Studying the Dynamic Patterns of OpenStreetMap Bugs in Great Britain

Amir Pourabdollah Jeremy Morley Steven Feldman Mike Jackson

The University of Nottingham
Nottingham Geospatial Building, Jubilee Campus
Nottingham, NG7 2TU
United Kingdom
amir.pourabdollah ; jeremy.morley ; steven.feldman ; mike.jackson@nottingham.ac.uk

Abstract

Studies of the quality of OpenStreetMap data have generally focused on limited quality measures applied to snapshots of the OSM database. In this project our objectives are to check a wider range of ISO-19157 quality types and to examine the dynamics of features and errors in OSM. In this paper we report on a study covering the OSM database for Great Britain using 17 quality rules assessed daily over a 50 day period from 28-10-2012 till 17-12-2012. In this period, the overall increase in bugs is 7.5 per 1000, compared with a historical rate of 14 per 1000. However an examination of the rates at which bugs are added and removed indicates a complex picture in which for most bugs the entry and removal rates are closely balanced. Line intersections without junction bugs however grew by approximately 6500 in the period and represent a particular area for further work in the OSM community.

Keywords: OpenStreetMap, data quality, dynamics, OSM, bugs.

1 Introduction

Since its inception in 2004, OpenStreetMap (OSM) has become the main free source of digital maps made by the crowd. Although OSM is rapidly growing in both contents and contributors, the belief that it is made by amateurs is perceived to limit trust in the value of this free data source within the traditional GIS community. The quality aspects of OSM have been investigated by different researchers and with different tools ([1-4]). We contend that to encourage uptake of data, not only must the OSM community produce better maps but the patterns of such map quality improvements should also be effectively demonstrated. It is therefore necessary to study the dynamics of the production and removal of *bugs* from the map over time. In the context of this paper, a "bug" is a deviation from the defined quality assurance rules.

This paper presents research conducted around static and dynamic analysis of data and bug growth in OpenStreetMap. The research is done within the framework of the OSM-GB project [5], in which the OSM data for Great Britain is analysed and redelivered for professional use [6]. The geographical scope of the research is Great Britain, mainly because it is the country where OSM started its work, and also because it has excellent authoritative mapping from the Ordnance Survey of Great Britain (OSGB) which can be used for "ground truthing". The quality measures are done within a rule-based geo-processing engine and are regularly repeated over time to demonstrate the dynamics of the bug patterns.

The following sections provide information on OSM-specific data quality, the research methodology and the outcome. Sample results are then discussed and the future work outlined.

2 Data Quality in OpenStreetMap

In addition to the general spatial data quality measures standardised in ISO 19157 [7], OSM has its own specific aspects of quality metrics. In part this relates to the free and open nature of feature attribution in OSM. Each feature in OSM can have an unlimited number of attributes in a key-value pair format (also called "tags"), in which both the key and value are free text. The OSM community has documented a list of standard key-value pairs that can be used to describe a real-world feature [8]. The standard set determines how the relevant features are graphically rendered by the common OSM tools, like the map rendering engine, Mapnik [9]. However, users are free to use their own tags to elaborate particular feature details. Such non-standard attribution can still be mapped by mappers using their own rendering tool and cartographic rules (e.g. TileMill [10]). The attribution standard set by the OSM community itself represents a reference for quality checking.

In general, OSM quality metrics are a combination of geometry and attribution checks. Both self-consistency and relative-to-reference rules can be applied, where each rule may originate from the standard spatial quality measures or from the OSM community standards. From the combination of all the above sources of quality metrics, a long set of checks can be defined for OSM. Some OSM editing tools, e.g. Potlatch and JOSM impose simple validation checks for data entry, however there are many other validation checks that are

either missed or not possible to be applied on the data entry phase.

Reviewing existing works on OSM quality, there are firstly a number of research articles that statistically compare an OSM snapshot with reference maps in order to reveal OSM's relative accuracy and/or completeness as a whole ([2] also followed up and extended in [11] and [12]). Secondly, other researchers have focused on a number of pilot self-checking methods, for example the node spacing on specified feature polygons in an OSM snapshot [13]. Thirdly there are a number of online or desktop tools that can discover a number of predefined self-checking bugs from the live OSM database in specific areas [14-17].

The observed gaps towards quality improvement are:

1- Effective serving of the individual detected bugs, instead of just reporting the statistics. Knowing the statistics on its own does not help the community to return and correct the particular errors.
2- Frequently iterating the bug detection process, instead of working on snapshots. Knowledge of the dynamics of bugs can help the community understand the rate of OSM quality improvement (or worsening). It can also promote trust in the data for the professionals.

In other words, we believe that knowing how good or how bad the OSM data is just the first step. The other steps are how to help the community to correct the data, and how to show potential adopters that it is (or it is not) improving, and how these changes are distributed geographically. These are the OSM-GB project ultimate targets [18].

We acknowledge that in some circumstances bugs may not be real mapping errors. Our aim is for bugs to be fed back to the OSM community for humans to check and fix as appropriate, in line with the OSM community's ethos. An example bug that would not necessarily be an error is a one-way road segment that does not terminate in another road, apparently trapping traffic. However if it terminates on the edge of a car park polygon the road segment could be correct (if the car park has another exit). In theory elaborate rules could be constructed to cover such eventualities but in a system of distributed volunteer mappers it may be simpler and more reliable to defer to manual checks.

While community bug fixing is the ultimate aim, some bugs are amenable to automatic fixing (e.g. spikes and undershoots). Our system applies automated fixes to produce an "OSM-GB" database which we maintain locally. The methods of partial fixing and effective serving of the individual detected bugs are part of our project but are beyond the scope of this short paper.

3 Methodology

3.1 System Design

The designed system capable of performing the following tasks:

- Downloading OSM data for Great Britain into a local database and frequently updating it with change;
- Detecting the data bugs within a rule-based engine for quality checking, and repeating this process as changes are merged into the database;
- Logging the results in each cycle;
- Serving the raw and analyzed data, current and historical detected bugs through a number of web services and in a variety of coordinate reference systems. The data delivery is done in Creative Commons (CC-BY-SA) License, for free, and can be downloaded in a number of vector and raster formats. Any corrected bug is not intended to automatically be fed to the OSM database, since we believe that the corrections shall be done by the OSM contributors.

A number of open-source tools have been utilized to perform the above tasks: PostGIS as the database, osm2pgsql and osmosis for importing, updating and replicating the OSM dataset, ogr2ogr for coordinate system transformations, and Mapnik and GeoServer for making the output through Web Services. The engine for rule-based quality checking is 1Spatial's Radius Studio [19]. With Radius Studio, we have been able to easily define and modify different sets of bug detection rules.

The frequency of updates, bug detecting and results logging has been set at one day. At the time of writing, the cycle has been repeated for about 100 consequent days but results presented in this paper are for a period of 50 days as will be explained in section 4.

Although the research scope is limited by time, update period, rules and geographical extent, the system is easily scalable. More rules and larger geographical extent need more computation times. The update period has been defined to be as frequent as possible whilst guaranteeing that each cycle is finished before the next iteration. At present we have not found a pressing need for more frequent updates.

3.2 Rules selection

With reference to ISO-19157 (Geographic information - Data quality) [7] the geospatial data quality elements are categorized as: Completeness, Logical Consistency, Positional Accuracy, Usability, Thematic Accuracy and Temporal Accuracy. In our research methodology at this stage, we have focused on completeness and logical consistency, since a number of errors in these categories are detectable without reference data. Accuracy checking which requires reference maps and/or datasets is not the focus of this paper but is a part of our further work.

Completeness validation consists of checking for commissions and omissions. Logical consistency may comprise conceptual or domain-related elements, format or topology of the geographical features. Among those groups, we focus on the conceptual and topological consistencies. We have categorized the detectable bugs into the following three groups: *Geometry bugs, Attribution bugs and combined Geometry/attribution bugs.*

3.3 Rules definition

The following bug detection rules are defined for this pilot study (some taken from [14]). They are applied to the OSM geometries of nodes, linear features ("ways"), and polygon features, which are defined as closed ways (i.e. forming rings) with appropriate identifying tags [8].

The rules comprise geometry and attribution checks (independent of a reference map):

- ***Spikes and Kickbacks*** (geometry bugs): unusual sharply acute corner in lines and polygons (figure 1).
- ***Self-intersected Lines*** (geometry bug): as a part of simple geometry checks according to OGC standards [20].
- ***Doubled Places*** (attribution bug): a point feature has the same name as the surrounding polygon.
- ***Different-layer Joints*** (geometry/attribution bug): when two roads have a common vertex, it is an inconsistency if their layer tags (indicating their relative elevations) are different.
- ***Intersection Without Junction*** (geometry/attribution bug): where two roads cross with no common vertex but they have the same layer tag. This and the previous bug are potential problems for routing algorithms.
- ***Overlapping Roads***: two roads in the same elevation having common edge(s).
- ***Ways Intersecting Buildings*** (geometry/attribution bug): roads cannot cross buildings unless at different elevations. (A bug that may not always be an error).
- ***Overlapping Buildings*** (geometry/attribution bug): buildings that are geometrically overlapped in small sections, which is a case of topological inconsistency. The overlapping structures are only valid if they are at different elevations.
- ***Unclosed Area*** (geometry/attribution bug): the OSM community has agreed to use an "area" tag to determine the closed areas. It is inconsistent to set the "area" tag for non-closed geometries.
- ***Invalid Motorway Connection*** (geometry/attribution bug): OSM has certain rules for connecting Motorways to the other types of road (country-dependant). For example a motorway cannot connect to a residential way (figure 1).
- ***One-way Roads (cul de sacs)***: it is also invalid for one-way roads to be dead-ends.
- ***Un-tagged Bridges/Tunnels*** (attribution bug): If the agreed OSM tagging for bridges/tunnel is missing (particularly the "highway" tag), the feature may not be shown on the map.
- ***Wrongly tagged Bridges/Tunnels*** (attribution bug): the OSM agreed rules for the "level" tag of the bridges (level > 0) and tunnels (level < 0) are not followed, which can cause wrong rendering.

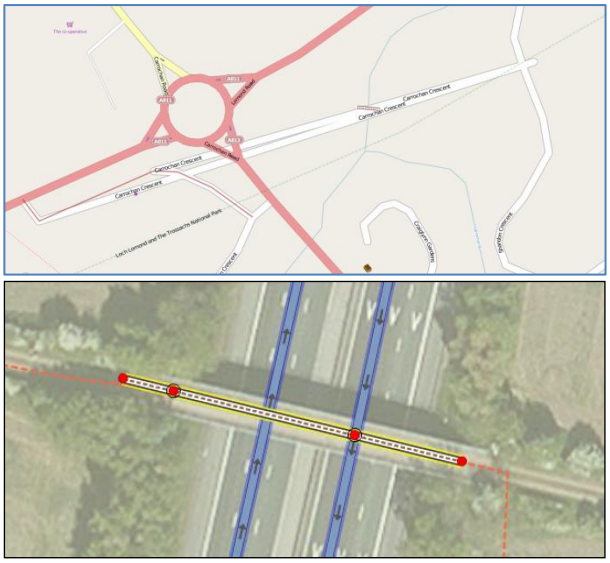

Figure 1: A sample bug: A road spike (above– the white road) and an invalid bridge to motorway connection (below)

4 Results

The above rules were applied on a daily iteration over 50 consequent days (28-10-2012 till 17-12-2012), producing the results summarised below. For a better understanding of the rate of change of bugs, the changes in numbers of features are presented first.

4.1 Feature Dynamics

Table 1 shows the statistics of the three OSM feature types.

Table 1: The dynamics of OSM feature types (in GB). POI = Point of Interest.

Feature	Current (17-12-2012)	Added per day			Changed per day		
		Average Number	Std. Dev.	Average Rate	Average Number	Std. Dev.	Average Rate
Lines	3,133,900	10,665	4,153	0.34%	7,486	3,205	0.23%
Polygons	2,276,354	8,726	3,560	0.38%	6,125	3,024	0.27%
POIs	1,212,283	2,241	1,314	0.18%	2,248	2,986	0.19%
Total	**6,622,537**	**21,632**	**3,429**	**0.33%**	**15,859**	**3,103**	**0.24%**

4.2 Bug Dynamics

Over the test period, the rules described in section 3.3 have been applied daily on the line and polygon features for the whole of Great Britain. The features which fail the rules are counted each day. For each bug type the results on day 1 have been compared to the results on day 50 in table 2, including the growth rate and numbers of remaining, added and removed bugs. Across the period shows an almost linear growth.

The growth rate and the number of remained, added and removed bugs have been shown in table 2.

5 Discussion

5.1 Features and bugs growth

Table 1 shows an average of 0.33% growth of OSM in GB per day, or about 120% per year. The growth rate of OSM worldwide in 2011 has been reported to be about 75% [21] equivalent to 0.2% per day.

Table 2: The pattern of bug detection

Feature type	Detected Bug	Buggy features in day 1	Buggy features in day 50	Bug growth	Remained bugs	New bugs in the 50 days	Removed bugs during the 50 days
Lines	One-way dead-ends (cul-de-sacs)	377	369	-2.12%	273 (72.41%)	96 (25.46%)	104 (27.59%)
	Different-layer joint	1,339	1,372	2.47%	1, 277 (95.37%)	95 (7.09%)	62 (4.63%)
	Kickback	249	254	2.01%	179 (71.89%)	75 (30.12%)	70 (28.11%)
	Spike	1,398	1,306	-6.58%	1,070 (76.54%)	236 (16.88%)	328 (23.46%)
	Intersection without junction	33,210	39,709	19.57%	30,136 (90.74%)	9,573 (28.83%)	3,074 (9.26%)
	Invalid motorway connection	45	46	2.22%	43 (95.56%)	3 (6.67%)	2 (4.44%)
	Overlapping roads	1,824	1,769	-3.02%	1,537 (84.27%)	232 (12.72%)	287 (15.73%)
	Self-intersected line	20,529	20,791	1.28%	19,963 (97.24%)	828 (4.03%)	566 (2.76%)
	Unclosed Area	2,340	2,411	3.03%	2,252 (96.24%)	159 (6.79%)	88 (3.76%)
	Un-tagged bridge	333	336	0.9%	331 (99.40%)	5 (1.50%)	2 (0.60%)
	Wrong-level bridge	634	646	1.89%	623 (98.26%)	23 (3.63%)	11 (1.74%)
	Wrong-level tunnel	168	169	0.6%	164 (97.62%)	5 (2.98%)	4 (2.38%)
Polygon	Doubled place	4,991	5,090	1.98%	4,845 (97.07)	245 (4.91%)	146 (2.93%)
	Kickback	73	69	-5.48%	67 (91.78)	2 (2.74%)	6 (8.22%)
	Spike	3,284	3,403	3.62%	3,249 (98.93%)	154 (4.96%)	35 (1.07%)
	Overlapping Buildings	17,964	18,851	4.94%	17,835 (99.28%)	1,016 (5.66%)	129 (0.72%)
	Ways intersecting Buildings	8,887	9,172	3.21%	8,725 (98.18%)	447 (5.03%)	162 (1.82%)
Summary		**97,645**	**105,763**	**8,118 (8.31%)**	**92,965 (94.8%)**	**13,194 (13.51%)**	**5,076 (5.21%)**

The bug generation and removal pattern in table 2 show that in the trial period, the number of detected bugs has grown by 8.31%. It also shows that this has been the result of 5.2% bug removal offset by 13.51% generation of new bugs. This shows that in this period bug removal is not keeping up with bug entering the database. This is illustrated in figure 2.

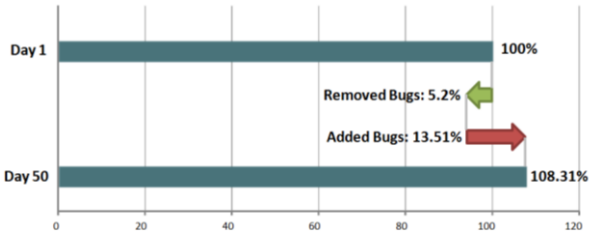

Figure 2: The pattern of bug additions and removals

5.2 Historical improvements in bugs production

The absolute number of bugs obviously dependents on the scope of the rules, so may not be meaningful by their own. However comparing the patterns of their growing from the past till now can provide meanings. According to table 2, overall there are 8118 more bugs at the end of the analysis period. This is equivalent to 162 more bugs per day. According to table 1, OSM grows daily by 21,632 features. This shows that for every 1000 features added, the bugs in the database have increased by 7.

On the other hand, on the day 1, buggy features were 97,645 out of 6,622,537 features, equivalent to 14 per 1000. This shows that while the history of data entry in OSM shows 14 bugs per thousand, the rate has currently decreased to 7 per thousand. However this includes bug fixing: looking only at the rate at which bugs are created (13,194 over 50 days, or 264 / day), the raw bug creation rate is 12 per 1000 new features.

In summary, the GB mappers overall produce bugs at about half the rate that they have since the beginning of OSM, within the scope of our rule base, but this is principally because of effective bug fixing.

5.3 Specific bugs dynamics

While figure 2 shows the pattern of bug production as a whole, each bug in table 2 shows a different bug growth pattern. Figure 3 illustrates those individual patterns sorted by overall growth rate.

Most of the bugs are growing but there are four that show negative growth because bug removals have overtaken bug creation. "Line Spike" for instance has a -6.58% growth (while "Polygon Spike" has grown by 3.62%). The fastest growing bug is "Intersection without junction", as the rate of creation is more than 3 times the rate of removal. It is noticeable that the overall growth rate is not necessarily consistent with the underlying creation and removal dynamics. "Overlapping building" for instance shows about 5% growth, while it is created about 8 times faster that it is

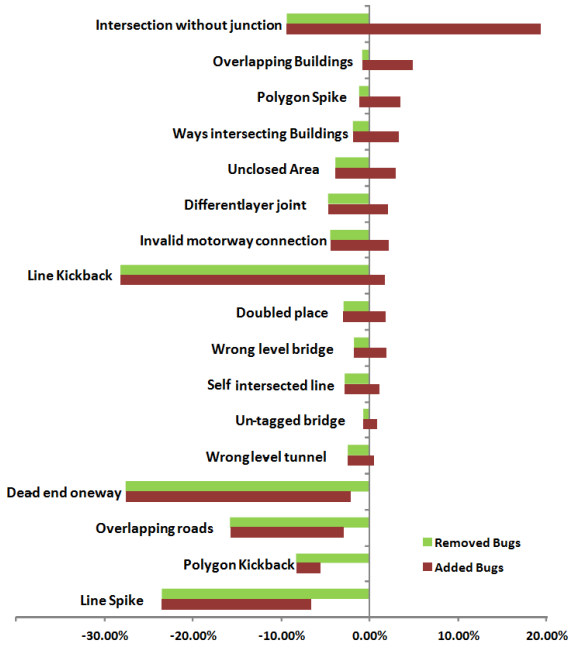

Figure 3: The pattern of bug additions and removals (The right-hand end of each red bar shows the total of removed and added)

removed. Line kickbacks grew by 2% but this is the marginal difference between 30% new bugs and 28% removal of bugs.

While some of the differences in bug rate between rules may be due to specific factors in the 50 day period, there are interested messages about the types of bugs that need particular consideration by OSM contributors. Also it shows the level of the contributors' attention to correcting each type of error.

Finally it is seen that the dynamics of different bug types have different behaviors and studying the sources of the differences are left for the future works.

6 Future Work

In the future the analysis can be done for more rules and for longer periods. Then the main future work is on studying the effect of sharing the individual bug reporting on decreasing the slope of the detected bugs in the short and long term. Other future work needed is on the scalability and adaptation of the method to other geographical areas, as well as studying the bug patterns according to the user's contributions.

Since the rules are developed based on a set of pre-defined logic, there is possibility of producing false-positives and false-negatives. Another study may be necessary to assess the level of those mistakes and the real-world validity of the developed rules.

7 Conclusions

The results here represent the changes in features across a relatively limited time period but show interesting characteristics. For most bugs, the rates of addition and removal of bugs are relatively well balanced (for 14 out of 17 rules the growth rate is within 5% of 0, figure 3). However this masks a wider variation in the addition and removal rates. In this sample, intersection without junction bugs are growing most in both relative and absolute terms while line spikes are being fixed quickest.

The future work will focus on examining the reasons for these patterns; issues of OSM tool design; feedback to the OSM community; and expanding the rule base with & without external reference data.

Acknowledgement

This research is funded by 1Spatial Ltd. and supported and KnowWhere Ltd. We also wish to acknowledge the collaborations from Ordnance Survey, Snowflake Software and Pitney Bowes Business Insight.

References

1. Zielstra, D. and A. Zipf. *A comparative study of proprietary geodata and volunteered geographic information for germany*. in *13th AGILE International Conference on Geographic Information Science*. 2010.
2. Haklay, M., *How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets.* Environment and planning. B, Planning & design, 2010. **37**(4): p. 682.
3. Mooney, P. and P. Corcoran. *Integrating volunteered geographic information into pervasive health computing applications*. in *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on*. 2011: IEEE.
4. Girres, J.F. and G. Touya, *Quality assessment of the French OpenStreetMap dataset.* Transactions in GIS, 2010. **14**(4): p. 435-459.
5. OSMGB. *OSM-GB Project Homepage - Measuring and Improving the Quality of OpenStreetMap for Great Britain*. 2012 [Accessed December 2012]; Available from: http://www.osmgb.org.uk
6. Morley, J., et al., *Formalizing the Crowd - applying formal quality assurance processes to OpenStreetMap*, in *AGI GeoCommunity*. 2012: Nottingham,UK.
7. ISO, *ISO/CD 19157 Geographic information – Data quality*. 2010.
8. OSM-Wiki. *OpenStreetMap Project Wiki*. 2012 [Accessed December 2012]; Available from: http://wiki.openstreetmap.org/wiki/Elements
9. Pavlenko, A. *Mapnik*. 2011 [Accessed December 2012]; Available from: http://mapnik.org/.
10. MapBox. *TileMill*. n.d. [Accessed December 2012]; Available from: http://mapbox.com/tilemill/.
11. Ather, A., *A quality analysis of openstreetmap data.* Master's thesis, University College London, 2009.
12. Kounadi, O., *Assessing the quality of OpenStreetMap data.* Msc geographical information science, University College of London Department of Civil, Environmental And Geomatic Engineering, 2009.
13. Mooney, P., P. Corcoran, and A.C. Winstanley. *Towards quality metrics for openstreetmap*. in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 2010: ACM.
14. KeepRight. *data consistency checks for OSM* 2012 [Accessed December 2012]; Available from: http://keepright.at
15. OpenStreetBugs. 2012 [Accessed December 2012]; Available from: http://openstreetbugs.schokokeks.org/
16. OSMInspector. *OSM Inspector, Geofabrik tools* 2012 [Accessed December 2012]; Available from: http://tools.geofabrik.de/osmi/
17. Scobbler. *MapDust*. 2012 [Accessed December 2012]; Available from: http://www.mapdust.com
18. Feldman, S., A. Pourabdollah, and J. Morley, *OSM-GB - Can quality improvement increase contribution?*, in *Society of Cartographers 48th Annual Conference*. 2012: London.
19. 1Spatial. *1Spatial Website*. 2012 [Accessed December 2012]; Available from: http://www.1spatial.com.
20. OGC. *Open Geospatial Consortium Web page* 2012 [Accessed December 2012]; Available from: http://www.opengeospatial.org/
21. BeyoNav. *OpenStreetMap's Growth Accelerates*. 2012 [Accessed December 2012]; Available from: http://www.beyonav.com/openstreetmaps-growth-accelerates