**Recommending similar listings on craigslist**

Final Report

|  |
| --- |
| Akarsh Sinha  Girish Sharma  Keerthana Nemili Mandeep Singh Rahi  Manmeet Walia |

Unstructured Data Analysts

# **Table of Contents**

**1. Background**2

**2. Problem Statement**3

**3. Proposed Solution**3

**4.** **Data Collection**4

**5. Exploratory Data Analysis**4

**6. Data Pre-Processing** 7

**7. Data Analysis**7

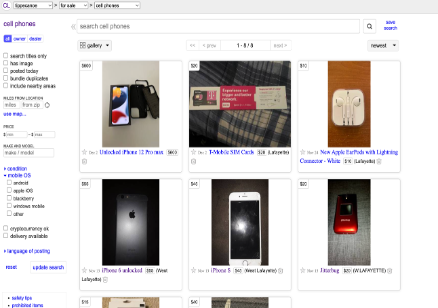
**8. Validation**9

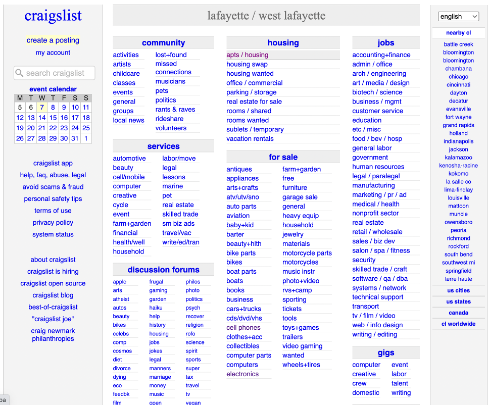
**9. Conclusion & Remarks**10

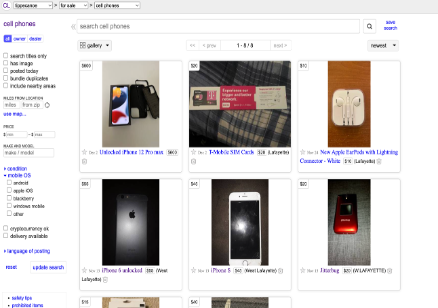
**References**…11

**Background**

Craigslist is a local posting website that lets people post any sort of classifieds on the platform. Most of Craigslist postings are free, except for specific postings in some or all locations in the US. Its major traffic source is the US, and most of it is direct, therefore people often skip search engines to reach the platform directly. By being straightforward and reliable over almost the last 3 decades, it has built a strong brand and thus a strong online presence.







|  |  |
| --- | --- |
| Fig 1.1: Craigslist home page | Fig 1.2: Listings of cellphones page |

On the main page of the Craigslist website, under the “for sale” we can find “cell phone”. These cellphone listings typically contain the title, price, maker/manufacturer, condition, mobile OS, location. Using the search bar, desired cellphones can be browsed.

**Problem Statement**

It is challenging for ad viewers to navigate Craigslist due to the unstructured format of its advertising which often have text descriptions and/or posted images. Additionally, this makes it challenging for platform managers to update and manage the website. The lack of cross-selling and up-selling recommendations on any ad page is one issue with this platform. Any such platform must monetize its visitors by providing a variety of options in order to retain them or risk losing them to another platform due to the enormous number of online platforms that customers can now access with only a few clicks. Most platforms provide other options to the customer if the visitor does not like the original option based on the product specifics, the user's and other similar users' browsing and purchase history. This has played a significant role in the success of online marketplaces like Amazon.com, which credits cross-selling for about 35% of its earnings. The next step for the mostly technologically antiquated platform to increase its market share is to improve the ad-viewer experience and giving people multiple options on ad pages until they find what they are searching for while still on the platform can help a lot with that.

**Proposed Solution**

The suggested enhancement of offering cross-selling and up-selling recommendations can be implemented by offering five or more ad recommendations on each ad page so that the user can easily continue their search without returning to the previous page or the main page. By leading them to ad recommendations like the one they're now viewing rather than making them browse through the complete list of ads again for what they specifically want, this offers them less interference in their path and refines their search. This ad recommendation system can be created for a specific category, such as cell phones, by scraping the text data (ad title, price, location, and description) for numerous ads, cleaning the data, reducing the dimensionality, normalizing the word frequency, splitting into training and test datasets, modeling, and comparing various models before selecting the best one. Its cross-selling recommendations can be based on the five tokenized ads that resemble the original tokenized ad the most. Before making the recommendations, it will also be explored during the project if it is possible to combine similarities between the picture data (uploaded photos) for the adverts and their related text data.

Fig 3.1: Proposed Design

**Data Collection**

For building the model that could learn from the input data to successfully identify relevant and recommendations for the product that user is interested in, we decided to scrape the Craigslist website for the listed cellphone data.

Graphical user interface

Description automatically generatedTo do this, we decided to filter the listings under 250 miles radius of Lafayette, with a focus on cellphones ([link](https://tippecanoe.craigslist.org/search/lafayette-in/moa?lat=40.44&lon=-86.877&sort=date&srchType=T&hasPic=1&search_distance=250&min_price=30&max_price=)). We found ~1200 listings at that point on the website. We decided to use Selenium to automate the control of website as it was more convenient then to derive the hyperlinks for the consecutive pages. Then we used BeautifulSoup to scrape data points relevant to our objective. The following are the fields that we scraped:

1. ID
2. Title
3. Description
4. Link
5. Location (Latitude, Longitude)
6. Price

Fig 4.1: Scraping

We plan to leverage Title to uniquely identify a particular cellphone model and map relevant recommendations based on this. The other fields can help model into a more effective classification of recommendations.

Post completion of the scraping exercise, we proceeded to transform the price and Location parameters using Normalization because the characteristics for these ranges (parameters) are different. With normalization, we are attempting to standardize the input so that few extreme values (outliers) in these inputs do not influence the conclusion of the analysis as they might strongly dominate other values. This becomes particularly important when working with algorithms (such as K Nearest Neighbor-KNN) that use Euclidean distance where extreme values might influence the distance strongly.

**Exploratory Data Analysis**

We have no null values in our scraped dataset. For the purposes of data analysis, we decided to filter key cellphone models from the description so that we can make an inference about the frequency distribution of these models.

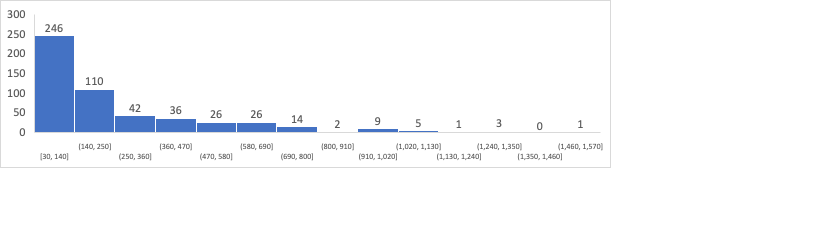
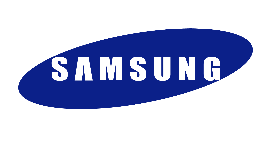


Fig 5.1: Frequency of prices

The product that has the maximum listed values is “Samsung Galaxy Z Fold” for $1500. There are very few smartphones priced at this price segment. The average price for a smartphone in this dataset is about $250. Most of the products listed are priced between $30 and $140.

Icon

Description automatically generated

**20%**

**42%**

**38%**

**Others**

Fig 5.2: Proportion of brands in cell phone listings

Fig 5.3: Frequency of cell-phone models

What we observed is that” iPhones” lead the listings by commanding 42% portion of the entire listings in the dataset, followed by “Samsung” phones that form 20% of the total listings. Additionally, non-popular cell phone models (“Others” category) form a significant portion of the entire listings at 38%. Thus, iPhones and Samsung phones are the most popular smartphones brands listed on Craigslist.

**LDA Results for Data Exploration**

We carried out LDA to find out what kind of different listings were there in the data. We got to know that there are two kinds of ads, selling and buying ads. Here are our findings from the analysis:

## Chart, bubble chart Description automatically generated**Topic or cluster 1:**

* About 46 percent of tokens belong to this cluster.
* Relative importance of words like unlocked, screen, carrier, Verizon is higher for this cluster.
* This cluster contains almost all the sales-ads that are about unlocked cell phones with a network carrier.

Fig 5.4: Topic or Cluster 1

## Chart, bubble chart Description automatically generated**Topic or cluster 3:**

* About 27 percent of tokens belong to this cluster
* Relative importance of words like *cash, buyer, repair* is higher for this cluster.
* This cluster **contains all the buyer-ads. It also contains ads that are related to repairs or screen replacement.**

Fig 5.4: Topic or Cluster 3

## **Data Pre-Processing**

Fig 6.1: Pre-Processing Flow

1. Tokenizing: Cell phone titles were tokenized to perform further analysis.
2. Removing stop-words: Apart from the stock stop-words in the nltk library, about 50 more stop-words were identified and removed by analysing the data.
3. TF-IDF word-matrix: This matrix was used as an input for unsupervised learning models to provide customer recommendations.

**Data Analysis:**

Before conducting the data analysis, we added additional features to factor in the location and the selling price mentioned for each ad to generate recommendations. It was necessary to localize the recommendations using the latitude and longitude coordinates for each ad as Craigslist is largely a peer-to-peer marketplace and the ad viewers often pickup or get the item delivered directly from the advertiser and thus, prefer local advertisers in making their purchase selections. It was also necessary to narrow down the recommendations to a similar price range of the original ad using the selling price mentioned for each ad that the ad viewer opens from the main screen because ads mostly showcase used/refurbished items, and the user who is willing to buy it over a new item may not be willing to click on an ad recommendation which is unusually low/high priced versus the original ad. After scraping these features, we also had to normalize them within range of -1 to 1 so that they can be used correctly with TF-IDF weights.

Then, the recommendations were generated from 2 models:

(i) K-Nearest Neighbor: This non-parametric, lazy learning algorithm was used in an unsupervised, item-item based collaborative filtering manner to generate recommendations in clusters of 6 ads each such that any ad from the cluster when viewed by the user would be accompanied by the other 5 ads in the cluster as recommendations. This algorithm does not assume underlying data distribution and relies on item feature similarity. Hence, it was effective when used with the high-dimensional TF-IDF weight vectors. Also, we had to use Cosine distance affinity as a similarity measure for searching nearest neighbors as the Euclidean is not useful with high dimensions.

|  |  |
| --- | --- |
| Cosine distance bucket | Proportion of ads |
| 0-0.1 | 13.6% |
| 0.1-0.2 | 32.8% |
| 0.2-0.4 | 48.7% |
| 0.4-0.7 | 5.0% |

As observed in the results csv file, we have generated the 5 top recommendations (the ad IDs in columns knnrec2, knnrec3, knnrec4, knnrec5, and knnrec6 are the ad recommendations for the respective ad IDs in column knnrec1) based on the shortest cosine distance (columns knndist2, knndist3, knndist4, knndist5, and knndist6 are the cosine distances) of the 5 recommended ads for the viewed ad.

On analyzing the proportion of all ad recommendations in buckets of cosine distances (0-0.1, 0.1-0.2, 0.2-0.4, and 0.4-0.7), we observe that almost 95% of the recommendations have some degree of similarity to the original ads and almost half the recommendations are very similar to the original ad as shown in Fig 7.1. From this, we can conclude that the ad recommendations are fairly similar to the original ad overall in terms of title words, price, and location. A sample ad and its recommendations from each bucket is shown in Fig 7.2.

Fig 7.1: Proportion of   
ad recommendations in each cosine distance bucket

|  |  |  |
| --- | --- | --- |
| Original Ad | NEW 10.1 PHABLET - 3G, 2 SIM Slots, 32 GB, Tablet, In Box, Orig $105 - $75 (St. Charles) | FACTORY UNLOCKED Pixel 7 Pro 128gb - $650 (St.Louis) |
| Recommended Ads | (i) NEW 10.1 PHABLET - 2 SIM Slots, 32 GB, Phone  Tablet, In Box - $75 (St Charles)  (ii) NEW 10.1 PHABLET - 2 SIM Slots, 32 GB, Phone Tablet, In Box - $75 (St Charles)  (iii) NEW 10.1 TABLET - 2 SIM Slots, 32 GB, Phone Tablet, In Box Orig $105 - $75 (St Charles)  (iv) Alcatel Joy Tablet - Wifi & Data Capable, 32 GB, W/Box, Sell - $50 (St charles)  (v) NEW 10.1 TABLET - 2 SIM Slots, 32 GB, Phone Tablet, In Box Orig $105 - $65 (St Charles) | (i) BRAND NEW SAMSUNG S22+ \*\*\*UNLOCKED\*\* - $650 (KIRKWOOD)  (ii) Google Pixel 7 128 Obsidian Black factory unlocked - $590  (iii) Samsung S22 Ultra 128gb Black Carrier Unlocked - $550 (Saint Louis)  (iv) Samsung Galaxy S22 ULTRA Verizon - $650 (Decatur)  (v) Samsung S21 ultra 5G - $500 (Fairview heights IL) |
| Cosine distance bucket | 0-0.2 | 0.2-0.4 |
| Proportion of recommendations in bucket | 55% | 42% |

Note: 3% ads belong to 0.4-0.7 bucket.

Fig 7.2 Sample ad and its recommendations from each bucket

Now, on analyzing the proportion of each of the 5 ad recommendations in buckets of cosine distances, we observe the following distribution which indicates that the first 1-2 recommendations are fairly similar to the original ad and this similarity slightly reduces in the next 4-5 recommendations. This can be visualized from the stacked bar chart of the proportion of each recommendation in each cosine distance bucket shown below (Fig 7.3).

Fig 7.3: Proportion of each recommendation in each cosine distance bucket

However, this is a naïve algorithm and cannot be utilized to recommend ads for the scale at which Craigslist operates. Hence, we have recommended to use it as the baseline as it provided us the most accurate recommendations based on the evaluation criteria for each algorithm.

(ii) Locality-Sensitive Hashing: This is a faster algorithm that basically “groups” the data before creating pairs by segmenting and hashing the same sample several times. We use the locality-sensitive hashing instead of the locality-preserving hashing so that the algorithm does not assume the data distribution. There are various methodologies of using this hashing and we use the MinHash approach in which: when the algorithm finds a pair of vectors hashed to the same value at least once, it tags them as candidate pairs. The math underlying the algorithm is based on the Jaccard Similarity Coefficient which is used to gauge the similarity and diversity of sample sets and calculated by taking the ratio of Intersection over Union.

Based on the recommendations generated from this algorithm, we can see that although they are only ~50% as accurate as the recommendations generated from K-NN, we can generate them faster with a time complexity of O(n\*log n) versus K-NN’s time complexity of O(n\*n) which can utilized for Craigslist’s scale where almost ~1 million ads get added every day. For example, in case of 60 million item-to-item comparisons, K-NN can take up to 11m years whereas LSH will take 14 years.

**Validation**

Most recommendation systems can use supervised or unsupervised learning methods. As most of our models have different evaluation criteria (shortest cosine distances for K-NN, shortest Euclidean distances for same-size K-means clustering, lowest RMSE for agglomerative clustering, and no easy method for LSH), it was difficult to evaluate the performance of each model against another. What we did is used the best naïve model (K-NN) as the baseline and evaluated the performance of other models against it by using hit accuracy (ratio of matching recommendations versus all recommendations) (Fig 1.3).

Fig 7.3: Relative hit accuracy of each LSH recommendation to each respective K-NN recommendation.

**Conclusion & Remarks**

Based on years of user interaction and user experience research, it is clear that most customers’ experience can be improved significantly by carrying out marginal improvements in any business’s offerings. Although Craigslist might prefer to not complicate the interface and thus the customer journey experience, it would be difficult to attract new and retain existing customers who already experience such interfaces on most other platforms. Hence, our first and most important recommendation would be to integrate a form of cross-selling and up-selling recommendation system in their ad view pages. The science is clear on that as most eCommerce and eMarketplace platforms already have integrated such systems and some big platforms like Amazon.com attribute up to 35% of their revenues to cross-selling.

Based on the recommendation systems presented by us above, we have elaborated how smart algorithms such as LSH can be designed to cost significantly lower computational time than naïve algorithms and we would recommend Craigslist to find the right algorithm to suit their scale of business and user traffic.

We would suggest Craigslist to incorporate both public data (longitude, latitude, price, etc.) as well as private data (customer segment, purchase history, etc.) to enhance the ad recommendations offered to their customers as was shown in the project with the features added from few publicly available data.

Should the system offer benefits in its pilot categories, we would recommend Craigslist to parallelize it to its most popular as well as revenue-generating categories such as job postings, rentals, cars and trucks, etc.

We observed that duplicate ads greatly reduce the quality of recommendations. For this reason, we would recommend Craigslist to change its 30-day duplicate deletion policy to a shorter duration to discourage sellers from posting them and maintain data integrity which can not just improve their recommendation systems but also other data analysis and machine learning systems.

While it was difficult for us to evaluate different unsupervised models with each of their different evaluation criteria, we would recommend Craigslist to put their resources behind doing so in a structured manner. We also recommend using real-time A/B testing using recommendations from different models to utilize one of the best and most real model evaluation tools there exists – remarkably improved customer experience.

**References**

1. **“Craigslist Business Model”**, <https://fourweekmba.com/how-does-craigslist-make-money/>
2. **“12 ways Amazon gets you to spend more”**, <https://www.cbsnews.com/media/12-ways-amazon-gets-you-to-spend-more/>
3. “**Locality Sensitivity Hashing”,** <https://towardsdatascience.com/understanding-locality-sensitive-hashing-49f6d1f6134>
4. **“ Machine Learning Basics with the** **K-Nearest Neighbors Algorithm**”, <https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>
5. **“Overview of text similarity metrics in Pyhton”,** <https://towardsdatascience.com/overview-of-text-similarity-metrics-3397c4601f50>
6. **“Guide to Latent Dirichlet Allocation”,** <https://towardsdatascience.com/latent-dirichlet-allocation-lda-9d1cd064ffa2>
7. **“Recommendation systems and the LSH algorithm”,** <https://www.di.fc.ul.pt/~vcogo/notes/2018_08_seteic_iff_lsh.pdf>