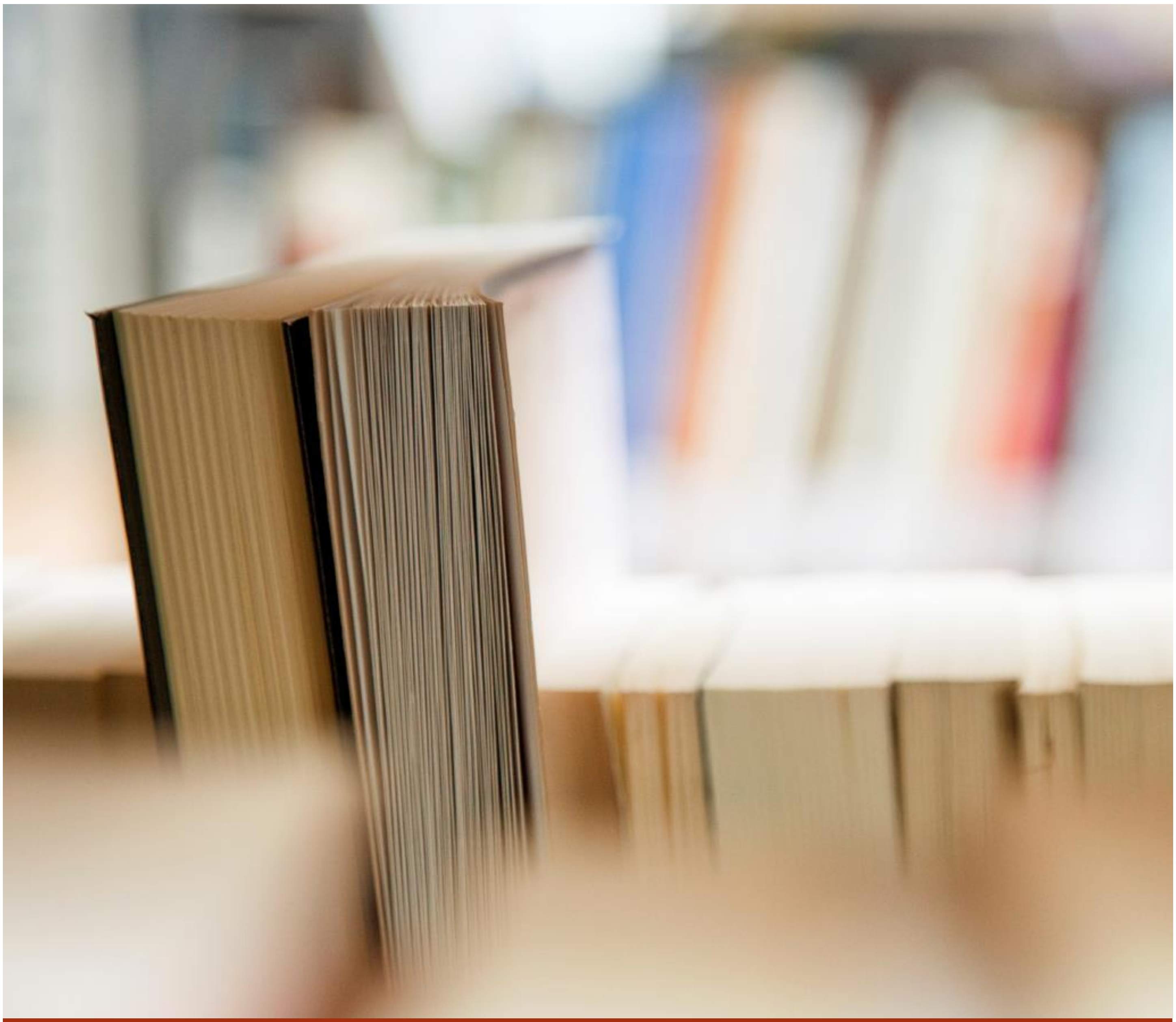


GENRE FILTER FOR BOOKS SECTION

Team ExtrAUDinary

craigslist

Shreyansh Jain
Vineet Suhas Soni
Adithya B Umamahesh
Vishal Shokeen
Sanjana Santhanakrishnan

December 8, 2021

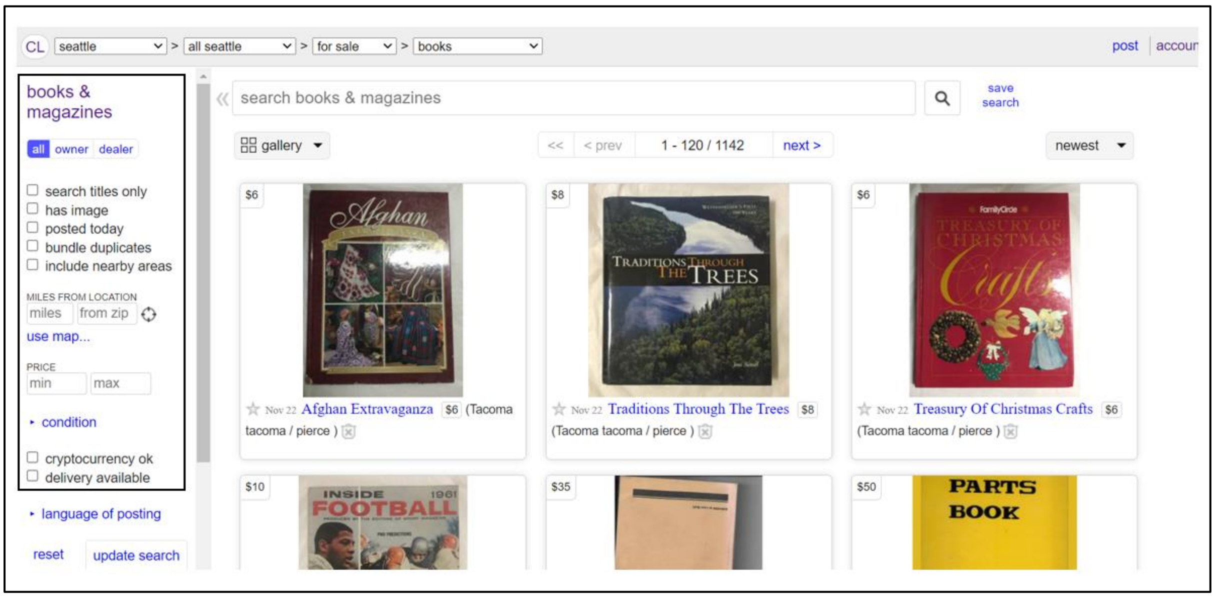
MGMT 59000-066
Analyzing Unstructured Data

Background on Craigslist

Started as a email distribution list in 1995 by Craig Newmark, Craigslist has become an American classified advertisements website with pages devoted to jobs, housing, for sale, items wanted, services, community service, gigs, résumés, and discussion forums. Since 2000, it started expanding to other locations and now covers more than 700 US cities and has also expanded to 70 countries. From 2008, it started supporting languages other than English and it also has sites specific to certain areas in larger cities. It boasts an annual revenue of 694 billion USD as of 2016 and also has more than 55 million monthly active visitors and 20 billion page views on a monthly basis.

Craigslist Books Subsection

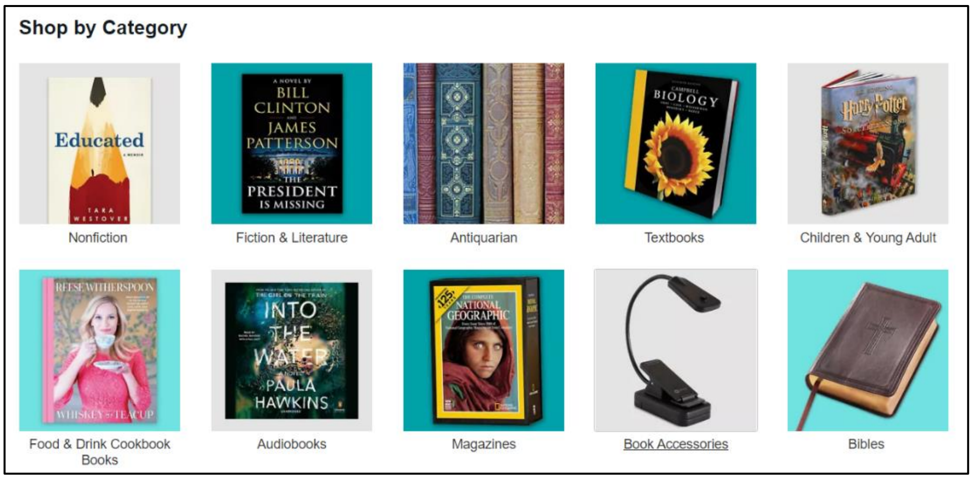
Books sub-section in ‘for sale’ section is one of the most popular sub-sections with thousands of books listed for various cities. Listing for books comprises of price, image, description, location and condition as major components of each listing as highlighted in the picture below and can be used to filter search results which enhances customer experience by displaying results pertinent to their needs.

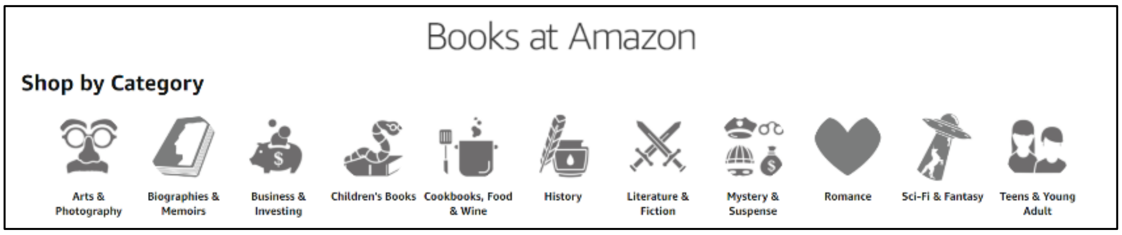


Problem Statement

One of the key aspects of any search functionality is how quickly the user is able to get the information he/she needs. In today’s fast paced internet dependent world, a user on any website has little patience to go through hundreds of listings to find what they are looking for. So, having a functionality to optimize the search plays a very important role in user experience and customer retention.

Along these lines, one of the key attributes while searching for books is its Genre and this information is currently missing on Craigslist. Books section on eBay and Amazon have the functionality as shown below for users to filter books based on its genre. We propose to add a similar functionality on Craigslist books section.





Proposed Solution

We have utilized the title of the book to build a machine learning algorithm to classify book listings into multiple genres. The predicted genre can be utilized as a filter in the books section which would help decrease buyers' search time hence improving their experience on Craigslist.

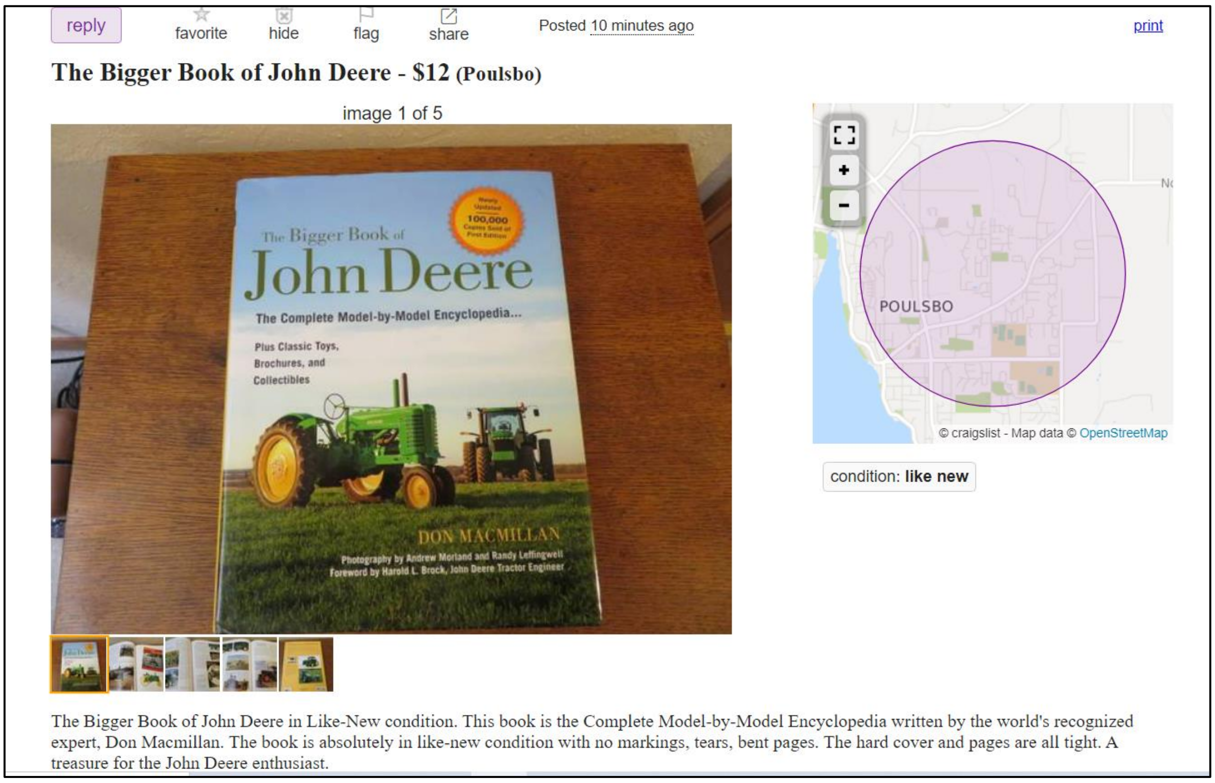
Business Analysis

The objective of our project is to solve the problem of not having genre information using text analysis and text classification methods so that we can improve user experience on Craigslist and increase the activity on the books sub-section.

There are several third-party sites like https://searchcraigslist.net/ offering search optimization on Craigslist listings. But even in these sites, the searches are customizable only for item types, location and other broad categories. They do not have the functionality to search for books of a specific genre on Craigslist.

This problem could also be solved by having owners or sellers list the genre as a mandatory field for books. But this is not a feasible solution to implement as there will have to be changes to existing listings and there may be resistance from users that have been selling on Craigslist for a long time. In addition, the main attraction for users to list their books on Craigslist is the unconventional style of utilizing unstructured information about the listing that is easy to add on the site. Another issue with this method is the subjective nature of genre. What one user tags as Education may be tagged as Children's books by another user. Hence, having the users perform the genre classification and tagging would be an unreliable solution.

Hence, we are proposing that Craigslist utilize text analytics and machine learning techniques to predict the genre of the book based on the title listed by the seller. So, once we have the genre information, this can be listed for each listing similar to how condition attribute is mentioned for the listing below.



Posted 10 minutes ago

The Bigger Book of John Deere - $12 (Poulsbo)

condition: **like new**

The Bigger Book of John Deere in Like-New condition. This book is the Complete Model-by-Model Encyclopedia written by the world's recognized expert, Don Macmillan. The book is absolutely in like-new condition with no markings, tears, bent pages. The hard cover and pages are all tight. A treasure for the John Deere enthusiast.

For the purpose of our analysis and modelling, we have chosen the book listings in Seattle as our data. The same modelling and tagging could be extended to other cities and any other categories that need categorization as well.

Data Analysis

The first step in analysing the data was to identify the data to be collected which was then followed by extracting the data from the Craigslist website.

Data Requirement

To analyze the books data, the first step was to identify a good predictor variable that could reasonably predict the genre. Hence, for this purpose book title was used as the predictor variable to create models that could classify books under various genres. And so, we required data on the book titles for each listing in Seattle.

Data Collection

The books data required was collected from the books sub-section in the Seattle Craigslist webpage by web scraping. The web scraping was done using the BeautifulSoup library in Python. Additionally, Selenium was used for browser automation with Chromedriver and getting all the listings from the Seattle books pages. The data obtained consisted of book titles, description, price and the condition for each listing. Similarly, we obtained the images posted by the sellers. The data consisted of 1106 rows of which 971 were usable and there were about 4358 images of the book listings. The next step was to manually label each of these 971 books.

Since there were only 971 total rows of data, we obtained a new dataset for training from github which was a dataset on books listed on Amazon which had 207,000 rows on book cover images, title, author, and genres for each respective book. These books were already categorized into 32 different genres. The data obtained from Craigslist was also manually labelled into these 32 categories. The next step was to clean and process the data before modelling.

Data Cleaning and Preprocessing

The training data on Amazon listings was first processed by combining the title and author columns into one. Then the Amazon training data and Craigslist data were combined to do the following cleaning and pre-processing steps –

- Grouped the 32 genres into 14 genres by combining related ones.
- Label encoded the genres so that there are 14 label values.
- Tokenized each book title using the ‘WhitespaceTokenizer’ function from the ‘nltk’ library.
- Removed all the stop words and punctuations in tokens generated
- Created a TF-TDF matrix using the ‘TfidfVectorizer’ with minimum document frequency 2.
- Split the training Amazon and test Craigslist data again.

Model Training and Validation

Once we had the training and test data, the first step was to create a validation set for training the model. For this, we used the ‘train_test_split’ function from scikit-learn and split the training data into 70% training and 30% validation sets. Now this 70% training data was used for building the models. For each of these models, we performed hyperparameter tuning to get the best fit for the data while ensuring there was no overfitting by comparing the training accuracy scores with those of validation. Here, the validation set was the remaining 30% of the Amazon books data. The details of the models we used are given below –

Naïve Bayes Model

The Naïve Bayes model is a simple probabilistic classifier that works on the principle of Bayes’ theorem. It is fairly easy to train with supervised learning and is scalable. This was one of the

models that took the least time to train since it required comparatively less computing power. Even though it is a simple model, it outperformed some of the other advanced ones.
For the genre classification, the vectorized Amazon training data was fed to MultinomialNB method from sklearn.naive_bayes and the model was trained. After tuning the smoothing parameter α, it yielded a training accuracy of 79.5% and validation score of 70.8%.

Logistic Regression Model

The next model that we trained was a simple Logistic Regression model. This model just uses the logit function to predict the genre classes. It is also computationally inexpensive and works well for a larger training data. For this model, we used the LogisticRegression method from sklearn.linear_model. This gave a training accuracy score of 73.7% and a validation score of 69.1%. In this model, we tuned using the parameter C.

Decision Tree Model

The decision tree model tries to predict the value of a class based on a set of input variables by building a tree-like flowchart structure. We used the DecisionTreeClassifier method from sklearn.tree and trained the model. We varied parameters like the maximum depth, the minimum number of samples at the leaf node etc. and finally reached a training accuracy of 71.2% and a validation score of 56%.

Random Forest Model

The random forest model works on an ensemble of decision trees and selects the class that is voted for by most trees. We used the RandomForestClassifier method from sklearn.ensemble and trained the model. We varied parameters like the maximum depth, the number of trees, the maximum number of features for splitting and finally obtained a training accuracy of 69.3% validation score of 54.7%. The challenge with this model was its computational complexity and training time.

Neural Networks Model

The next model that we trained was the neural network model using the MLPClassifier method within sklearn.neural_network. This model is a Multi-Layer Perceptron classifier that works well for very large datasets. By tuning the hidden layer size, activation function and weight optimization, we obtained a training accuracy score of 79.6% and a validation score of 71.9%.

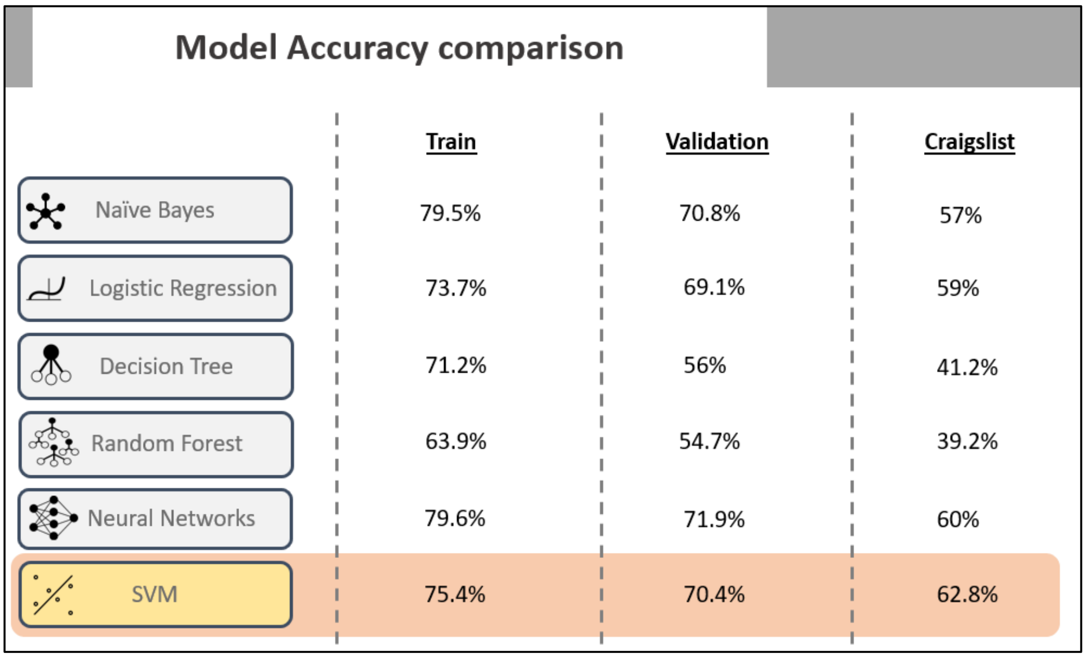
Support Vector Machine Model

The last model that we trained was the Support Vector Machine model which works by constructing a hyperplane to classify the data. It works well even with high-dimensional data. By varying the regularization parameter C for optimal fit, we obtained a training accuracy of 75.4% and a validation score of 70.4%.

The next step was to see if these models performed well on unseen Craigslist set data.

Test Data Validation

The real test for our trained models was to see how they performed on the Craigslist books data. The Amazon training, validation and Craigslist test accuracy scores are given below –



Model Accuracy comparison

	Train	Validation	Craigslist
Naïve Bayes	79.5%	70.8%	57%
Logistic Regression	73.7%	69.1%	59%
Decision Tree	71.2%	56%	41.2%
Random Forest	63.9%	54.7%	39.2%
Neural Networks	79.6%	71.9%	60%
SVM	75.4%	70.4%	62.8%

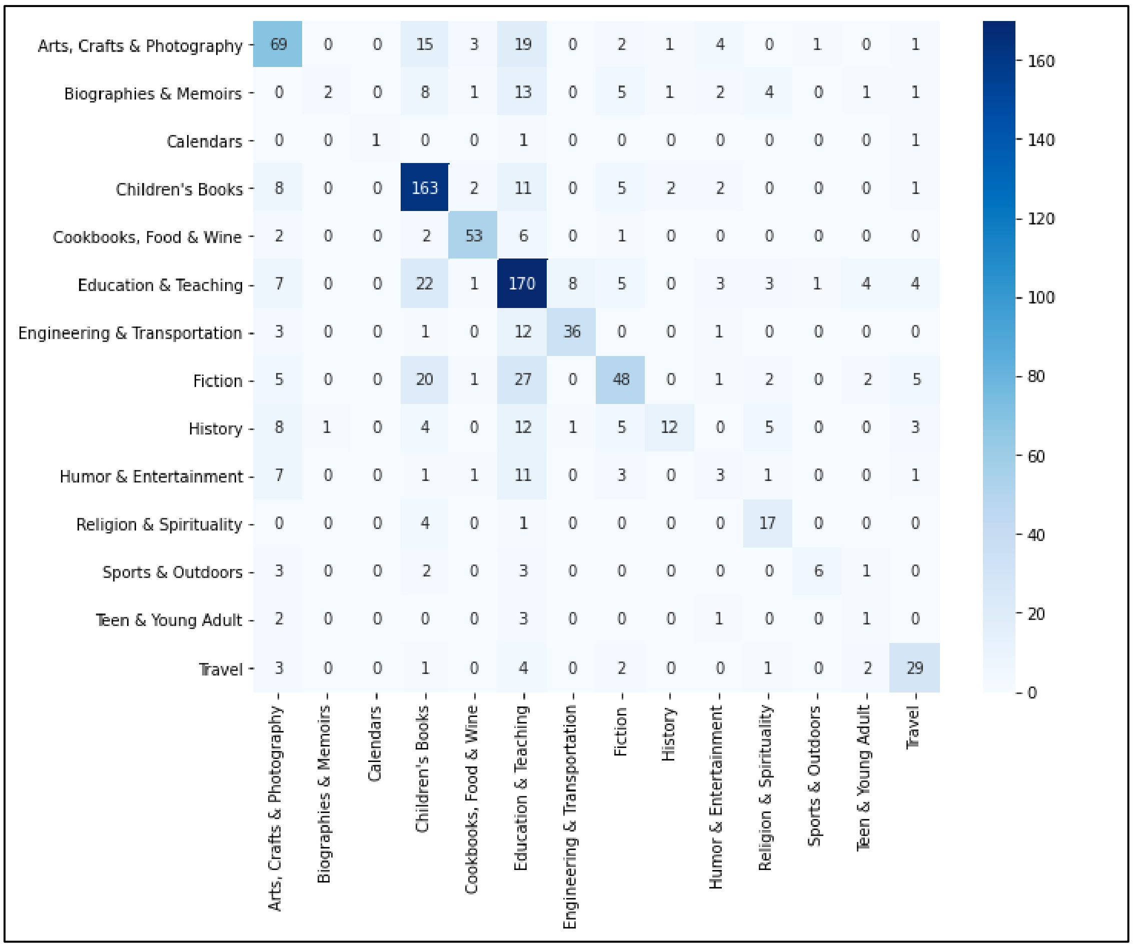
As seen above, almost all models have a good accuracy score for the training data and the value drops for the validation dataset. Based on the Craigslist data accuracy scores, we can see that the accuracy scores have dropped even further. Since we are predicting 14 different classes, the Naïve Bayes model and the Logistics Regression model have not worked well. With decision tree, it seems like a classic case of overfitting on the training data. With random forest, to avoid overfitting the parameters were tuned but the model then started underfitting to the data. Neural network model would have worked better if we had had a much larger training dataset and there seems to be quite a bit of difference in the 3 scores.

Hence, the best model obtained here is the Support Vector Machine model which had little difference between the training and validation scores and also performed the best on the Craigslist test data with an accuracy of 62.8%.

Challenges

Even though SVM performed pretty well, there was still a significant difference in accuracy scores between validation and test sets and so we decided to look at the confusion matrix to understand which books were misclassified the most.

Based on the confusion matrix given below, we can see that Fiction and History books were misclassified the most.



	Arts, Crafts & Photography	Biographies & Memoirs	Calendars	Children's Books	Cookbooks, Food & Wine	Education & Teaching	Engineering & Transportation	Fiction	History	Humor & Entertainment	Religion & Spirituality	Sports & Outdoors	Teen & Young Adult	Travel
Arts, Crafts & Photography	69	0	0	15	3	19	0	2	1	4	0	1	0	1
Biographies & Memoirs	0	2	0	8	1	13	0	5	1	2	4	0	1	1
Calendars	0	0	1	0	0	1	0	0	0	0	0	0	0	1
Children's Books	8	0	0	163	2	11	0	5	2	2	0	0	0	1
Cookbooks, Food & Wine	2	0	0	2	53	6	0	1	0	0	0	0	0	0
Education & Teaching	7	0	0	22	1	170	8	5	0	3	3	1	4	4
Engineering & Transportation	3	0	0	1	0	12	36	0	0	1	0	0	0	0
Fiction	5	0	0	20	1	27	0	48	0	1	2	0	2	5
History	8	1	0	4	0	12	1	5	12	0	5	0	0	3
Humor & Entertainment	7	0	0	1	1	11	0	3	0	3	1	0	0	1
Religion & Spirituality	0	0	0	4	0	1	0	0	0	0	17	0	0	0
Sports & Outdoors	3	0	0	2	0	3	0	0	0	0	0	6	1	0
Teen & Young Adult	2	0	0	0	0	3	0	0	0	1	0	0	1	0
Travel	3	0	0	1	0	4	0	2	0	0	1	0	2	29

In addition, we found the following issues with the listings on Craigslist –

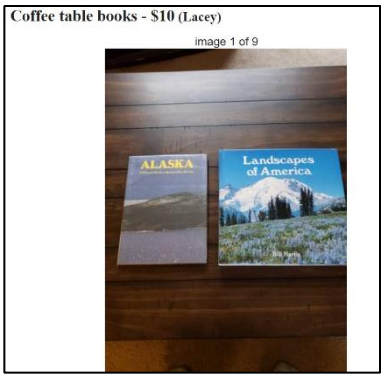
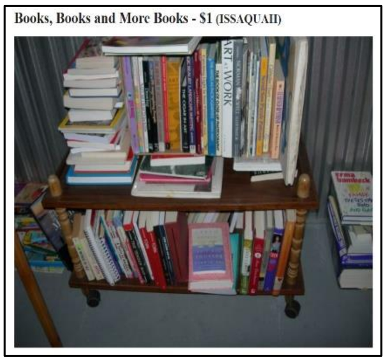

Coffee table books - $10 (Lacey)

image 1 of 9

1. **Ambiguous titles (15%)**
   For example, the title here says 'Coffee Table books' so the models were classifying this as a Cookbook instead of the actual Travel genre.
   A solution could be identifying the titles from the images.


Books, Books and More Books - $1 (ISSAQUAH)

2. **Multiple books in one listing (5%)**
   Many listings have multiple books listed and the model has not been able to classify these.
   One solution for this could be tagging the books as a new category but this category would have to be added to the training data for the models to work.

3. **Misspelled titles (2%)**
   Certain listings did not have the title listed with the right spelling. This also caused the model to misclassify these. Here, the word 'Children' is misspelled, and the model was unable to recognize it. For this also, images could be used for retrieving titles.

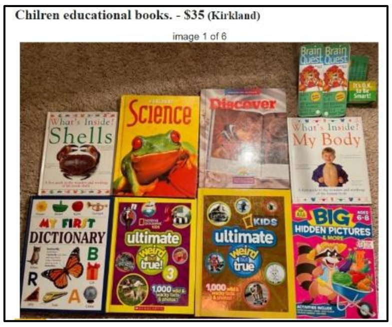

Chilren educational books. - $35 (Kirkland)

image 1 of 6

4. **Multiple genres (2%)**
   Certain books could fall into multiple genres, and these were misclassified by the model. For example, 'Alice in Wonderland' could be classified as Children's books as well as Fiction.
   A solution for this could be soft-tagging the books under both categories and classifying based on probabilities.

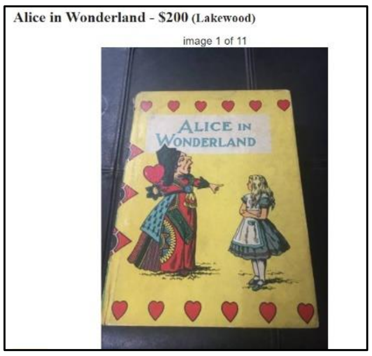

Alice in Wonderland - $200 (Lakewood)

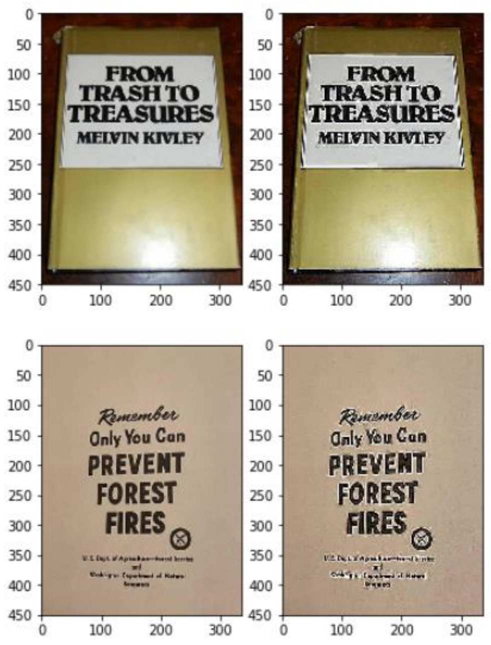
image 1 of 11

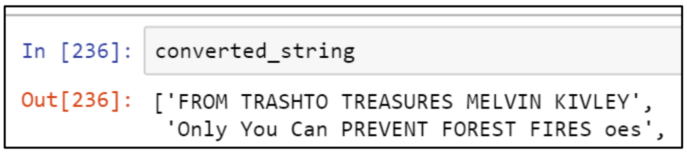
Additional Analysis

In order to solve the problem with the titles, we used OCR (Optical Character Recognition) to identify the book titles from the listing images and compared it with the listing titles. The following steps were involved –

- Preprocess the images to improve clarity
- Use Tesseract to convert image to text
- Tokenize and process text
- Validate it with listing title

Here, the ImageEnhance method from PIL was used to improve the sharpness of the image and the corresponding title was obtained using pytesseract with customized options.





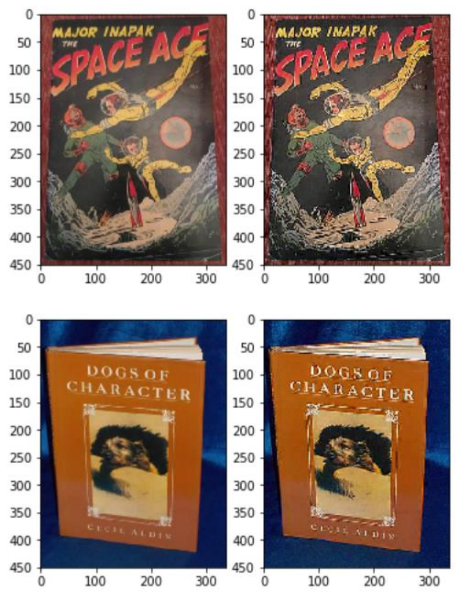
```
In [236]: converted_string

Out[236]: ['FROM TRASHTO TREASURES MELVIN KIVLEY',
           'Only You Can PREVENT FOREST FIRES oes',
```

But even this method had a few pitfalls and was not accurate due to the following issues –

- Not all listings had images
- Poor quality images posted
- Unconventional fonts on titles were not recognized
- Titles of multiple books in a listing could not be identified

These issues caused poor results as shown below –



'OS ppg hese **MAJOR INAPAK** PE Mh THE jaye ZA el AGE ea eds Lan hi gg Ny Ra ae SES Seen Sn fe yA sige San NCAA Saar',

'**DOGS OF CHARACTER** E Poe oe eo SE CUES'

This OCR text recognition is just a start and needs additional image processing and training using image data to give better accuracy. This could be one of the ideas that could be an add-on to the genre classification that Craigslist could look into.

Conclusion and Future Scope

Book genre classification would be an important value add to Craigslist's books sub-section. It would reduce the time spent by buyers on searching for books drastically and this would improve user experience and customer retention as customers are more likely to come back if they find what they are looking for quickly. Additionally, utilizing OCR for retrieving book titles would enhance the quality of information that is available to buyers on Craigslist and reduce the burden on advertisers to be accurate in their listings. Further, the same classification concept can be extended to other sections of Craigslist such as arts and crafts, electronics etc. Implementing these functionalities would still retain the simple form of the Craigslist site which has been one of its core principles while still enhancing both buyer and advertiser experience in the long run.

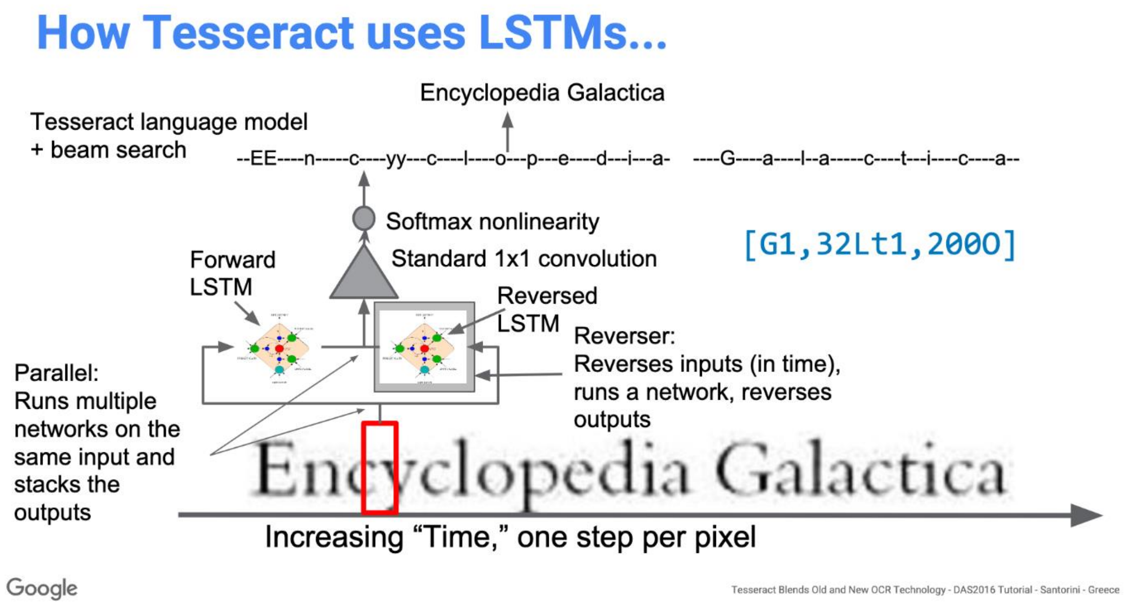
Appendix

Py-Tesseract is a wrapper for ***Google's Tesseract-OCR Engine.***

Tesseract OCR

Convert IMG to TXT using Pytesseract
JPG
–>
–>
TXT

It is a standalone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries.



How Tesseract uses LSTMs...
Encyclopedia Galactica
Tesseract language model
+ beam search
--EE----n-----c----yy---c----l----o---p---e----d---i---a-   ----G----a----l--a-----c----t---i----c----a--
Softmax nonlinearity
Standard 1x1 convolution
[G1,32Lt1,200O]
Forward
LSTM
Reversed
LSTM
Reverser:
Reverses inputs (in time), runs a network, reverses outputs
Parallel:
Runs multiple networks on the same input and stacks the outputs
Encyclopedia Galactica
Increasing "Time," one step per pixel
Google
Tesseract Blends Old and New OCR Technology - DAS2016 Tutorial - Santorini - Greece

References

https://en.wikipedia.org/wiki/Craigslist

https://www.forbes.com/sites/forbesagencycouncil/2017/10/30/the-value-of-search-results-rankings/?sh=624ea3f444d3