# Yelp Recruiting Competition

How many "useful" votes will a Yelp review receive? Show off your skills to land an interview for a position on a Yelp data mining team!

350 teams · 4 years ago



Competition Data

Edit

- sample_submission.cs...
- eligibility_release....
- yelp_test_set.tgz
- yelp_training_set.tg...
- yelp_test_set.zip
- yelp_training_set.zi...

**eligibility_release.pdf** 130.88 KB

Download

Data Description

## Data Description

In the training set:

- 11,537 businesses
- 8,282 checkin sets
- 43,873 users
- 229,907 reviews

Each file is composed of a single object type, one JSON object per line. The training data was recorded on 2013-01-19. The testing data contains reviews, businesses, users, and checkins from the period between 2013-01-19 and 2013-03-12.

**Many user and business records referenced in the test set can be found in the training data.**

**Training set** = yelp_training_set.tgz OR yelp_training_set.zip
**Testing set** = yelp_test_set.tgz OR yelp_test_set.zip
**Sample submission format** = sample_submission.csv

## business

{
'type': 'business',
'business_id': (encrypted business id),
'name': (business name),
'neighborhoods': [(hood names)],
'full_address': (localized address),
'city': (city),
'state': (state),
'latitude': latitude,
'longitude': longitude,
'stars': (star rating, rounded to half-stars),
'review_count': review count,
'categories': [(localized category names)]
'open': True / False (corresponds to permanently closed, not business hours),
}

## review

{
'type': 'review',
'business_id': (encrypted business id),
'user_id': (encrypted user id),
'stars': (star rating),
'text': (review text),
'date': (date, formatted like '2012-03-14'),
'votes': {'useful': (count), 'funny': (count), 'cool': (count)}
}

## user

{
'type': 'user',
'user_id': (encrypted user id),
'name': (first name),
'review_count': (review count),
'average_stars': (floating point average, like 4.31),
'votes': {'useful': (count), 'funny': (count), 'cool': (count)}
}

## checkin

{
'type': 'checkin',
'business_id': (encrypted business id),
'checkin_info': {

'0-0': (number of checkins from 00:00 to 01:00 on all Sundays),
'1-0': (number of checkins from 01:00 to 02:00 on all Sundays),
...
'14-4': (number of checkins from 14:00 to 15:00 on all Thursdays),
...
'23-6': (number of checkins from 23:00 to 00:00 on all Saturdays)

} # if there was no checkin for a hour-day block it will not be in the dict
}