Road Report AI – Weekly Report

Traffic Data + Maps API Research

1. Summary

This week I focused on validating our options for the "Traffic" input in our model and clarifying what we should (and should not) rely on from Google Maps APIs. The main outcome is that we can support a credible Traffic feature for the MVP without relying on expensive or unreliable real-time traffic calls by using TxDOT AADT counts and time-of-day patterns from CRIS.

2. What I Looked Into

- TxDOT Open Data traffic datasets (AADT counts and congestion lists) and how they could map onto our crash dataset.
- Whether TxDOT traffic sources provide "real-time" traffic in a way we can safely integrate into our model.
- Google Maps Geocoding (reverse geocoding) vs Roads API (nearest road / snap-to-road) for working with coordinates.
- How we should determine county/region from coordinates without paying per Google API call.
- Billing / quota considerations that could impact our scope.

3. Key Findings – Traffic Data

TxDOT's public traffic datasets are valuable, but they are primarily "average volume" datasets rather than a true real-time feed. That is still useful for us because it gives an exposure/volume feature, which helps us avoid labeling highways as "more dangerous" simply because more vehicles drive there.

3.1 Recommended MVP Approach (Traffic)

- Use TxDOT Annual Average Daily Traffic (AADT) counts as our main "traffic volume / exposure" feature.
- Join CRIS crash points to the nearest AADT count point using PostGIS (nearest-neighbor spatial join).
- Use derived features such as AADT itself and a crash-rate proxy (e.g., crash_count normalized by AADT).
- Use the crash timestamp (already in CRIS) to capture time-of-day/rush-hour effects instead of trying to buy historic traffic.

3.2 Optional Enhancements (Traffic)

- Flag if a location is near a 'Top 100 Congested Roadway' segment as an additional risk indicator (useful for demo overlays).

- Investigate TxDOT's STARS II / continuous count station ecosystem only as a stretch goal; we should not depend on it for MVP.

3.3 Risk / Limitation Notes

- AADT is not real-time and won't tell us current congestion at a specific minute, but it is a strong baseline feature.
- Real-time traffic calls are hard to use for training labels (past crashes) because we generally do not have historic minute-by-minute traffic states.

4. Key Findings – Google Maps APIs

Google Maps can help with UI polish and coordinate-to-road mapping, but we should be careful about cost. For the MVP, we should avoid large-scale per-crash Google API calls during training.

4.1 Reverse Geocoding (Geocoding API)

- Reverse geocoding is best for turning a clicked point into a human-readable address for the UI.
- It can help identify admin areas (city/county) in the response, but it costs per request and shouldn't be used at training scale.

4.2 Nearest Road / Map Matching (Roads API)

- If we need the nearest road segment for a coordinate, Roads API is the correct tool (Nearest Roads / Snap to Roads).
- This is more appropriate than reverse geocoding when our goal is road alignment rather than an address label.
- We can treat road-segment mapping as a stretch goal unless we decide road-segment scoring is our MVP unit.

4.3 Traffic From Google (Routes / Advisories)

- Google Routes / RouteTravelAdvisory can provide current traffic-aware routing information.
- It does not solve historic traffic for training, and frequent polling can increase cost quickly.
- Recommendation: only use Google current-traffic as a live 'advisory' feature, not as a training dependency.

4.4 County From Coordinates (Avoid Billing)

Instead of calling Google to determine county/region, we can compute county locally using a county boundary polygon dataset and a point-in-polygon query in PostGIS.

- Do NOT rely on 'county centroid' datasets for county lookup (centroids are not boundaries).
- Use a county boundary polygon layer and run ST_Contains/ST_Intersects queries in PostGIS.

5. Decisions I Recommend We Make in the Next Team Meeting

- Traffic MVP = TxDOT AADT volume feature + CRIS time-of-day features (rush-hour derived from timestamps).

- Google APIs MVP usage = map display + (optional) reverse geocode for UI labels only.
- County resolution = PostGIS point-in-polygon using county boundary polygons (no per-request billing).
- Road type = prefer CRIS / TxDOT roadway system codes (IH/US/SH/FM/etc.), not Google inference.
- Stretch goals = Roads API nearest-road mapping and Google current traffic advisories (if we have budget).

6. Risks / Blockers

- Google billing: we should set quotas and avoid any design that requires per-crash calls during training.
- Data alignment: AADT points may not perfectly match every crash location; we should document our join strategy (nearest within threshold).
- If we commit to road-segment scoring as MVP, we will need a reliable segmentation method (grid-cell scoring is safer for MVP).

7. Next Steps (My Work Items)

- Create PostGIS tables for AADT points and (optionally) congested roadway segments; load a small sample region first.
- Implement a spatial join pipeline: CRIS crash point -> nearest AADT point (with distance threshold and fallback behavior).
- Add fields to our risk-score output to surface 'traffic volume' and 'time-of-day' contributions clearly.
- Draft an API contract note documenting that Google reverse geocoding is UI-only and not used for training.
- Bring a short billing plan to the team meeting (quotas + what calls are allowed).

8. References

1. TxDOT Open Data – Annual Average Daily Traffic (AADT) Counts (Public): https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-annual-average-daily-traffic-counts-public/about
2. TxDOT Open Data – 5-Year Statewide AADT Traffic Counts (Public): https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-5-year-statewide-aadt-traffic-counts-public/about
3. TxDOT Open Data – Top 100 Congested Roadways: https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::txdot-top-100-congested-roadways/about
4. Google Maps Platform – Geocoding API (Reverse Geocoding): https://developers.google.com/maps/documentation/geocoding/overview

5. Google Maps Platform – Roads API Overview:
https://developers.google.com/maps/documentation/roads/overview
6. Google Maps Platform – Routes API / Travel Advisories:
https://developers.google.com/maps/documentation/routes_preferred/reference/rest/Shared.Types/RouteTravelAdvisory
7. TxDOT – Traffic Count Maps / STARS II entry point:
https://www.txdot.gov/data-maps/traffic-count-maps.html
8. TxDOT Open Data – Texas County Boundaries (polygon layer):
https://gis-txdot.opendata.arcgis.com/datasets/TXDOT::texas-county-boundaries/about
9. TxDOT – CR-100 Crash Record instructions (roadway system codes):
https://www.txdot.gov/content/dam/docs/division/trf/crash-records/cr100-v29.pdf