tensorflow negotiator supports associative degree obligatory sub-set of tensorflow managers that are optimized for on-contraceptive use. This includes 2 or 3 models to figure with tensorflow that need further measures. TensorFlow is second generation framework for google brain, version 1.0.0 was discharged on eleven feb 2017 through the reference implementations runs on single computer, tensorflow can operate multiple cpu’s and gpu’s (with elective CUDA and SYCL extensions for general purpose graphics process computing units) TensorFlow is obtainable on 64 bit platforms for UNIX, macos, windows and mobile computing as well as automation and ios. Its strong design allows computation to be simply distributed across a variety of platforms (cpu’s, gpu’s, Tpu’s) and from desktop to server clusters to mobile and edge devices.

Location Manager

Until your application can restore suffering area, some sensible walks around setting up need to be played. You will understand what these procedures mean in this operation. The termination advance of setting up zone update would be to verbalize the show’s real support. Should assent feel the loss, the application will obtain a SecurityException at runtime. Subordinate upon the LocationManager methods used, either ACCESS_COARSE_LOCATION or ACCESS_FINE_LOCATION assent is required. For example, you need to verbalize the ACCESS_COARSE_LOCATION support if your application uses a framework based zone provider as it were. The more exact GPS requires the ACCESS_FINE_LOCATION support. Note that telecom the ACCESS_FINE_LOCATION support proposes ACCESS_COARSE_LOCATION starting at now. While not required, most current Android-filled contraptions can get zone invigorates through different secured propels.
which are isolated to an application as LocationProvider objects. District providers may have obvious execution properties to the degree time-to-fix, precision, cash related cost, control use, and so forth. All around, a territory provider with a sensibly recognizable exactness, like the GPS, requires a more drawn out fix time than a less precise one, for instance, a structure based region provider.

Machine Learning

We use cognitive learning API’s to construct models of self learning. Machine learning is one of the best technology that we use in our application to make it sometimes more accurate. PC based understanding (ML) is a depiction that draws in programming applications to end up being reliably precise in foreseeing results without being unequivocally balanced. The fundamental explanation for AI is to make counts that can get input data and use quantifiable examination to envision a yield while invigorating yields as new data gets open.

The system related with AI resembles that of data mining and sensible appearing. Both require exploring data to channel for models and changing experience rehearses as necessities be. Various people think about AI from shopping on the web and being served headways related to their purchase. This happens considering the way wherein that proposal engines use AI to modify online business advancement is fundamentally steady. Past redo advancing, other significant AI use cases join reshaping zones, spam filtering, sort out security peril insistence, watchful sponsorship and building news sources.

Maps JavaScript API

The Maps JavaScript API lets you take people on your guide using the TransitLayer object in a city's general travel network. The guide will display substantial transit lines as dense, shaded lines at the point where the Transit Layer is activated and the guide focuses on a city that supports transit data.

The line shade is set based upon the travel line administrator's details. Empowering the Transit Layer will change the base guide design to accentuate travel courses all the more likely.

Travel data is accessible in selected areas only. Seeing a breakdown of urban communities where open travel data is now available, if this breakdown is not too difficult to advise.

In case you're an open organization that directs open transportation for your city and might want your information to be incorporated, it would be ideal if you visit the Google Transit Partner Program site to find out additional. The Traffic, Transit, and Bicycling layers alter the base guide layer to show current traffic conditions, nearby travel systems, or bicycling course data. These layers are accessible in select areas. The Maps JavaScript API enables you to include ongoing traffic data (where bolstered) to your maps utilizing the TrafficLayer object.

Traffic data is invigorated every now and again, yet not in a flash. Fast successive solicitations for a similar region are probably not going to yield various outcomes. With the Maps JavaScript API you can increase a guide with an assortment of overlays, for example, markers, polylines, polygons, and so on. Every one of these explanations joins styling data with area information. The google.maps.Data class is a holder for discretionary geospatial information. Rather than including these overlays, you can utilize the Data layer to add subjective topographical data to your guide. In the event that that information contains geometries, for example, focuses, lines or polygons, the API will render these as a matter of course as markers, polylines and polygons. You can style these highlights as you would an ordinary overlay, or apply styling rules dependent on different properties contained in your informational index.

- Data Polygon - The class Data. Polygon manages twisting of polygons for you. You may move it through a variety of at least one straight circles, defined as ordered by scope/longitude. The principal direct ring characterizes the polygon's outer boundary. The second and subsequent direct rings are used to describe inward ways (gaps) in the polygon, on the off chance that you enter more than one straight line.

- GeoJSON - GeoJSON is a typical standard for sharing geospatial information on the web. It is lightweight and effectively intelligible, making it perfect for sharing and teaming up. With the Data layer, you can add GeoJSON information to a Google map in only one line of code.

  Each guide has a map.data object, which goes about as an information layer for discretionary geospatial information, including GeoJSON. You can load and show a GeoJSON record by calling the loadGeoJSON() technique for the information object. The underneath model tells the best way to include a guide and burden outer GeoJSON information.

- Layers - Layers are questions on the guide consisting of at least one separate object, but they are managed as a single entity. For the most part, the layers represent assortments of articles you attach to give a standard affiliation over the guide. The Maps JavaScript API deals with the introduction of articles inside layers by rendering their constituent things into one item (commonly a tile overlay) and showing them as the guide's viewport changes. Layers may likewise adjust the introduction layer of the guide itself, somewhat changing the base tiles in a manner predictable with the layer. Note that most layers, by configuration, may not be gotten to by means of their individual articles, however may just be controlled as a unit.

  You can compute bearings (utilizing an assortment of techniques for transportation) by
utilizing the DirectionsService object. This item speaks with the Google Maps API Directions Service which gets bearing solicitations and returns a proficient way. Travel time is the essential factor which is improved, however different factors, for example, separation, number of transforms and a lot more might be considered. You may either deal with these headings results yourself or utilize the DirectionsRenderer item to render these outcomes.

While indicating the cause or goal in a way demanded, you can determine a question string (for instance, "Chicago, IL" or "Darwin, NSW, Australia"), a LatLng esteem, or a google.maps.Place object.

- Distance Matrix - Getting to the Distance Matrix administration is nonconcurrent, since the Google Maps API needs to make a call to an outside server. Consequently, you have to pass a callback technique to endless supply of the solicitation, to process the outcomes.

  You get to the Distance Matrix administration inside your code through the google.maps.DistanceMatrixService constructor object. The DistanceMatrixService.getDistanceMatrix() strategy starts a solicitation to the Distance Matrix administration, passing it a DistanceMatrixRequest object exacting containing the causes, goals, and travel mode, just as a callback technique to endless supply of the reaction. Use the drivingOptions article to decide a starting time to measure your goal's best path under normal traffic conditions. You can also decide if you need to be pessimistic, idealistic, or the best gage based on verifiable traffic conditions and live traffic in the rush hour gridlock.

- HeatMap - A heatmap is a perception used to delineate the power of information at land focuses. At the point when the Heatmap Layer is empowered, a shaded overlay will appear over the guide. As a matter of course, territories of higher power will be shaded red, and zones of lower power will seem green.

  The Heatmap Layer is a component of the google.maps.visualization library, and is not, of course, stacked. The Visualization classes are an independent library, separate from the JavaScript API code for primary Maps. You will initially stack it using the libraries parameter in the Maps JavaScript API bootstrap URL to exploit the utility found within this library.

  Using the DirectionsService items, you can find bearings (using an assortment of transport strategies). This article speaks to the Google Maps API Directions Service that gets requests for headings and returns an efficient way. Travel time is the critical factor that is improved, but multiple factors may be considered, for example, separation, number of transforms and much more. You can either manage the effects of these bearings yourself, or use the DirectionsRenderer article to make those effects.

Objective

The goal of this android application is to build a certain framework that is sufficiently existing to solve all traffic management issues. I will clarify my android application project focused on real-time traffic information engine. It is focused on parsing of the IOT, ML, MVVM, cloud computing, microservices and JSON.

II. METHODOLOGY

This application uses the accelerometer and gps sensors to obtain the 3 axis coordinates and later converts them to speed and then the calculated speed is taken into the memory. This happens after every 30 seconds so as to add or remove new vehicles entering or leaving the current path respectively. The basic idea is to calculate the average of the speeds of vehicles on the current path and then predicting the traffic as per the average obtained. The traffic gets classified into three types i.e. LOW, MEDIUM or HIGH traffic. There exists a particular range defined by the developer to predict the traffic into any one of the three categories. The application gains the access of the smartphone’s sensors the moments one opens the app and sends the 3 axis coordinates for further calculation. It seems similar to google maps but is a lot different from it because google maps uses the gps sensor to calculate the number of vehicles present on the road in real time whereas this application uses the average speed calculated on the basis of the moving speed of all the vehicles moving on the same path. This is one of the most accurate method to predict road traffic.
A. Workflow

The above activity diagram shows the workflow in the application, when a user installs the application, GPS is enabled and then traffic data from the Google server is requested. The Map API is invoked and that particular response is received by the device and after rendering, the data is received by the application.

B. Workflow of Google APIs in the application
The above diagram shows the pictorial representation of the workflow of APIs in the application and this entire diagram is only depiction the relation between the Google APIs in the application.

**Steps to use application:**
- Install application.
- Tap on the application icon.
- Type location where you want to see traffic status.
- Click on the wheel icon.
- Click on the play button and check all the data.

### III. RESULT AND DISCUSSION

Any person can access this app from anywhere. This application uses accelerometer, chronometer and GPS of the mobile phone. User has to enter the location they want to check the traffic. Application shows the traffic as low, moderate and high. This application also displays the max speed, average speed, distance covered and accuracy too.

**Screenshots:**

The above screenshot shows the max speed, average speed, distance travelled and accuracy.

The above screenshot shows the location input for the traffic prediction.
I. This is the main screen for traffic status of your current location by which users can get to know how much traffic he or she can face during the driving.

In the above image, the green colour on the route predicts lowest traffic.

The image above shows the moderate traffic scenario. In this the yellow color of the route predicts that there is moderate traffic on that route.
In the above, there is only red color on the route, this predicts that there is high traffic on that particular route.

IV. CONCLUSION

This traffic prediction application provides a platform for the user to check for the traffic to their destination as well as check their speed of driving. It will help users to reach their destination and find the best route to save their time. The accuracy of the application shows how much the user should depend on the result. In our case we are totally dependent on mobile phone sensors to implement IOT and cloud computing operations for better results. In the future we will include information like under construction, road break-down and other required features for user’s accessibility.

V. REFERENCES

1. https://cloud.google.com/docs/tutorials
2. https://www.tensorflow.org/learn
5. Sakitha Kumarage, Use of Crowdsourced Travel Time Data in Traffic Engineering Applications
6. Attila M.Nagy, Vilmos Simon “Survey on traffic prediction in smart cities” 
7. https://www.w3schools.com/js/js_json_parse.asp
10. https://www.w3schools.com/graphics/google_maps_intro.asp