16 Spring, CSE 6242 **The Incredibles #20**

GT Student Rental Solution, a Web Application

Presented by:

Kai Lu

Cheng Chen

Qi Ge

Jiahao Luo

Shuo Huang

Website:http://gtrent2016.herokuapp.com/gtrent/

GitHub:https://github.gatech.edu/klu65/CSE6242-GTRent

Georgia Institute of Technology

1. Introduction

Along with increasing student population in Georgia Tech, those students who prefer living off campus may be confronted with an accommodation puzzle. Rents, distances to school, life convenience and crime rates are several major concerns in decision making. Thus in this project, we mine data sources including Zillow, Google, Yelp and Atlanta local crime database for all-round details of Atlanta rental properties within I-285 highway. Moreover, we polish and integrate details into a web application with data visualization and analytics skills we learned in class, which aims to provide best local rental information and recommendation for Georgia Tech students in need.

2. Problem definition

This project is inspired by our painful experiences in searching for suitable accommodations as first-year students. Details about these rental properties' locations, rents, neighborhoods and residents' reviews are scattered throughout several major data sources such as Google Maps, Yelp, and rental housings' official websites, making searching process tedious and difficult. On the other hand, some existing tools like Zillow and Trulia are unnecessarily complicated for GT students. Thus, our project aims to create an application that combines all necessary features and information in a more concise and dedicated way.

3. Survey

Songnian's book, by reviewing development of web-based GIS and mapping services [1], inspired us to implement different data sources and services on Web-GIS.

According to Zhou's study of commuting in the University of Los Angeles [2], the most prevalent transit tools are car (41.2%), public transportation (30.9%) and walking (24.8%). Since transportation in Atlanta is similar to LA's, we collected distances between school and properties in three aspects: driving, public transportation, and walking.

Since accessibility to retailing services are quite important for evaluating properties' value[4], we searched each property's neighborhood for these services and combine them with crime rates and distances to campus to build a ranking system.

Amy Hicks's research examines people's motivations for using Yelp and their differences, including methods and needs [3], giving us data correlated with students. Weijia Dai's paper introduces cleaning data method for those containing restaurants on Yelp[8]. Mukherjee's paper [4] focuses on circumstances of possible fake reviews and how Yelp reacts to them, helping us evaluate accuracy of Yelp reviews.

Crawling addresses' concepts are based on paper [5], from which we utilize similar idea about spatial range queries. Hybrid cumulated kNN searches searched for places, covering areas surrounded by I-285.

After learning basic crime data mapping concepts (repeat victimisation) in Hirschfield's book [6], we converted addresses into geographic coordinates and making calculating crime rate in a specific area easier. Ultimately, we focus on relationships between crime rates and locations.

Devin's papers demonstrate the inverse relationship between crime rates and property values: "the estimated elasticities of property values with respect to crime range from -0.15 to -0.35" [7]. Leigh's research provides more detailed price gradient of distance from criminal offenses: a crime event has a -4% price reduction on houses within 0.1 miles, yet has no influence on houses farther away [8]. Thus, it is reasonable to rank final rental result with elastic numbers and distance gradient considering their linear relationship.

By Gelman's paper [9], which studies difference between data qualities based on user-contributed information, Zillow improves completeness and integrity instead of accuracy. Because of large and complex geographic data, we may apply data mining techniques in Miller's book [10], including classification, clustering, association, trends and regression analysis.

4. Proposed method

4.1 Intuition

As GT students, we combined information from several major data sources including Zillow, Google Maps, Yelp, and local police crime database, and built a ranking and recommendation system and a web application, both precisely meeting GT students' need, especially when compared with popular applications like Zillow and Trulia.

4.2 Description of Approaches

4.2.1 Property data scraping from Zillow.com:

We were trying to collect property data with Zillow API and Google Map API, however, there were some limitations of these two approaches. For example, with Zillow API, no current rent of a property but an estimated one was returned on our request. On the other hand, no rental house was listed on the results returned by Google Map API. Thus, we have switched our method to scraping data from Zillow.

We used urllib2 to request and read related rental information from web pages on Zillow by going through all compatible zip codes within I-285 highway. The procedure is shown below:

1. By using Google Map API, we can easily obtain a zip code list file (**zipcodeList.txt**) which is within the I-285 highway.
2. After that, we need to loop through specific zip code on Zillow.com and scrape the information by running the script **scrapeProperty.py**.
3. Assuming we are finding rental information with zip code 30339.

response=urllib2.urlopen("http://www.zillow.com/homes/for_rent/GA-30339/house,condo,apartment_duplex,mobile,townhouse_type/1_p/")
page_text = response.read()

The *page_text* contains the first page of the search results of rental properties whose zip code is 30339 from Zillow.com. As you can see from the below figure, the right-hand side bar shows the information of the purple dots on the map.

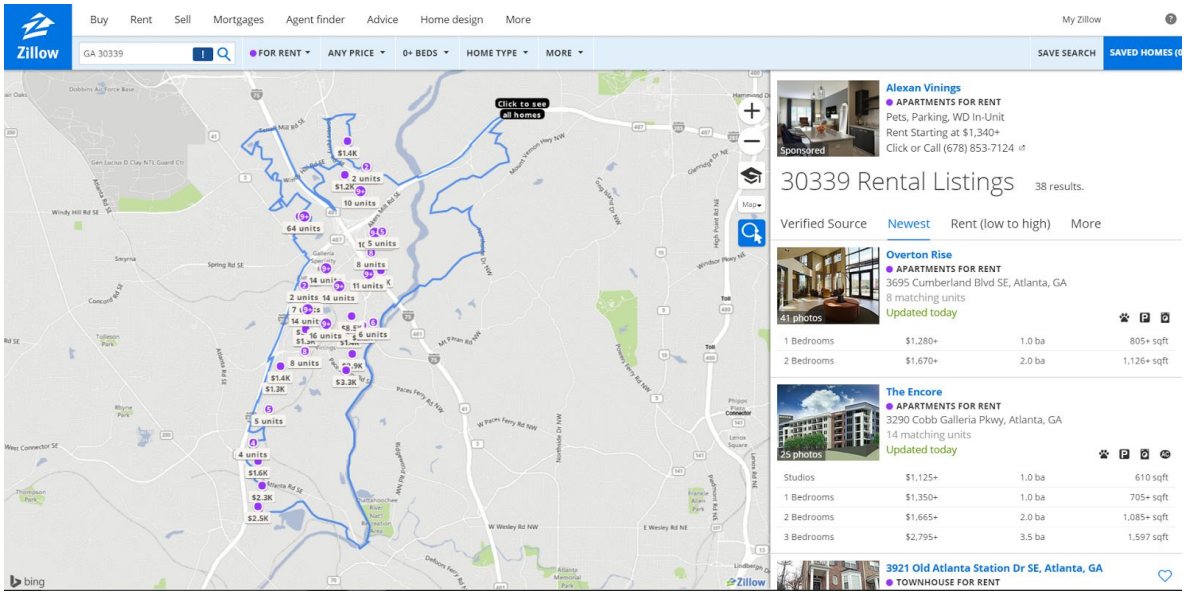


Figure 1. Screenshot of rental information on Zillow.com

4. We could then collect the name, coordinates, address, type and link of each property with the **findInfo** function in script **findInfo.py** by going over all the pages of search results.
5. By going over all compatible zip code, we have raw information of all rental properties listed on Zillow.com within I-285 highway.

4.2.2 Property data cleaning with OpenRefine:

Since data from Zillow.com does not have an uniform return for each request. The original data we scraped had many duplicates, blank and messy tuples. So we utilized OpenRefine to get rid of all dirty tuples and convert raw data into a preferred format for us. In the end, total number of rental properties result after filtering is 1098, and all those properties are located within I-285 highway in Atlanta.

4.2.3 Commuting time collection with Google Map Distance Matrix API:

In order to obtain approximate commuting time from rental properties to school, we utilized Google Map Distance Matrix API to request travel durations between two given locations. Also, we considered three main modes of transportation.

(1) For students who have a vehicle, we set the transportation mode as *driving* and calculated the average duration between school and the property.

(2) For students who usually walk or take public transportation, the average duration calculation was the same except setting the transportation mode as *walking* or *transit*.

The commuting time collection is run by function **findRouteTime** defined in script **findDistance.py.**

4.2.4 Life convenience evaluation with Yelp API

Data for convenience evaluation is downloaded from Yelp API. Every estate property is chosen as a searching center. Several category data have been involved to evaluate convenience around searching center. For instance, there are two searching radiuses for food which are 300 meters (for walking) and 2000 meters (for driving). For gas station and entertainment,the search radius is 2000 meters. 300 meters takes an adult 5 minutes or less on foot which is a reasonable time for us to walk to a restaurant. The average number of search results of every property for 2000 meters are 96.03 for food, 4.80 for gas and 31.99 for entertainment. So the scale of data is suitable under such search radius. The following formulas are our basic evaluation standard.

$$sum\_rating = \sum_{all\_results\_for\_the\_target\_property} \frac{rating}{5.0}$$

$$food\_score = \frac{sum\_rating_{target\_property}}{\max_{among\_all\_properties}(sum\_rating)} * 100$$

$$entertainment\_score = \frac{sum\_rating_{target\_property}}{\max_{among\_all\_properties}(sum\_rating)} * 100$$

$$gas\_score = \frac{number\_of\_gas\_station\_around\_target\_property}{\max imun\_number\_of\_gas\_station\_among\_all\_properties} * 100$$

The sum_rating is the sum of rating result around our target property from Yelp. After we obtained the sum_rating, we can calculate the score of relative service. In order to normalize the rating, we introduced maximum value of sum_rating as the denominator. For instance, there are two restaurants near desirable property and one's rating is 5.0 and the other one is 4.0. Then its

sum of rating is (4+5) / 2 = 4.5. For the other properties, assumed the highest "sum of rating" is 20 for a specific property, then food score of target property is 4.5 / 20*100 = 22.5.

4.2.5 Crime data collection:

Crime data are downloaded from Atlanta Police Department and scraped from crimemapping.com. We used Google Maps Geocoding API, converting addresses into geographic coordinates, and cleaned data with OpenRefine. Because the relationship between property price and distance from offender is within 0.15 miles, that means there's no impact when people choosing a house more than 0.15 miles from the offender's location(Figure 2). We could use the following formula to calculate crime impact factor:

$$Impact = \sum((-14\%/0.15 \text{ mile}) \times (\text{distance from crime location}))$$

Finally, crime impact factor is normalized to range of 0-100.

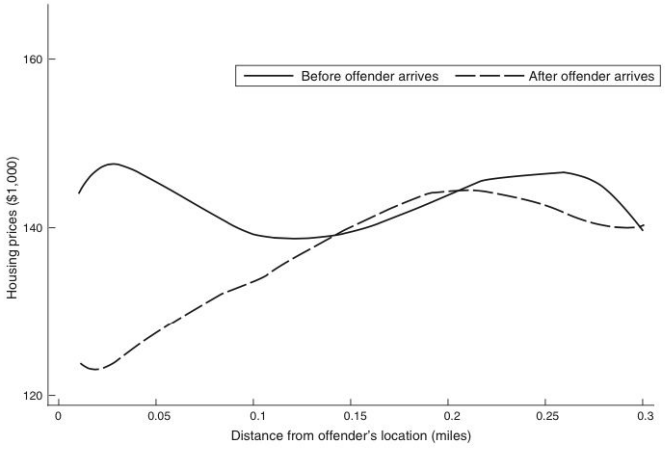


FIGURE 2B. PRICE GRADIENT OF DISTANCE FROM OFFENDER
(Sales during year before and after arrival)

Figure 2: Price Gradient of Distance from Offender (Sales during the year before and after arrival) by Leigh Linden [8].

4.2.6 Database building

After collection and cleaning all details of properties from Google Map, Zillow, Yelp and Atlanta local crime database, and calculating derived attributes that we need (For example, Crime_Grade and Yelp_Grade), we integrated all those data and generated a PostgreSQL database. The following figure is overall database schema for our project:

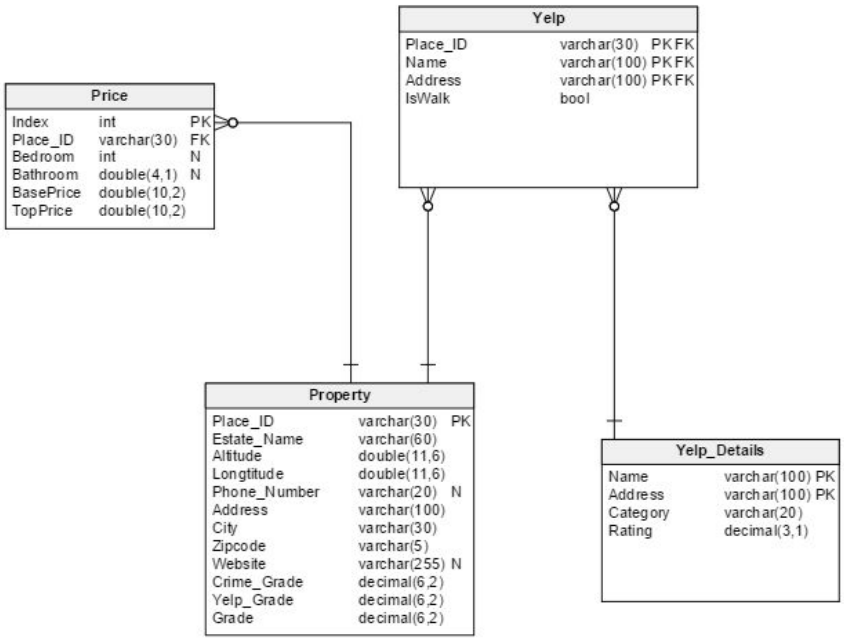


Figure 3. Relational Database Scheme

Place_ID is the primary key of the Table *Property*. *Place_ID* referencing Table *Property* is the foreign key of Table *Yelp*. *Name* and *Address* referencing Table *Yelp_Details* are the foreign keys of Table *Yelp*. The combination of *Place_ID*, *Name* and *Address* is the primary key of Table *Yelp*. *Place _ID* referencing Table *Property* is the foreign key of Table *Price*. The primary key of Table *Price* is the *index*, which is an automatically increasing number.

4.2.7 Property clustering with K-means algorithm

To better meet our custom needs and build recommendation system, we decided to cluster our properties data into several different groups. Because lacking training data, we focused on clustering algorithms for unlabeled data such as K-means, Hierarchical clustering and DBSCAN. Based on our app function, we prefer even cluster size and fewer clusters, which makes K-means our best choice. The basic idea of K-means algorithm is shown below:

$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} ||x_i - \mu_k||^2.$$

Our final property clustering is based on position, distance to GT, surrounding facilities and security. So we used “latitude”, “longitude”, “distance to GT”, “crime score”,“gas score”, “food score”and “entertainment score” as seven attributes to cluster properties.

To decide the best number of clusters, K is screened from 1 to 49. From the elbow method result, the percentage of variance explained has little increase when K is larger than 10 (**Figure 4:Left**). So we choose 10 as elbow point and separate 1097 properties into 10 different clusters.

A scatter plot is used to evaluate the K-means clustering results, different clusters are labeled with different colors. On scatter plot(**Figure 4:Right**), red and orange spots are located mainly in the center region whereas teal spots are in surrounding area, so we could tell the clusters are separated based on their latitude and longitude. Some properties having different locations but similar facilities and security are also been clustered into same group(e.g. purple spots).

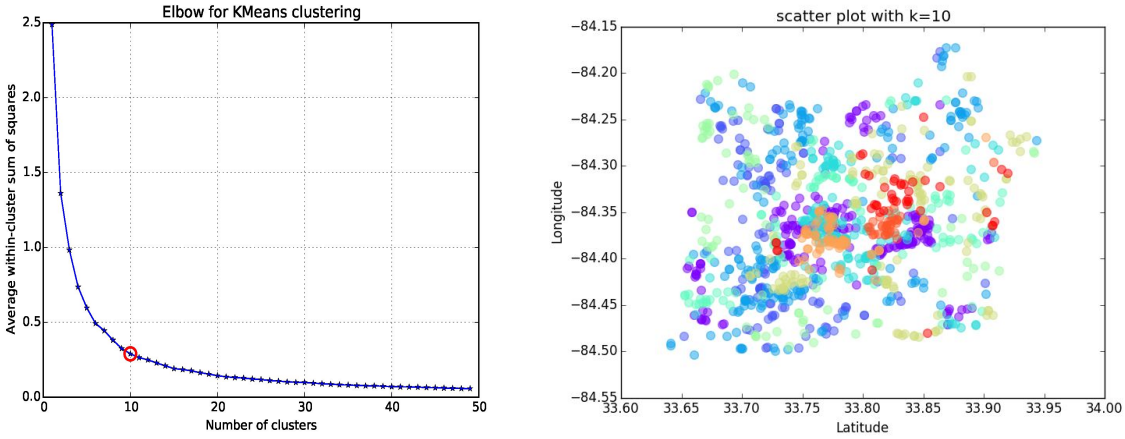


Figure 4. Properties clustering result with K-Means

4.2.8 Data visualization in form of Web application

Our web application is available online, serving users via Heroku cloud platform. The website address is https://gtrent2016.herokuapp.com/gtrent/.

We set up the page skeleton based on a open-source template from cssmoban.com. As to the front end part, we’ve been using **HTML, CSS, jQuery** and **D3.js** to construct the fundamentals of our website. For the back end part, in order to retrieve data for the user, the website connects database built on **Django** through **Ajax**. The following figures show some features of our website.

Figure 5 shows homepage of our website. On this page, we’ve designed a search box for searching property locations. As long as user types in an unfinished address, it will automatically complete rest of address and offer a list of candidates for user to choose from.

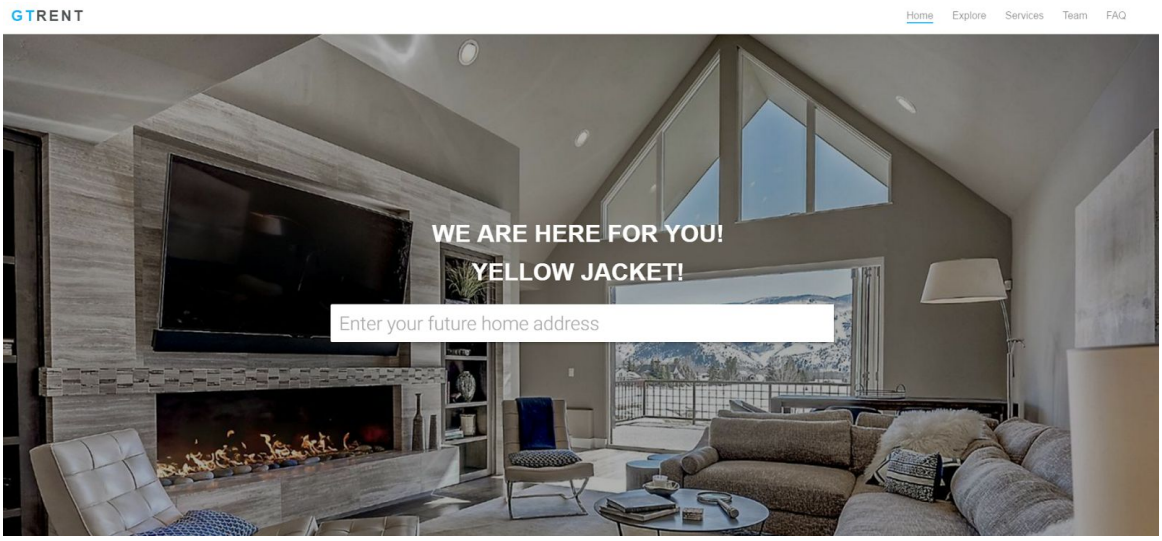


Figure 5. Homepage screenshot

After user selects a desired address, it will direct user to "Explore" page shown in **Figure 6**. There is a filter panel on the right side where users can apply filter based on their desired commuting style, commuting time, floor plan, range of price, property types and life convenience options. After "Apply" button is clicked, left side panel will roll out, displaying a list of properties satisfying filter, ordered by their grades. Moreover, each property will be shown as blue markers on map. If user moves his mouse over any property on the list, the map would pan to location of property and corresponding marker would turn red with a information window showing its name and minimum price.

If a property on the list or its corresponding marker is clicked as shown in **Figure 7**, there would appear a window with detailed information of property, including name, address, driving/walking/transit time and Food/Gas/Entertainment score on the right, and similar properties according to our K-means clustering on the left. In addition, there would be a link directing the users to original Zillow website.

The "Service" page explains techniques and data source that we use, "Team" page introduces every member of our group. "FAQ" page shows answers for some common questions.

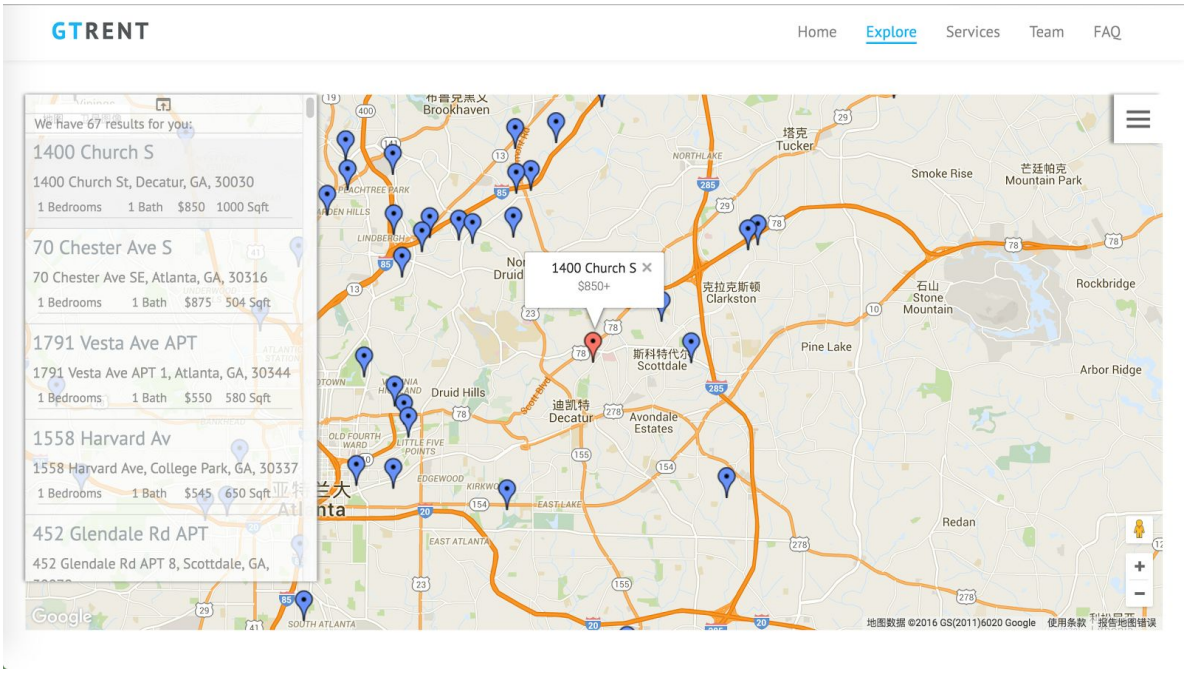


Figure 6. Explore page screenshot

In **Figure 8**, we attached a user investigation survey link on bottom of the website for further evaluation and improvement. All result will be stored and analyzed to improve our website in future.

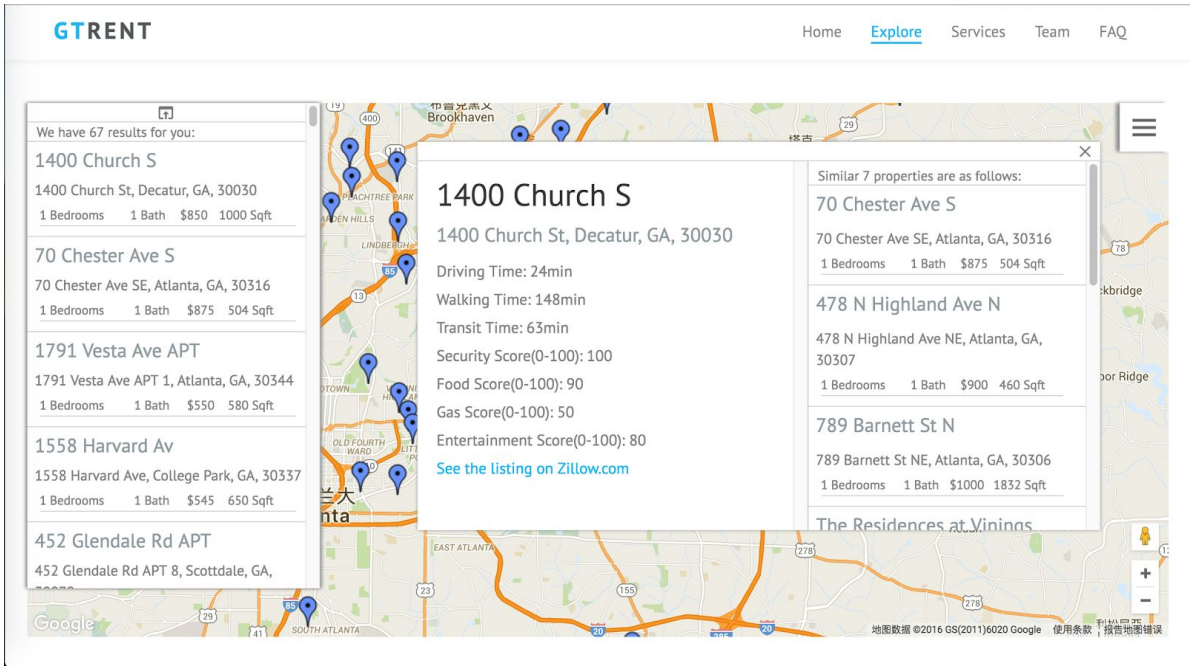


Figure 7. Detail Information window screenshot

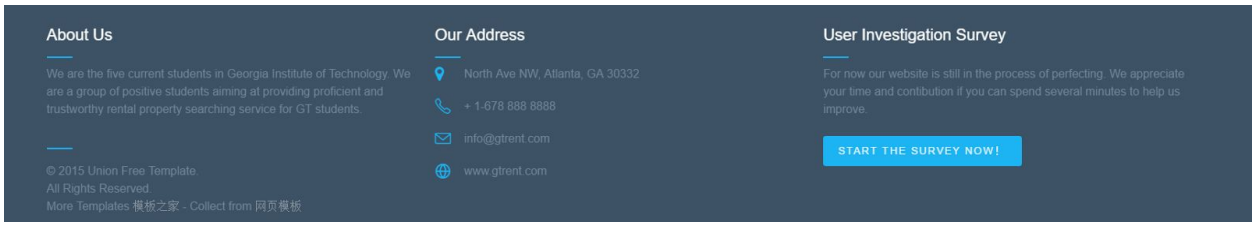


Figure 8. User investigation survey screenshot

5. List of innovations

5.1 Clustering all rental properties with K-means algorithm and providing the users with a recommendation of similar properties.

We used "latitude", "longitude", "distance to GT", "crime score","gas score", "food score"and "entertainment score" as attributes to cluster the properties with K-means algorithm.Then implemented the Elbow method to find the optimum number of clusters and visualized the clusters with K=10.

5.2 Building the web framework on Django and serving the users with AWS based cloud platform Heroku.

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Heroku is the best platform for building with modern webapp architectures. The Django plus Heroku combination provides considerable convenience and stable for building database and backend scripts.

6. Experiments and Evaluation

6.1 Testbed and list of questions

In order to evaluate if our work is valid and successful, we have done three series of experiments to answer the following questions:

1) Are the property data we scraped from Zillow.com consistent with what people directly get from Zillow?
2) Is the commuting time we got from Google Map Distance Matrix API reliable?
3) Do the potential users satisfy with the performance of our application?

6.2 Experiments and observation

6.2.1 Comparing with Zillow results to test the integrity of our rental property data.

To verify the data integrity, we compared the numbers of properties returned by Zillow Search Engine with the numbers we scraped from Zillow one month ago.

Zipcode	# of Results on Zillow.com(5/1/2016)	# of Results scraped from Zillow(3/25/2016)
30021	10	10
30030	35	38
30032	64	49
30033	29	28
30034	43	51
30079	5	5
30084	22	21
30303	3	5
30305	56	52
30306	47	37
30307	31	29
30308	56	53
30309	78	58
30310	60	60
30311	32	50
30312	41	46
30313	8	8
30314	22	15

30315	50	44
30316	64	45
30317	13	14
30318	60	43
30319	59	29
30324	60	48
30326	22	18
30327	23	29
30329	30	23
30337	8	10
30339	37	38
30340	16	11
30341	28	23
30342	37	32
30344	64	38
30345	11	9
30354	22	25
30360	6	11
Total	1252	1097

The table demonstrates that the percentage error of our scraping work is approximately 15%. And it is highly possible that such error results from the regular updates of rental listings on Zillow, since the searching results are collected one month later than scraping work.

6.2.2 Comparing results with different commuting style to verify the correctness of commuting time of properties.

To verify the commuting time we got from Google Map Distance Matrix API, we did some experiments by applying different commuting styles but fixing other conditions.

Figure 9 shows the searching results when we set the commuting style to "Driving" and commuting time to "45min". There are 67 results in this case.

Figure 9. Searching result when commuting style is *Driving*

Figure 10 shows the searching results when we change the commuting style to "Transit". The number of results dramatically drops to 13.

Figure 10. Searching result when commuting style is *Transit*

Figure 11 shows the searching results when we change the commuting style to "Walking". The number of results is only 2.

Figure 11. Searching result when commuting style is *Walking*

In conclusion, the experiments on commuting styles illustrate that the commuting time we have is reliable and thus make reasonable searching results.

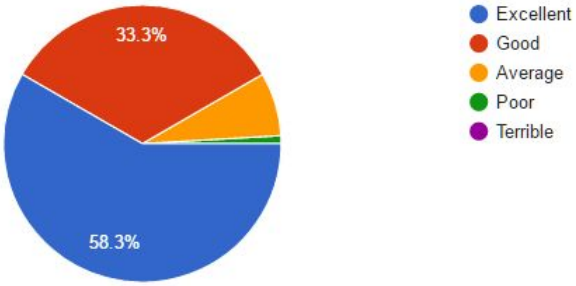
6.2.3 Inviting GT students to test the recommendation and ranking system.

In the last page of our website, we attach an user investigation survey link for user to evaluate their experience with our web application. Here is the link of our survey questions.

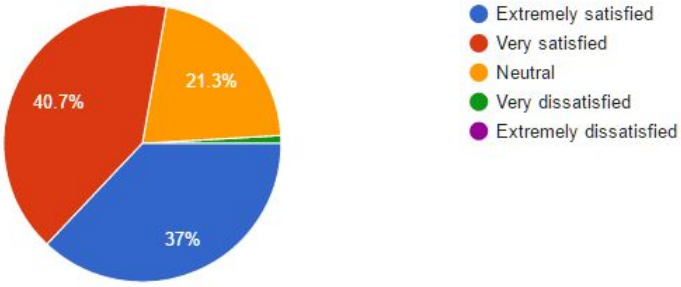
Here is the final result for the survey started from 4/29/2016 to 5/1/2016. There are totally 108 respondents. According to the survey result, overall, most of the users are satisfied with GTRent and rate us a quite high grades. Besides, most of the users are willing to recommend our web application to their friends. Compared to the similar website, Zillow and Trulia, user are more comfortable to use our website. And based on the pie chart, there is 75% of user accessing our website through PC, while the rest of them are using mobile devices.

However, there are also weakness according to this survey. First, our contents about the rental property should be more ample, for example, pictures for the property. Second, since our web application is built for PC user. The experience with our website is not that comfortable for mobile device user. All of this recommendation will help us improve in the future.

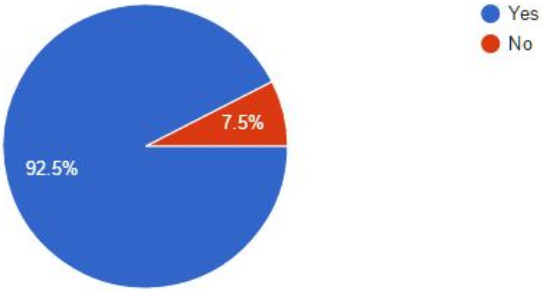
Overall, how would you rate GTRent? (108 responses)

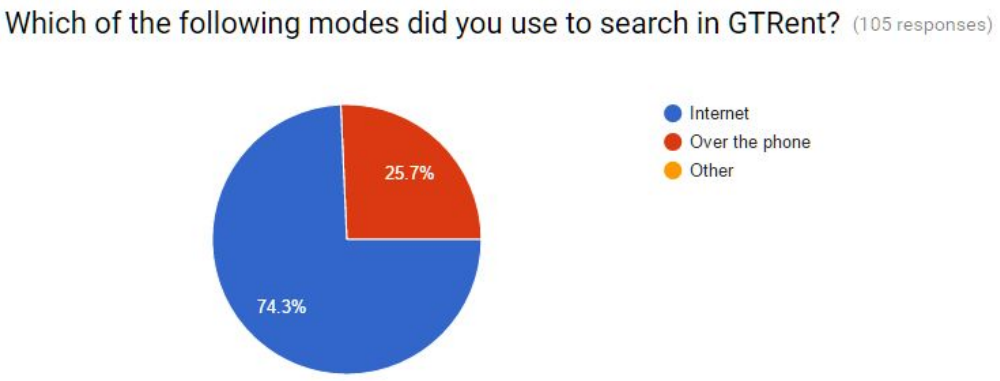


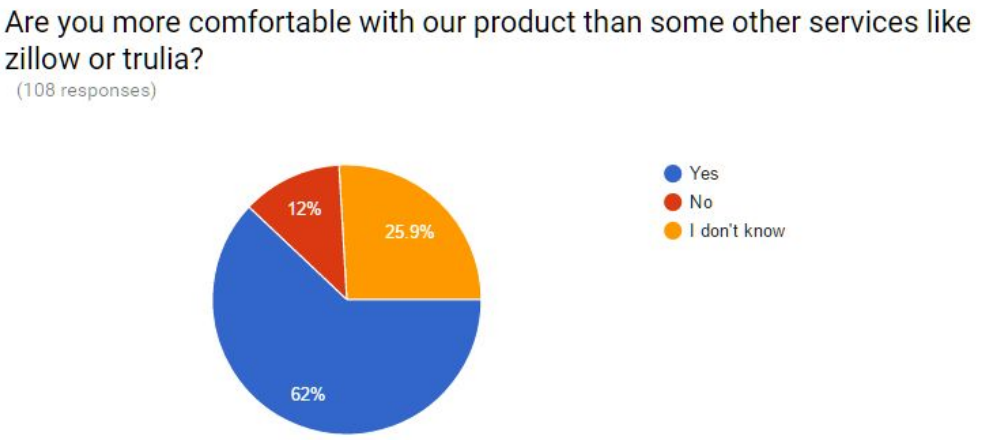
How satisfied are you with your service from GTRent? (108 responses)



Would you recommend GTRent to a friend? (106 responses)







7. Conclusion and discussion

7.1 Conclusion

1) We obtained a large scale of data from different data sources and successfully filtered out the abundant and spurious tuples.
2) We designed our own property ranking and recommendation system to satisfy users with different needs.
3) We constructed a relational database for data storage and an interactive web application for data visualization. They are well connected and performed.

7.2 Discussion

1) The rental properties listings change everyday. To present up-to-date rental information, a regular update mechanism should be established.

2) The web application is currently not mobile-friendly. More improvements should be made in our following steps.

8. Team member contributions

Data collection and cleaning:
Kai Lu & Qi Ge(Google Map), Shuo Huang(Crime Data), Jiahao Luo(Zillow) and Cheng Chen(Yelp)
The work distribution for Web application development：

1) Construct recommendation and ranking system. (by Shuo and Cheng)
2) Web design. (by Kai and Jiahao)
3) Create web app and add functions. (by Kai, Jiahao and Qi)
4) Connect database to our web app with Django. (by Cheng and Qi)
5) Building a cloud server on Heroku. (by Qi)

All the workloads are relatively evenly distributed.

Reference

1. Hardie, Andrew. "The development and present state of Web-GIS." *Cartography 27.2 (1998): 11-26.*
https://www.crcpress.com/Advances-in-Web-based-GIS-Mapping-Services-and-Applications/Li-Dragicevic-Veenendaal/9780415804837

2. Zhou, J., 2012. Sustainable commute in a car-dominant city: Factors affecting alternative mode choices among university students. *Transportation research part A: policy and practice, 46(7), pp.1013-1029*
http://www.sciencedirect.com/science/article/pii/S0965856412000651

3. Hicks, Amy, et al. "Why people use Yelp. com: An exploration of uses and gratifications." *Computers in Human Behavior 28.6 (2012): 2274-2279.*
http://www.sciencedirect.com/science/article/pii/S0747563212001951

4. Mukherjee, Arjun, et al. "What yelp fake review filter might be doing?."*ICWSM. 2013*.
http://www2.cs.uh.edu/~arjun/papers/ICWSM-Spam_final_camera-submit.pdf

5. Bae, W.D., Alkobaisi, S., Kim, S.H., Narayanappa, S. and Shahabi, C., 2009. Web data retrieval: solving spatial range queries using k-nearest neighbor searches. *Geoinformatica, 13(4), pp.483-514.*
http://link.springer.com/article/10.1007/s10707-008-0055-2

6. Mapping and Analysing Crime Data: Lessons from Research and Practice *(edited by Alex Hirschfield, Kate Bowers, April 26, 2001 by CRC Press, ISBN 9780748409228 - CAT# TF2275)*

7. Crime and property values: Evidence from the 1990s crime drop. *Devin G. Pope a, Jaren C. Pope.Regional Science and Urban Economics 42 (2012) 177–188*
http://www.sciencedirect.com/science/article/pii/S0166046211001037

8. Estimates of the Impact of Crime Risk on Property Values from Megan's Laws. *Leigh Linden and Jonah E. Rockoff. American Economic Review 2008, 98:3, 1103–1127*
https://www.aeaweb.org/articles.php?doi=10.1257/aer.98.3.1103

9. Gelman, Irit Askira, and Ningning Wu. "Combining structured and unstructured information sources for a study of data quality: a case study of Zillow. com." *System Sciences (HICSS), 2011 44th Hawaii International Conference on. IEEE, 2011.*

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=5718619&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D5718619

10. Miller, Harvey J., and Jiawei Han, eds. *Geographic data mining and knowledge discovery. CRC Press, 2009.* https://www.crcpress.com/Geographic-Data-Mining-and-Knowledge-Discovery-Second-Edition/Miller-Han/9781420073973

11. James J. Nolan III. *Establishing the statistical relationship between population size and UCR crime rate: Its impact and implications. Journal of Criminal Justice 32 (2004) 547 – 555.* http://theipti.org/wp-content/uploads/2012/02/covariance.pdf