

# Data Wrangle OpenStreetMaps Data

by Rica Enriquez, July 1, 2015

In this project, the OpenStreetMap data for Cambridge, United Kingdom is explored. It was downloaded from <https://mapzen.com/data/metro-extracts> on July 1, 2015. It was prepared for MongoDB using `nanoproject_2.py` and then added into a local 'udacity' database as the 'cambridge' collection.

## Import the database for querying below

```
In [1]: from pymongo import MongoClient
import pprint
import os

client = MongoClient()
db = client["udacity"]
```

## Section 1. Problems Encountered in the Map

There were a few problems with the street names. "chieftain" and "sweetpea" were not capitalized. This was fixed in "nanoproject\_2\_prep.py" using the mapping scheme similar to the "Improving Street Names" script from Lesson 6.11.

When importing the data, all labels were included. Many of these labels were only available for a small fraction of the documents. So in the following sections, underutilized upper labels and lower labels are removed. If a subset of the dataset is used in the future, removal of labels may need to be more judiciously.

## Removing underutilized labels

Using MongoDB, only upper labels that had at least 1000 documents were kept. The "nanoproject\_2\_prep.py" file prints the list of upper labels. This list is then looped through in "nanoproject\_2\_query.py" to check the number of documents it is used in. If there are less than 1000 documents, that label is removed and the database is updated.

```
In [2]: all_labels = ['real_ale', 'fhrrs', 'anglican', 'dance', 'ss052', 'demodified', 'maxspeed', 'smoking', 'openplaques',
'is_in', 'max_age', 'created_by', 'fax', 'cctv', 'mph', 'icao', 'automatic_door', 'motor_vehicle',
'school', 'level', 'notes', 'bus_stop', 'ncn_1', 'real_cider', 'disused', 'clothes', 'bicycle',
'cost',
'exit_to', 'leaf_type', 'access', 'fast_food', 'eligibility', 'water', 'tracks', 'address', 'hoops',
'used_to_be', 'permit_holders', 'microbrewery', 'survey', 'military', 'amenity', 'alt_name', 'fee',
'lwn',
'vehicle', 'have_riverbank', 'type', 'start_date', 'entrance', 'drinkable', 'club',
'campaign_for_by',
'give_way', 'visibility', 'site', 'phone', 'traffic_calming', 'room', 'tunnel', 'det', 'roof', 'male',
'history', 'estate', 'lock', 'currency', 'ncn_ref', 'pond', 'species', 'information', 'monitoring',
'gate', 'uk_postcode_centroid', 'FIXME', 'description', 'alt_ref', 'hazard', 'leisure', 'date',
'rental',
'natural', 'lcn_ref', 'wheelchair', 'outdoor_seating', 'healthcare', 'patio', 'office', 'trade',
'postal_code', 'motorcycle', 'int_ref', 'pitch', 'covered', 'derelict', 'old_ref', 'junction', 'food',
'material', 'foot', 'tourism', 'smoothness', 'fixme', 'name', 'designation', 'osmarender',
'embankment',
'crossing', 'kerb', 'name_1', 'frequency', 'naptan', 'access_land', 'loc_name', 'network', 'bus_bay',
'highways_agency', 'ref', 'brewery', 'highway', 'barrier', 'post_box', 'cars', 'maxweight',
'electrified',
'was_called', 'old_amenity', 'accommodation', 'tenant', 'noexit', 'segregated', 'route', 'atm',
'box_type', 'turn', 'place', 'high_capacity', 'support', 'note', 'owner', 'horse', 'service',
'priority',
'motorcar', 'park_ride', 'enforcement', 'noname', 'est_width', 'artist_name', 'old_old_name', 'ncn',
'population', 'multi_storey', 'royal_cypher', 'aeroway', 'landuse', 'tracktype', 'builder', 'bridge',
'occupier', 'nga', 'sidewalk', 'hgv', 'lit', 'takeaway', 'overall_site', 'payment', 'old_shop',
'aerodrome', 'url', 'medical', 'tactile_paving', 'shop', 'golf', 'indoor', 'social_facility',
'last_survey', 'gauge', 'mapillary', 'wood', 'fuel', 'iata', 'abutters', 'bracket_ref', 'tourist_bus',
'artist', 'motorboat', 'public_transport', 'power_source', 'automatic', 'int_name', 'locale',
'lamp_type',
'route_ref', 'parking', 'sport', 'power_supply', 'capacity', 'maxwidth', 'wikipedia', 'state',
'boundary',
'email', 'screen', 'denomination', 'key', 'substation', 'junction_ref', 'bar_billiards', 'railway',
'genus', 'comment', 'maintainer', 'wall', 'loading_gauge', 'outside_seating', 'recycling', 'height',
'ele', 'alt_description', 'boat', 'speech_output', 'mkgmap', 'waste', 'bicycle_parking', 'website',
'direction', 'lanes', 'building_1', 'craft', 'official_name', 'mail', 'grills', 'replaces', 'busway',
'parking_space', 'replaced', 'overtaking', 'layer', 'ons_code', 'backrest', 'telephone', 'surface',
'guided_busway', 'beer_garden', 'waterway', 'cuisine', 'education', 'surveillance',
'collection_times',
'status', 'wires', 'cyclestreets_id', 'fence_type', 'fruit', 'ownership', 'colour', 'contact',
'one-way',
'landmark', 'left', 'taxi', 'livestock', 'proposed', 'hour_off', 'not', 'voltage', 'seats',
'guest_house',
'isced', 'toilets', 'generator', 'TODO', 'bench', 'source', 'bollard', 'usage', 'emergency',
'historic',
'lcn', 'psv', 'furniture', 'vending', 'tower', 'internet_access', 'right', 'twitter', 'platforms',
'local_ref', 'man_made', 'religion', 'artwork_type', 'power', 'trees', 'Comment', 'incline',
'footway',
'industry', 'taxon', 'supervised', 'step_count', 'female', 'operator', 'area', 'unisex',
'opening_hours',
'museum', 'width', 'occupier3', 'occupier2', 'admin_level', 'bus', 'brand', 'delivery',
'construction',
'diaper', 'courts', 'old_name', 'real_fire', 'circuits', 'books', 'dispensing', 'display',
'crossing_ref',
'cinema', 'carriageway_ref', 'maxheight', 'cafe', 'cables', 'recycling_type', 'hour_on', 'locality',
'interior_decoration', 'cycleway', 'department', 'denotation', 'shelter', 'latest_survey_date',
'diet',
'min_age', 'maxstay', 'opened', 'building', 'yelp', 'wifi', 'traffic_signals']
```

```
# Number of documents in the collection
```

```
N = db.cambridge.find().count()
```

```
removed = []
```

```
kept = []
```

```
# Remove labels used in less than 1000 documents
```

```
for label in all_labels:
```

```
    pipeline = [{"$group": {"_id": "$" + label, "count": {"$sum": 1}}}, {"$match": {"_id": None}}]
```

```
    result = list(db.cambridge.aggregate(pipeline))
```

```
    if len(result) > 0:
```

```
        n = result[0]["count"]
```

```
        if n >= N - 1000:
```

```
            db.cambridge.update({}, {"$unset": {label: ""}}, multi=True)
```

```
            removed.append(label)
```

```
        else:
```

```
            kept.append(label)
```

```
print len(removed), "labels were removed and", len(all_labels) - len(removed), "labels were kept."
```

```
print "The labels kept are:", kept
```

```
336 labels were removed and 11 labels were kept.
```

```
The labels kept are: ['address', 'amenity', 'entrance', 'natural', 'foot', 'highway', 'barrier', 'landuse', 'source', 'ope
rator']
```

## Removing underutilized sublabels

Similarly, only lower labels that had at least 500 documents were kept. The "kept" list from the MongoDB query above is used in "nanoproject\_2\_prep.py" to print a dictionary of the list of kept upper labels and their sublabels. This dictionary is then looped through in "nanoproject\_2\_query.py" to check the number of documents each sublabel is used in, using MongoDB. If there are less than 500 documents, that sublabel is removed and the database is updated. If an upper label does not contain any sublabels, it is also removed. The final structure of the collection is printed out.

```
In [3]: kept_sublabels = {'building': ['name', 'level', 'levels', 'min_level', 'material', 'levels:underground'],
                        'maxspeed': ['type', 'ype'],
                        'name': ['cy', 'eo', 'ru', 'sr', 'uk', 'zh', 'en', 'zh_pinyin', 'he', 'de'],
                        'service': ['bicycle:pump', 'bicycle:chain_tool'], 'access': ['conditional'],
                        'source': ['crossing', 'addr', 'name', 'phone', 'population', 'maxwidth', 'ref', 'detail', 'info',
                                   'location', 'start_date', 'fhrrs:id', 'opening_hours', 'housenumber', 'postcode', 'access',
                                   'ele', 'cost', 'taxon', 'database', 'position', 'geometry', 'maxspeed', 'lit', 'oneway',
                                   'traffic_calming', 'designation', 'operator', 'width', 'bus:backward', 'taxi:backward',
                                   'bicycle:backward', 'tourist_bus:backward', 'occupier', 'description', 'highway', 'noname',
                                   'hgv', 'outline', 'maxspeed:date', 'addr:housenumber', 'ons_code', 'wifi', 'pkey',
                                   'bridge', 'addr:postcode', 'tracktype', 'height'],
                        'address': ['street', 'postcode', 'housenumber', 'housename', 'full', 'city', 'country',
                                   'interpolation', 'flat', 'flats', 'place', 'town'],
                        'ref': ['university_of_cambridge', 'observado']}

# Remove sublabels used in less than 500 documents
removed_sub = {}
kept_sub = {}
for label in kept_sublabels.keys():
    for sublabel in kept_sublabels[label]:
        pipeline = [{"$group": {"_id": " ".join(["$", label, ".", sublabel]), "count": {"$sum": 1}}},
                    {"$match": {"_id": None}}]
        result = list(db.cambridge.aggregate(pipeline))
        if len(result) > 0:
            n = result[0]["count"]
            if n >= N - 500:
                db.cambridge.update({}, {"$unset": {" ".join([label, '.', sublabel]): ""}}, multi=True)
            try:
                removed_sub[label].append(sublabel)
            except:
                removed_sub[label] = [sublabel]
        else:
            try:
                kept_sub[label].append(sublabel)
            except:
                kept_sub[label] = [sublabel]

# Remove the upper labels that no longer have sublabels
for label in removed_sub.keys():
    if label not in kept_sub.keys():
        db.cambridge.update({}, {"$unset": {label: ""}}, multi=True)

# Print the final structure of the collection
final_labels = {}
for label in kept:
    if label in kept_sub:
        final_labels[label] = kept_sub[label]
    elif label not in removed_sub:
        final_labels[label] = None

print "The final structure of the collection is:"
pprint.pprint(final_labels)
```

The final structure of the collection is:

```
{'address': ['street',
             'postcode',
             'housenumber',
             'housename',
             'city',
             'country',
             'interpolation'],
 'amenity': None,
 'barrier': None,
 'entrance': None,
 'foot': None,
 'highway': None,
 'landuse': None,
 'natural': None,
 'operator': None,
 'source': ['name']}
```

## Remove postcodes that do not start with "CB"

All postcodes in Cambridge start with CB. Listing the postcodes in the collection, documents with a postcode of "SG8 5TF" is discovered and should be removed since it is for a place in Royston and the Stevenage postcode area. Since there are only two documents in the collection with this postcode, it is an error and not an approach to extend the collection the surrounding area. Additionally the postcode "CB1" is incomplete - there should be a second set of three characters. Documents with this are also removed.

```
In [4]: db.cambridge.remove({"address.postcode": "CB1"})
db.cambridge.remove({"address.postcode": "SG8 5TF"})
```

```
Out[4]: {'u'n': 0, 'u'ok': 1}
```

## Update Cities to Cambridge

Some cities were "cambridge" and not "Cambridge", overspecified to "Girton" or "South Cambridgeshire", or listed as "11". However, the other information does show that each entry is in Cambridge. Therefore, the "city" is updated to "Cambridge".

```
In [5]: db.cambridge.update({"address.city": "cambridge"}, {"$set": {"address.city": "Cambridge"}}, upsert=False,
      multi=True)
db.cambridge.update({"address.city": "South Cambridgeshire"}, {"$set": {"address.city": "Cambridge"}}, upsert=False,
      multi=True)
db.cambridge.update({"address.city": "Girton"}, {"$set": {"address.city": "Cambridge"}}, upsert=False, multi=True)
db.cambridge.update({"address.city": "11"}, {"$set": {"address.city": "Cambridge"}}, upsert=False, multi=True)
```

```
Out[5]: {'u'n': 0, 'u'nModified': 0, 'u'ok': 1, 'updatedExisting': False}
```

## Pare down barrier, entrance, highway, landuse, and operator

Some entries for these sublabels were the same, but in a different format. They were updated to be more consistent.

```

In [6]: # Pare down barrier
pipeline = [{"_id": "$barrier", "count": {"$sum": 1}},
            {"$sort": {"_id": -1}}]
result = list(db.cambridge.aggregate(pipeline))
pprint.pprint(result)

db.cambridge.update({"barrier": "fence;wall"}, {"$set": {"barrier": "fence"}}, upsert=False, multi=True)
db.cambridge.update({"barrier": "fence;wall"}, {"$set": {"barrier": "fence"}}, upsert=False, multi=True)
db.cambridge.update({"barrier": "fedr"}, {"$set": {"barrier": None}}, upsert=False, multi=True)
db.cambridge.update({"barrier": "bollards"}, {"$set": {"barrier": "bollard"}}, upsert=False, multi=True)

# Pare down entrance
pipeline = [{"_id": "$entrance", "count": {"$sum": 1}},
            {"$sort": {"_id": -1}}]
result = list(db.cambridge.aggregate(pipeline))
pprint.pprint(result)

db.cambridge.update({"entrance": "secondary_entrance"}, {"$set": {"entrance": "secondary"}}, upsert=False,
                    multi=True)
db.cambridge.update({"entrance": "main_entrance; porters"}, {"$set": {"entrance": "main_entrance;porters"}},
                    upsert=False, multi=True)
db.cambridge.update({"entrance": "porters;main_entrance"}, {"$set": {"entrance": "main_entrance;porters"}},
                    upsert=False, multi=True)
db.cambridge.update({"entrance": "main"}, {"$set": {"entrance": "main_entrance"}}, upsert=False, multi=True)
db.cambridge.update({"entrance": "emegency"}, {"$set": {"entrance": "emergency"}}, upsert=False, multi=True)
db.cambridge.update({"entrance": "main_entrance;porters;"}, {"$set": {"entrance": "main_entrance;porters"}},
                    upsert=False, multi=True)

# Pare down highway
pipeline = [{"_id": "$highway", "count": {"$sum": 1}},
            {"$sort": {"_id": -1}}]
result = list(db.cambridge.aggregate(pipeline))
pprint.pprint(result)

db.cambridge.update({"highway": "bus_stand"}, {"$set": {"highway": "bus_stop"}}, upsert=False, multi=True)

# Pare down landuse
pipeline = [{"_id": "$barrier", "count": {"$sum": 1}},
            {"$sort": {"_id": -1}}]
result = list(db.cambridge.aggregate(pipeline))
pprint.pprint(result)
db.cambridge.update({"landuse": "institutional"}, {"$set": {"landuse": "institutional"}}, upsert=False,
                    multi=True)

# Pare down operator
pipeline = [{"_id": "$operator", "count": {"$sum": 1}},
            {"$sort": {"_id": -1}}]
result = list(db.cambridge.aggregate(pipeline))
pprint.pprint(result)
db.cambridge.update({"operator": "YourSpace"}, {"$set": {"operator": "Your Space Apartments"}}, upsert=False,
                    multi=True)
db.cambridge.update({"operator": "Your Space"}, {"$set": {"operator": "Your Space Apartments"}}, upsert=False,
                    multi=True)
db.cambridge.update({"operator": "Trinity College"},
                    {"$set": {"operator": "Trinity College (University of Cambridge)"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "St John's College"},
                    {"$set": {"operator": "St John's College (University of Cambridge)"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "Lucy Cavendish College"},
                    {"$set": {"operator": "Lucy Cavendish College (University of Cambridge)"}}, upsert=False,
                    multi=True)
db.cambridge.update({"operator": "Lloyds"}, {"$set": {"operator": "Lloyds TSB"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "King's College"},
                    {"$set": {"operator": "King's College (University of Cambridge)"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "King's College (University Of Cambridge)"},
                    {"$set": {"operator": "King's College (University of Cambridge)"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "Needham Institute"}, {"$set": {"operator": "Needham Research Institute"}},
                    upsert=False, multi=True)
db.cambridge.update({"operator": "Gonville and Caius College (University of Cambridge)"},
                    {"$set": {"operator": "Gonville & Caius College (University of Cambridge)"}}, upsert=False,
                    multi=True)
db.cambridge.update({"operator": "EDF"}, {"$set": {"operator": "EDF Energy"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "Clare College"},
                    {"$set": {"operator": "Clare College (University of Cambridge)"}}, upsert=False, multi=True)
db.cambridge.update({"operator": "Christ's College"},
                    {"$set": {"operator": "Christ's College (University of Cambridge)"}}, upsert=False, multi=True)

[{u'_id': u'yes', u'count': 286},
 {u'_id': u'wall', u'count': 669},
 {u'_id': u'tree', u'count': 1},
 {u'_id': u'swing_gate', u'count': 1},
 {u'_id': u'stile', u'count': 6},
 {u'_id': u'steps', u'count': 1},
 {u'_id': u'rising_kerb', u'count': 19},
 {u'_id': u'ramp', u'count': 1},
 {u'_id': u'railing', u'count': 19},

```

```

{u'_id': u'pram_handles', u'count': 2},
{u'_id': u'lift_gate', u'count': 59},
{u'_id': u'kissing_gate', u'count': 12},
{u'_id': u'kerb', u'count': 34},
{u'_id': u'height_restrictor', u'count': 1},
{u'_id': u'hedge', u'count': 490},
{u'_id': u'gravestone', u'count': 1},
{u'_id': u'gate', u'count': 807},
{u'_id': u'full-height_turnstile', u'count': 1},
{u'_id': u'fence', u'count': 5405},
{u'_id': u'entrance', u'count': 155},
{u'_id': u'ditch', u'count': 1},
{u'_id': u'cycle_barrier', u'count': 39},
{u'_id': u'chicane', u'count': 2},
{u'_id': u'cattle_grid', u'count': 36},
{u'_id': u'car_trap', u'count': 9},
{u'_id': u'boom', u'count': 1},
{u'_id': u'bollard', u'count': 186},
{u'_id': u'block', u'count': 1},
{u'_id': u'barrier', u'count': 3},
{u'_id': None, u'count': 298180}]
[ {u'_id': u'yes', u'count': 273},
  {u'_id': u'unused', u'count': 1},
  {u'_id': u'staff', u'count': 9},
  {u'_id': u'site_entrance', u'count': 5},
  {u'_id': u'service', u'count': 57},
  {u'_id': u'secondary', u'count': 428},
  {u'_id': u'residence;porters', u'count': 1},
  {u'_id': u'residence', u'count': 969},
  {u'_id': u'private', u'count': 201},
  {u'_id': u'primary_entrance', u'count': 1},
  {u'_id': u'porters;main_entrance;', u'count': 9},
  {u'_id': u'porters', u'count': 38},
  {u'_id': u'plant', u'count': 1},
  {u'_id': u'parking', u'count': 7},
  {u'_id': u'other', u'count': 7},
  {u'_id': u'office', u'count': 1},
  {u'_id': u'main_entrance;porters', u'count': 1},
  {u'_id': u'main_entrance', u'count': 608},
  {u'_id': u'locked', u'count': 2},
  {u'_id': u'internal', u'count': 4},
  {u'_id': u'home', u'count': 1},
  {u'_id': u'goods', u'count': 45},
  {u'_id': u'gate', u'count': 1},
  {u'_id': u'fire_exit', u'count': 169},
  {u'_id': u'exit', u'count': 10},
  {u'_id': u'emergency', u'count': 5},
  {u'_id': u'disabled', u'count': 3},
  {u'_id': u'archway', u'count': 2},
  {u'_id': None, u'count': 303569}]
[ {u'_id': u'unclassified', u'count': 219},
  {u'_id': u'turning_circle', u'count': 87},
  {u'_id': u'trunk_link', u'count': 41},
  {u'_id': u'trunk', u'count': 41},
  {u'_id': u'traffic_signals', u'count': 163},
  {u'_id': u'traffic_calming', u'count': 11},
  {u'_id': u'track', u'count': 172},
  {u'_id': u'tertiary', u'count': 176},
  {u'_id': u'street_lamp', u'count': 25},
  {u'_id': u'steps', u'count': 872},
  {u'_id': u'spiral_staircase', u'count': 29},
  {u'_id': u'speed_camera', u'count': 8},
  {u'_id': u'service', u'count': 3024},
  {u'_id': u'secondary', u'count': 47},
  {u'_id': u'road', u'count': 2},
  {u'_id': u'residential', u'count': 1685},
  {u'_id': u'proposed', u'count': 2},
  {u'_id': u'primary_link', u'count': 12},
  {u'_id': u'primary', u'count': 281},
  {u'_id': u'pedestrian', u'count': 166},
  {u'_id': u'path', u'count': 99},
  {u'_id': u'passing_place', u'count': 1},
  {u'_id': u'motorway_link', u'count': 14},
  {u'_id': u'motorway_junction', u'count': 10},
  {u'_id': u'motorway', u'count': 19},
  {u'_id': u'mini_roundabout', u'count': 31},
  {u'_id': u'milestone', u'count': 2},
  {u'_id': u'living_street', u'count': 15},
  {u'_id': u'footway', u'count': 3756},
  {u'_id': u'cycleway', u'count': 679},
  {u'_id': u'crossing', u'count': 231},
  {u'_id': u'construction', u'count': 20},
  {u'_id': u'bus_stop', u'count': 541},
  {u'_id': u'bus_guideway', u'count': 14},
  {u'_id': u'bridleway', u'count': 26},
  {u'_id': None, u'count': 293907}]

```

```
[{'u_id': 'u\'yes\'', 'u_count': 286},
{'u_id': 'u\'wall\'', 'u_count': 669},
{'u_id': 'u\'tree\'', 'u_count': 1},
{'u_id': 'u\'swing_gate\'', 'u_count': 1},
{'u_id': 'u\'stile\'', 'u_count': 6},
{'u_id': 'u\'steps\'', 'u_count': 1},
{'u_id': 'u\'rising_kerb\'', 'u_count': 19},
{'u_id': 'u\'ramp\'', 'u_count': 1},
{'u_id': 'u\'railing\'', 'u_count': 19},
{'u_id': 'u\'pram_handles\'', 'u_count': 2},
{'u_id': 'u\'lift_gate\'', 'u_count': 59},
{'u_id': 'u\'kissing_gate\'', 'u_count': 12},
{'u_id': 'u\'kerb\'', 'u_count': 34},
{'u_id': 'u\'height_restrictor\'', 'u_count': 1},
{'u_id': 'u\'hedge\'', 'u_count': 490},
{'u_id': 'u\'gravestone\'', 'u_count': 1},
{'u_id': 'u\'gate\'', 'u_count': 807},
{'u_id': 'u\'full-height_turnstile\'', 'u_count': 1},
{'u_id': 'u\'fence\'', 'u_count': 5405},
{'u_id': 'u\'entrance\'', 'u_count': 155},
{'u_id': 'u\'ditch\'', 'u_count': 1},
{'u_id': 'u\'cycle_barrier\'', 'u_count': 39},
{'u_id': 'u\'chicane\'', 'u_count': 2},
{'u_id': 'u\'cattle_grid\'', 'u_count': 36},
{'u_id': 'u\'car_trap\'', 'u_count': 9},
{'u_id': 'u\'boom\'', 'u_count': 1},
{'u_id': 'u\'bollard\'', 'u_count': 186},
{'u_id': 'u\'block\'', 'u_count': 1},
{'u_id': 'u\'barrier\'', 'u_count': 3},
{'u_id': 'None', 'u_count': 298180}]
[{'u_id': 'u\'Your Space Apartments\'', 'u_count': 5},
{'u_id': 'u\'YMCA\'', 'u_count': 1},
{'u_id': 'u\'Wolfson College (University of Cambridge)\'', 'u_count': 160},
{'u_id': 'u\'Whitbread plc\'', 'u_count': 2},
{'u_id': 'u\'Westcott House\'', 'u_count': 7},
{'u_id': 'u\'Wesley House\'', 'u_count': 6},
{'u_id': 'u\'Vue Cinemas\'', 'u_count': 1},
{'u_id': 'u\'University of Cambridge\'', 'u_count': 1397},
{'u_id': 'u\'United Reformed Church Westminster College\'', 'u_count': 1},
{'u_id': 'u\'UK Power Networks\'', 'u_count': 9},
{'u_id': 'u\'Tyndale House\'', 'u_count': 1},
{'u_id': 'u\'Trinity Hall (University of Cambridge)\'', 'u_count': 147},
{'u_id': 'u\'Trinity College (University of Cambridge)\'', 'u_count': 238},
{'u_id': 'u\'Travelodge\'', 'u_count': 4},
{'u_id': 'u\'The Junction\'', 'u_count': 1},
{'u_id': 'u\'The Hospital Management Trust\'', 'u_count': 1},
{'u_id': 'u\'The Co-operative Group\'', 'u_count': 2},
{'u_id': 'u\'The Cambridge Union Society\'', 'u_count': 1},
{'u_id': 'u\'The Busway (Cambs CC)\'', 'u_count': 3},
{'u_id': 'u\'Tesco Express\'', 'u_count': 1},
{'u_id': 'u\'Tesco\'', 'u_count': 2},
{'u_id': 'u\'TLC Group\'', 'u_count': 1},
{'u_id': 'u\'Subway\'', 'u_count': 1},
{'u_id': 'u\'Streetcar\'', 'u_count': 3},
{'u_id': 'u\'Steve\'', 'u_count': 1},
{'u_id': 'u\'Station Cycles\'', 'u_count': 2},
{'u_id': 'u\'Starbucks\'', 'u_count': 2},
{'u_id': 'u\'Stagecoach\'', 'u_count': 38},
{'u_id': 'u\'St John\'s Innovation Park"', 'u_count': 1},
{'u_id': 'u\'St John\'s College School"', 'u_count': 6},
{'u_id': 'u\'St John\'s College (University of Cambridge)\'', 'u_count': 286},
{'u_id': 'u\'St Edmund\'s College (University of Cambridge)\'', 'u_count': 86},
{'u_id': 'u\'St Catharine\'s College (University of Cambridge)\'', 'u_count': 135},
{'u_id': 'u\'St Andrew the Great Cambridge (CofE parish)\'', 'u_count': 1},
{'u_id': 'u\'Spar\'', 'u_count': 1},
{'u_id': 'u\'Sidney Sussex College (University of Cambridge)\'', 'u_count': 92},
{'u_id': 'u\'Shell\'', 'u_count': 1},
{'u_id': 'u\'Selwyn College (University of Cambridge)\'', 'u_count': 180},
{'u_id': 'u\'Schlumberger\'', 'u_count': 2},
{'u_id': 'u\'Savills\'', 'u_count': 1},
{'u_id': 'u\'Salvation Army\'', 'u_count': 1},
{'u_id': 'u\'Sainsbury\'s Local"', 'u_count': 1},
{'u_id': 'u\'Sainsbury\'s"', 'u_count': 2},
{'u_id': 'u\'Ryman\'', 'u_count': 1},
{'u_id': 'u\'Royal Society of Chemistry\'', 'u_count': 1},
{'u_id': 'u\'Royal Mail\'', 'u_count': 163},
{'u_id': 'u\'Robinson College (University of Cambridge)\'', 'u_count': 143},
{'u_id': 'u\'Riverside ECHG\'', 'u_count': 5},
{'u_id': 'u\'Ridley Hall\'', 'u_count': 19},
{'u_id': 'u\'Red Lion\'', 'u_count': 1},
{'u_id': 'u\'Ranc Care Homes\'', 'u_count': 1},
{'u_id': 'u\'Queens\' College (University of Cambridge)\'', 'u_count': 171},
{'u_id': 'u\'Pret a Manger\'', 'u_count': 1},
{'u_id': 'u\'Peterhouse (University of Cambridge)\'', 'u_count': 193},
{'u_id': 'u\'Pembroke College (University of Cambridge)\'', 'u_count': 86},
```

```
{u'_id': u'Parkside Community College', u'count': 1},
{u'_id': u'Parcelforce', u'count': 1},
{u'_id': u'Palamas House', u'count': 7},
{u'_id': u'Open University', u'count': 1},
{u'_id': u'Nicholson's", u'count': 1},
{u'_id': u'Newnham College (University of Cambridge)', u'count': 148},
{u'_id': u'Needham Research Institute', u'count': 3},
{u'_id': u'Nationwide', u'count': 1},
{u'_id': u'National Express; Trainseurope; Hertz', u'count': 1},
{u'_id': u'National Express', u'count': 3},
{u'_id': u'National Blood Service', u'count': 2},
{u'_id': u'NatWest', u'count': 1},
{u'_id': u"Nando's", u'count': 1},
{u'_id': u'NIAB', u'count': 8},
{u'_id': u'NHS', u'count': 2},
{u'_id': u'NCP', u'count': 1},
{u'_id': u'Murray Edwards College (University of Cambridge)', u'count': 106},
{u'_id': u'Ministry of Defence', u'count': 13},
{u'_id': u'Milton FC / CC', u'count': 1},
{u'_id': u'Milton Brewery', u'count': 2},
{u'_id': u'Medical Research Council', u'count': 3},
{u'_id': u'McDonald's", u'count': 2},
{u'_id': u'Marks & Spencer', u'count': 2},
{u'_id': u'Margaret Beaufort Institute of Theology', u'count': 3},
{u'_id': u'Magdalene College (University of Cambridge)', u'count': 205},
{u'_id': u'Machine Mart', u'count': 1},
{u'_id': u'Lucy Cavendish College (University of Cambridge)', u'count': 114},
{u'_id': u'Lloyds TSB', u'count': 5},
{u'_id': u'Lifecraft', u'count': 1},
{u'_id': u'Leys and St Faith\u2019s schools Foundation', u'count': 21},
{u'_id': u'LNWR', u'count': 1},
{u'_id': u'Korean', u'count': 1},
{u'_id': u"King's College School", u'count': 11},
{u'_id': u"King's College (University of Cambridge)", u'count': 81},
{u'_id': u'KFC', u'count': 1},
{u'_id': u'Justin Drake', u'count': 1},
{u'_id': u'Jesus College (University of Cambridge)', u'count': 142},
{u'_id': u'JD Wetherspoon', u'count': 2},
{u'_id': u'Intercontinental Hotels Group', u'count': 1},
{u'_id': u'Institute of Astronomy (University of Cambridge)', u'count': 2},
{u'_id': u'IUCN', u'count': 2},
{u'_id': u'Hungry Horse', u'count': 1},
{u'_id': u'Hundred Houses Society', u'count': 6},
{u'_id': u'Hughes Hall (University of Cambridge)', u'count': 28},
{u'_id': u'Housing 21', u'count': 1},
{u'_id': u'Homerton College (University of Cambridge)', u'count': 131},
{u'_id': u'Holiday Inn', u'count': 2},
{u'_id': u'Hilton Worldwide', u'count': 1},
{u'_id': u'Hills Road Sports & Tennis Centre', u'count': 1},
{u'_id': u'Highways Agency', u'count': 82},
{u'_id': u'HSBC', u'count': 2},
{u'_id': u'HM Government', u'count': 2},
{u'_id': u'Greene King', u'count': 1},
{u'_id': u'Green Man', u'count': 2},
{u'_id': u'Granta Housing Society', u'count': 2},
{u'_id': u'Gonville & Caius College (University of Cambridge)',
  u'count': 150},
{u'_id': u'Girton College (University of Cambridge)', u'count': 139},
{u'_id': u'Forestdale Hotels', u'count': 1},
{u'_id': u'Fitzwilliam College (University of Cambridge)', u'count': 85},
{u'_id': u'Felix Hotel', u'count': 2},
{u'_id': u'Excelcare Holdings', u'count': 2},
{u'_id': u'Excelcare', u'count': 2},
{u'_id': u'Excel Care', u'count': 1},
{u'_id': u'Evangelical Lutheran Church of England', u'count': 1},
{u'_id': u'Emmanuel College (University of Cambridge)', u'count': 172},
{u'_id': u'Embassy CES', u'count': 1},
{u'_id': u'Edward Storey Foundation', u'count': 2},
{u'_id': u'East Of England Ambulance Service NHS Trust', u'count': 1},
{u'_id': u'EDF Energy', u'count': 3},
{u'_id': u'Downing College (University of Cambridge)', u'count': 65},
{u'_id': u'De Vere', u'count': 1},
{u'_id': u'Darwin College (University of Cambridge)', u'count': 83},
{u'_id': u'Costa', u'count': 1},
{u'_id': u'Corpus Christi College (University of Cambridge)', u'count': 175},
{u'_id': u'Co-Op', u'count': 1},
{u'_id': u'Clare Hall (University of Cambridge)', u'count': 141},
{u'_id': u'Clare College (University of Cambridge)', u'count': 220},
{u'_id': u'City Sightseeing', u'count': 17},
{u'_id': u'City Screen Limited', u'count': 1},
{u'_id': u'Cineworld', u'count': 1},
{u'_id': u'Churchill College (University of Cambridge)', u'count': 316},
{u'_id': u'Church of England', u'count': 1},
{u'_id': u"Christ's College (University of Cambridge)", u'count': 133},
{u'_id': u'Cambridgeshire County Council', u'count': 28},
```



```
{u'_id': u'Cambridgeshire Constabulary', u'count': 1},
{u'_id': u'Cambridge University Hospitals NHS Foundation Trust',
 u'count': 14},
{u'_id': u'Cambridge University Catholic Society', u'count': 1},
{u'_id': u'Cambridge Science Park', u'count': 1},
{u'_id': u'Cambridge Regional College', u'count': 12},
{u'_id': u'Cambridge Preservation Society', u'count': 1},
{u'_id': u'Cambridge Leisure', u'count': 1},
{u'_id': u'Cambridge International College', u'count': 1},
{u'_id': u'Cambridge Cyrenians', u'count': 1},
{u'_id': u'Cambridge City Council', u'count': 35},
{u'_id': u'Cambridge Centre for Sixth Form Studies', u'count': 1},
{u'_id': u'Cambridge Business Park', u'count': 1},
{u'_id': u'Caffe Nero', u'count': 1},
{u'_id': u'CRM Students', u'count': 2},
{u'_id': u'CRM Student Accommodation Management', u'count': 1},
{u'_id': u'CPPF', u'count': 1},
{u'_id': u'CHS Homes', u'count': 2},
{u'_id': u'CHS Group', u'count': 5},
{u'_id': u'CCSS', u'count': 1},
{u'_id': u'CCDC', u'count': 1},
{u'_id': u'CATS,CSVPA', u'count': 1},
{u'_id': u'Browns', u'count': 1},
{u'_id': u'British Antarctic Survey', u'count': 8},
{u'_id': u'Boots UK', u'count': 1},
{u'_id': u'Birdlife International', u'count': 1},
{u'_id': u'Best Western', u'count': 1},
{u'_id': u'Bella Italia', u'count': 1},
{u'_id': u'Barclays Bank PLC', u'count': 5},
{u'_id': u'Bank Machine', u'count': 1},
{u'_id': u'BT', u'count': 2},
{u'_id': u'BP', u'count': 2},
{u'_id': u'Arundel House Hotel', u'count': 1},
{u'_id': u'Argyle Street Housing Cooperative', u'count': 1},
{u'_id': u'Anglian Water', u'count': 29},
{u'_id': u'Anglia Ruskin University', u'count': 25},
{u'_id': u'Anchor Trust', u'count': 2},
{u'_id': u'Age Concern', u'count': 1},
{u'_id': u'Adnams', u'count': 1},
{u'_id': u'Abbeyfield UK', u'count': 1},
{u'_id': u'Abbeyfield Cambridgeshire Vietnamese Society Ltd', u'count': 1},
{u'_id': u'Abbey', u'count': 1},
{u'_id': u'AVEVA Group plc', u'count': 1},
{u'_id': None, u'count': 299723}]
```

```
Out[6]: {u'n': 0, u'nModified': 0, u'ok': 1, 'updatedExisting': False}
```

## Section 2. Overview of the Data

A statistical overview of the dataset with the MongoDB queries used to obtain such statistics are below.

```
In [7]: print "The size of 'cambridge_england.osm' is", os.stat("cambridge_england.osm").st_size / 1e6, "MB."
print "The size of 'cambridge_england.osm.json' is", os.stat("cambridge_england.osm.json").st_size / 1e6, "MB."
N2 = db.cambridge.find().count()
print "There are", N, "documents in the original set and", N2, "documents in the cleaned set."
pipeline = [{"$group": {"_id": "$created.user", "count": {"$sum": 1}}}]
print "There are", len(list(db.cambridge.aggregate(pipeline))), "unique users."
print "There are", db.cambridge.find({"type": "node"}).count(), "nodes."
print "There are", db.cambridge.find({"type": "way"}).count(), "ways."
pipeline = [{"$group": {"_id": "$created.user", "count": {"$sum": 1}}},
            {"$sort": {"count": -1}},
            {"$limit": 1}]
print list(db.cambridge.aggregate(pipeline))[0]['_id'], "contributed the most to this collection with", \
        list(db.cambridge.aggregate(pipeline))[0]['count'], "documents."
pipeline = [{"$group": {"_id": "$created.user", "count": {"$sum": 1}}},
            {"$group": {"_id": "$count", "num_users": {"$sum": 1}}},
            {"$sort": {"_id": 1}},
            {"$limit": 1}]
print list(db.cambridge.aggregate(pipeline))[0]['num_users'], "users contributed once."
pipeline = [{"$group": {"_id": "$amenity", "count": {"$sum": 1}}},
            {"$match": {"_id": {"$ne": None}}},
            {"$sort": {"count": -1}},
            {"$limit": 5}]
print list(db.cambridge.aggregate(pipeline))[0]['_id'], list(db.cambridge.aggregate(pipeline))[1][
    "_id"], "are the top two amenities."
```

The size of 'cambridge\_england.osm' is 61.828774 MB.  
The size of 'cambridge\_england.osm.json' is 0.0 MB.  
There are 306428 documents in the original set and 306428 documents in the cleaned set.  
There are 453 unique users.  
There are 257058 nodes.  
There are 49351 ways.  
smb1001 contributed the most to this collection with 81443 documents.  
120 users contributed once.  
university bicycle\_parking are the top two amenities.

## Section 3. Additional Ideas

Some addresses have house names. It would be interesting to know if there's a certain postal code with the most and if there is an operator that is popular. Additionally, it would be interesting where the top amenities are located.

```
In [8]: # Find the top postcodes with housenames
pipeline = [{"$match": {"address.housename": {"$exists": True}}},
            {"$group": {"_id": "$address.postcode", "count": {"$sum": 1}}},
            {"$sort": {"count": -1}}]
pprint.pprint(list(db.cambridge.aggregate(pipeline)))

# Find the top operators with housenames
pipeline = [{"$match": {"address.housename": {"$exists": True}}},
            {"$group": {"_id": "$operator", "count": {"$sum": 1}}},
            {"$sort": {"count": -1}}]
pprint.pprint(list(db.cambridge.aggregate(pipeline)))

# Find the top amenities with postcodes and sort by postcode
pipeline = [{"$match": {"amenity": {"$ne": None}}},
            {"$match": {"amenity": {"$ne": "university"}}},
            {"$match": {"address.postcode": {"$ne": None}}},
            {"$group": {"_id": {"amenity": "$amenity",
                                "postcode": "$address.postcode"},
                        "count": {"$sum": 1}}},
            {"$sort": {"count": -1}},
            {"$group": {"_id": "$_id.amenity",
                        "info": {"$push": {
                            "postcode": "$_id.postcode",
                            "count": "$count"}},
                        "count": {"$sum": "$count"}}},
            {"$sort": {"count": -1}},
            {"$limit": 5}]
pprint.pprint(list(db.cambridge.aggregate(pipeline)))
```

```
[{'u_id': None, 'u_count': 200},
 {'u_id': u'CB4 1HG', 'u_count': 17},
 {'u_id': u'CB3 0EY', 'u_count': 11},
 {'u_id': u'CB1 2LJ', 'u_count': 9},
 {'u_id': u'CB5 8HU', 'u_count': 8},
 {'u_id': u'CB4 1HH', 'u_count': 8},
 {'u_id': u'CB1 2LG', 'u_count': 7},
 {'u_id': u'CB1 8RG', 'u_count': 7},
 {'u_id': u'CB3 9NF', 'u_count': 5},
 {'u_id': u'CB2 3QZ', 'u_count': 5},
 {'u_id': u'CB3 0AE', 'u_count': 5},
 {'u_id': u'CB4 1AJ', 'u_count': 5},
```

{u'\_id': u'CB2 8EX', u'count': 5},  
{u'\_id': u'CB1 7PH', u'count': 5},  
{u'\_id': u'CB2 1NS', u'count': 4},  
{u'\_id': u'CB3 0AJ', u'count': 4},  
{u'\_id': u'CB5 8HT', u'count': 4},  
{u'\_id': u'CB2 3EA', u'count': 4},  
{u'\_id': u'CB1 8QL', u'count': 4},  
{u'\_id': u'CB4 1ST', u'count': 4},  
{u'\_id': u'CB3 9NQ', u'count': 3},  
{u'\_id': u'CB1 7RU', u'count': 3},  
{u'\_id': u'CB4 3LG', u'count': 3},  
{u'\_id': u'CB4 3JD', u'count': 3},  
{u'\_id': u'CB2 1QJ', u'count': 3},  
{u'\_id': u'CB4 1AL', u'count': 3},  
{u'\_id': u'CB5 8AF', u'count': 3},  
{u'\_id': u'CB5 8AQ', u'count': 3},  
{u'\_id': u'CB4 3QS', u'count': 3},  
{u'\_id': u'CB4 1HJ', u'count': 3},  
{u'\_id': u'CB2 3QG', u'count': 3},  
{u'\_id': u'CB2 8BG', u'count': 3},  
{u'\_id': u'CB3 9NH', u'count': 2},  
{u'\_id': u'CB3 0NL', u'count': 2},  
{u'\_id': u'CB1 1DT', u'count': 2},  
{u'\_id': u'CB2 7EJ', u'count': 2},  
{u'\_id': u'CB4 3PT', u'count': 2},  
{u'\_id': u'CB3 0DL', u'count': 2},  
{u'\_id': u'CB5 8DW', u'count': 2},  
{u'\_id': u'CB3 9BL', u'count': 2},  
{u'\_id': u'CB5 8EG', u'count': 2},  
{u'\_id': u'CB4 1BP', u'count': 2},  
{u'\_id': u'CB5 8HN', u'count': 2},  
{u'\_id': u'CB1 2JB', u'count': 2},  
{u'\_id': u'CB2 3RA', u'count': 2},  
{u'\_id': u'CB1 3NF', u'count': 2},  
{u'\_id': u'CB2 3RF', u'count': 2},  
{u'\_id': u'CB1 1JP', u'count': 2},  
{u'\_id': u'CB3 9NB', u'count': 2},  
{u'\_id': u'CB4 1LY', u'count': 2},  
{u'\_id': u'CB4 1NR', u'count': 2},  
{u'\_id': u'CB3 9DQ', u'count': 2},  
{u'\_id': u'CB4 1DT', u'count': 2},  
{u'\_id': u'CB2 3RQ', u'count': 2},  
{u'\_id': u'CB2 1QP', u'count': 2},  
{u'\_id': u'CB2 3EB', u'count': 2},  
{u'\_id': u'CB1 7DY', u'count': 2},  
{u'\_id': u'CB4 3QU', u'count': 2},  
{u'\_id': u'CB4 1PA', u'count': 2},  
{u'\_id': u'CB2 3DZ', u'count': 2},  
{u'\_id': u'CB4 3BB', u'count': 2},  
{u'\_id': u'CB4 3QW', u'count': 2},  
{u'\_id': u'CB2 8PS', u'count': 2},  
{u'\_id': u'CB2 1QW', u'count': 2},  
{u'\_id': u'CB2 1NF', u'count': 2},  
{u'\_id': u'CB2 3DY', u'count': 2},  
{u'\_id': u'CB4 1GL', u'count': 2},  
{u'\_id': u'CB1 3ND', u'count': 2},  
{u'\_id': u'CB2 1AG', u'count': 2},  
{u'\_id': u'CB2 7EH', u'count': 2},  
{u'\_id': u'CB1 3AQ', u'count': 2},  
{u'\_id': u'CB1 2FJ', u'count': 1},  
{u'\_id': u'CB5 8SR', u'count': 1},  
{u'\_id': u'CB23 7PH', u'count': 1},  
{u'\_id': u'CB3 9HX', u'count': 1},  
{u'\_id': u'CB3 9NJ', u'count': 1},  
{u'\_id': u'CB1 2QR', u'count': 1},  
{u'\_id': u'CB3 9NT', u'count': 1},  
{u'\_id': u'CB4 3BW', u'count': 1},  
{u'\_id': u'CB4 3EH', u'count': 1},  
{u'\_id': u'CB3 9EG', u'count': 1},  
{u'\_id': u'CB4 3DD', u'count': 1},  
{u'\_id': u'CB3 9JZ', u'count': 1},  
{u'\_id': u'CB1 8PS', u'count': 1},  
{u'\_id': u'CB1 3RW', u'count': 1},  
{u'\_id': u'CB1 7ST', u'count': 1},  
{u'\_id': u'CB1 7SZ', u'count': 1},  
{u'\_id': u'CB1 8AP', u'count': 1},  
{u'\_id': u'CB2 8RS', u'count': 1},  
{u'\_id': u'CB1 1AL', u'count': 1},  
{u'\_id': u'CB2 1LY', u'count': 1},  
{u'\_id': u'CB2 1HX', u'count': 1},  
{u'\_id': u'CB2 1HF', u'count': 1},  
{u'\_id': u'CB2 1LR', u'count': 1},  
{u'\_id': u'CB2 1NE', u'count': 1},  
{u'\_id': u'CB5 8DN', u'count': 1},  
{u'\_id': u'CB2 8BW', u'count': 1},  
{u'\_id': u'CB2 8BB', u'count': 1},

{u'\_id': u'CB2 8RN', u'count': 1},  
{u'\_id': u'CB4 3QN', u'count': 1},  
{u'\_id': u'CB5 8RL', u'count': 1},  
{u'\_id': u'CB4 1JB', u'count': 1},  
{u'\_id': u'CB4 1AD', u'count': 1},  
{u'\_id': u'CB1 2FG', u'count': 1},  
{u'\_id': u'CB1 1JU', u'count': 1},  
{u'\_id': u'CB4 1JZ', u'count': 1},  
{u'\_id': u'CB3 0ET', u'count': 1},  
{u'\_id': u'CB1 7UB', u'count': 1},  
{u'\_id': u'CB4 3DU', u'count': 1},  
{u'\_id': u'CB4 3QX', u'count': 1},  
{u'\_id': u'CB1 8PW', u'count': 1},  
{u'\_id': u'CB2 1JB', u'count': 1},  
{u'\_id': u'CB2 1JA', u'count': 1},  
{u'\_id': u'CB2 8NY', u'count': 1},  
{u'\_id': u'CB4 2QR', u'count': 1},  
{u'\_id': u'CB3 9AQ', u'count': 1},  
{u'\_id': u'CB5 8AT', u'count': 1},  
{u'\_id': u'CB3 0HH', u'count': 1},  
{u'\_id': u'CB3 0JG', u'count': 1},  
{u'\_id': u'CB1 2FL', u'count': 1},  
{u'\_id': u'CB2 3RD', u'count': 1},  
{u'\_id': u'CB3 0DX', u'count': 1},  
{u'\_id': u'CB3 9DA', u'count': 1},  
{u'\_id': u'CB4 3RW', u'count': 1},  
{u'\_id': u'CB4 3AL', u'count': 1},  
{u'\_id': u'CB1 3YH', u'count': 1},  
{u'\_id': u'CB1 2HF', u'count': 1},  
{u'\_id': u'CB1 2HS', u'count': 1},  
{u'\_id': u'CB1 2HA', u'count': 1},  
{u'\_id': u'CB4 3HB', u'count': 1},  
{u'\_id': u'CB1 2BX', u'count': 1},  
{u'\_id': u'CB1 8QN', u'count': 1},  
{u'\_id': u'CB2 1QN', u'count': 1},  
{u'\_id': u'CB2 1GA', u'count': 1},  
{u'\_id': u'CB4 2GR', u'count': 1},  
{u'\_id': u'CB1 3ET', u'count': 1},  
{u'\_id': u'CB1 3BN', u'count': 1},  
{u'\_id': u'CB4 2WR', u'count': 1},  
{u'\_id': u'CB3 9DP', u'count': 1},  
{u'\_id': u'CB2 3QA', u'count': 1},  
{u'\_id': u'CB4 2HY', u'count': 1},  
{u'\_id': u'CB1 7TL', u'count': 1},  
{u'\_id': u'CB2 1RB', u'count': 1},  
{u'\_id': u'CB2 3EL', u'count': 1},  
{u'\_id': u'CB4 1BY', u'count': 1},  
{u'\_id': u'CB1 1JX', u'count': 1},  
{u'\_id': u'CB4 1DF', u'count': 1},  
{u'\_id': u'CB4 1LN', u'count': 1},  
{u'\_id': u'CB2 3EQ', u'count': 1},  
{u'\_id': u'CB1 3DE', u'count': 1},  
{u'\_id': u'CB3 0LU', u'count': 1},  
{u'\_id': u'CB2 3BU', u'count': 1},  
{u'\_id': u'CB4 1UU', u'count': 1},  
{u'\_id': u'CB4 1UN', u'count': 1},  
{u'\_id': u'CB4 2JE', u'count': 1},  
{u'\_id': u'CB4 2HX', u'count': 1},  
{u'\_id': u'CB1 3AE', u'count': 1},  
{u'\_id': u'CB4 1XQ', u'count': 1},  
{u'\_id': u'CB3 9HJ', u'count': 1},  
{u'\_id': u'CB1 2QG', u'count': 1},  
{u'\_id': u'CB4 3BY', u'count': 1},  
{u'\_id': u'CB3 0AH', u'count': 1},  
{u'\_id': u'CB2 7EB', u'count': 1},  
{u'\_id': u'CB4 3EE', u'count': 1},  
{u'\_id': u'CB4 1GR', u'count': 1},  
{u'\_id': u'CB4 1DL', u'count': 1},  
{u'\_id': u'CB2 1SJ', u'count': 1},  
{u'\_id': u'CB5 8DQ', u'count': 1},  
{u'\_id': u'CB4 1TU', u'count': 1},  
{u'\_id': u'CB5 8JJ', u'count': 1},  
{u'\_id': u'CB2 1RD', u'count': 1},  
{u'\_id': u'CB1 7PN', u'count': 1},  
{u'\_id': u'CB1 1ES', u'count': 1},  
{u'\_id': u'CB2 3EJ', u'count': 1},  
{u'\_id': u'CB4 1HB', u'count': 1},  
{u'\_id': u'CB4 3AB', u'count': 1},  
{u'\_id': u'CB1 2LZ', u'count': 1},  
{u'\_id': u'CB1 2LX', u'count': 1},  
{u'\_id': u'CB4 1NT', u'count': 1},  
{u'\_id': u'CB1 3DF', u'count': 1},  
{u'\_id': u'CB1 1BH', u'count': 1},  
{u'\_id': u'CB1 1DJ', u'count': 1},  
{u'\_id': u'CB2 1QS', u'count': 1},

```

{u'_id': u'CB1 7GX', u'count': 1},
{u'_id': u'CB1 1LB', u'count': 1},
{u'_id': u'CB2 8PQ', u'count': 1},
{u'_id': u'CB2 3EN', u'count': 1},
{u'_id': u'CB1 2FH', u'count': 1},
{u'_id': u'CB2 8NZ', u'count': 1},
{u'_id': u'CB2 1QH', u'count': 1},
{u'_id': u'CB2 3EH', u'count': 1},
{u'_id': u'CB4 3PH', u'count': 1},
{u'_id': u'CB1 1DU', u'count': 1},
{u'_id': u'CB2 3EG', u'count': 1},
{u'_id': u'CB4 1NZ', u'count': 1},
{u'_id': u'CB4 3AN', u'count': 1},
{u'_id': u'CB2 8ET', u'count': 1},
{u'_id': u'CB4 1HL', u'count': 1},
{u'_id': u'CB2 3HG', u'count': 1},
{u'_id': u'CB1 2EZ', u'count': 1},
{u'_id': u'CB2 8NX', u'count': 1},
{u'_id': u'CB2 3NH', u'count': 1},
{u'_id': u'CB1 8DB', u'count': 1},
{u'_id': u'CB2 1UF', u'count': 1},
{u'_id': u'CB1 7AL', u'count': 1},
{u'_id': u'CB4 1DU', u'count': 1},
{u'_id': u'CB4 3HP', u'count': 1},
{u'_id': u'CB2 2EG', u'count': 1},
{u'_id': u'CB4 1AF', u'count': 1},
{u'_id': u'CB3 0HD', u'count': 1},
{u'_id': u'CB2 1AB', u'count': 1},
{u'_id': u'CB2 1AA', u'count': 1},
{u'_id': u'CB4 1BW', u'count': 1},
{u'_id': u'CB4 0WS', u'count': 1},
{u'_id': u'CB1 1LQ', u'count': 1},
{u'_id': u'CB24 6AY', u'count': 1},
{u'_id': u'CB3 0BT', u'count': 1},
{u'_id': u'CB2 3ER', u'count': 1},
{u'_id': u'CB2 1PZ', u'count': 1},
{u'_id': u'CB1 2BY', u'count': 1},
{u'_id': u'CB1 1DR', u'count': 1},
{u'_id': u'CB4 1HA', u'count': 1},
{u'_id': u'CB4 3QA', u'count': 1},
{u'_id': u'CB1 1DY', u'count': 1},
{u'_id': u'CB4 1HF', u'count': 1},
{u'_id': u'CB1 2HN', u'count': 1},
{u'_id': u'CB2 8PH', u'count': 1},
{u'_id': u'CB1 2UB', u'count': 1},
{u'_id': u'CB3 0FA', u'count': 1},
{u'_id': u'CB5 8RG', u'count': 1},
{u'_id': u'CB4 3NP', u'count': 1},
{u'_id': u'CB24 6AT', u'count': 1},
{u'_id': u'CB2 8AE', u'count': 1},
{u'_id': u'CB1 3ER', u'count': 1},
{u'_id': u'CB5 8HQ', u'count': 1},
{u'_id': u'CB2 1PG', u'count': 1},
{u'_id': u'CB1 3BZ', u'count': 1},
{u'_id': u'CB4 0WX', u'count': 1},
{u'_id': u'CB2 3HW', u'count': 1},
{u'_id': u'CB4 1QU', u'count': 1},
{u'_id': u'CB5 8DT', u'count': 1},
{u'_id': u'CB1 2PY', u'count': 1},
{u'_id': u'CB4 0PP', u'count': 1},
{u'_id': u'CB1 3XB', u'count': 1},
{u'_id': u'CB5 8AD', u'count': 1},
{u'_id': u'CB2 1UB', u'count': 1},
{u'_id': u'CB2 1QY', u'count': 1},
{u'_id': u'CB2 3RH', u'count': 1},
{u'_id': u'CB3 0JB', u'count': 1},
{u'_id': u'CB2 1UA', u'count': 1},
{u'_id': u'CB4 1HQ', u'count': 1},
{u'_id': u'CB2 3LL', u'count': 1}]
[{u'_id': None, u'count': 514},
{u'_id': u'University of Cambridge', u'count': 53},
{u'_id': u'Clare College (University of Cambridge)', u'count': 6},
{u'_id': u'Selwyn College (University of Cambridge)', u'count': 4},
{u'_id': u'Anglia Ruskin University', u'count': 4},
{u'_id': u'Riverside ECHG', u'count': 3},
{u'_id': u'CRM Students', u'count': 2},
{u'_id': u'Murray Edwards College (University of Cambridge)', u'count': 2},
{u'_id': u'Newnham College (University of Cambridge)', u'count': 2},
{u'_id': u'Girton College (University of Cambridge)', u'count': 2},
{u'_id': u'Christ's College (University of Cambridge)", u'count': 2},
{u'_id': u'British Antarctic Survey', u'count': 1},
{u'_id': u'Clare Hall (University of Cambridge)', u'count': 1},
{u'_id': u'St John's College (University of Cambridge)", u'count': 1},
{u'_id': u'Evangelical Lutheran Church of England', u'count': 1},
{u'_id': u'Travelodge', u'count': 1},
{u'_id': u'St Andrew the Great Cambridge (CofE parish)', u'count': 1},

```

```

{u'_id': u'King's College (University of Cambridge)', u'count': 1},
{u'_id': u'CCSS', u'count': 1},
{u'_id': u'The Junction', u'count': 1},
{u'_id': u'Whitbread plc', u'count': 1},
{u'_id': u'Trinity Hall (University of Cambridge)', u'count': 1},
{u'_id': u'Peterhouse (University of Cambridge)', u'count': 1},
{u'_id': u'CHS Group', u'count': 1},
{u'_id': u'Justin Drake', u'count': 1},
{u'_id': u'The Co-operative Group', u'count': 1},
{u'_id': u'Hundred Houses Society', u'count': 1},
{u'_id': u'Cambridge City Council', u'count': 1},
{u'_id': u'Magdalene College (University of Cambridge)', u'count': 1},
{u'_id': u'Tesco Express', u'count': 1},
{u'_id': u'Trinity College (University of Cambridge)', u'count': 1},
{u'_id': u'Gonville & Caius College (University of Cambridge)', u'count': 1},
{u'_id': u'Costa', u'count': 1}]
[{u'_id': u'restaurant',
  u'count': 104,
  u'info': [{u'count': 8, u'postcode': u'CB2 1DP'},
    {u'count': 6, u'postcode': u'CB1 2AS'},
    {u'count': 5, u'postcode': u'CB2 1AB'},
    {u'count': 4, u'postcode': u'CB3 0AH'},
    {u'count': 3, u'postcode': u'CB2 1TB'},
    {u'count': 3, u'postcode': u'CB2 1DB'},
    {u'count': 3, u'postcode': u'CB2 3AR'},
    {u'count': 3, u'postcode': u'CB2 1SJ'},
    {u'count': 3, u'postcode': u'CB1 2LD'},
    {u'count': 2, u'postcode': u'CB3 9EY'},
    {u'count': 2, u'postcode': u'CB2 1NT'},
    {u'count': 2, u'postcode': u'CB4 1JY'},
    {u'count': 2, u'postcode': u'CB4 2JQ'},
    {u'count': 2, u'postcode': u'CB2 3JX'},
    {u'count': 2, u'postcode': u'CB2 3QN'},
    {u'count': 2, u'postcode': u'CB1 2BD'},
    {u'count': 2, u'postcode': u'CB5 8LG'},
    {u'count': 2, u'postcode': u'CB2 8NX'},
    {u'count': 2, u'postcode': u'CB2 1UF'},
    {u'count': 2, u'postcode': u'CB1 1LH'},
    {u'count': 2, u'postcode': u'CB2 1QA'},
    {u'count': 2, u'postcode': u'CB1 1DG'},
    {u'count': 2, u'postcode': u'CB1 1BG'},
    {u'count': 2, u'postcode': u'CB3 0AF'},
    {u'count': 2, u'postcode': u'CB2 3NB'},
    {u'count': 2, u'postcode': u'CB2 1UJ'},
    {u'count': 1, u'postcode': u'CB5 8EG'},
    {u'count': 1, u'postcode': u'CB4 1UY'},
    {u'count': 1, u'postcode': u'CB1 3NF'},
    {u'count': 1, u'postcode': u'CB3 0DL'},
    {u'count': 1, u'postcode': u'CB4 3JP'},
    {u'count': 1, u'postcode': u'CB4 3LF'},
    {u'count': 1, u'postcode': u'CB1 2AZ'},
    {u'count': 1, u'postcode': u'CB2 1AW'},
    {u'count': 1, u'postcode': u'CB2 1NW'},
    {u'count': 1, u'postcode': u'CB5 8WR'},
    {u'count': 1, u'postcode': u'CB3 0AD'},
    {u'count': 1, u'postcode': u'CB5 8AB'},
    {u'count': 1, u'postcode': u'CB2 3NE'},
    {u'count': 1, u'postcode': u'CB1 1LN'},
    {u'count': 1, u'postcode': u'CB2 3QF'},
    {u'count': 1, u'postcode': u'CB1 1HR'},
    {u'count': 1, u'postcode': u'CB5 8DZ'},
    {u'count': 1, u'postcode': u'CB2 1TW'},
    {u'count': 1, u'postcode': u'CB2 3QJ'},
    {u'count': 1, u'postcode': u'CB2 3PP'},
    {u'count': 1, u'postcode': u'CB1 1ER'},
    {u'count': 1, u'postcode': u'CB2 1UG'},
    {u'count': 1, u'postcode': u'CB2 3RF'},
    {u'count': 1, u'postcode': u'CB1 7DY'},
    {u'count': 1, u'postcode': u'CB2 1LA'},
    {u'count': 1, u'postcode': u'CB1 1DJ'},
    {u'count': 1, u'postcode': u'CB2 1QY'},
    {u'count': 1, u'postcode': u'CB1 3RX'},
    {u'count': 1, u'postcode': u'CB2 1PG'},
    {u'count': 1, u'postcode': u'CB2 8PB'},
    {u'count': 1, u'postcode': u'CB2 8EX'},
    {u'count': 1, u'postcode': u'CB4 1HA}]]},
{u'_id': u'pub',
  u'count': 74,
  u'info': [{u'count': 3, u'postcode': u'CB2 3AR'},
    {u'count': 3, u'postcode': u'CB1 1LN'},
    {u'count': 2, u'postcode': u'CB4 3AX'},
    {u'count': 2, u'postcode': u'CB2 3QN'},
    {u'count': 2, u'postcode': u'CB3 0AJ'},
    {u'count': 2, u'postcode': u'CB4 1NG'},
    {u'count': 2, u'postcode': u'CB5 8JE'},

```

```

    {u'count': 2, u'postcode': u'CB2 1DB'},
    {u'count': 2, u'postcode': u'CB3 9NQ'},
    {u'count': 2, u'postcode': u'CB3 9NF'},
    {u'count': 2, u'postcode': u'CB1 2PW'},
    {u'count': 1, u'postcode': u'CB4 1HA'},
    {u'count': 1, u'postcode': u'CB1 1LD'},
    {u'count': 1, u'postcode': u'CB1 1HA'},
    {u'count': 1, u'postcode': u'CB4 3JB'},
    {u'count': 1, u'postcode': u'CB4 2DG'},
    {u'count': 1, u'postcode': u'CB2 1RX'},
    {u'count': 1, u'postcode': u'CB3 9JZ'},
    {u'count': 1, u'postcode': u'CB1 3NW'},
    {u'count': 1, u'postcode': u'CB4 1SP'},
    {u'count': 1, u'postcode': u'CB4 2LG'},
    {u'count': 1, u'postcode': u'CB4 2QA'},
    {u'count': 1, u'postcode': u'CB1 2QF'},
    {u'count': 1, u'postcode': u'CB1 8QN'},
    {u'count': 1, u'postcode': u'CB2 1AA'},
    {u'count': 1, u'postcode': u'CB3 0AH'},
    {u'count': 1, u'postcode': u'CB1 3AX'},
    {u'count': 1, u'postcode': u'CB1 1JT'},
    {u'count': 1, u'postcode': u'CB2 3PN'},
    {u'count': 1, u'postcode': u'CB1 2EA'},
    {u'count': 1, u'postcode': u'CB1 2BH'},
    {u'count': 1, u'postcode': u'CB1 1JX'},
    {u'count': 1, u'postcode': u'CB1 2DW'},
    {u'count': 1, u'postcode': u'CB3 0AE'},
    {u'count': 1, u'postcode': u'CB1 2NU'},
    {u'count': 1, u'postcode': u'CB1 2LL'},
    {u'count': 1, u'postcode': u'CB2 3DB'},
    {u'count': 1, u'postcode': u'CB1 1BG'},
    {u'count': 1, u'postcode': u'CB5 8AF'},
    {u'count': 1, u'postcode': u'CB2 1AW'},
    {u'count': 1, u'postcode': u'CB4 1HB'},
    {u'count': 1, u'postcode': u'CB1 1DY'},
    {u'count': 1, u'postcode': u'CB2 1NW'},
    {u'count': 1, u'postcode': u'CB4 3BA'},
    {u'count': 1, u'postcode': u'CB2 1EN'},
    {u'count': 1, u'postcode': u'CB1 1LH'},
    {u'count': 1, u'postcode': u'CB5 8DD'},
    {u'count': 1, u'postcode': u'CB1 2JB'},
    {u'count': 1, u'postcode': u'CB1 7AW'},
    {u'count': 1, u'postcode': u'CB4 1XG'},
    {u'count': 1, u'postcode': u'CB2 1HL'},
    {u'count': 1, u'postcode': u'CB1 2LG'},
    {u'count': 1, u'postcode': u'CB2 1HW'},
    {u'count': 1, u'postcode': u'CB4 3DZ'},
    {u'count': 1, u'postcode': u'CB1 7AA'},
    {u'count': 1, u'postcode': u'CB1 1DU'},
    {u'count': 1, u'postcode': u'CB4 1NZ'},
    {u'count': 1, u'postcode': u'CB1 3AA'},
    {u'count': 1, u'postcode': u'CB1 2AZ'},
    {u'count': 1, u'postcode': u'CB1 2LF'},
    {u'count': 1, u'postcode': u'CB3 0AF'}}],
{u'_id': u'fast_food',
 u'count': 73,
 u'info': [{u'count': 4, u'postcode': u'CB4 1JY'},
            {u'count': 4, u'postcode': u'CB1 2AD'},
            {u'count': 4, u'postcode': u'CB1 7AW'},
            {u'count': 3, u'postcode': u'CB1 7AA'},
            {u'count': 3, u'postcode': u'CB1 3AH'},
            {u'count': 2, u'postcode': u'CB1 7AG'},
            {u'count': 2, u'postcode': u'CB1 2AW'},
            {u'count': 2, u'postcode': u'CB2 1NW'},
            {u'count': 2, u'postcode': u'CB2 3NA'},
            {u'count': 2, u'postcode': u'CB2 1AW'},
            {u'count': 2, u'postcode': u'CB1 3LP'},
            {u'count': 2, u'postcode': u'CB4 1XE'},
            {u'count': 2, u'postcode': u'CB2 3LL'},
            {u'count': 2, u'postcode': u'CB2 1LA'},
            {u'count': 2, u'postcode': u'CB5 8RG'},
            {u'count': 2, u'postcode': u'CB1 2LD'},
            {u'count': 1, u'postcode': u'CB4 3AX'},
            {u'count': 1, u'postcode': u'CB2 1DB'},
            {u'count': 1, u'postcode': u'CB5 8JE'},
            {u'count': 1, u'postcode': u'CB2 1AB'},
            {u'count': 1, u'postcode': u'CB1 2AS'},
            {u'count': 1, u'postcode': u'CB4 3BU'},
            {u'count': 1, u'postcode': u'CB2 3NB'},
            {u'count': 1, u'postcode': u'CB1 2NJ'},
            {u'count': 1, u'postcode': u'CB5 8PB'},
            {u'count': 1, u'postcode': u'CB4 1EH'},
            {u'count': 1, u'postcode': u'CB2 1DP'},
            {u'count': 1, u'postcode': u'CB1 8QJ'},
            {u'count': 1, u'postcode': u'CB4 3JD'},
            {u'count': 1, u'postcode': u'CB4 3HG'},

```

```

        {u'count': 1, u'postcode': u'CB5 8LG'},
        {u'count': 1, u'postcode': u'CB4 2NF'},
        {u'count': 1, u'postcode': u'CB1 1LN'},
        {u'count': 1, u'postcode': u'CB1 1DG'},
        {u'count': 1, u'postcode': u'CB1 1BG'},
        {u'count': 1, u'postcode': u'CB4 3HL'},
        {u'count': 1, u'postcode': u'CB1 2QF'},
        {u'count': 1, u'postcode': u'CB2 3RD'},
        {u'count': 1, u'postcode': u'CB1 2JB'},
        {u'count': 1, u'postcode': u'CB2 1NT'},
        {u'count': 1, u'postcode': u'CB2 3NP'},
        {u'count': 1, u'postcode': u'CB1 3AW'},
        {u'count': 1, u'postcode': u'CB2 8PB'},
        {u'count': 1, u'postcode': u'CB1 7AJ'},
        {u'count': 1, u'postcode': u'CB2 1UF'},
        {u'count': 1, u'postcode': u'CB1 2BD'},
        {u'count': 1, u'postcode': u'CB1 3BE'},
        {u'count': 1, u'postcode': u'CB5 8WL'},
        {u'count': 1, u'postcode': u'CB4 1RJ'}}],
{u'_id': u'cafe',
 u'count': 59,
 u'info': [{u'count': 5, u'postcode': u'CB1 3NF'},
            {u'count': 4, u'postcode': u'CB1 1DG'},
            {u'count': 3, u'postcode': u'CB1 2AS'},
            {u'count': 3, u'postcode': u'CB2 1LA'},
            {u'count': 2, u'postcode': u'CB2 1DP'},
            {u'count': 2, u'postcode': u'CB2 1SP'},
            {u'count': 2, u'postcode': u'CB1 3AH'},
            {u'count': 2, u'postcode': u'CB2 1RG'},
            {u'count': 2, u'postcode': u'CB1 1ER'},
            {u'count': 1, u'postcode': u'CB1 3LP'},
            {u'count': 1, u'postcode': u'CB2 3QN'},
            {u'count': 1, u'postcode': u'CB1 2BD'},
            {u'count': 1, u'postcode': u'CB2 3PF'},
            {u'count': 1, u'postcode': u'CB2 1UB'},
            {u'count': 1, u'postcode': u'CB3 9ND'},
            {u'count': 1, u'postcode': u'CB2 1QA'},
            {u'count': 1, u'postcode': u'CB1 8QJ'},
            {u'count': 1, u'postcode': u'CB2 3NE'},
            {u'count': 1, u'postcode': u'CB1 1LH'},
            {u'count': 1, u'postcode': u'CB3 0AJ'},
            {u'count': 1, u'postcode': u'CB1 7AW'},
            {u'count': 1, u'postcode': u'CB1 2LJ'},
            {u'count': 1, u'postcode': u'CB4 1EP'},
            {u'count': 1, u'postcode': u'CB5 8AD'},
            {u'count': 1, u'postcode': u'CB3 7BD'},
            {u'count': 1, u'postcode': u'CB1 1LN'},
            {u'count': 1, u'postcode': u'CB2 1UA'},
            {u'count': 1, u'postcode': u'CB2 8PB'},
            {u'count': 1, u'postcode': u'CB2 1SU'},
            {u'count': 1, u'postcode': u'CB4 0PP'},
            {u'count': 1, u'postcode': u'CB2 3LL'},
            {u'count': 1, u'postcode': u'CB2 3HW'},
            {u'count': 1, u'postcode': u'CB1 3AA'},
            {u'count': 1, u'postcode': u'CB2 3PP'},
            {u'count': 1, u'postcode': u'CB2 3QY'},
            {u'count': 1, u'postcode': u'CB1 2AD'},
            {u'count': 1, u'postcode': u'CB2 1TW'},
            {u'count': 1, u'postcode': u'CB1 2AW'},
            {u'count': 1, u'postcode': u'CB2 3PJ'},
            {u'count': 1, u'postcode': u'CB2 3JU'},
            {u'count': 1, u'postcode': u'CB2 3NA'},
            {u'count': 1, u'postcode': u'CB5 8WR'},
            {u'count': 1, u'postcode': u'CB2 3BZ'}}],
{u'_id': u'place_of_worship',
 u'count': 34,
 u'info': [{u'count': 2, u'postcode': u'CB1 2DG'},
            {u'count': 1, u'postcode': u'CB5 8AQ'},
            {u'count': 1, u'postcode': u'CB2 1UF'},
            {u'count': 1, u'postcode': u'CB2 8RN'},
            {u'count': 1, u'postcode': u'CB1 8QJ'},
            {u'count': 1, u'postcode': u'CB1 3TH'},
            {u'count': 1, u'postcode': u'CB1 4AJ'},
            {u'count': 1, u'postcode': u'CB5 8DT'},
            {u'count': 1, u'postcode': u'CB3 0AE'},
            {u'count': 1, u'postcode': u'CB1 7UB'},
            {u'count': 1, u'postcode': u'CB1 8QN'},
            {u'count': 1, u'postcode': u'CB2 1QY'},
            {u'count': 1, u'postcode': u'CB2 3AX'},
            {u'count': 1, u'postcode': u'CB1 2QA'},
            {u'count': 1, u'postcode': u'CB4 3QS'},
            {u'count': 1, u'postcode': u'CB1 3AN'},
            {u'count': 1, u'postcode': u'CB4 1LN'},
            {u'count': 1, u'postcode': u'CB4 2JE'},
            {u'count': 1, u'postcode': u'CB3 0AQ'},

```



```
{u'count': 1, u'postcode': u'CB2 1UB'},  
{u'count': 1, u'postcode': u'CB5 8LD'},  
{u'count': 1, u'postcode': u'CB2 3PP'},  
{u'count': 1, u'postcode': u'CB2 1HS'},  
{u'count': 1, u'postcode': u'CB1 3LP'},  
{u'count': 1, u'postcode': u'CB2 1QG'},  
{u'count': 1, u'postcode': u'CB2 1RG'},  
{u'count': 1, u'postcode': u'CB1 3DS'},  
{u'count': 1, u'postcode': u'CB1 2LT'},  
{u'count': 1, u'postcode': u'CB2 1JP'},  
{u'count': 1, u'postcode': u'CB5 8RS'},  
{u'count': 1, u'postcode': u'CB5 8LF'},  
{u'count': 1, u'postcode': u'CB2 3AR'},  
{u'count': 1, u'postcode': u'CB5 8BA'}}}]
```

## Section 4. Conclusions

The Cambridge, England OpenStreetMap dataset is full of information. However, this can also be cumbersome to analyze. During the cleaning up stages, many labels and sublabels were removed. It may be useful to move the information to an existing label rather than having it removed. For example, the information in "have\_riverbank" and "trees" can be moved to the kept "natural" label. Additionally, there are some labels that would have been useful for analysis. For example, "cuisine" was prematurely removed. The top cuisine for this area couldn't be examined with MongoDB.