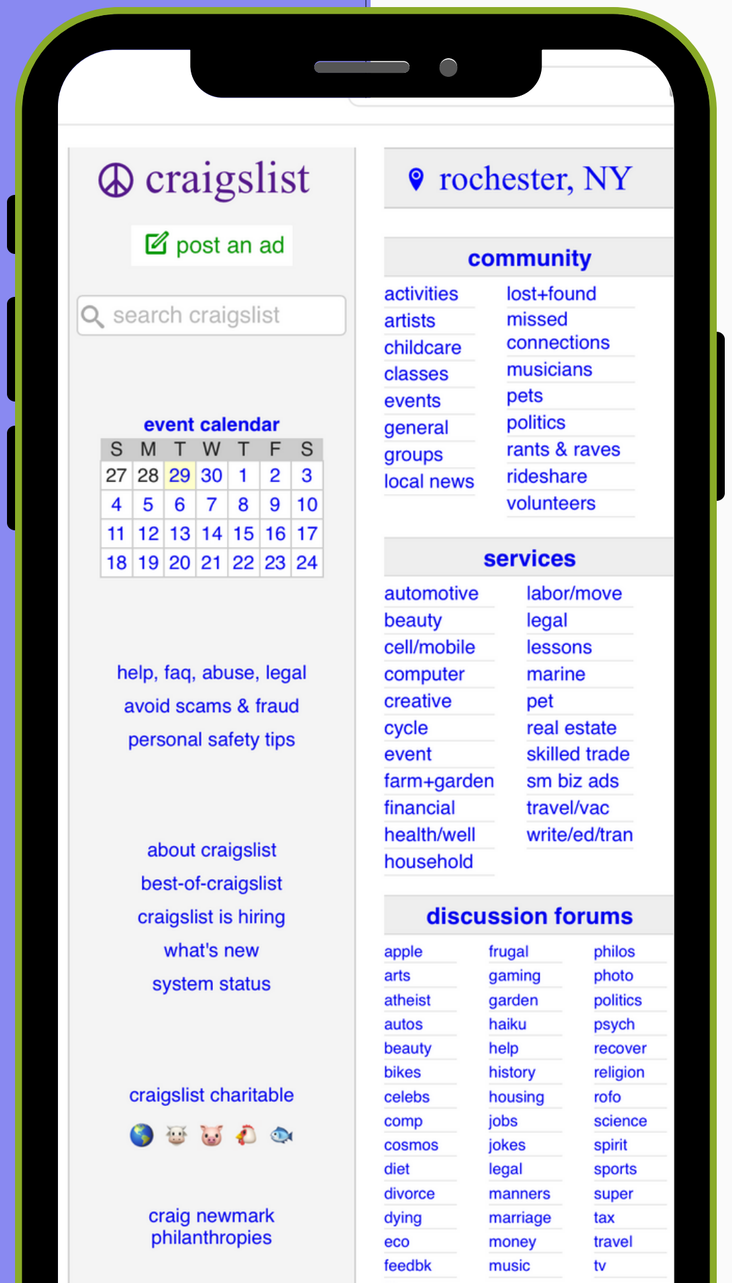
Smart Categorization of Craigslist Ads

Because the right ad belongs in the right place

Presented by Group 9
4/30/2025

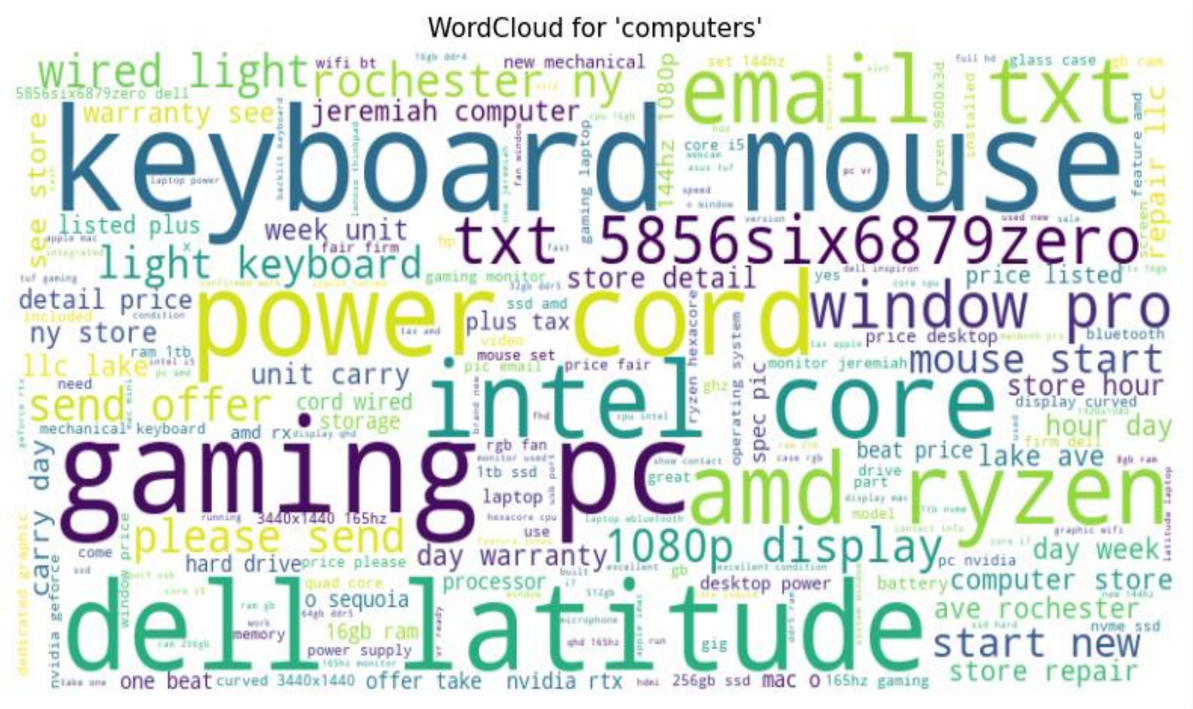


Craigslist allows free-form posting — but that flexibility leads to problems.

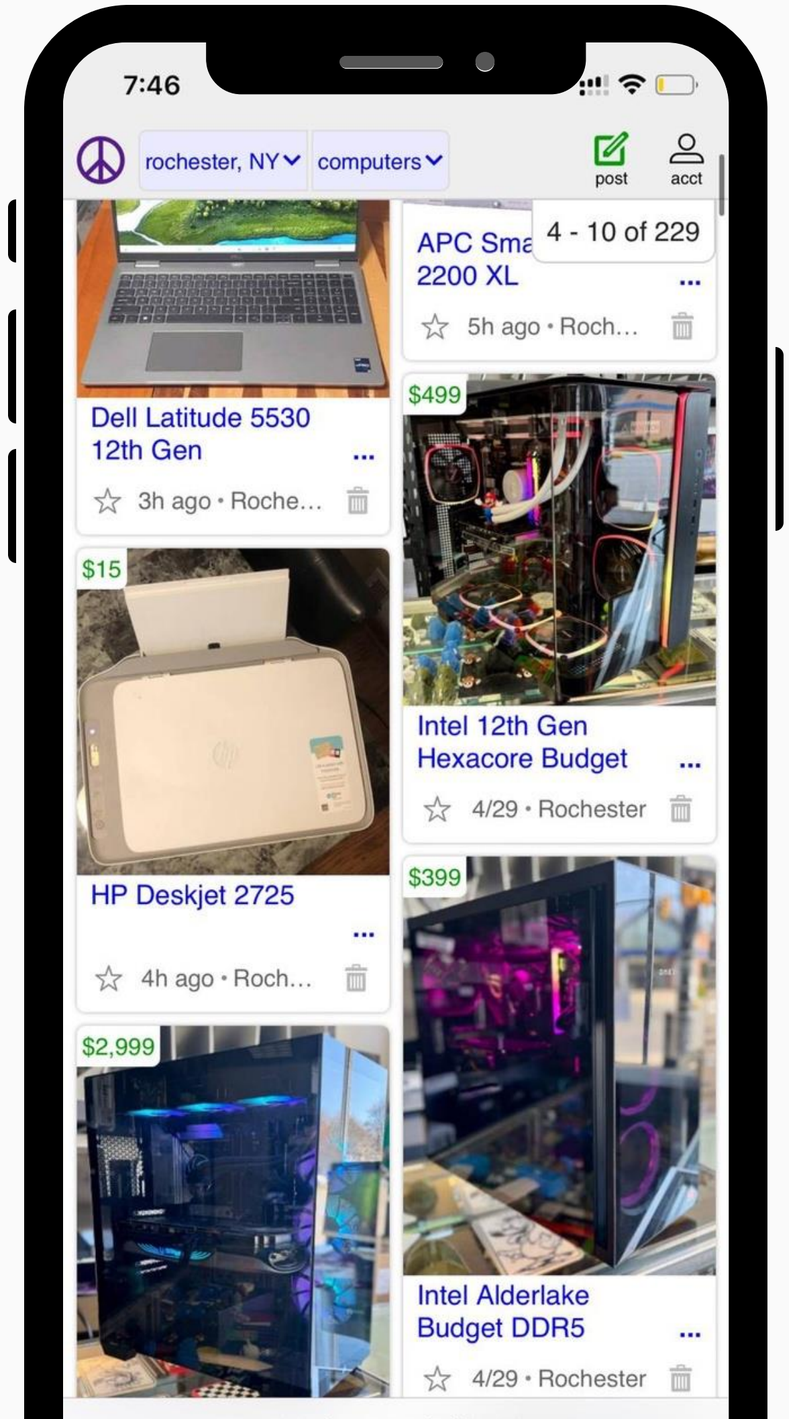
30%

Tech-related posts are posted under the wrong category.

Examples: Keyboards listed as “Computers”, Monitors and cables mixed with laptops



Result:Cluttered browsing, poor user experience, more moderation effort.



Problem

Thousands of Craigslist ads are misclassified, disrupting search and increasing moderation effort.

- Especially prevalent in the Computers and Computer Parts categories.
- Listings for accessories often appear alongside full systems leading to frustration and abandoned sessions.
- The absence of structured subcategories may reduce user trust in Craigslist's reliability as a buying platform, especially when compared to competitors with structured marketplaces

Our Solution:

A model that separates Computers from Computer-parts.

Accurate classification. Cleaner categories. Smarter Craigslist.

The Objective

Build a Smart Classifier

Automatically identify whether a listing is a Computer or a Computer Part.

Enhance Search Quality

Help users find relevant results faster with cleaner categories.

Support Moderators

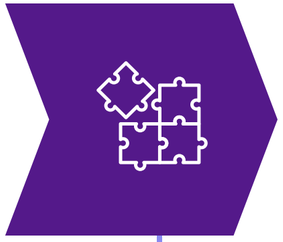
Reduce manual effort by assisting content moderation through automation.

Process Adopted

01

Web Scraping

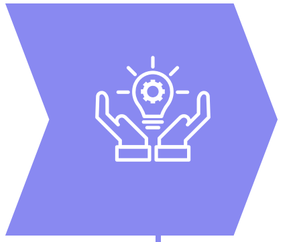
- Scraped ~700 listings from Craigslist (Rochester, NY) using Selenium and BeautifulSoup
- Focused on two categories: Computers and Computer Parts
- Extracted titles and descriptions for each listing



02

Data Cleaning

- Combined datasets and added ”human_label" column to identify the original category
- Dropped rows with missing or empty text
- Stripped whitespace and converted all text to lowercase
- Saved a combined dataset for further processing



03

Text Preprocessing

- Used NLTK to tokenize, clean, and remove stop words
- Manually selected and consolidated keywords to boost interpretability
- Created a unified "text" column for modeling (title + description)

04

Feature Engineering

- Applied TF-IDF Vectorization (max 3000 features, unigrams only)
- Manually selected domain-relevant keywords for a second model
- Created sparse matrices for both approaches

05

Data Set-up

- Performed an 80/20 Train-Test Split
- Ensured reproducibility by setting random_state = 42
- Balanced class sizes post-deduplication (302 entries total)

06

Modeling & Evaluation

- Trained and evaluated 4 models:
  - Naive Bayes (Multinomial)
  - Logistic Regression
  - Support Vector Machine
  - Random Forest
- Evaluated using accuracy, precision, recall, and F1-score
- Final Logistic Regression model achieved 87% accuracy and balanced F1-scores across both categories

Data Analysis

1 Scraping Listings

- Used BeautifulSoup to parse Craigslist search result pages from Rochester, NY.
- Used Selenium to open each listing and extract the title and description fields.
- Collected and saved ~700 listings across two categories: Computers and Computer Parts.

2 Extracting Clean Data

- Saved raw data into two CSVs: computers.csv and computer_parts.csv.
- Concatenated title + description into a unified "text" column.
- Removed rows with missing/blank text and saved the cleaned file as combined_data.csv.

3 Manual Labeling

- Created a label column based on the original category.
- Generated model predictions and compared them with manual labels.
- Added a flagged column to identify potential misclassifications for human review.

Model Performance (TF-IDF Features)

01

Multinomial Naive Bayes

02

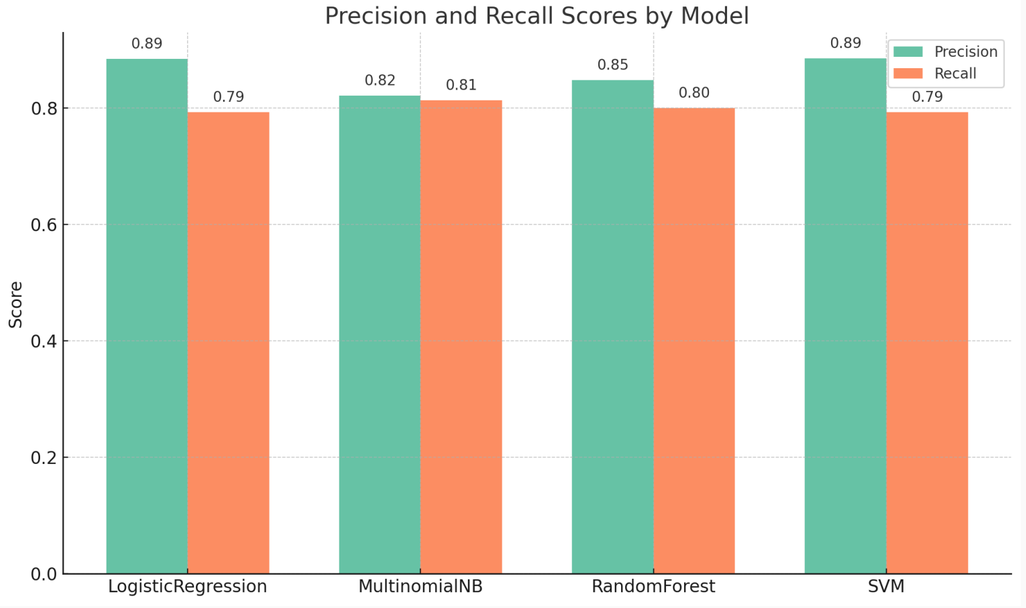
Logistic Regression

03

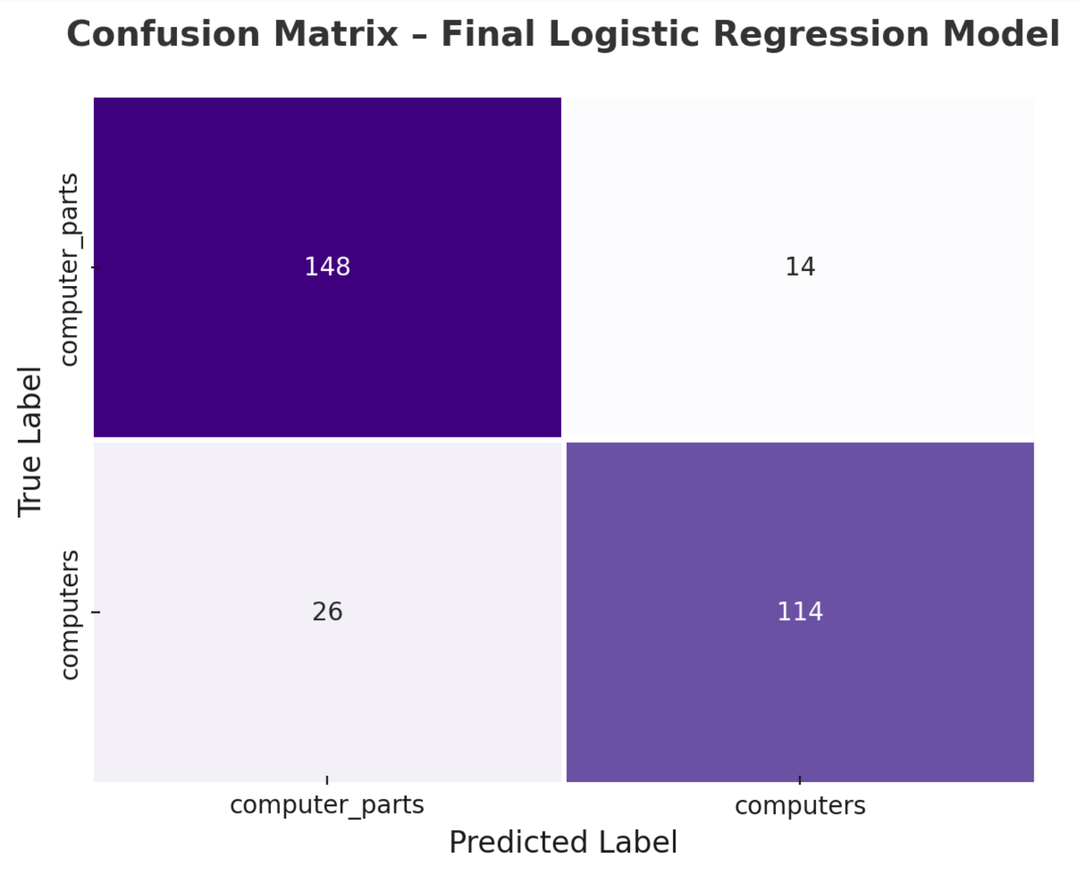
Random Forest Classifier

04

Support Vector Machine



Model Evaluation



- **The Logistic Regression model achieved an overall accuracy of 85.4% on the test set.**
- **It delivered a balanced performance with 88.5% precision and 79.3% recall, resulting in an F1 score of 83.4%.**
- **Logistic Regression was chosen as the final model due to its strong consistency, interpretability, and efficient deployment potential.**

VALUE TO CRAIGSLIST

Better Search Experience

Enhanced user experience through faster, more relevant search results.

Efficient Moderation

Reduced moderation burden, allowing teams to focus on edge cases.

Scalable Across Categories

Scalable solution applicable to other major sections like Jobs, Cars, and Housing.

Thank You!

Questions?

We’d love to hear your thoughts

Presented by Group 9