University of Nebraska at Omaha

Thesis-Equivalent Project

Bridging the Data: Adding Highly-Accurate Bridge Data to OpenStreetMaps

William Heller

Department of Computer Science

12/7/2023

Supervisory Committee:

Brian Ricks

Andrea Grover

Pei-Chi Huang

Abstract

Large open-source geographic datasets, such as OpenStreetMaps (OSM), contain a vast amount of data on roads, buildings, bridges, and other locales across the entire globe. Other geographic databases, such as The National Bridge Inventory (NBI), provide more specialized geographic data about bridges and bridge health. While both these datasets provide information on real-world places and infrastructure, a much richer dataset can be created by combining the two. This merged dataset would provide all geographical information from OSM, contain detailed bridge information from NBI, and could be used to perform vehicle routing that focuses on preserving bridge health while routing large vehicle loads, such as military convoys. This project introduces such a dataset by combining OSM and NBI through multiple iterations of a data matching process. The final result uses data querying to find relevant data and employs and heuristics that mimic human analysis to avoid false positives and employs a light API architecture to facilitate the process. This process can be generalized to enrich OSM with other data such as restaurant reviews and updated speed limits. In general, this process specializes in enhancing existing geographic information by performing informed and confident decision making, rather than adding new data points.

1 Introduction

Bridge health has been a growing topic of interest, and many new technologies are being developed to analyze, predict, and maintain bridge health. Many systems are developed with the intent to describe current bridge health, predict future conditions, and estimate necessary repair dates. Other systems have taken a preventative approach to bridge health, which is key to improving overall bridge health and lifespans. One preventative measure that remains largely unexplored is bridge-aware vehicle routing, where one or several vehicles are routed to avoid structurally unsound bridges. For instance, when routing military convoys with heavy multi-ton loads, we must consider the integrity of the bridges traversed en route, ensuring load-bearing structures will not be damaged or destroyed by the weight of its carried convoy. To create such a system, data must be present that accurately defines roadways, bridge locations, and bridge health ratings. To reduce the potential for errors and increase the number of use cases, the data must be confined to a single standardized source.

Many world map datasets, such as Google Maps or OpenStreetMaps (OSM), contain a vast amount of reliable data about roads, buildings, sidewalks and more. However, they contain minimal information about bridges. At best, these datasets provide a bridge's location, name, and structure type; at worst, a given bridge may not *exist* in these datasets. To add richer bridge information and reduce inconsistencies in a map dataset, a bridge-oriented dataset must be introduced. Enter, The National Bridge Inventory (NBI), a large database which contains inspection records for every bridge in the United States. Its data is collected once every two years from various agencies that report bridge health and condition ratings. While researchers have analyzed the accuracy and usefulness of datasets like OSM and NBI, there is little research on combining these datasets together. Tools exist for importing big data from various sources into OSM, but they often focus on importing similar geographical data and not vastly differing data types, such as bridge inspection records from NBI. This project proposes a method of adding NBI data to the OSM database using specialized heuristics that mimic human decision, custom visualization tools, and various open-source APIs. The success of this implementation will be measured by the confidence in our data import via automated measurements and visualisations. The percentage of bridges successfully imported is of large importance, though the precision of the added data takes precedent.

2 Background

Before exploring approaches to the data combining process, the nuances of format and geo-spatial issues of OSM and NBI must be understood. While both datasets are publicly available, they differ greatly in the data they provide, and storage format. OSM is a map of the world with user-volunteered data, while NBI features government-managed bridge health and repair records. Common fields must be discerned that can correlate OSM and NBI data points. The most logical choice is matching bridges between datasets by location and name. However, both datasets struggle from inaccuracies and inconsistencies that must be identified to avoid imprecise matching.

2.1 OpenStreetMaps Data

OpenStreetMaps is an open-source global mapping project founded in 2004 by Steve Coast [9]. Since its conception, OSM has seen exponential growth and increasingly accurate data. OSM is often cited as being highly reliable thanks to its open-source nature and growing user base of volunteers[1, 16]. However, limitations – such as incomplete or inaccurate data – exist in rural or less populated areas[11]. While Google Maps is often found to be more accurate, OSM provides finer detail in many instances and has no restrictions on usage [15]. Tools have been made to import geographical data from various sources and allow users to fine-tune specific data points. Though OSM is generally accurate and detailed, bridges are often lacking in information. Most bridges in OSM are labelled correctly but may be missing fields such as the type of bridge, construction material, year made, etc. These fields are superfluous to the average OSM user, but are vital when performing bridge-aware routing.

When working with OSM data, there are several different file formats available. Two formats in particular, OSM XML and PBF (Protocolbuffer Binary Format), are vital. The OSM XML format can be considered a "raw data" format, listing all data in a human-readable XML format. This format is generally larger in size compared to other OSM formats and slower to work with due to the nature of its XML structure. To increase performance when reading and writing OSM data, we use the PBF format. The PBF format is a binary format that is much faster and more compact than OSM XML, but is not human readable [19]. Therefore, we opt to use the OSM XML format for correctness checking and data analysis, and the PBF format for data parsing and editing when possible with tools.

OSM data, regardless of the file format, is comprised of a collection of objects:

Nodes, Ways, and Relations. In OSM, Nodes are the basis of all data. A Node has latitude and longitude coordinates and an ID, describing a single point on a map. Ways are a collection of Nodes, which can describe roads, buildings, highways, and importantly, bridges. Relations can contain Ways and Nodes and are used to define geological relations between OSM objects. All OSM objects can have tags with a key-value pair used to define metadata for the object such as street name, highway number, bridge information, etc. In OSM, bridges can be identified as a Way object with a bridge tag, such as `"bridge"="yes"`. Name tags are applicable to most OSM objects and are stored similarly, e.g. `"name"="Highway 72"`.

2.2 National Bridge Inventory Data

The National Bridge Inventory (NBI) is a database created and maintained by the Federal Highway Administration that contains a complete record of bridges and bridge repairs in the United States. It was made publicly available in 2021 and has since been used in various applications, typically pertaining to bridge health. Though the dataset is publicly available, only the Federal Highway Administration can update it, and most bridges are only updated once every two years or longer.

The NBI database is vast, containing data on all bridges in the United States over 20 feet long used for traffic [24]. It contains over 620,000 entries, each having over 100 fields. Each entry describes a single real-world bridge including year constructed, health ratings, materials, etc. Since bridges will be matched between the datasets by location and name, the most crucial fields are `latitude` and `longitude` for location and `carried-by` for street name. It is important to note that in NBI, only one point of the bridge is recorded, which could be at any point on or near the real-world bridge. NBI entries are updated by the bridge's owning agency and several standards put in place by the Federal Highway Administration must be followed when collecting bridge data. Unfortunately, these standards enforce little when collecting a bridge's location; It is fully up to the bridge's owning agency to best decide how the bridge's latitude and longitude should be measured [3, 17]. A standard from 1995 indicates a required amount of precision, but does not describe which point of the bridge to use for data collection [24]. This means there is no way to be certain *how* this data is collected and consequently, how to correct any inconsistencies in latitude and longitude. A bridge's positional data is collected at the time of the bridge's initial inspection, and is rarely updated afterward [3].

Though NBI data is frequently used for bridge health analysis, inaccuracies occur frequently in the data due to the lack of standards during the data collection process. This is especially apparent when inspecting NBI positional data against the OSM dataset. In some cases, the positional data of NBI was found to be invalid in 28% of entries [4]. Inaccuracies in NBI have even led to the collapsing of some bridges that may not have been properly inspected [6]. Our implementation must find a novel method of avoiding these inaccuracies. We do so by relying on the strength and accuracy of OSM to find errors in NBI and determine where they lay.

2.3 Methodology and Mentality

When working with two datasets of varying quality and format, it is important to understand the nuances of each, and where data discrepancies may halt progress. To account for limitations and keep the project in scope, the methodology and mentality of this project must be discussed first to set a reasonable baseline for expectations and describe the purpose of this project.

This project seeks to add highly-accurate bridge data to the OpenStreetMaps database using various open-source services to encourage data privacy, process transparency, and architectural maintainability. The process of doing so should not be considered "combining" or "merging" of two datasets. These words imply all data across each dataset is put together to create a new dataset. This also generally means that the two datasets are similar in format and content. In our case, not all data can be used, nor are the two datasets similar. The process here should be considered "data enrichment," though the words "merging" and "combining" are used interchangeably in this report. Essentially, this process improves existing data in the OpenStreetMaps dataset by adding more detailed and enriching information. The added data is sourced from a vastly different "outside" dataset, NBI. Adding *all* data from NBI into OSM is unlikely due to the greatly differing natures of the two data sources. In conclusion, this project focuses on adding highly-accurate bridge data from NBI into OSM, rather than completely combining these two datasets together.

This subtle distinction is important as it lays a foundation for goals. Though adding as much data as possible into OSM is important, the higher priority is the accuracy of these additions. Since only a subset of the outside data source (NBI) is being added, it must be guaranteed that this subset of data is introduced and included correctly. The focus is on maintaining the integrity of OSM data, while adding as much usable data from NBI as possible. By following this goal, the resulting dataset will have a selection of bridges that have highly detailed and accurate NBI data.

NBI data is added to corresponding OSM bridges in the form of tags. For example the tag `nbi=yes` can be applied to an OSM bridge to indicate matching NBI data was found and applied. Other tags can be added to describe NBI metadata such as `nbi:sub-cnd=5` to denote a score of 5 for a bridge's NBI substructure rating.

2.4 Related Work

Very few researchers have attempted to use these two datasets together. It is difficult to find relevant research on the subject and therefore, difficult to gauge the comparative success of the taken approaches. Most related research analyzes the NBI and OSM datasets individually to reveal their strengths and weaknesses. The lack of related literature proves that this is a novel idea, but provides little foundation to compare against.

Since 2004, OSM has seen great improvements in completeness and accuracy. It is less developed than larger commercial sources like Google Maps, but continues to improve thanks to its expanding user base [16]. It also sees larger commercial and research-based usage to avoid costs of commercial APIs [25].

NBI data is consistently found to contain invalid, incomplete, or missing data [4, 12]. Some researchers have suggested methods to remove invalid data from NBI or estimate missing bridge condition ratings [12]. This project focuses on the accuracy of bridge location rather than bridge health.

2.5 Visualizing Data

Performing a statistical analysis of the data can showcase where holes may lie in the data, or in each iteration of the system. Unfortunately, it is difficult to verify the validity of matched data without *seeing* where the data lies to understand what the real-world implications are. In fact, a visual analysis is the preferred method of explaining and understanding the differences and similarities between datasets. To address this, a custom software is created that visualizes each match made by the merging process. This helps identify when edge cases occur and how to handle them.

This visualization process uses the Folium library to retrieve satellite-view images to show geographical information around the data points. It is capable of plotting segmented shapes representing OSM ways, placing nodes representing NBI entries, and storing other relevant data such as OSM tags, NBI fields, and heuristic scores. With this data combined, a full suite for visually analyzing the effectiveness of the matching process is made available to assess and fix issues across iterations.

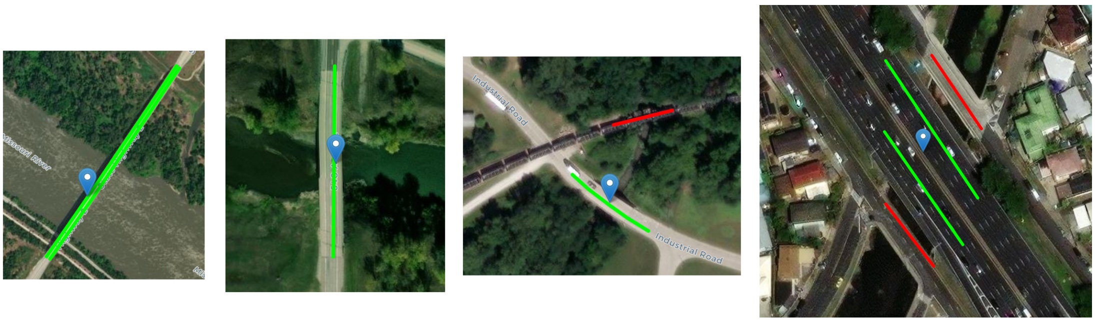

Figure 1: A select few examples created from the visualizer tool. These examples showcase how NBI and OSM datums are represented (points and lines resepc-tively), as well as which OSM bridges are chosen to receive data (green lines are matching OSM Objects, red lines are irrelevant bridges).

Figure 1 shows a few examples of visualizations created from this software. These visualizations can be found throughout this report and are an important part of understanding results from iterations of the process. Within these visualizations, three types of data are present: blue markers which represent NBI entries (or points), green lines which represent an OSM bridge that is declared as a match to the NBI point, and red lines which represent an OSM bridge that is found near the NBI point (this usually implies it is not declared a match). The importance of a visualization lies in whether OSM bridges are deemed a "match" or not for a given NBI point. This helps visualize and understand edge cases when attempting to match data.

All created maps center on the NBI data point, showing nearby bridges in OSM. The created visual analyses are saved as HTML files and contain nothing more than map data. They are not ideal for mathematically describing the accuracy of the merging process, but visualizations provide valuable human-readable information that is easy to parse. These files have much potential for performing crowd-sourced data affirmation for the individual and combined datasets.

3 Project Iterations

The project has been approached and implemented through multiple methods. All of the methods introduced use a Python-based algorithm to append desired NBI data to an OSM object. The underlying libraries, APIs, and techniques are generally identical across techniques, but using different methods of data collection and comparative analysis can grant drastically different results. These pieces are fine-tuned to provide clearer results in the merged data.

The reason for using various approaches is to explore the difference in data quality and the speed at which it can be created. Additionally, this provides insight to where inconsistencies lie and how to avoid them. Some merging methods maintain stricter matching criteria, while others are more tolerant to erroneous data. Since there is no work to compare against, approaches are assessed primarily by speed, accuracy, and transparency. When further iterations are created they must improve upon previous iterations in terms of accuracy and transparency.

In all approaches, the PyOsmium library is used. PyOsmium is highly efficient at reading and writing OSM data in nearly any format, making it the prime candidate for using PBF data [22]. Conversely, NBI data is comparatively easy to work with. It is stored in a basic database format, and can be downloaded as a CSV file. Reading and writing this data is relatively trivial.

3.1 First Iteration: Reverse-Geocoding

The first approach to merge the two datasets – and the baseline for all future iterations to improve upon – relies on a technique known as reverse geocoding. Using a reverse geocoding algorithm, a user can supply coordinates in latitude and longitude format and receive an address and details about the approximate location from a map database. For this implementation, the open-source API Nominatim was used. Nominatim provides a reverse geocoding endpoint and returns OSM data to the user in various formats [18]. This method of matching is called "Reverse-Geocoded Matching," or simply, "RG Matching."

Using Python, each entry of NBI (or a subset of NBI) is inspected. For each NBI entry, a Nominatim API call is made to use reverse geocoding and retrieve information about the OSM item that is closest to its coordinates. This, however, may not always return the correct OSM object; It may not even return a bridge. Since Nominatim does not return tags of the object, a relation between the NBI and OSM objects is saved to cross reference tags later. To parse and edit the OSM data, PyOsmium is used to iterate through each OSM way. If an OSM Object is found to have a relation to one of the NBI entries, the respective NBI data is added to that OSM Object in the form of tags. To confirm that the relation between NBI and OSM is correct, the data is merged if and only if the OSM Way has a bridge tag, and is a roadway (the Way isa bridge and intended for vehicles).

When using reverse-geocoding via Nominatim, 15,336 NBI entries and 10,384 valid OSM bridges are present for Nebraska. However, only 4,697 of the NBI entries were successfully reverse geocoded to a valid bridge in OSM. This means less than a third (30.6%) of the NBI entries are being matched, and less than half (40.6%) of the valid OSM bridges are receiving NBI data. To summarize, more than two-thirds of the NBI data (69.4%) could not be properly applied to OSM using reverse geocoding.

This approach provides a baseline for what all future work must accomplish. All bridges that are properly merged between NBI and OSM are trivially similar based on location and tags. However, this method has no contingency plans for incorrectly matched data (false positives) or data in which no match was found. This approach guarantees that matches are mostly correct, but is only able to

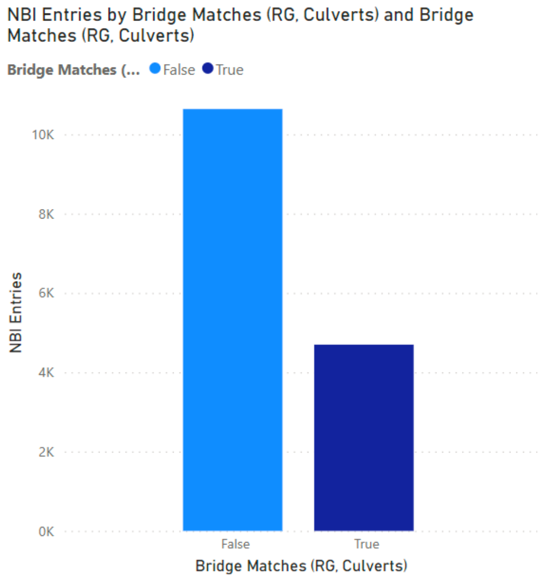

Figure 2: Using reverse-geocoding, less than a third of all NBI entries for Nebraska – 4,697 of 15,336, or 30.6% – could be properly matched to OSM data. While these matches are confident, the match rate is terribly low and must be improved. This proves that there is a necessity to iterate on the project to improve the matching algorithm.

match about less than a third of the NBI data. This can be attributed to a few factors:

1. Reverse Geocoding only retrieves one OSM object. This is problematic for bridges with split roads, as in Figure 3.

2. There is no possible method of correcting incorrect matches (false positives) since only one result is returned. With no other nearby bridges provided, the first and only option returned by reverse geocoding must be used.

3. Nominatim provides very few tags from the requested OSM object, excluding all bridge tags. There is little transparency about why the OSM object was picked and what other tags it contains. Extra analysis and cross-referencing is required using PyOsmium.

4. Most NBI entries labeled as "culverts" are not matched to relevant OSM data, heavily lowering the match rate. This issue is addressed in Section 3.2.

Because of these issues, this approach was abandoned in favor of one that could more freely search and match data. However, the reverse-geocoding approach is still relevant, as it serves as a baseline for future iterations, and can be used to analyze how much of the data between NBI and OSM can be directly correlated, an important statistic for understanding the shortcomings of each dataset. Though this approach is not the final implementation for data addition, it is still an important step in the project.

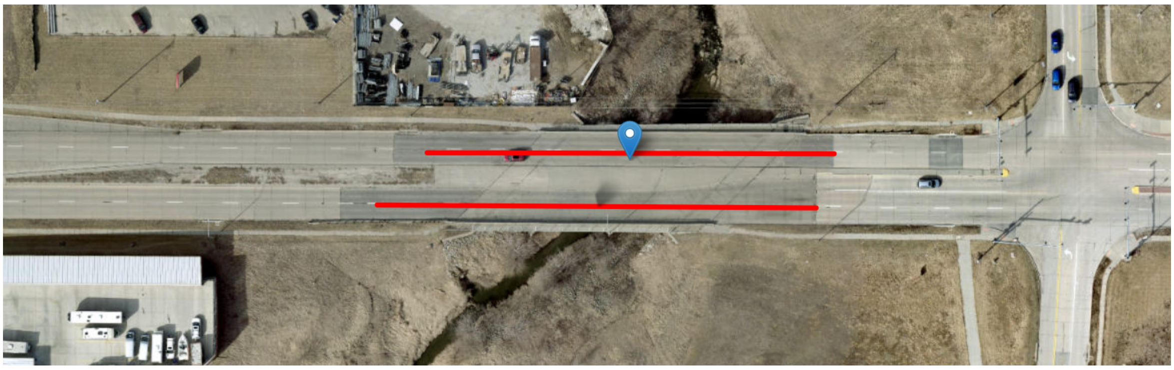

Figure 3: A visualization of a two lane bridge. A single blue marker exists near the center of the bridge, representing an NBI entry. However, OSM represents this single bridge with two different way objects (represented by two red lines). The merging process must consider the possibility of one NBI entry correlating to multiple OSM objects, as it would here.

To determine the precision of the introduced data, the accuracy of the reverse-geocoder must be questioned. Some researchers have questioned the use and accuracy rates of reverse geocoding algorithms. Fundamentally, a geocoding algorithm can only be as strong as its provided map [20]. Many have found Nominatim to provide lesser quality results, but still serviceable, when compared to competitors like Google maps [5]. The work of Kounadi et al. [14] defines accuracy success rates of various reverse-geocoders, looking for near exact similarity between a retrieved address and the original address. Their research only looked at Nominatim for OSM-based reverse geocoding, but it proved to be better than most other solutions. Several other APIs exist that can reverse-geocode OSM data, namely Pelias and Gisgraphy. These solutions both use OSM data and are open-source [10, 21]. In particular, Pelias has the ability to return multiple addresses for one coordinate location, solving two of the three problems. The third problem persists, however, as neither Pelias nor Gisgraphy return tags of the OSM object.

The fourth issue is unrelated to the process of reverse-geocoding. Within NBI, bridges can be labelled as a culvert. In the first iteration – and all future iterations – the match rates of culvert entries are shockingly low. These data points must be analyzed closer and assessed to improve the performance of the matching algorithm.

3.2 Data Cleaning

NBI contains hundreds of thousands of bridges, all varying in shape, structure, and size. These details are stored in various fields for each bridge entry. Due to the unreliability of NBI data [4, 6], the data must be cleansed to avoid erroneous merges by inspecting the accuracy and consistency of bridge locations (in latitude and longitude) and bridge types (such as culverts, suspended bridges, tunnels, etc.).

To organize the massive amount of data in NBI, bridges can be divided into categories. One such category is culverts. Culverts are defined as "a drainage structure beneath an embankment (e.g., corrugated metal pipe, concrete box culvert)" by the National Highway Institute [3]. In real world construction, culverts are very different from typical bridges, often not needing support structures or frequent maintenance. In the topic of bridge-aware routing, culverts are unlikely to be problematic due to the nature of their construction.

Numerous issues with culvert entries arise when inspecting the datasets. In OSM, culverts are often defined along the tunneling waterway, rather than along the road it passes through. Because of this, the road or embankment is often missing any sort of identifying bridge tag. Furthermore, underlying waterways are often inconsistently or incorrectly tagged in OSM. This makes it difficult to determine which OSM way, if any, should be given NBI tags. There are also issues with culverts positionally when comparing NBI and OSM. When looking at the position of culverts between datasets, NBI culverts are frequently far away from the corresponding culvert in OSM, as seen in Figure 4.

An analysis of the merged data using reverse-geocoding in Figure 5 confirms these suspicions. In NBI, a culvert entry has a culvert rating from 0-9 while a non-

Figure 4: A visualization of OSM and NBI data overlayed onto Google Maps. A bright blue dot exists far north of Vane St, which represents a culvert entry from NBI. This NBI entry is about 66 meters from the corresponding OSM culvert, which is the creek running underneath Vane Street (the red highlighted line intersecting Vane Street). NBI culverts are frequently far from the corresponding creek in OSM, leading to mismatched data.

culvert entry has a culvert rating of "N". The analysis shows that most matches come from non-culvert entries (a.k.a. bridges). It also shows that culverts have incredibly low match rates. This proves that culverts are very difficult to match properly and confirms the issues with their inclusion in the data.

Considering all of the inconsistencies found in culverts across both datasets, the decision to exclude culverts from the merging process is important. The difficulty of correctly matching culverts between the datasets is far greater than the value received since there is little concern for damaging culverts in vehicle routing. Because of this, culverts are excluded entirely from all data and the merging process in future iterations.

For NBI's Nebraska data, there are 4,214 culverts (27.47% of the 15,336 entries). Removing culverts from the data leaves 11,122 valid NBI entries. These

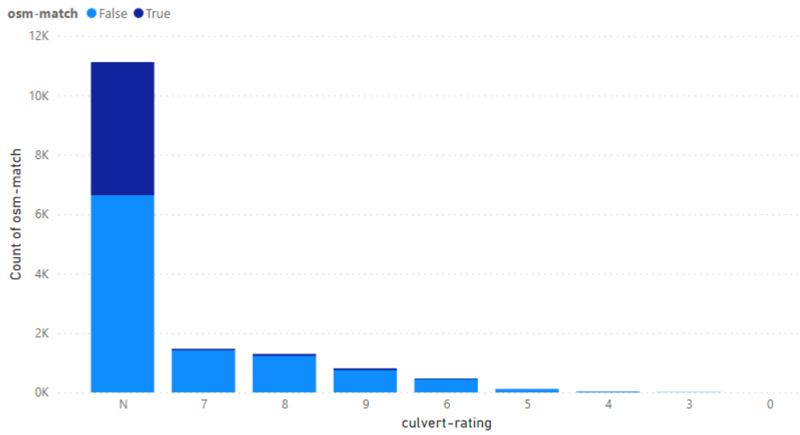

Figure 5: A graph depicting the frequency of NBI entries receiving matching OSM data based on whether the entry is a culvert or not. Culverts have a culvert rating from 0-9, while bridges have a culvert rating of "N". The data shows that bridges have a much higher match rate than culverts do. This confirms that assumption that NBI culverts are more difficult to match than bridges.

remaining entries will be used in the matching process exclusively in the following iterations to avoid bad data input from culverts. For Nebraska, OSM has 10,384 non-footway bridges. This leaves a discrepancy of 738 bridges between the cleaned NBI data and complete OSM data, a difference of less than 10%. Ideally, the OSM data would account for 93% of NBI's Nebraska data. This project does not account for the possibility of OSM bridges being cycleways or NBI containing non-roadway bridges.

3.3 Second Iteration: Data Querying

The second approach is very similar to the first in concept. Python is used in a similar manner to iterate through the NBI entries and call an API to collect relevant OSM data. Where this method differs greatly is in the chosen API. In this method, Overpass is utilized. Overpass is an open-source API which uses a query language that can gather OSM data in an area. Various restraints can be imposed to gather specific data, which is used to search for bridges. This method of data merging is referred to as "Query Matching," or simply, "Querying."

This method accounts for every issue that reverse-geocoding has. Since Overpass can query an area and potentially find multiple bridges, the algorithm can account for scenarios where one NBI entry correlates to multiple OSM bridges. If the closest OSM bridge to the NBI entry is incorrect, other bridges in the queried area can be considered for matches. When querying using Overpass raw data is returned, which includes tags for every object, providing full transparency of the results and removing the need for cross-referencing data later.

In practice, this method matches more bridges than reverse-geocoding and produces more accurate results due the tolerance of querying multiple bridges. This approach does not help define the accuracy of either dataset, but serves as a strong base for an error-tolerant merging algorithm.

Overpass serves as a highly-flexible method of querying for OSM data around a point from NBI. The Overpass API allows queries to filter other important factors of the collected data such as tags, area codes, and street names. With an area of queried data around the point in question, further deductions can be made to pick which bridges the NBI data should be applied to. It is extremely flexible and allows all decision making to be performed independently without relying on the correctness of the dataset and performance of a hidden algorithm, unlike RG Matching.

Using Overpass fixes several issues present with reverse-geocoding, but introduces a new issue: false positives. When matching data, a false positive is defined as a declared match between an OSM Object and an NBI entry that do not represent the same bridge in the real world. These false positives are present in a large amount of the matched data, specifically in scenarios where multiple OSM bridges are present around an NBI point. Examples of this can be seen in Figure 6.

False positives are difficult to identify by data alone, hence the need for visualizations. This method of matching is an improvement, being more error tolerant and increasing the match rate, but the false positives muddle the precision of the data. To create a stronger and richer dataset, further improvements must be made. A decision-making process must exist that can identify which bridges, if any, near an NBI entry should be considered a match. Enter: Heuristics.

Figure 6: Using query matching introduces false positives to the data. Four examples of this are shown here. In each visualization, the NBI point lies close to one OSM bridge, yet other OSM bridges exist nearby. The other bridges are declared as a match even though they are far away and represent a completely different bridge.

3.4 Final Iteration: Query-Heuristic Matching

Overpass on its own will not suffice as a method of determining the correct OSM object to add NBI data to. When querying, it is likely that multiple bridges – or no bridges at all – will be found. How should the correct bridge be determined when multiple bridges are found near a singular NBI datum? How can the algorithm guarantee false positives are avoided during the matching process? The answers lie in the decision-making process. Scoring and heuristics make up the basis of this solution, creating a decision-making process that mimics human analysis. By applying good scoring, a consistent model is created for describing the likelihood of the OSM object being a good fit to receive data. With heuristics, scoring is used to pick the OSM objects that are a confident match. A strong combination of these two aspects are necessary to ensure the added data is accurate and reliable, removing as many false positives as possible. This method of matching is deemed "Query-Heuristic Matching," or simply, "QH Matching."

When shifting to QH Matching, the improvements are clear! Over half of NBI's entries were added to OSM (6,329 of 11,122, or 56.9%, of entries from the NBI

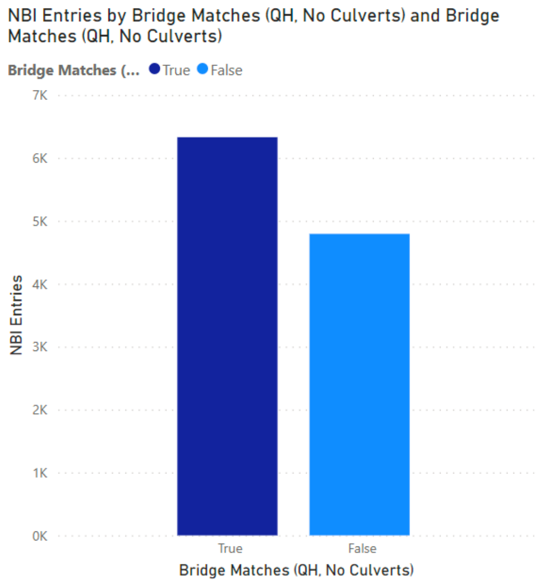

Figure 7: Query-Heuristic matching successfully matches over half of the NBI data (56.9%) while avoiding false positive matches in OSM data. This is a drastic improvement over previous iterations and has the most potential for creating a fully-enriched OSM dataset.

data were properly matched), with the majority of unmatched data being caused by missing relevant data in OSM. 4,793 (43.1%) of NBI entries had no nearby bridges in OSM! This missing data largely exists in rural areas of Nebraska which are notoriously incomplete in OSM. Using heuristics lowers the potential for total matches found when using query-matching, but it ensures confidence in the found matches.

To better understand the heuristics portion of the QH Matching algorithm, all considered scoring algorithms and heuristic intuitions are described. Two methods of scoring are used when bridges are found near an NBI datum: distance scoring and name scoring (also referred to as string scoring). The distance score is a value applied to an OSM object based on its distance to the relevant NBI datum. The name score is a value applied to an OSM object based on the similarity between its name and the NBI datum's name. Heuristics are much harder to concretely define, but are created and adjusted to make decisions as close to human analysis as possible.

3.4.1 Distance Scoring - Centroids

When calculating distances on a map, geodesic distances must be used. A geodesic distance is the shortest path between two points on an ellipsoid, usually the earth [13]. For this project, the common *WGS-84* world model is used, which is accurate and follows OSM and NBI's data standards. From here on, "distance" and "geodesic distance" will be used interchangeably. All distances will be considered in meters. Most bridges in OSM are defined by two points, though many bridges may curve and thus need multiple points (or segmented lines) to describe their unique shape, as shown in Figure 8. To accommodate, distance calculation algorithms must consider an entire shape rather than just a line segment.

Figure 8: Bridges in the real world can have slight curvature, as seen here. To accommodate for this, OSM allows Ways (in this case, bridges) to be described by multiple nodes. This means that OSM bridges can come in any shape and size!

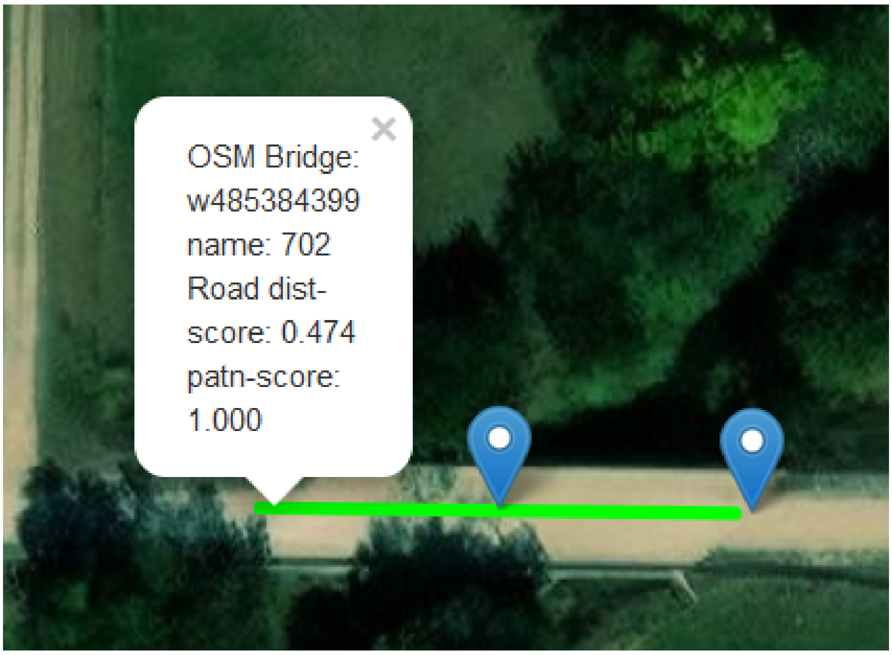

Figure 9: A visualisation of the selection process when using centroids. The green line represents a bridge in OSM. The leftmost blue marker is its centroid and the rightmost blue marker is the corresponding NBI entry. For this case, the distance score of the NBI point and OSM Object is 0.474.

The first approach explored for distance scoring is centroid distance. A centroid is the center of a set of points found by taking the average of all the points. This applies to any shape with multiple vertices, including lines and polygons. The centroid for a shape, P, constructed from a set of points can be defined as follows:

$$\sum_{n=1}^{t} \frac{P_n}{t}$$

Where P is an ordered set of 2-D vectors (points defined as $[0, 1]$, $[2, 3]$, etc.), P_n is the n^{th} item of P, and t is the cardinality of P. The resulting vector describes the center, or *centroid*, of the shape.

In OSM, Ways are defined using nodes and can be mathematically described as a segmented line defined by multiple coordinates. This structure works well for calculating centroids. With a centroid for the Way defined, finding the geodesic distance between the centroid and the NBI point is trivial using a Python library.

This method of distance calculation is easy to implement, but is lacking in descriptiveness. As seen in Figure 9, though an OSM object may seemingly lie close to the NBI datum, the distance score is below the expected threshold score of `0.5` since its centroid is far from the NBI datum. To fix this issue, a far more expressive method of distance scoring is used.

3.4.2 Distance Scoring - Shortest Distance

With the shortest distance method of distance calculation, the shortest distance from the NBI datum to the OSM object is calculated. This does not rely on using a singular point from the shape – like a centroid – rather, it considers the entirety of the shape and is much closer to what humans would expect when measuring distance from a point to a shape. The shortest distance from a point to a shape must be calculated by considering each segment of the shape as a line, and finding the shortest distance to each of those lines. The smallest distance among them is considered the shortest distance. The shortest distance between a point, X, and a segmented line defined by two points P_1 and P_2 on a world map is defined by the following function:

$$f(X, P_1, P_2) = \begin{cases} geo\_dist(X, P_1) & \text{if } r < 0 \\ geo\_dist(X, P_2) & \text{if } r > 1 \\ \sqrt{|geo\_dist(X, P_1)^2| - |(r * geo\_dist(P_2, P_1)^2|} & \text{otherwise} \end{cases}$$

$$r = \frac{(P_2 - P_1) \cdot (X - P_1)}{|P_2 - P_1|^2}$$

Where P_1, P_2, and X are 2-D vectors representing coordinates and geo_dist is the function that defines the geodesic distance between two points (in meters). The vectors P_1 and P_2 represent the start and end point of the line segment, and X is the point to calculate the distance from. To find the shortest distance from a point to a shape defined by an ordered set of points, P, the following algorithm is used:

$$shortest\_dist = min\{f(X, P_i, P_{i+1}) : i = 1, ..., n - 1\}$$

Where P_i is the i^{th} point (2-D vector) of shape P, X is the point to calculate the shortest distance to, and n is the number of points in shape P. This function has some accuracy loss when calculating large distances due to not using geodesic distances when calculating r. However, all queried bridges are within 50 meters of the NBI point, meaning the error loss is negligible. An exact distance is not necessary and this function works well for general distance scoring.

When calculating distances, no matter the method, a shorter distance must to correspond to a greater score and a longer distance must correspond to a lesser score. To accomplish this, the calculated distance is applied to a simple function that generates a score:

$$dist\_score = \frac{t}{d+t}$$

Where t is a threshold value that describes the desired maximum distance in meters, and d is the actual calculated distance. This function returns a score in the range (0..1]. Any score above 0.5 means the distance is less than the threshold value, while any score below 0.5 means the distance is greater than the threshold value. When defined in code, t is a "magic number" and $t = 20$ was found to be suitable, prioritizing bridges found within 20 meters of the NBI entry.

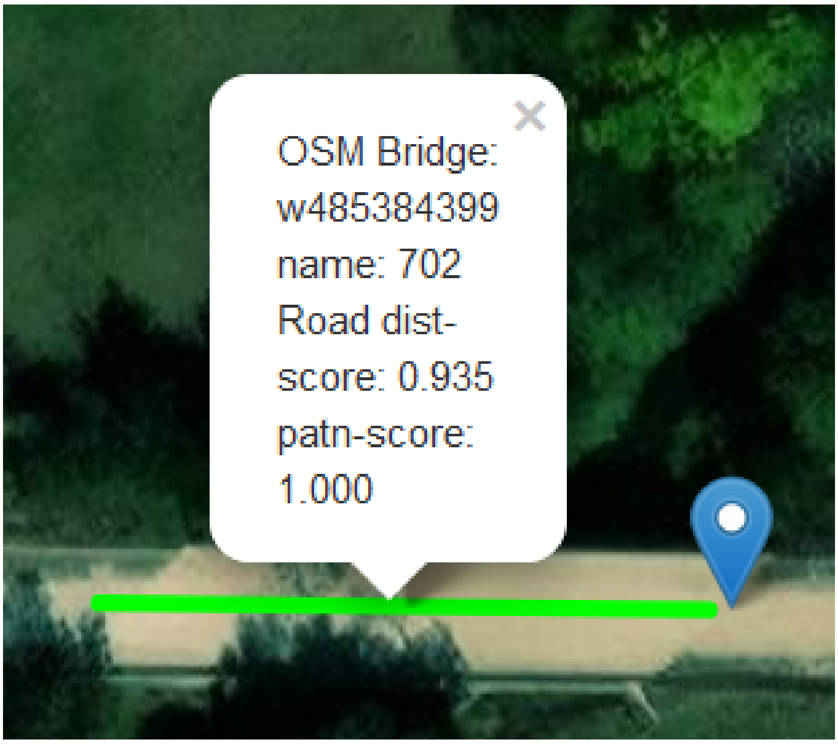

Figure 10: A visualization of shortest-distance scoring. The green line represents an OSM bridge, and the blue marker represents the NBI datum. Using this method, the shortest distance is found between a point and a shape defined by a set of points. This method score the same scenario as Figure 9 at a much more reasonable 0.935.

The results provide a method of distance scoring close to human analysis. As seen in Figures 9 and 10, the NBI datum lies next to the edge of the bridge in OSM. With centroid scoring (Figure 9), this scenario was scored at a low `0.474`, while shortest-distance scoring (Figure 10), scored the OSM object's distance highly at `0.935`, indicating it is extremely close to the NBI point. This is much closer to what humans would expect when analyzing distance.

3.4.3 Name Scoring - Gestalt Matching

To score the names of OSM objects against NBI entries, similarity scoring algorithms are used. Character case is always ignored when calculating string similarity for this project. Gestalt Pattern matching is an algorithm for calculating string similarity which creates a score between two strings of varying length by imitating the way humans analyze patterns in strings [23]. Given two strings, s_1 and s_2, their Gestalt score is defined by:

$$gestalt\_score = \frac{2C_m}{|s_1| + |s_2|}$$

where C_m is the number of matching characters between the strings and $|s_1|$ and $|s_2|$ are the lengths of strings s_1 and s_2 respectively. The resulting similarity metric is a value between 0 and 1, with zero indicating no similarities and 1 indicating identical strings. [2].

This algorithm works well for general string comparison, relying on a percentage of common letters between the two strings. The Python implementation of Gestalt pattern matching was used for this project, known as *SequenceMatcher*. The *SequenceMatcher* class is slightly more advanced than typical gestalt pattern matching, relying on recursively finding the longest common substrings, and ignoring "trash" characters. It more closely follows methods of sequence-based scoring and is *not* commutative. To make the *SequenceMatcher* algorithm commutative, a score is calculated for each possible order and the average is taken:

$$avg\_match = \frac{SequenceMatcher(s_a, s_b) + SequenceMatcher(s_b, s_a)}{2}$$

Where $SequenceMatcher(a, b)$ is the score calculated by the *SequenceMatcher* class using strings s_a and s_b. This change makes the SequenceMatcher class commutative. However, further methods of pattern matching exist and must be considered for this project.

3.4.4 Name Scoring - Sørensen-Dice Coefficient

The Sørensen-Dice Coefficient is an algorithm used for calculating a similarity matrix between two strings. The original formula was published independently by both Thorvald Sørensen and Lee Raymond Dice [7, 26]. The Sørensen-Dice coefficient for two strings, s_A and s_b, can be defined as such:

$$sd\_score = \frac{2|A \cap B|}{|A| + |B|}$$

Where A and B are the sets of characters found in strings s_a and s_b respectively, and $|A|$ and $|B|$ are the cardinalites of sets A and B respectively. This formula is quite similar to gestalt matching with one key difference: it uses sets, rather than strings. This eliminates duplicate characters, and thus provides scores equal to or higher than ones from the Gestalt algorithm. Higher scores are not a necessity, though the Sørensen-Dice coefficient generates scores closer to what humans might expect when analyzing two strings.

Comparison of String Scoring Algorithms			
NBI Name	OSM Name	Gestalt Score	Sørensen-Dice Score
ROAD 702	702 Road	0.5	1.0
654 AV-707 / 708 R	654 Avenue	0.429	0.571
US136	Highway 136	0.375	0.4
US75	639 Boulevard	0.118	0.118
FAS 3695	South 72nd Road	0.087	0.3

Table 1: Comparison of string scoring algorithms

Table 1 shows a comparison of the two algorithms on a select few cases. The first case is the most important, showing two strings that describe the same name but in a reversed order. The Sørensen-Dice coefficient scores these two strings at

`1.0` (a perfect match!) whereas Gestalt Matching scores them at `0.5`. The second case has two drastically different strings referring to the same street (”654 AV” should be intuitively considered the same as ”654 Avenue”), which Sørensen-Dice scores above `0.5`. When inspecting these cases using the Sørensen-Dice Coefficient, any score over `0.5` is generally a close match. The third case shows some issues with using these relatively simple algorithms, as neither score it above `0.5`, even though ”US136” is conceptually the same as ”Highway 136”. There also exist many cases where the OSM name may be empty, or the two differ vastly in terms of content, despite the data referring to the same road. Consequently, string scoring is a good supplemental score to confirm that data points are equivalent, but should not be relied on as the sole method of determining match confidence.

3.4.5 General Scoring

Once both scores have been calculated, a general score can be generated. This is done by averaging the two scores:

$$general\_score = \frac{dist\_score + name\_score}{2}$$

This general score works well for comparing the likelihood of matching between OSM objects. However, it removes the expressiveness of using separate scores and is only used to compare match confidence between multiple OSM bridges. The OSM bridge with the highest general score for an NBI entry is most likely to be a match for the data.

3.4.6 Heuristics

After deciding what scoring mechanisms to use, heuristics must be defined that determine when scores quantify a match. These heuristics read in scores and pick matches in a way that is designed to mimic human reasoning. When deciding what data pieces match, there are four possible cases for our data, all of which assume there is a queried section of OSM data around an NBI point. Each case requires different reasoning to pick appropriate matches. The methodologies for each case are discussed to explain the reasoning, but low-level descriptions of numbers and thresholds are omitted.

Case 1 Zero bridges found in OSM:

This is the easiest case to handle. When there are zero bridges found in OSM around the NBI point, there is nothing to add data to. This case is ignored and the process continues. It is possible to add a new bridge to OSM along with its necessary NBI data, but this is far outside the scope of this project.

Case 2 One bridge found in OSM:

This case is also quite easy to handle. If one bridge is found in OSM near the NBI point, it is highly likely that bridge is a precise match. To be confident in this decision, distance scoring should still be considered. If the distance is past our threshold, the bridge is considered too far away to be possible match and is ignored. In this case, a distance score above 0.5 is desirable.

Case 3 Two bridges found in OSM:

This case introduces most of the edge cases when attempting to match data. Most likely is the case that both OSM objects are two different bridges, either carrying separate streets or opposing directions of traffic. Distance scoring is used as the primary distinguisher between matches here, with string scoring used as a supplemental method of improving confidence. If both bridges have the same name and are close to the point, it is typically safe to assume they both refer to the same bridge. This case must also consider the possibility that one or neither bridge has high enough scores to be considered a match.

Case 4 Three or more bridges found in OSM:

This case is one of the most difficult to handle. It generally follows all of the assumptions in case 3, but must also assess the likelihood of bridges being segmented in OSM. In rare cases, one OSM Bridge can be comprised of two or more contiguous Ways. In some cases, like around airports, the OSM query may contain over 25 bridges, so the algorithm must be careful to select only one or two bridges that closely match the NBI data.

4 Results

Using RG Matching, Query Matching, and QH Matching, several versions of the merged dataset have been created to compare. It is important to note that the merged datasets only contain data for Nebraska to reduce the time, size, and scope of the tests. A smaller combined dataset provides results that are easier to parse, but provides a shallow understanding of the algorithm's performance.

4.1 Analysis of Iterations

The details of each iteration have been described in Section 3 in isolated views. To show the growth brought by each iteration, a more in-depth comparison is required.

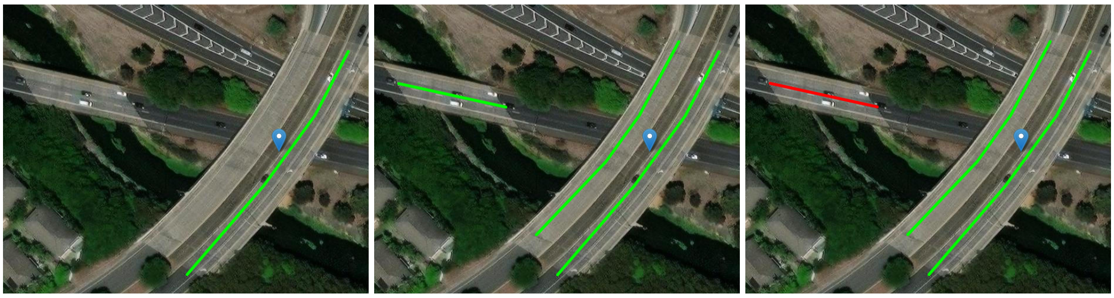

Figure 11: A sequence of visualizations that describe system's choices when deciding matching bridges. From left to right, these methods are: RG Matching (Nominatim), Query Matching (Overpass), and QH Matching (Overpass and heuristics). Notice that reverse-geocoding only finds one bridge, and just querying will select all bridges in the area. By applying heuristics, only relevant nearby bridges are chosen.

Figure 11 provides visual examples to show improvements of the data merging process. The first iteration used reverse-geocoding to retrieve an OSM object from a coordinate of NBI data. This worked well as a baseline and gave confident answers, but had several issues. Most importantly, it could only retrieve one bridge at a time and had no tolerance for incorrect data. The next iteration used Overpass to query for bridge data around a point. This was able to observe multiple bridges around a singular NBI point, but would often apply NBI data to unrelated OSM bridges. The final iteration uses Overpass and heuristics. This functions similarly to the previous iteration, but applies data scoring and heuristics to create a selection process that mimics human intelligence.

Visual representations of data are sufficient when describing match confidence. However, confidence is harder to define when looking at statistical analyses. Conversely statistics can show how the algorithms improve the match rate of data.

Iteration Match Rates				
	NBI		OSM	
Iteration	Entries	Added	Entries	Received
RG Matching	11,122	4,483	10,384	4,483
Query Matching	11,122	6,329	10,384	7,905
QH Matching	11,122	6,329	10,384	6,509

Table 2: A comparison of match rates for all three iterations of the project. There are 11,122 valid (non-culvert) NBI entries and 10,384 valid OSM bridges to match. The NBI "Added" column describes how many NBI entries were applied to the OSM dataset, and the OSM "Received" column describes how many OSM bridges received NBI data.

As seen in Table 2 the match rates improve across iterations. The working data contains 11,122 non-culvert NBI entries and 10,384 valid OSM Bridges (vehicle-traversable bridges). When using RG Matching, an NBI entry can be matched to, at most one, OSM bridge. Hence, the equal "Added" and "Received" numbers between NBI and OSM. The match rate for RG matching is poor, only matching 40.31% of NBI entries and enhancing 43.17% of OSM Bridges. This is primarily due to reverse-geocoding only returning the single closest object to an NBI point.

When shifting to Query Matching, the match rate improves greatly. 6,329 (56.91%) of NBI's entries were matched and added to the OSM data, and 7,905 (76.13%) of OSM's valid bridges received NBI data! However, this method of matching data introduces false positives, and some OSM entries have been erroneously given incorrect NBI information. To fix this issue, a new method of querying is introduced: QH Matching.

QH Matching has the best results overall, though this is difficult to show with statistics alone. Its NBI "Added" rate is the same as Query Matching. However, its "Received" rate for OSM is lower, giving NBI data to only 6,509 (62.68%) of the OSM Bridges. This lower rate is due to the heuristics. It isn't lowering the overall rate of NBI data added – that remains the same. Rather, it is removing false positives through the use of heuristics and only adding NBI data to relevant OSM bridges. This can be proved through visual analysis. By employing heuristics, nearly 1,400 false positives are removed from the OSM Bridges. The results from QH Matching are confident and match as much data between the two sets as possible while avoiding the "false positive" issue.

With both visual and statistical data present, Query-Heuristic matching proves to be the best option, adding the highest amount of data to OSM while having the best accuracy. These results can be improved greatly by increasing the completeness and correctness of both the NBI and OSM datasets, and refining the heuristics. This is discussed further in Section 6.

4.2 Reliability of NBI Data

To better analyze and understand discrepancies in the original datasets, an in-depth analysis was performed. Certain intuitions can be made about correlations between NBI precision and match rates.

There are few standards when collecting location information for a bridge in NBI. The Federal Highway Administration lets the owning agency decide the best practice when collecting the location information of a bridge. Consequently, one could assume some agencies employ better practices than others and determine which ones produce problematic data. However, there is little correlation between the NBI entry's owner agency and the likelihood of it being matched with OSM data. An analysis reveals that over 10,000 of Nebraska's bridges are owned by a highway county agency according to NBI. Most other agencies have too small a sample size to confidently draw any conclusions. A similar assumption could be made based on the year of the bridge's construction. Since location information for a bridge is rarely – if ever – updated, one could predict a correlation between a bridge's age and the likelihood of it being matched. Essentially, the older a bridge the less likely it is to be matched due to older technology and a lack of updates. However, the analysis shows that there is little correlation between a bridge's age and its likelihood of being matched.

Unfortunately, the Nebraska NBI dataset is too small to draw any conclusions related to the locational precision of bridges. With a larger dataset, these intuitions can be confirmed.

5 Generalized Theory

This project focuses on adding rich bridge data into OpenStreetMaps using several custom techniques and programs. Using Overpass, data is queried and applied to specialized scoring and heuristic algorithms that find matches between the two sets to add relevant information. This process can be generalized to fit other scenarios in which "outside" geographical data must be added to a complete map dataset of a vastly different format and standard. The heuristics and scoring algorithms must be individually specialized to meet the needs of each scenario. The generalization of this project has two main steps: Preparation and the System Process. Proper execution of these two steps allow for a system that can enhance a map with "outside" data. These steps can be described as so:

Prepare Steps that must be completed before implementing the system.

- **Sanitize** Sanitize and clean the "outside" dataset.
- **Scoring** Research and implement appropriate scoring techniques.
- **Heuristics** Create a basic heuristic algorithm to determine matches. This must be iterated upon to improve match rates.

Process Steps that the system must accomplish.

- **Query** Query relevant data from the map dataset for each entry of the "outside" data

Match Use specialized scoring and heuristics algorithms to determine which entries should be considered matches.

Apply Apply "outside" data to the map dataset based on the found matches.

Analyze Create a method of analyzing the match results and why those matches were picked.

A few example scenarios that can use this generalized process include: Mass updating road speed limits in OSM from a national list of speed limits, adding contact information to houses and addresses for personal use and convenience, and adding review scores to local restaurants and shops. Note that this process can only enrich datasets, not add new data points. This generalized process is perhaps the most important contribution of this project, providing a method of enhancing open-source map data with various outside data sources that have geographical components.

6 Limitations

While developing this process, several limitations arose. Most of which stem from the data sources used. OSM data was used as the primary world map, which is an ever-growing database of crowd-sourced geographical data. It is generally quite complete, but is lacking in rural areas across the US. Further complicating this issue, most of the used data is from Nebraska, which is a largely rural state muddled with missing, incomplete, and incorrect data. 72% of its communities have populations below 800 people, and 73% of counties lost population in the years 2010-2020 [8]. NBI also happens to be limited in terms of correctness with bridges often being located far from their true location. The limitations of both datasets however, was necessary to work around to keep the entirety of the project open-sourced. Other methods of finding matches could be explored. One paper [4] uses the National Highway Planning Network to explore bridge location accuracy.

This project was designed with the intent of adding NBI data to OSM. The process can be generalized, but still focuses solely on enriching data. New data points could be created in OSM to correlate to NBI datums, but this introduces too many issues that are outside the scope of this project.

Due to time constraints, the heuristics had limited testing but still provided sufficient results. With more time and research this process could be expanded further by adding additional scoring methods refining the specifications of the heuristics. Improving quality of the data sources will also improve effectiveness of the process overall.

A large improvement to both match rate and accuracy could be made using crowd-sourced data affirmation and machine learning. With crowd-sourcing, more definitive matches can be found and the results can be used to train a machine-learning algorithm, which can be used in place of data scoring and heuristics.

7 Conclusion

Bridge-aware routing is a fairly new technology that must be highly considerate of dynamic changes in bridge health and passing vehicles. Technologies exist with proper routing capabilities, but a complete dataset to facilitate this process does not currently exist. The dataset introduced in this project serves as a strong backend for any bridge-aware routing services, or problems focused on using rich bridge data in a real-world setting. This dataset is created by combining two publicly available datasets, OpenStreetMaps and the National Bridge Inventory, using open-source APIs and a few custom Python scripts.

The merging of this data is not as trivial as matching common fields between the two datasets. Due to numerous inconsistencies from each dataset, a location-tolerant matching method must be created which can merge outside bridge data into map data. Using the Overpass API, OSM data can be queried around an NBI data point to allow for tolerance when matching data. A set of scores and heuristics are applied when matching the data to mimic human analysis and create precise matches.

This process can be generalized to develop a method of enhancing map datasets using vastly differing "outside" data. The key feature is employing heuristics to mimic human decision making to find matches between the two datasets. This has several uses besides adding enhanced bridge data to OSM, and is the most important contribution of this project.

References

[1] Khalid El-Ashmawy. "Testing the positional accuracy of OpenStreetMap data for mapping applications". In: *Geodesy and cartography* 42 (Apr. 2016), pp. 25–30. DOI: 10.3846/20296991.2015.1160493.

[2] Edward Blurock. *String Similarity Metrics: Sequence Based — Baeldung on Computer Science*. en-US. Mar. 2021. URL: https://www.baeldung.com/cs/string-similarity-sequence-based (visited on 11/22/2023).

[3] *Bridge Inspector's Reference Manual (BIRM)*. en. Mar. 2023. URL: https://www.fhwa.dot.gov/bridge/nbis/pubs/nhi23024.pdf.

[4] Steven B. Chase, Chris Nutakor, and Edgar P. Small. "AN IN-DEPTH ANALYSIS OF THE NATIONAL BRIDGE INVENTORY DATABASE UTILIZING DATA MINING, GIS AND ADVANCED STATISTICAL METHODS". In: 2000. URL: https://api.semanticscholar.org/CorpusID:54968322.

[5] Błażej Ciepłuch et al. "Comparison of the accuracy of OpenStreetMap for Ireland with Google Maps and Bing Maps". en. In: *Proceedings of the Ninth International Symposium on Spatial Accuracy Assessment in Natural Resuorces and Enviromental Sciences 20-23rd July 2010* (July 2010). Publisher: University of Leicester, p. 337. URL: https://eprints.nuim.ie/2476/ (visited on 08/09/2023).

[6] Bill Dedman. *Bridge collapse revealed holes in federal data*. en. Aug. 2008. URL: https://www.nbcnews.com/id/wbna25956395 (visited on 08/04/2023).

[7] Lee R. Dice. "Measures of the Amount of Ecologic Association Between Species". In: *Ecology* 26.3 (1945), pp. 297–302. DOI: https://doi.org/10.2307/1932409. eprint: https://esajournals.onlinelibrary.wiley.com/doi/pdf/10.2307/1932409. URL: https://esajournals.onlinelibrary.wiley.com/doi/abs/10.2307/1932409.

[8] David Drozd and Jerry Deichert. *Nebraska County Population Projections: 2010 to 2050*. en. Tech. rep. 258. University of Nebraska at Omaha, Dec.

2015. URL: https://digitalcommons.unomaha.edu/cparpublications/258/ (visited on 11/24/2023).

[9] Giles Foody et al. *Mapping and the Citizen Sensor*. Ubiquity Press, 2017. ISBN: 9781911529163. URL: http://www.jstor.org/stable/j.ctv3t5qzc (visited on 08/31/2023).

[10] *Gisgraphy*. URL: https://www.gisgraphy.com/index.php (visited on 08/29/2023).

[11] Mordechai Haklay. "How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets". en. In: *Environment and Planning B: Planning and Design* 37.4 (Aug. 2010), pp. 682–703. ISSN: 0265-8135, 1472-3417. DOI: 10.1068/b35097. URL: http://journals.sagepub.com/doi/10.1068/b35097 (visited on 08/24/2023).

[12] Akshay Kale, Brian Ricks, and Robin Gandhi. *New Measure to Understand and Compare Bridge Conditions Based on Inspections Time-Series Data*. Vol. 27. Journal Abbreviation: Journal of Infrastructure Systems Publication Title: Journal of Infrastructure Systems. Aug. 2021. DOI: 10.1061/(ASCE)IS.1943-555X.0000633.

[13] Charles F. F. Karney. "Algorithms for geodesics". In: *Journal of Geodesy* 87.1 (Jan. 2013), pp. 43–55. ISSN: 1432-1394. DOI: 10.1007/s00190-012-0578-z. URL: https://doi.org/10.1007/s00190-012-0578-z.

[14] Ourania Kounadi et al. "Accuracy and privacy aspects in free online reverse geocoding services". In: *Cartography and Geographic Information Science* 40 (Mar. 2013), pp. 140–153. DOI: 10.1080/15230406.2013.777138. URL: https://www.researchgate.net/publication/236221915_Accuracy_and_privacy_aspects_in_free_online_reverse_geocoding_services.

[15] D Lemke et al. "[Who Hits the Mark? A Comparative Study of the Free Geocoding Services of Google and OpenStreetMap]". ger. In: *Gesundheitswesen (Bundesverband der Arzte des Offentlichen Gesundheitsdienstes (Germany))* 77.8-9 (Sept. 2015), e160–5. ISSN: 1439-4421. DOI: 10.1055/s-0035-1549939. URL: https://doi.org/10.1055/s-0035-1549939 (visited on 08/09/2023).

[16] Abhishek Nagaraj and Henning Piezunka. "How Competition Affects Contributions to Open Source Platforms: Evidence from OpenStreetMap and Google Maps". en. In: ().

[17] *National Bridge Inspection Standards*. May 2022. URL: `https://www.federalregister.gov/documents/2022/05/06/2022-09512/national-bridge-inspection-standards` (visited on 08/04/2023).

[18] *Nominatim*. URL: `https://nominatim.openstreetmap.org` (visited on 09/05/2023).

[19] *OSM file formats - OpenStreetMap Wiki*. URL: `https://wiki.openstreetmap.org/wiki/OSM_file_formats` (visited on 08/31/2023).

[20] Claudio Owusu et al. "Geocoding Fundamentals and Associated Challenges". In: Sept. 2017, pp. 41–62. ISBN: 978-1-138-62644-7. DOI: `10.1201/9781315228396-3`.

[21] *Pelias Geocoder*. URL: `https://www.pelias.io/` (visited on 08/29/2023).

[22] *PyOsmium - osmcode*. URL: `https://osmcode.org/pyosmium/` (visited on 08/29/2023).

[23] John W. Ratclif. "Pattern Matching: the Gestalt Approach". In: *Dr. Dobb's* (July 1988). URL: `http://www.drdobbs.com/database/pattern-matching-the-gestalt-approach/184407970` (visited on 11/22/2023).

[24] "Recording and Coding Guide for the Structure Inventory and Appraisal of the Nation's Bridges". en. In: (May 1995).

[25] Sukhjit Singh Sehra, Jaiteg Singh, and Hardeep Singh Rai. "A Systematic Study of OpenStreetMap Data Quality Assessment". In: *2014 11th International Conference on Information Technology: New Generations*. Apr. 2014, pp. 377–381. DOI: `10.1109/ITNG.2014.115`.

[26] Tage Sørensen et al. "A method of establishing group of equal amplitude in plant sociobiology based on similarity of species content and its application to analyses of the vegetation on Danish commons". In: 1948. URL: `https://api.semanticscholar.org/CorpusID:135206594`.