# Assessing semantic information of Volunteered Geographic Information

Gursimar Kaur*[1], Sukhjit Singh Sehra†[2] and Sumeet Kaur Sehra‡[3]

[1]Department of Computer Science & Engineering, GNDEC, Ludhiana, Punjab, India

**KEYWORDS:** Volunteered Geographic Information (VGI), semantic, OpenStreetMap (OSM), completeness, perception

## 1 Introduction

Although high quality geospatial information is in great demand but limited due to short supply, OpenSteetMap (OSM) is striving hard to compete with the commercial vendors providing worldwide map data-sets free of cost. The users can now contribute to fine cartographic mapping and enrich it with attribute details. This area was traditionally limited to professionals and expert cartographers further imposing technical and legal restrictions over its access to end-users(Haklay and Weber, 2008).

With the introduction of Web 2.0, this new version of World Wide Web allowed the non-expert contributors to help in map completion and quality improvement. OpenStreetMap (OSM) is the most successful example of VGI paradigm which has main focus on the collection of geographic information from people having local knowledge and make it publicly available to allow a two-way interaction of information flow. With the enormous potential of the ever-rich dataset, it has its own limitations. The individuals are contributing to OSM either constructively or damagingly(Arsanjani et al., 2013). The community-based efforts collaborating at global level result in the semantic heterogeneity for attributes which itself highlights deeper research issues. The choice of tags is usually dependent on the perception of the contributors leading to a different choice of word phrases although all of the contributions are semantically similar. The basic attributes evolve in a bottom-up way, which are further challenged due to typos and redundancies introducing unwanted noise in data quality(Codescu et al., 2011). The main focus was on identification of the intrinsic data quality elements for analysis of data under observation using various similarity measures.

The remainder of this paper is organized as follows: Section 2 discusses related work for analysing the semantic information of geographic features. Section 3 outlines the methodology of designing

*gursimar.sokhi@gmail.com
†sukhjitsehra@gmail.com
‡sumeetsehra@gmail.com

the matching algorithms for feature correspondence. Section 4 cover the results obtained with the algorithms developed. Section 5 draws a conclusion and directions for future work. Sections 6 briefly covers the academic background of the authors.

## 2 Related Work

The significant approaches to evaluate semantic accuracy of the real world features includes following authors as shown in Table 1.

Table 1: Literature Review

| **Authors** | **Description** |
|---|---|
| Haklay (2010) | Focused on analysis of OSM information quality through a comparison with Ordnance Survey(OS) dataset. |
| Koukoletsos (2012) | Proposed a framework for quality evaluation of linear VGI datasets and evaluated semantic (or attribute) accuracy and completeness. Further, it analysed the data by using an automated matching procedure for text comparison. |
| Mashhadi et al. (2015) | Focused on the positional and semantic accuracy as well as completeness of OSM information. The study determined the impact of the socio-economic factors on the quality of OSM data. |
| Ballatore et al. (2013) | Devised a mechanism for computing the semantic similarity of the OSM geographic classes to alleviate the prevailing semantic gap. It consist of the development of the OSM Semantic Network by means of a web crawler tailored to the OSMWiki website. |
| Vandecasteele and Devillers (2015) | Analysed various quality aspects and developed an open source plug-in called OSMantic (OSM Semantic) for the Java OpenStreetMap (JOSM) editor. This plug-in automatically suggests related tags to the contributors to enhance the user experience and to reduce the semantic heterogeneity. |

## 3 Methodology

For the purpose of this study, we considered two datasets namely, test and reference datasets which focused on feature correspondence for the attribute constraints (as shown in Table 2). Computing a score for the given pair of attribute names, we quantify the quality of OSM data depicted as lexicographical errors by Mashhadi et al. (2015), describing the index of similarity of names in accordance

with their semantics. Further, we attempted to generalize the output with the frequency distribution to explore the hidden redundancies and ambiguous data. In order to analyse the semantic

Table 2: Test and Reference Dataset

| **Id** | **Test dataset name** | **Reference dataset name** |
|---|---|---|
| 1 | MC-GRATH | MC GRATH |
| 2 | KOYUK | KOYUK HILL |
| 3 | HOMER | MAST HOMER HARBOUR |
| . | . | . |
| . | . | . |
| . | . | . |
| 70 | ANVIK | ANVIK INC MAINS |
| 71 | SOLDOTNA | SOLDOTNA |

relationship between the given names, it is crucial to analyse measure of similarity by computing distance or metric between two given name strings. Amongst the well known string searching algorithms, the suitable similarity measures for the analysis includes three edit-based algorithms : Levenshtein distance, Jaro-Winkler distance and Longest Common Sub-string algorithm.

### 3.1 Levenshtein Distance

Also called as Edit distance, Levenshtein distance is a commonly used text similarity algorithm. The algorithm is used to compute the minimum number of operations to measure the similarity between two strings from both the datasets. The basic operations which are performed to make one string identical to other have cost one for one edit and are defined as insertion, deletion and substitution. A value of one to three is regarded to minor misspelling errors (Girres and Touya, 2010). According to Will (2014), normalized Levenshtein distance is defined as Equation 1.

$$\frac{LevenshteinDistance(string1, string2)}{maximum(string1length, string2length)} \tag{1}$$

The computation considers two strings equivalent if the calculated distance is less than 0.35(Mashhadi et al., 2015). The distance results in '0' if the string variables are same and '1' if they are totally unique.

### 3.2 Jaro-Winkler distance

The Jaro-Winkler distance is the extension to the original Jaro string comparison along with the Winkler's modification which recomputes the score considering the initial characters of the given pair of strings. The idea behind the metric is that if you can calculate the number of transpositions and

the number of unique characters in the given pair of strings, then the distance computed between '0' (unique) and '1' (exact match) will represent the index of similarity. It is most preferably used for place names or person names. The basic operations performed to compute Jaro-Winkler distance includes insertion, deletion and transposition.

### 3.3 Longest Common Sub-string Algorithm

The longest common sub-string (LCS) algorithm is a string similarity algorithm which is used to decide how far or close the meaning of name attribute is related to the other one. It can efficiently access the common elements between given character strings. Meanwhile, the Levenshtein distance only deals with spelling mistakes and rectifies it, the LCS algorithm works much efficient as it works in iterative steps. It recursively finds and removes the common sub-string and then removes the next longest common substring appearing in same order. The idea is to find the common length of suffix for all substrings of both strings using dynamic programming.

## 4 Results

The contributions made by the non-expert volunteers can be misunderstood due to various ways used for spelling an entity which deviates from the general understanding of how it should apply. Table 3 represents the results obtained as the normalized values of computed Levenshtein distance, Longest Common Sub-string algorithm and Jaro-Winkler distance. To evaluate the performance of

Table 3: Results

| **Id** | **Levenshtein distance** | **Longest Common Substring** | **Jaro-Winkler distance** |
|---|---|---|---|
| 1 | 0.125 | 0.625 | 0.917 |
| 2 | 0.5 | 0.5 | 0.833 |
| 3 | 0.722 | 0.278 | 0.693 |
| . | . | . | . |
| . | . | . | . |
| . | . | . | . |
| 70 | 0.677 | 0.333 | 0.788 |
| 71 | 0 | 1 | 1 |

the implemented string searching algorithms, it is important to understand the meaning in which the names of attributes describing the tags (key-value pairs) are used. The contributions made by the amateur and intermediate mappers, lack of verification and observation to check its credibility

and ambiguous entries to describe the OSM object are the main reasons of semantic heterogeneity. The results obtained from the designed algorithms depend on two important aspects, which are the scores computed for finding the correlation in the attribute names and the status which depicts if they are acceptably similar or not. Figure 1 represents the interpretation of scores obtained from developed similarity measures using Levenshtein Distance, Longest Common Sub-string Algorithm and Jaro-Winkler distance as frequency distribution plot. For each score value, the height of the bar in the frequency distribution depicts count of its occurrence as computed for the corresponding pair of attributes.

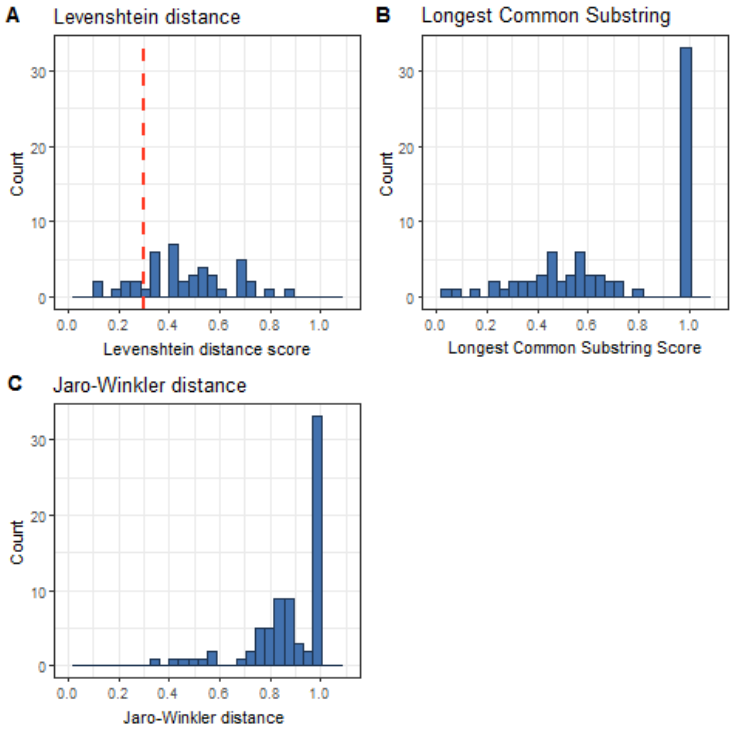


Figure 1: Normalized semantic similarity score for (A) Levenshtein distance, (B) Longest Common Sub-string algorithm, (C) Jaro-Winkler distance

Despite the ubiquity of the well known string similarity metric named Levenshtein distance in terms of its accuracy, it was outperformed by Jaro-Winkler distance and Longest Common Sub-string Algorithm. Considering the Jaro-Winkler distance, the values computed lie closer to '1' mostly falling in range of '0.7' to '1' which signifies that the corresponding names are either semantically similar or can be considered as an exact match. This signifies that the scores computed result in similar values for approximately similar set of names. On the other hand, Levenshtein Distance considers those corresponding attributes as similar which have their score less than or equal to '0.3' ('0' as exact match). The Longest Common Sub-string Algorithm focuses on substring that is necessarily contiguous in both the strings. The scores obtained ranged from '0' to '1' where the

count of occurrences of '1'(a perfect match) is highest amongst other values.

### 4.1 Levenshtein distance

The results obtained by comparison of attributes of two datasets as shown in Table 3 depicts the normalized distance which is computed considering the minimum number of operations required to convert one name similar to another.

Value as '0' depicts a total match, whereas value as '0.5' depicts slight deviation in both the names. Considering the distance obtained, the values which are less than or equal to '0.35' are acceptable and are shown as blue, and other greater values are shown as red which are used to depict names with higher variation and thus considered not similar. To gain an insight to the overall trend of how may attributes were acceptable or not are generalized in Figure 1 (A). It shows efficient results when strings only contain small differences but is unable to tackle the diversity in which feature names can be written. It focuses more on automatically correcting the spelling mistakes prevailing in the given strings.

### 4.2 Longest Common Sub-string algorithm

This algorithms is preferably used for compound names. It repeatedly locates the longest common sub-string in the corresponding attributes to a minimum length. The time complexity of the computation is $O(|s1|X|s2|)$ using $O(min(|s1|, |s2|))$ space.

It is based on the approach of fuzzy string matching in which the computed values depicts the degree of similarity. The normalized values in the range from '0' to '1' help in rephrasing the logic of "Are the strings similar?" as "How similar are the given strings?". It is considered as a better technique than Levenshtein distance for matching these attribute names (also refer Figure 1 (B)). The frequency count as '1' for strings which are computed as a match is much higher than the normalized Levenshtein distance. It is observed that the values calculated are considered as similar (shown in blue color in Table 3), if it is more than or equal to 0.30 and the score for names not so similar are depicted in red colour.

### 4.3 Jaro-Winkler distance

The distance computed for string matching is used to measure similarity between two strings. The higher the computed distance for the corresponding attributes, the more similar they are.

The algorithm considers that fewer errors typically occur at the beginning of the names. Table 3 represents the normalized Jaro-Winkler distance and the status assigned to names. We obtained values from '0' to '1' which assigns the colour as blue if score is more than or equal to 0.74 to label it as similar. It is preferably used for short names. Figure 1 (C) depicts the frequency count of distance calculated and signifies that most of the feature names compared are approximately similar in semantics.

## 5 Conclusion and Future Scope

A simple method has been proposed to explore the data type information accompanied with hidden semantics. It is observed that the people tag the attributes for map features differently due to different perceptions and mapping habits influenced by past experiences. The procedure compared the test and reference datasets using string searching algorithms. It focused on reducing semantic heterogeneity and resolving uncertainties which prevails due to usage of semantically similar tags. The performance of the algorithms was acceptable when the corresponding features had similar geometric representation. It further helped to reduce the imbalance in spatially rich information and its attribute details in Volunteered Geographic Information.

The evaluation of the matching results showed that the designed scripts produced a low mismatching error, but on the other hand failed to match a considerable amount of features. Therefore, improvements are suggested by using the buffer method. We can further take help of DBpedia to fill the semantic gap prevailing in the spatial information which extracts structured information from Wikipedia. Completeness (omission) and positional accuracy of OSM dataset are acceptable.

## 6 Biography

**Gursimar Kaur** is currently pursuing her master's degree in Computer Science & Engineering with research interest in semantic analysis of OpenStreetMap (OSM) data.

**Sukhjit Singh Sehra** and **Sumeet Kaur Sehra** are presently working in GNDEC at Department of Computer Science & Engineering as assistant professor. Their research interest is focused on assessment and improvement of VGI quality.

## References


Arsanjani, J. J., Barron, C., Bakillah, M., and Helbich, M. (2013). Assessing the Quality of OpenStreetMap Contributors together with their Contributions. In *16th AGILE Conference on Geographic Information Science*, Leuven, Belgium.

Ballatore, A., Bertolotto, M., and Wilson, D. C. (2013). Geographic knowledge extraction and semantic similarity in openstreetmap. *Knowledge and Information Systems*, 37(1):61.

Codescu, M., Horsinka, G., Kutz, O., Mossakowski, T., and Rau, R. (2011). DO-ROAM: activity-oriented search and navigation with openstreetmap. In *GeoSpatial Semantics*, volume 6631 of *Lecture Notes in Computer Science*, pages 88–107. Springer.

Girres, J.-F. and Touya, G. (2010). Quality assessment of the french openstreetmap dataset. *Transactions in GIS*, 14(4):435–459.

Haklay, M. (2010). How good is volunteered geographical information A comparative study of openstreetmap and ordnance survey datasets. *Environment and Planning B: Planning and Design*, 37(4):682–703.

Haklay, M. and Weber, P. (2008). Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18.

Koukoletsos, T. (2012). *A Framework for Quality Evaluation of VGI linear datasets*. phdthesis, University College London (UCL).

Mashhadi, A., Quattrone, G., and Capra, L. (2015). *OpenStreetMap in GIScience*, chapter The Impact of Society on Volunteered Geographic Information: The Case of OpenStreetMap, pages 125–141. Lecture Notes in Geoinformation and Cartography. Springer International Publishing.

Vandecasteele, A. and Devillers, R. (2015). *OpenStreetMap in GIScience*, chapter Improving Volunteered Geographic Information Quality Using a Tag Recommender System: The Case of OpenStreetMap, pages 59–80. Lecture Notes in Geoinformation and Cartography. Springer International Publishing.

Will, J. (2014). Development of an automated matching algorithm to assess the quality of the OpenStreetMap road network. Master Degree Thesis, Department of Physical Geography and Ecosystem Science, Lund University, Slvegatan, Sweden.