Final Report

Unlocking New Business Opportunities in San Diego

Blake Crowther, Alan Li, Wesley Schiller

Summary

This project developed an automated business opportunity detection system using knowledge graphs and AI to analyze market data for expansion and investment opportunities. The solution addresses the challenges of manual business opportunity identification by providing faster, more accurate matching and early detection capabilities for competitive advantage.

The Challenge

- Manual business opportunity identification is time-consuming and error-prone
- Need for data-driven insights to discover areas for expansion and investment
- Market analysis requires processing complex relationships between businesses, geographic areas, and demographics

Value Proposition

- Faster opportunity identification
- Higher accuracy in matching
- Competitive advantage through early detection
- Efficient identification of underserved areas
- Improved decision-making for new business locations

1. Data Sources and Integration

1.1 Primary Data Sources

- **Business Data**

    - Google Places API providing verified real-time business information
    - Comprehensive business categorization system focused on key categories (fast-food restaurants, grocery stores, bakeries)
    - Additional fields: business address, ratings, price levels
    - Scalable approach allowing targeted start with flexibility to grow

- **Administrative Topology Data**

    - Block Groups (Census data) with spatial geometries
    - Cities and Neighborhoods data
    - Zipcode boundaries with spatial geometries
    - Hierarchical spatial relationships

- **Geo-Enrichments**

    - Population statistics
    - Age distribution
    - Wealth indices
    - Education levels
    - Fast food spending patterns

1.2 Integration Challenges

Spatial Data Complexity

- Multiple coordinate systems (EPSG:2230, EPSG:4326)
- Limited native spatial support in Neo4j
- Complex geospatial topology handling
- Direct and indirect relationship mapping

Data Quality Management

- Missing or incomplete data handling procedures
- Cross-validation between multiple sources
- Automated validation rules implementation

2. Schema Mapping and Data Transformations

The Neo4j schema was designed to analyze geographic and business-related data for identifying business opportunities. It consists of various nodes representing entities like businesses, geographic areas (block groups, zip codes, cities, neighborhoods), and demographic data (e.g., population, wealth index, crime index). The schema includes spatial layers for each entity type, facilitating geographic queries such as identifying nearby businesses or understanding business distribution in relation to socio-economic factors.

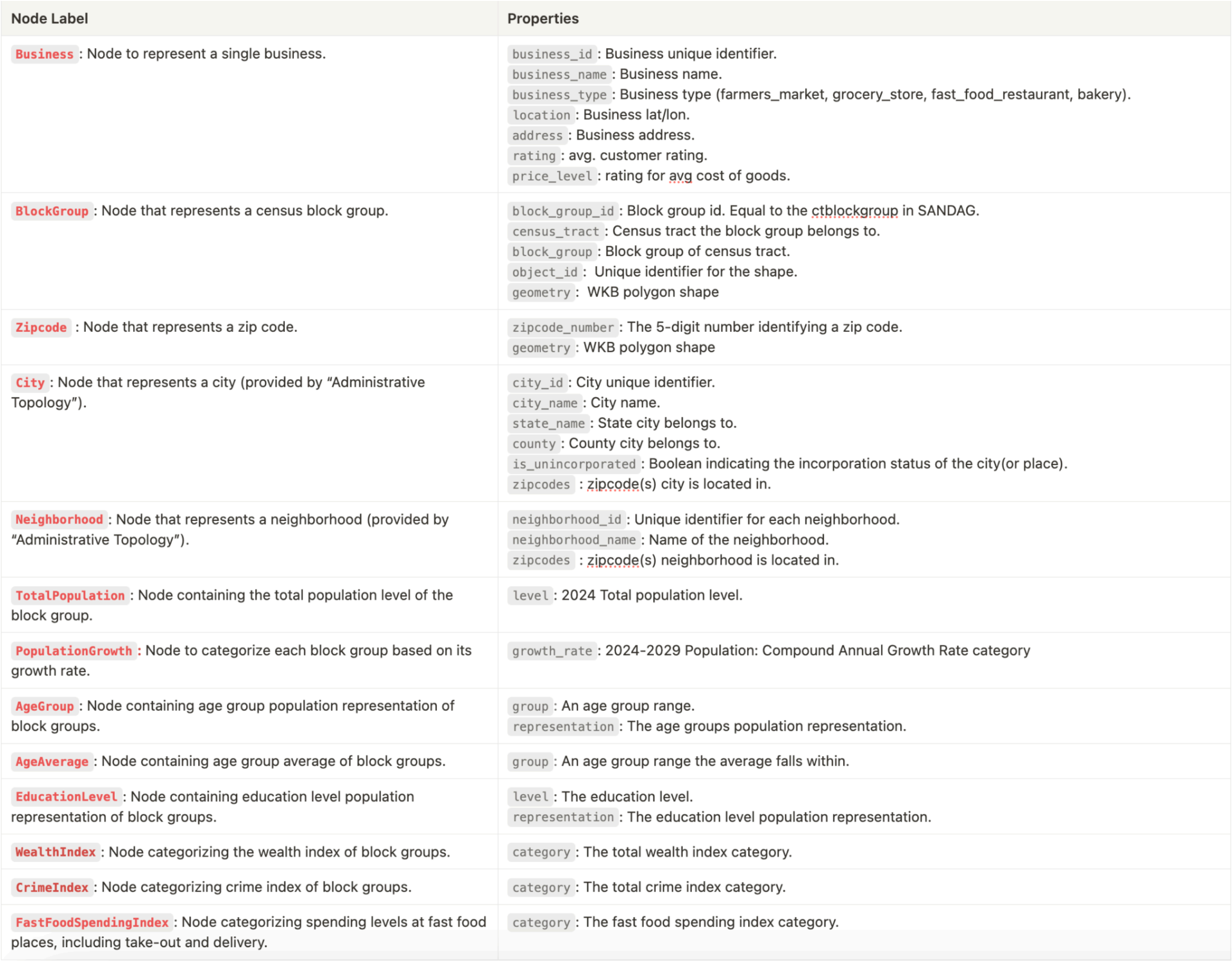
The schema's purpose is to enhance business opportunity identification by combining geographic, socio-economic, and business data. It can support entrepreneurs, business owners, and government agencies like the USDA in locating business opportunities.

Large Language Models (LLMs) were integrated into the schema's development process, aiding in:

1. **Schema design and refinement**: LLMs helped define node types, properties, and constraints to ensure data consistency.
2. **Creating relationships and queries**: LLMs assisted in generating relationships between entities and creating complex graph queries.
3. **Data analysis**: LLMs enabled insights into business trends.
4. **Natural language interaction**: LLMs allow non-technical users to interact with the data via natural language queries.
5. **Schema evolution**: As needs change, LLMs help suggest new features, optimizations, and improvements.
6. **Documentation and communication**: LLMs support generating schema documentation and communicating changes.

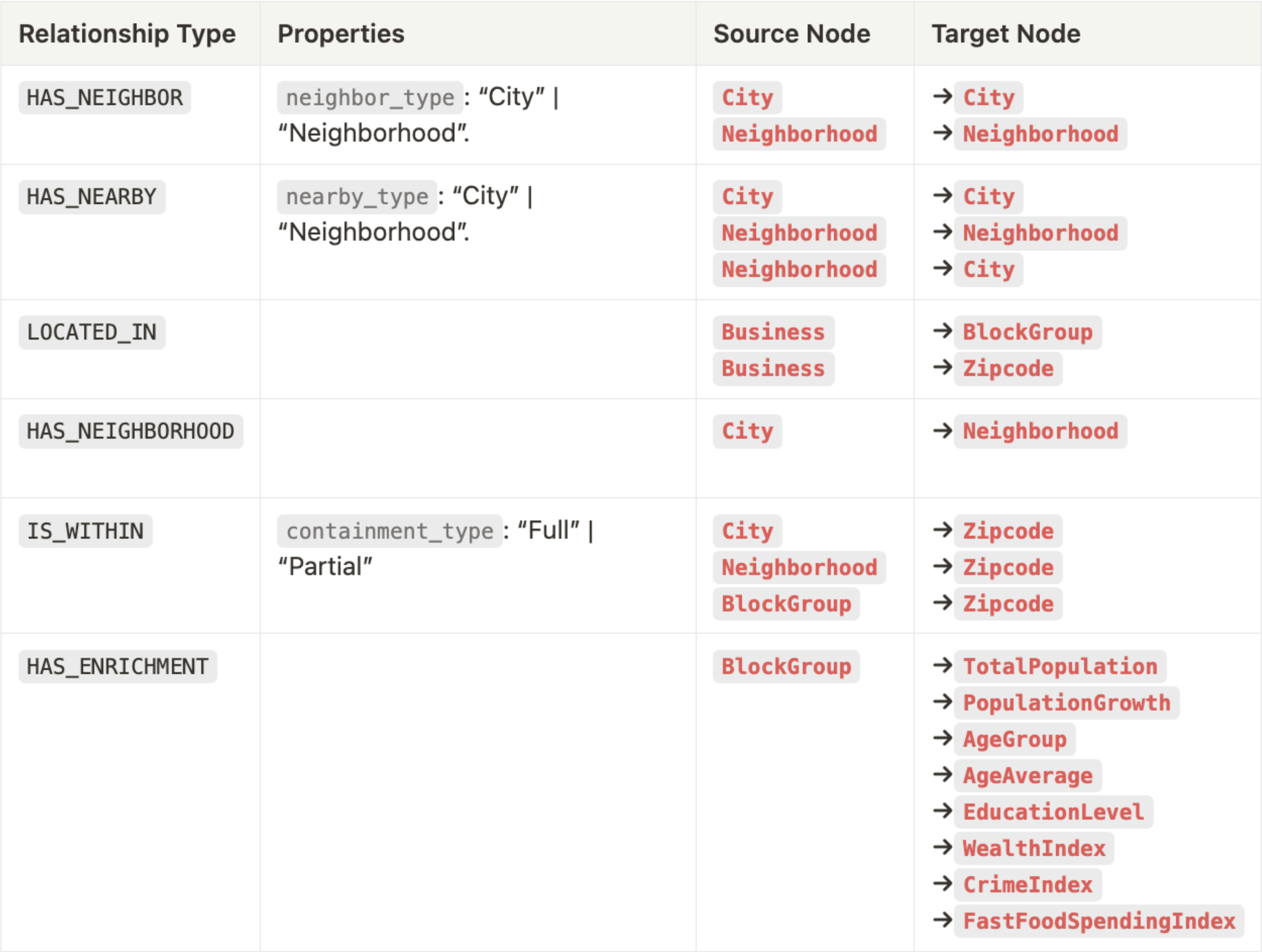
The integration of LLMs ensures the schema evolves effectively, optimizing data analysis and interactions while meeting user needs.

2.1 Node Labels



Node Label	Properties
`Business` : Node to represent a single business.	`business_id` : Business unique identifier. `business_name` : Business name. `business_type` : Business type (farmers_market, grocery_store, fast_food_restaurant, bakery). `location` : Business lat/lon. `address` : Business address. `rating` : avg. customer rating. `price_level` : rating for avg cost of goods.
`BlockGroup` : Node that represents a census block group.	`block_group_id` : Block group id. Equal to the ctblockgroup in SANDAG. `census_tract` : Census tract the block group belongs to. `block_group` : Block group of census tract. `object_id` : Unique identifier for the shape. `geometry` : WKB polygon shape
`Zipcode` : Node that represents a zip code.	`zipcode_number` : The 5-digit number identifying a zip code. `geometry` : WKB polygon shape
`City` : Node that represents a city (provided by "Administrative Topology").	`city_id` : City unique identifier. `city_name` : City name. `state_name` : State city belongs to. `county` : County city belongs to. `is_unincorporated` : Boolean indicating the incorporation status of the city(or place). `zipcodes` : zipcode(s) city is located in.
`Neighborhood` : Node that represents a neighborhood (provided by "Administrative Topology").	`neighborhood_id` : Unique identifier for each neighborhood. `neighborhood_name` : Name of the neighborhood. `zipcodes` : zipcode(s) neighborhood is located in.
`TotalPopulation` : Node containing the total population level of the block group.	`level` : 2024 Total population level.
`PopulationGrowth` : Node to categorize each block group based on its growth rate.	`growth_rate` : 2024-2029 Population: Compound Annual Growth Rate category
`AgeGroup` : Node containing age group population representation of block groups.	`group` : An age group range. `representation` : The age groups population representation.
`AgeAverage` : Node containing age group average of block groups.	`group` : An age group range the average falls within.
`EducationLevel` : Node containing education level population representation of block groups.	`level` : The education level. `representation` : The education level population representation.
`WealthIndex` : Node categorizing the wealth index of block groups.	`category` : The total wealth index category.
`CrimeIndex` : Node categorizing crime index of block groups.	`category` : The total crime index category.
`FastFoodSpendingIndex` : Node categorizing spending levels at fast food places, including take-out and delivery.	`category` : The fast food spending index category.

2.2 Relationships



Relationship Type	Properties	Source Node	Target Node
`HAS_NEIGHBOR`	`neighbor_type` : "City" | "Neighborhood".	`City` `Neighborhood`	→ `City` → `Neighborhood`

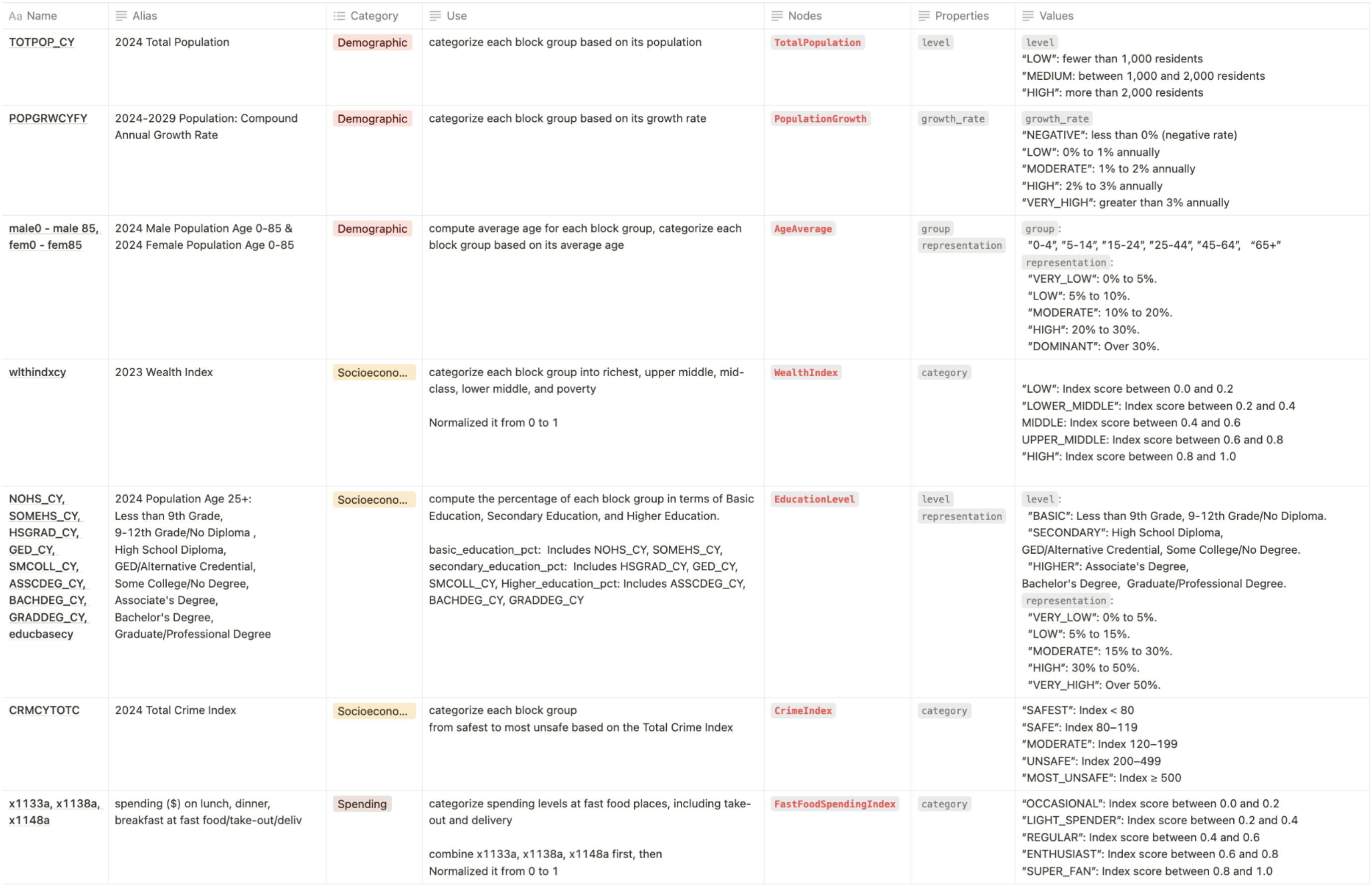
`HAS_NEARBY`	`nearby_type` : "City" | "Neighborhood".	`City`
		`Neighborhood`
		`Neighborhood`
			→ `City`
			→ `Neighborhood`
			→ `City`

`LOCATED_IN`		`Business` `Business`	→ `BlockGroup` → `Zipcode`
`HAS_NEIGHBORHOOD`		`City`	→ `Neighborhood`

`IS_WITHIN`	`containment_type` : "Full" | "Partial"	`City`
		`Neighborhood`
		`BlockGroup`
			→ `Zipcode`
			→ `Zipcode`
			→ `Zipcode`

`HAS_ENRICHMENT`		`BlockGroup`	→ `TotalPopulation` → `PopulationGrowth` → `AgeGroup` → `AgeAverage` → `EducationLevel` → `WealthIndex` → `CrimeIndex` → `FastFoodSpendingIndex`

2.3 Geoenrichment Mappings



Name	Alias	Category	Use	Nodes	Properties	Values
TOTPOP_CY	2024 Total Population	Demographic	categorize each block group based on its population	TotalPopulation	level	level "LOW": fewer than 1,000 residents "MEDIUM: between 1,000 and 2,000 residents "HIGH": more than 2,000 residents
POPGRWCYFY	2024-2029 Population: Compound Annual Growth Rate	Demographic	categorize each block group based on its growth rate	PopulationGrowth	growth_rate	growth_rate "NEGATIVE": less than 0% (negative rate) "LOW": 0% to 1% annually "MODERATE": 1% to 2% annually "HIGH": 2% to 3% annually "VERY_HIGH": greater than 3% annually
male0 - male 85, fem0 - fem85	2024 Male Population Age 0-85 & 2024 Female Population Age 0-85	Demographic	compute average age for each block group, categorize each block group based on its average age	AgeAverage	group representation	group : "0-4", "5-14", "15-24", "25-44", "45-64", "65+" representation : "VERY_LOW": 0% to 5%. "LOW": 5% to 10%. "MODERATE": 10% to 20%. "HIGH": 20% to 30%. "DOMINANT": Over 30%.
wlthindxcy	2023 Wealth Index	Socioecono...	categorize each block group into richest, upper middle, mid-class, lower middle, and poverty Normalized it from 0 to 1	WealthIndex	category	"LOW": Index score between 0.0 and 0.2 "LOWER_MIDDLE": Index score between 0.2 and 0.4 MIDDLE: Index score between 0.4 and 0.6 UPPER_MIDDLE: Index score between 0.6 and 0.8 "HIGH": Index score between 0.8 and 1.0
NOHS_CY, SOMEHS_CY, HSGRAD_CY, GED_CY, SMCOLL_CY, ASSCDEG_CY, BACHDEG_CY, GRADDEG_CY, educbasecy	2024 Population Age 25+: Less than 9th Grade, 9-12th Grade/No Diploma , High School Diploma, GED/Alternative Credential, Some College/No Degree, Associate's Degree, Bachelor's Degree, Graduate/Professional Degree	Socioecono...	compute the percentage of each block group in terms of Basic Education, Secondary Education, and Higher Education. basic_education_pct: Includes NOHS_CY, SOMEHS_CY, secondary_education_pct: Includes HSGRAD_CY, GED_CY, SMCOLL_CY, Higher_education_pct: Includes ASSCDEG_CY, BACHDEG_CY, GRADDEG_CY	EducationLevel	level representation	level : "BASIC": Less than 9th Grade, 9-12th Grade/No Diploma. "SECONDARY": High School Diploma, GED/Alternative Credential, Some College/No Degree. "HIGHER": Associate's Degree, Bachelor's Degree, Graduate/Professional Degree. representation : "VERY_LOW": 0% to 5%. "LOW": 5% to 15%. "MODERATE": 15% to 30%. "HIGH": 30% to 50%. "VERY_HIGH": Over 50%.
CRMCYTOTC	2024 Total Crime Index	Socioecono...	categorize each block group from safest to most unsafe based on the Total Crime Index	CrimeIndex	category	"SAFEST": Index < 80 "SAFE": Index 80–119 "MODERATE": Index 120–199 "UNSAFE": Index 200–499 "MOST_UNSAFE": Index ≥ 500
x1133a, x1138a, x1148a	spending ($) on lunch, dinner, breakfast at fast food/take-out/deliv	Spending	categorize spending levels at fast food places, including take-out and delivery combine x1133a, x1138a, x1148a first, then Normalized it from 0 to 1	FastFoodSpendingIndex	category	"OCCASIONAL": Index score between 0.0 and 0.2 "LIGHT_SPENDER": Index score between 0.2 and 0.4 "REGULAR": Index score between 0.4 and 0.6 "ENTHUSIAST": Index score between 0.6 and 0.8 "SUPER_FAN": Index score between 0.8 and 1.0

3. Knowledge Graph Construction

3.1 ETL Pipeline Architecture

- Modular design with separate domain components
- Automated data validation against JSON schema
- Configurable graph cleanup and reprocessing
- Robust error handling and progress reporting
- Parameterized execution for targeted updates

3.2 Business Data Integration

- **Data Source Interaction**

    1. **Google Places API Integration**

- Asynchronous client implementation
- Field mask optimization for targeted data retrieval
- Language code specification for consistent results
- Business type filtering capabilities

2. **Spatial Query Optimization**

    - Block group geometry extraction from Neo4j
    - Minimum enclosing circle calculation for each block group
    - Radius optimization for API coverage
    - Coordinate system handling

- **Business Data Collection Process**

    1. **Systematic Area Coverage**

        - Block group-based querying strategy
        - Center point and radius calculation
        - Spatial containment verification
        - Business type categorization:
            - Fast food restaurants
            - Grocery stores
            - Bakeries

    2. **Data Field Collection**

        - Required fields:
            - Unique business ID
            - Business name
            - Business type
            - Latitude/longitude coordinates
        - Optional enrichments:
            - Formatted address
            - Rating information
            - Price level indicators

- **Spatial Integration Process**

    1. **Point Layer Management**

        - Dedicated business spatial layer initialization
        - Point geometry creation from coordinates
        - Spatial index utilization
        - Layer-specific configurations

2. **Location Validation**

    - Polygon containment checks
    - Coordinate validation
    - Block group boundary verification
    - Zipcode area determination

- **Relationship Establishment**

    1. **Primary Relationships**

        - LOCATED_IN relationships to block groups
        - LOCATED_IN relationships to zipcodes
        - Automated relationship property assignment
        - Spatial validation checks

    2. **Location Resolution**

        - Block group containment verification
        - Zipcode lookup for each business
        - Coordinate-based relationship validation
        - Multiple containment handling

- **Quality Control Implementation**

    1. **Data Validation**

        - Schema constraint enforcement
        - Required field verification
        - Spatial data validation
        - Business type verification

    2. **Error Management**

        - Individual business failure tracking
        - API error handling
        - Transaction management
        - Detailed error logging

    3. **Performance Monitoring**

        - Success/failure metrics tracking
        - API quota management

    - Processing statistics collection
    - Progress reporting

This implementation ensures:

- Efficient API utilization
- Accurate spatial placement
- Comprehensive data collection
- Robust error handling

3.3 Administrative Topology Data Integration

- **Data Source Extraction**

  1. **Block Group Data**

     - Fetched from PostgreSQL using GeoPandas
     - SQL queries with configurable filters
     - Automatic coordinate system conversion from EPSG:2230 to EPSG:4326
     - Geometry column handling with WKB format

  2. **Administrative Data**

     - City and neighborhood data from PostgreSQL
     - Zipcode data from CSV files
     - Combined data validation and deduplication
     - Relationship data extraction (neighboring areas, contained areas)

- **Node Creation Process**

  1. **Spatial Layer Initialization**

     - Created dedicated spatial layers for block groups and zipcodes
     - Configured WKT geometry property names
     - Initialized spatial indices for efficient querying
     - Established layer-specific configurations

  2. **Node Creation and Validation**

     - Validated properties against JSON schema constraints
     - Implemented match keys for deduplication
     - Added spatial geometries where available
     - Tracked success/failure metrics for each node type

  3. **Property Assignment**

    - Block Groups: census tract, block group ID, object ID, WKT geometry
    - Zipcodes: zipcode number, optional WKT geometry
    - Cities: city ID, name, state, county, incorporation status
    - Neighborhoods: neighborhood ID, community name

- **Relationship Construction**

  1. **Containment Relationships**

     - IS_WITHIN relationships between geographic entities
     - Computed containment types (Full/Partial)
     - Calculated overlap ratios for partial containment
     - Validated spatial relationships

  2. **Proximity Relationships**

     - HAS_NEIGHBOR relationships for adjacent entities
     - HAS_NEARBY relationships for proximate areas
     - Relationship properties for entity types
     - Bidirectional relationship creation

  3. **Spatial Intersection Processing**

     - Used Neo4j spatial procedures for intersection detection
     - Shapely geometry operations for precise calculations
     - Overlap ratio computation for containment classification
     - Automated relationship property assignment

- **Quality Control Measures**

  1. **Data Validation**

     - Schema constraint checking
     - Required field verification
     - Spatial validity checks
     - Relationship consistency validation

  2. **Error Handling**

     - Detailed error logging for failed operations
     - Success/failure count tracking
     - Individual failure reporting
     - Transaction management

3. **Cleanup and Maintenance**

    - Configurable node cleanup
    - Spatial layer reset capabilities
    - Incremental update support
    - Progress monitoring and reporting

This generalized approach ensures:

- Scalable geographic data integration
- Consistent relationship modeling
- Robust quality control
- Maintainable system architecture

3.4 Geo-Enrichments Data Integration

- **Data Preparation and Transformation**

1. **Demographic Calculations**

    - Age distribution analysis using midpoint calculations
    - Gender-based population aggregation
    - Weighted age averages computation
    - Population totals normalization

2. **Index Normalization**

    - Wealth index standardization
    - Fast food spending normalization
    - Education level aggregation:
        - Basic (no high school, some high school)
        - Secondary (high school grad, GED, some college)
        - Higher (associate's, bachelor's, graduate degrees)

- **Enrichment Categories**

1. **Population Metrics**

    - Total Population (LOW, MEDIUM, HIGH)
    - Population Growth (NEGATIVE, LOW, MODERATE, HIGH, VERY_HIGH)
    - Age Average (0-4, 5-14, 15-24, 25-44, 45-64, 65+)
    - Age Group Representation (VERY_LOW to DOMINANT)

2. **Socioeconomic Indicators**

        - Wealth Index (LOW to HIGH)
        - Education Level (BASIC, SECONDARY, HIGHER)
        - Education Representation (VERY_LOW to VERY_HIGH)
        - Crime Index (SAFEST to MOST_UNSAFE)

    3. **Consumer Behavior**

        - Fast Food Spending Index:
            - OCCASIONAL
            - LIGHT_SPENDER
            - REGULAR
            - ENTHUSIAST
            - SUPER_FAN

- **Node Creation Process**

    1. **Category Node Generation**

        - Automated creation from schema constraints
        - Enumerated value validation
        - Property combinations generation
        - Index creation for efficient querying

    2. **Data Quality Controls**

        - Zero population handling
        - Value range validation
        - Category boundary checks
        - Null value management

- **Relationship Construction**

    1. **HAS_ENRICHMENT Relationships**

        - Block group to enrichment category linking
        - Source value preservation
        - Multiple category associations
        - Relationship property assignment

    2. **Quality Assurance**

        - Success/failure tracking
        - Error logging and reporting

        - Transaction management
        - Data consistency verification

- **Implementation Features**

    1. **Technical Components**

        - Schema-driven node creation
        - Automated index management
        - Configurable cleanup processes
        - Progress monitoring

    2. **Performance Optimization**

        - Batch relationship creation
        - Efficient data transformation
        - Index utilization
        - Memory management

This implementation provides:

- Comprehensive demographic analysis
- Standardized categorization
- Efficient data processing
- Robust error handling
- Flexible enrichment framework

4. Knowledge Graph Querying

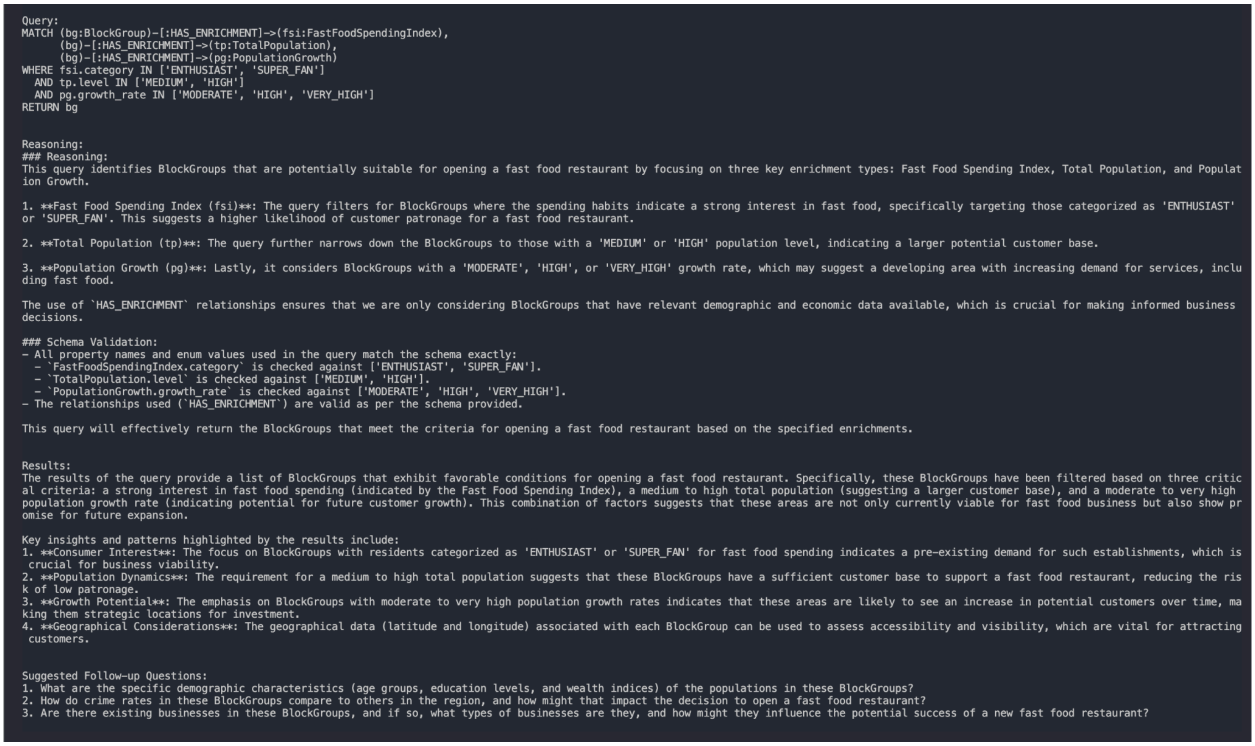
4.1 Two-Phase LLM Approach

- **Phase 1: Graph Querying**

    - Combines prompt engineering with spatial procedures
    - Leverages detailed data model
    - Executes complex spatial queries

- **Phase 2: Results Interpretation**

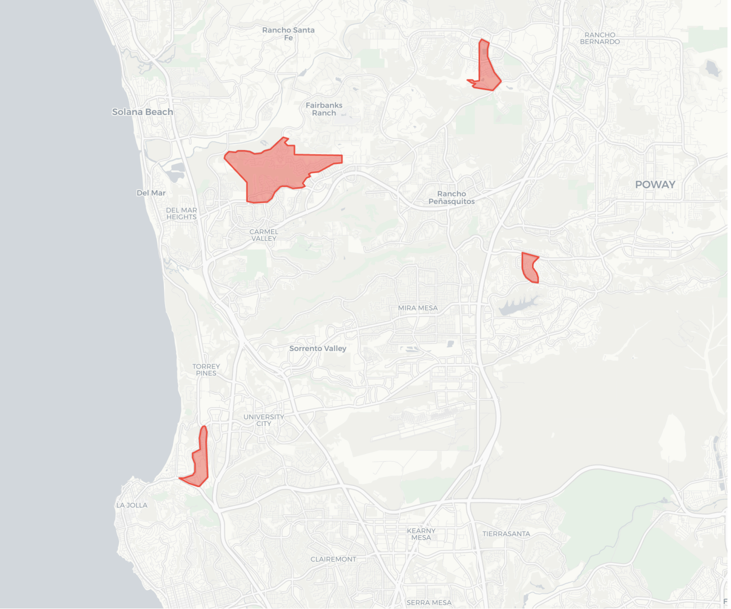
    - LLM processes query results
    - Provides intuitive explanations
    - Helps users understand implications
    - Contextualizes findings for business decisions



```
Query:
MATCH (bg:BlockGroup)-[:HAS_ENRICHMENT]->(fsi:FastFoodSpendingIndex),
      (bg)-[:HAS_ENRICHMENT]->(tp:TotalPopulation),
      (bg)-[:HAS_ENRICHMENT]->(pg:PopulationGrowth)
WHERE fsi.category IN ['ENTHUSIAST', 'SUPER_FAN']
  AND tp.level IN ['MEDIUM', 'HIGH']
  AND pg.growth_rate IN ['MODERATE', 'HIGH', 'VERY_HIGH']
RETURN bg


Reasoning:
### Reasoning:
This query identifies BlockGroups that are potentially suitable for opening a fast food restaurant by focusing on three key enrichment types: Fast Food Spending Index, Total Population, and Populat
ion Growth.

1. **Fast Food Spending Index (fsi)**: The query filters for BlockGroups where the spending habits indicate a strong interest in fast food, specifically targeting those categorized as 'ENTHUSIAST'
or 'SUPER_FAN'. This suggests a higher likelihood of customer patronage for a fast food restaurant.

2. **Total Population (tp)**: The query further narrows down the BlockGroups to those with a 'MEDIUM' or 'HIGH' population level, indicating a larger potential customer base.

3. **Population Growth (pg)**: Lastly, it considers BlockGroups with a 'MODERATE', 'HIGH', or 'VERY_HIGH' growth rate, which may suggest a developing area with increasing demand for services, inclu
ding fast food.

The use of `HAS_ENRICHMENT` relationships ensures that we are only considering BlockGroups that have relevant demographic and economic data available, which is crucial for making informed business
decisions.

### Schema Validation:
- All property names and enum values used in the query match the schema exactly:
  - `FastFoodSpendingIndex.category` is checked against ['ENTHUSIAST', 'SUPER_FAN'].
  - `TotalPopulation.level` is checked against ['MEDIUM', 'HIGH'].
  - `PopulationGrowth.growth_rate` is checked against ['MODERATE', 'HIGH', 'VERY_HIGH'].
- The relationships used (`HAS_ENRICHMENT`) are valid as per the schema provided.

This query will effectively return the BlockGroups that meet the criteria for opening a fast food restaurant based on the specified enrichments.


Results:
The results of the query provide a list of BlockGroups that exhibit favorable conditions for opening a fast food restaurant. Specifically, these BlockGroups have been filtered based on three critic
al criteria: a strong interest in fast food spending (indicated by the Fast Food Spending Index), a medium to high total population (suggesting a larger customer base), and a moderate to very high
population growth rate (indicating potential for future customer growth). This combination of factors suggests that these areas are not only currently viable for fast food business but also show pr
omise for future expansion.

Key insights and patterns highlighted by the results include:
1. **Consumer Interest**: The focus on BlockGroups with residents categorized as 'ENTHUSIAST' or 'SUPER_FAN' for fast food spending indicates a pre-existing demand for such establishments, which is
 crucial for business viability.
2. **Population Dynamics**: The requirement for a medium to high total population suggests that these BlockGroups have a sufficient customer base to support a fast food restaurant, reducing the ris
k of low patronage.
3. **Growth Potential**: The emphasis on BlockGroups with moderate to very high population growth rates indicates that these areas are likely to see an increase in potential customers over time, ma
king them strategic locations for investment.
4. **Geographical Considerations**: The geographical data (latitude and longitude) associated with each BlockGroup can be used to assess accessibility and visibility, which are vital for attracting
 customers.


Suggested Follow-up Questions:
1. What are the specific demographic characteristics (age groups, education levels, and wealth indices) of the populations in these BlockGroups?
2. How do crime rates in these BlockGroups compare to others in the region, and how might that impact the decision to open a fast food restaurant?
3. Are there existing businesses in these BlockGroups, and if so, what types of businesses are they, and how might they influence the potential success of a new fast food restaurant?
```

4.2 Visualization Capabilities

- Dynamic map generation based on query results
- Clear visualization of spatial relationships
- Intuitive presentation of business patterns
- Helps users quickly identify promising opportunities
- Eliminates need to parse raw data



Query: What block groups might be best for opening a fast food restaurant?

Additional context: "Try to return the blockgroup nodes in the result.

5. Future Directions

5.1 Planned Enhancements

- Machine learning model improvements
- Prompt engineering for more complex queries
- Additional data source integration
- Scalability optimizations
- New feature development

References

1. Google Places API Documentation
2. Neo4j Spatial Documentation
3. SANDAG GIS Data Portal
4. Census Bureau Data Documentation