Car Value Evaluator

Spencer Horrocks

Executive Summary

Classified ads are still among the most popular methods of selling and purchasing used items. However, there are often discrepancies between sellers and buyers as to making a good deal. This program was created to aid buyers in being able to identify good deals and thus saving time in their searches for new cars.

The basis of the program searches local classified car advertisements from the popular website auto.yahoo.com based on a given ZIP code. The results of the query are then transfered into an excel sheet and key information such as the year, miles and price of the car are extracted. Customer criteria such as make, model, and price range are taken into consideration. Listings that meet criteria and seem generally seem like "good deals" will be marked as such colored into one of three categories. The program will also look for abnormalities in the listing, such as mistakes in mileage, and warn users that a possible problem could exist in a certain listing and further research will still need to be done for more accurate information on a certain listing. Those listing that do not meet customer criteria or generally seem like "bad deals" will be noted as such in the program and colored and sorted for the user to be aware of.

The end goal has been to improve the experience for buyers of used cars and try to quickly and visually determine if a car is a good deal. This program will aid in that process and may even help sellers to create listings of higher quality.

Implementation Documentation

In order to implement the best solution, it was imperative to find an adequate site to browse for local listings based on user defined specifications. Sites that were considered included Craigslist, Yahoo Autos, KSL and Kelly Blue Book. For the purposes of this project, Yahoo Autos was chosen as the sources for classified listings of automobiles. This site was selected due to the ease of locating listings from a web query and also for the ability to search for national results as opposed to only local listings.

After the sources site had been selected, a basic user form was created in order to help facilitate the specific specifications such as the user's preference to the make and model of a car as well as the price range and zip code for where the user would be looking in. This user form would aid in either making a broad search of any type of car within the user’s requested zip code or narrowing search results to match a specific type of car.

Next, the code was created in order to perform the actual process of setting up a web query that would pull in the information from the Yahoo Auto site and list only the relative information from the listings such as mileage and price and ignore less relevant information such as distance from zip code. This information is put into a new table and then checked to ensure accuracy and integrity. If the information is found to be incomplete or incorrect, it is removed from the table.

Once the data is collect and integrity is ensured, the program runs a defined algorithm wherein the data is tested against the user specifications and data is colored in one of three categories to categorize listings as "good deals", "bad deals", or as having problems with the. A green color denotes a listing that has a reasonable amount of miles and price, a yellow color signifies that there is something suspicious about one or more of the numbers listed, or that the car might suffer from high mileage which could mean a higher probability of a break down, and a red listing means that the car is outside the user's specified budget.

Lastly, a link is posted at the bottom of the table which will direct the user to the page that the information was taken from. This link will allow the user quick access to other offerings on the site that are outside the scope of this project such as listing contact information for the car and generating a CARFAX report.

Difficulties encountered

The first difficult that needed to be overcome within the scope of this project was attempting to find a good classified listing site to pull the web query from. The site needed to have a large amount of listings both locally and nationally, and also needed to be simple to create and organize the data. Sites like Kelly Blue Book required a lot of information in order to search for listings and delivered complicated results. This posed a problem as it would need more error handling and would require additional research about cars to be performed. KSL was fairly simple for web queries, however the site only contained local listings. This would require a bit more error handling to ensure aid the users to only input valid Utah zip codes and could be frustrating if someone in Utah wanted to search for a car outside of the state. After a bit of experimentation with Yahoo Autos, I found that it wouldn't be completely ideal, however it would be a decent option due to the layout of the site and the ease of manipulating the URL.

The next difficulty that was faced in the score of this project was Yahoo Auto's page changing the format on me. Much of the code was based on the fact that the relevant information would be in the same general location on the page. However shortly after creating my loops to grab the listings Yahoo Autos changed the layout of their webpage which in turn changed the location of the relevant information. In order to deal with this problem search capabilities and general error handling has been added to help maintain the integrity of the data and usefulness of the program.

Another problem that occurred during the course of the project was simply how to qualify listings as being a "good" or "bad" listing. Originally the idea involved taking the data from the listing and comparing it to information given on a website like Kelly Blue Book. As mentioned before the biggest problem was the sheer amount of information needed in order to perform such a web query. In some cases, the amount of information needed simply wouldn't be available from the listings given. This would be a consistency problem with the program lowering the perceived value by the user. This issue was resolved by the creation of a user form which allowed the user to specify what they were looking for in a used car.

One of the biggest problems in the creation of this problem was the amount of time required to debug the program. Nearly half the bugs encountered in this program would cause Excel to crash and needed to be restarted. For extremely difficult bugs, it wasn't rare for Excel to need to be restored upwards of 15 times. It would usually take around 60 seconds to perform this task which substantially increased overall time spent on this project.

One final issue that needed to be dealt with was every so often the seller would have impossible or improbable values. These cases were not necessarily "bad" listings but couldn't be considered "good" either. To resolve the problem incurred by this use case, a third category was added that would inform the user of incomplete or possibly incorrect data. That way, if they so desired they would be able to contact the seller to investigate the listen further.

Assistance

Apart from sites such as Stack Overflow for general knowledge on VBA, no additional outside assistance was used in this project.