TxDOT

All these exist for the 2024 year, but records exist going back to 2003

Crashes by road type and county (fatal):
https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/12.pdf

Crashes by hour + day: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/17.pdf

Holiday Crashes: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/18.pdf

Cause of Crash: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/21.pdf

Crash by lighting: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/23.pdf

Crash by weather: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/24.pdf

Crash by road surface condition: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/25.pdf

Work Zone Crashes: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/33.pdf

Previous Years: https://www.txdot.gov/data-maps/crash-reports-records/motor-vehicle-crash-statistics/archive.html

Notes:

Depending on previous years and real time data that is accessible, we could feed in each year and each attribute to an AI and just have it come up with the projected stats of traffic incidents this year and apply them to roads. Also, given county info, the information could be more personalized. With time sensitive information like weather, time of day, work zone, that would need to be more dynamic. If some form of real time data polling exists, sending the state of the world/weather/road to the AI as well as the local score of the area could generate a new safety score for the road, whatever that looks like.

My main concern with this data is that there is a lack of individual roads. While counties are polled as a whole in some places, none of the data in TxDOT represents individual roads. Furthermore, some of this data just splits the crash report into rural versus urban and then statewide which is less helpful for fine tuning of data than country wide.

To maybe help make county data more accessible as maybe a percent, some form of census data might be needed. Then we can see crashes per number of registered drivers.

2023 data:
General Crash Visualization:
https://tableau.txdot.gov/views/CRISCrashTreeDashboardforTxDOT_gov20240607a/CrashTreeAnalysis?%3Aembed=y&%3Atoolbar=n

This will permit us to get more specific data, assuming we want to build the queries manually:
https://cris.dot.state.tx.us/public/Query/app/query-builder
As in, we can get intersections, addresses, dates, times, damage, and everything else, as well as filter by whatever we want. Just a bit more manual. Doesn't appear to have a separate API, but we can export our data as CSV files to parse and train off of. Max entries per query caps at 50,000. Should be sufficient. Also goes back to 2016, though some details only work for the past couple years. This is the case for all of our data, but it's only considered reportable and thus reported (mandatorily) if it causes injury, death, or >= $1,000 in apparent damages. Also gives us latitude and longitude for crash locations
This may detail how to place mass queries:
https://www.txdot.gov/content/dam/docs/division/trf/crash-records/cris-guide.pdf

Census

2024 vehicle registration:
https://www.txdmv.gov/sites/default/files/report-files/Registered-Vehicles-by-County-Dec2024.pdf

This should give the total amount of vehicles registered per month. This would be a useful resource to compare to crashes. It is not a perfect comparison as many drivers probably had repeat crashes and cars may have come from out of state, but it is a point of comparison.

Other years seem to exist, but are difficult to find. I saw 2022 and 2021 at some point so the others are probably out there. The site associated with it (https://data.texas.gov/browse?sortBy=relevance&pageSize=20&limitTo=datasets&q=Register) seems to not have an easy way to locate each year of car registration.

Weather

https://api.weather.gov/ seems to be free so long as there is a registration process. One of the API calls: /alerts/active/zone/{zoneId} seems to return what we want, the JSON responses are inside of the site. I believe that either messageType or event. Either way this should work for our needs.

We will need to register for this service using a real email so that we can be contacted in case of abuse. There is a registration/authentication process. At this time I have no idea what the id for the area codes are nor how to sign up for the service.

Links with: https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/24.pdf

This https://data.democratandchronicle.com/weather-data/brown-county-tx/48049/2024-12-01/ should get us a view of how many days it rained or snowed on a per month basis. This should allow us to compare the amount of wrecks based on rain/wet roads based on how much it crashed to get a better real risk factor.

Maps API

Google maps:
https://developers.google.com/maps/documentation/geocoding/overview

We would need to reverse geocode, based on my understanding of this API. This involves taking longitude/latitude coordinates to get an address back. This may be good if we want to do local like local services. I have read that there isn't like any automatic like 'what is this road type' but there are like road prefixes that can be used by the code to determine what type of road it is (linking with https://www.txdot.gov/content/dam/docs/division/trf/crash-records/2024/12.pdf)

This seems to be the same google map platform but seems to be better at the reverse geocoding and getting the nearest road for a set of coordinates: https://developers.google.com/maps/documentation/roads/overview

This seems to be the pricing: https://developers.google.com/maps/billing-and-pricing/pricing#routes-pricing

It might be good to look more into geocoding and seeing how else google maps works with its API key. We may need to bring up billing in our meeting tomorrow. Maybe also what data this gives, see if we can use GPS or coordinates or something to get the region/county to match up with crash records

I'll just put this down: https://developers.google.com/maps/documentation/geocoding/geocoding-v4-overview
It's the geocoding v4 preview. It may be useful, but it's still in preview, and not sure if it'll be much better than the v3 listed above.
Texas also has a counties dataset that we can calculate which county we're in from coordinates:
https://data.texas.gov/dataset/Texas-Counties-Centroid-Map/ups3-9e8m/about_data

We can use this as an api as well if we want

Traffic

More detail per road/entry:
https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-annual-average-daily-traffic-counts-public/about
More entries:
https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-5-year-statewide-aadt-traffic-counts-public/about
This may be useful as well, but unsure of how useful. Top 100 most congested roadways in Texas:
https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-top-100-congested-roadways/about

This is a database that talks about the traffic in an area at any given time. This does not line up with any actual data from TxDot crash data. However, we may be able to do something to flag higher traffic areas as in more risk. We can likely extrapolate the required data from the tables though

I don't know if there is an API to pull this from in real time. We may be able to get around this through a backend and the earlier hourly crash reports. A rush hour thing can be grabbed as well, manually telling the AI that on this hour, we should think of highways at least as rush hour. This could use more looking into.

Google Maps RouteTravelAdvisory (https://developers.google.com/maps/documentation/routes_preferred/reference/rest/Shared.Types/RouteTravelAdvisory) from the above listed Google Maps Routes API permits us to fetch current traffic data for routes/streets, but doesn't get historic data. Going along with some of the pricing things above, this could get fairly expensive in training, depending on how often we want to poll the data for the training, and how many sample places we want to poll for.
I'll write this down here: https://www.openwebninja.com/api/waze