1. Problems encountered in the map

Over-abbreviated Street Names

The main issue was with abbreviated street names (eg. Blvd., St. Ave.). I decided that it was best to remove these in order to maintain consistency. The abbreviations which appear at the end of the string were easy enough to deal with. The challenge was those that appear in the substring (eg. Jerry Blvd N). Luckily, small update to the code from problem set 6 solved my issues.

The most challenging part of this was going through the huge variety of road names like “Treeway”, “Harbour” and “Gateway”. There were a lot more than we saw in the problem sets. Definitely was interesting going through the dataset!

For example, after cleaning the data
<tag k="name" v="225 Platten Blvd" /> becomes
<tag k="name" v="225 Platten Boulevard" />

Inconsistent use of underscores and spaces
Throughout the data there were many cases where underscores and spaces were used interchangeably to separate individual words. For example, in some amenity would use “book_store” while others used “book store”. Since spaces were the more common convention, I decided to split tag values by “_” and then join them back together with spaces.

For example, after cleaning the data
<tag k="highway" v="motorway_link" /> becomes
<tag k="highway" v="motorway link" />

In other cases, spaces were not used when they should have been. For example, all Canadian postal codes are six alphanumeric characters long (eg. M1S 1S4). However, I found that there was an inconsistency in formatting. Some nodes had a space in the postcode while others did not (eg. M4MT4X). The easiest way to fix this was to go through all the postcodes, check if there was a space in them and if not to add that space.

<tag k="addr:postcode" v="M5B1L4" /> becomes
<tag k="addr:postcode" v="M5B 1L4" />

Missing road meta tags
Another problem with the map are highways and roads missing meta tags with road information that would be important for any apps that are built on top of OSM. Some examples are max speed, whether the road is one way, whether it has sidewalks, number of lanes etc. Unfortunately, since most of these roads are missing positional (lat and long) information as well, I don't see how we could fix this programmatically.

2. Data Overview

This section contains basic statistics about the dataset and the MongoDB queries used to gather them.

File sizes
Toronto_canada.osm 1.24gb
Sample_toronto.osm.json 1.76gb

However in the interest of saving my macbook from exploding, I used a smaller subset of the data for this assignment (about 550mb)

```
# Number of documents
> db.cities_features.find().count()
2273700

# Number of Nodes
> db.cities_features.find({“type”: “node”}).count()
1747114

# Number of ways
> db.cities_features.find({“type”: “way”}).count()
526586

# Number of unique users
> db.cities_features.distinct(“created.user”).length
1500

# Top 1 contributing user
> db.cities_features.aggregate([{"$group": {"_id": "$created.user", "count": {"$sum":
1}}}, {"$sort": {"count": -1}}, {"$limit": 1}])
{ "_id" : "andrewpmk", "count" : 1403806 }


# Number of users appearing only once (having 1 post)
> db.cities_features.aggregate([{"$group":{"_id":"$created.user", "count":{"$sum":1}}},
{"$group":{"_id":"$count", "num_users":{"$sum":1}}}, {"$sort":{"_id":1}}, {"$limit":1}])
```

500

3. Additional Ideas

There are a lot of issues with the data can perhaps be fixed most easily by drawing on the power of open source contributors. In terms of how exactly to encourage more contributors, social feedback and gamification is the best way to accomplish this. Having user data more prominently displayed, a leaderboard with scores based on quality and number of contributions, different levels and rewards for the best contributors would be good incentives.

In addition, OSM should take a page from Wikipedia and focus on building a stronger community around its product. Start hosting meetups to bring contributors together, make it easy for contributors to connect with each other online and start recognizing milestones and accomplishments of long time members.

Of course, implementing gamification and community is not without drawbacks. Firstly, if gamification is not implemented carefully it can unintentionally incentivize users with external motivation. This is bad because when externally motivated, users start expecting rewards and external validation for their actions. In fact, they might start contributing even less as much as they otherwise would have since they start expecting some tangible benefit for their work. However, if executed carefully gamification can be extremely powerful (See Reddit, Quora, Yahoo! Answers etc.)

Building a community must be done carefully. It's obvious when it is done insincerely and its only purpose is to get people to work more. OSM must reflect upon why they're maintaining their project and do a good job of communicating this. Only then can you make contributors feel like they're a part of something important, which will intrinsically motivate them to contribute.

Top 10 appearing amenities

```
db.cities_features.aggregate([{"$match":{"amenity":{"$exists":1}}},
{"$group":{"_id":"$amenity", "count":{"$sum": 1}}}, {"$sort": {"count": -1}}, {"$limit":
10}])

{ "_id" : "parking", "count" : 2550 }
{ "_id" : "fast_food", "count" : 298 }
{ "_id" : "restaurant", "count" : 269 }
{ "_id" : "school", "count" : 259 }
{ "_id" : "bench", "count" : 206 }
{ "_id" : "place_of_worship", "count" : 184 }
{ "_id" : "post_box", "count" : 175 }
{ "_id" : "cafe", "count" : 142 }
{ "_id" : "waste_basket", "count" : 114 }
```

{ "_id" : "fuel", "count" : 96 }

Biggest religion (no surprise here)

```
db.cities_features.aggregate([{"$match":{"amenity":{"$exists":1},
"amenity":"place_of_worship"}},{"$group":{"_id": "$religion", "count": {"$sum": 1}}},
{"$sort": {"count": -1}}, {"$limit": 1}])
```

{ "_id" : "christian", "count" : 149 }

Conclusion

After a second review of this data, there are a lot more issues than I previously thought. Although it's amazing that so much data has already been provided through users' efforts, a testament to the great open source community here in Toronto. Once more GPS information is provided, in the future it would be more possible to write a bot to clean up the OSM data on Toronto.