Senior Project – Shopping Assistant

Hsin-Ting Chou (Justin)
Department of Electronic Engineering
National Taipei University of Technology
Taipei, Taiwan
and
Department of Computer Science
University of Cincinnati
Ohio, United State
t106360211@ntut.org.tw

Chia-Wei Hsu (Jarwy)
Department of Electronic Engineering
National Taipei University of Technology
Taipei, Taiwan
and
Department of Computer Science
University of Cincinnati
Ohio, United State
t106360229@ntut.org.tw

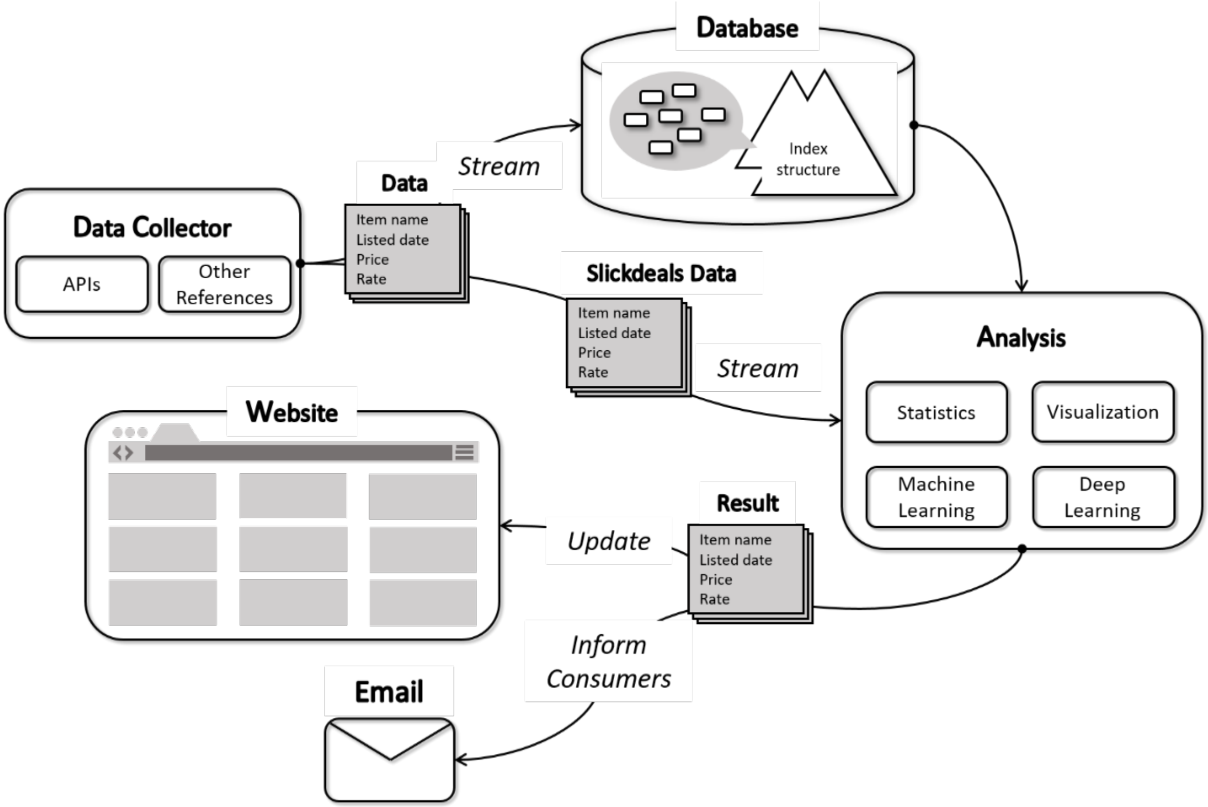
ABSTRACT

Imagine that today you are going to buy a product, as usual, you go to the browser and search for the item that you yearn day and night. After you stay in front of the computer all day without doing another thing, you realized that there are lots of excellent deals in the enormous internet market which are not only high-quality but also cheap. However, to find a perfect deal will spend a lot of time and people cannot always find a great deal while there are too many deals to look over and compare. There should be someone telling you where could you get a better deal. In this project, we are going to establish a shopping assistant system, which can help you to find cheaper and high-quality deals without spending your time so that you can enjoy the time with your family and friends, have some drinks and have a wonderful life.

1. INTRODUCTION

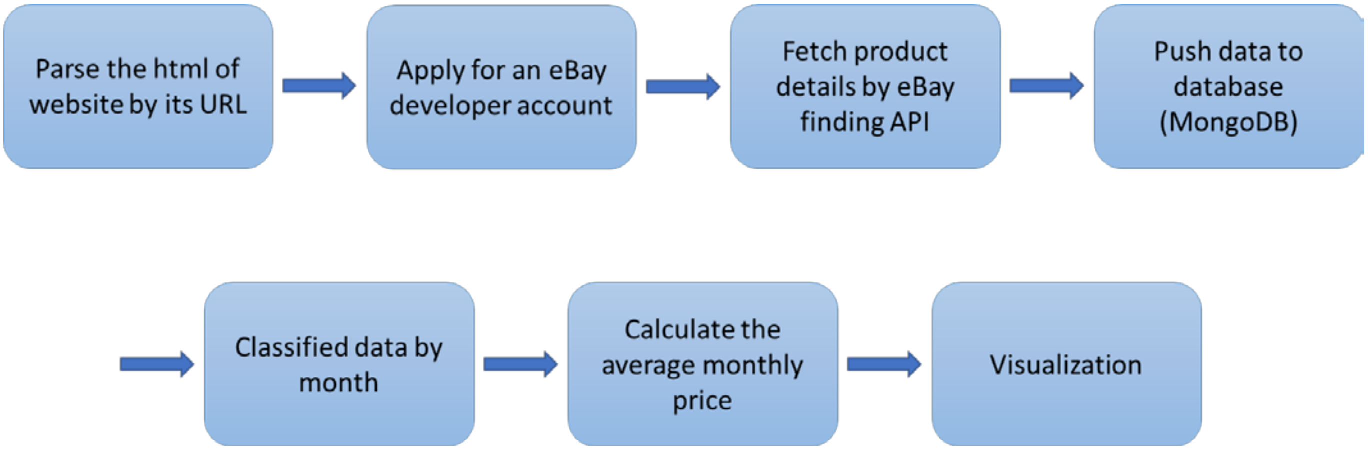
To compare the prices, we first need to find the prices of the product in different sources, collect and store them into a database, so that we can analyze for consumers. There are several websites are doing similar work, such as booking.com, Agoda, Airbnb, and Expedia. However, all of them are comparing housing and hotel prices. None of them can provide price information for other products. What they have done is doing web crawler and show them to the consumer. What if those websites get a price which is the lowest on the market right now but the highest in the price history? Consumers may buy it because of the lowest price but they do not realize that actually they spend too much money on this product. In our shopping assistant system, we are going to show the price history and the consumer reports of the product, and then analyze the quality of the product, also, telling consumers when will be the best time to buy it. There are several difficulties in our project, First, we need lots of price information to make sure we can analyze, so we need to find a methodology which can keep collecting the price of any certain product from different sources, such as web crawler or analyzing RSS. Second, we need a database that can store a huge amount of data while it will not be crashed at the same time. Third, we need to analyze the prices from different sources, and then show the price history and possible trend by statistics or data mining algorithms such as linear regression. Fourth, we need to inform the consumers when it will be a better time to buy the product by sending them emails or other approaches. The only thing consumers need to do is telling the shopping assistant system what you are looking for, then it will provide a chart in which the x-axis represents time and the y-axis represent the price. In this graph, the historical price will be represented by a line, and real-time items collected from major shopping website will represent as dots, one dot represents one item. If a consumer takes price as the only criterion to judge whether an item worth buying or not, then those item points above the historical price line are not worth buying because their price is higher than the market price. On the other hand, those dots under the line are worth considering. With the help of a shopping assistant, consumers can find a cheaper item without spending time looking at the deals by themselves.

2. FLOW CHART



3. API

3.1. eBay API.


Parse the html of website by its URL
Apply for an eBay developer account
Fetch product details by eBay finding API
Push data to database (MongoDB)
Classified data by month
Calculate the average monthly price
Visualization

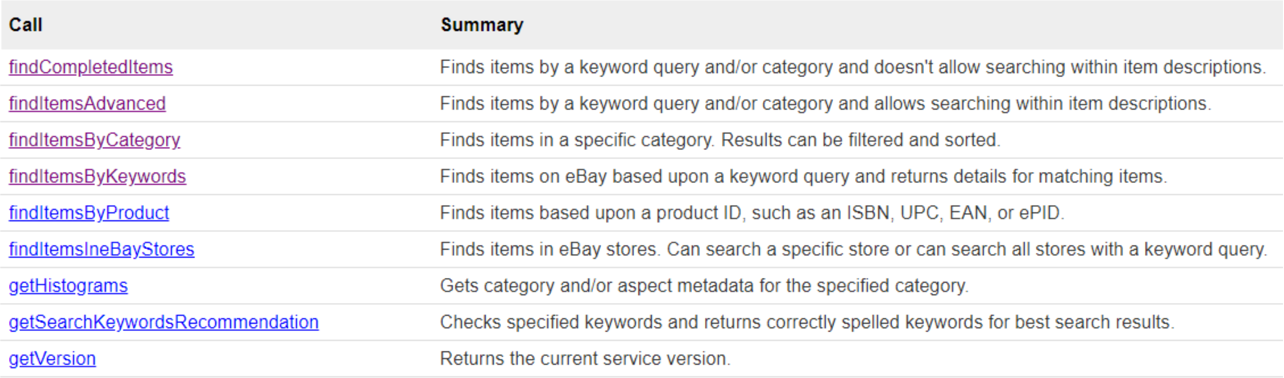
The flow chart above shows the steps we took for eBay analyzing part. In the beginning, we tried to parse the whole web page and used tools to fetch the specific information about the products we are really interested in with the URL of the web page. It works, but it is not efficient enough. We then turned to find the API of eBay, hope to enhance the efficiency.

In order to get the key to access API, we created an eBay developer account. After we dived in the world of API, we found out that there are multiple calls of Finding API. We have tried every call, and finally we pick two which meet our require. One is "findItemsAdvanced" which has many filters that can help us to find specific product. Another is “findCompletedItems”, this call searches for the items which are completed (include those sold items and expired items). These two calls can help us to reach our goal. The diagram of API parameters is shown in Fig 1.

Every good have its’ own selling status. We get the data of those goods with the selling status we want (Active, Ended, EndedWithSales and EndedWithoutSales), and use a different number to represent each status. We use “0” to represents those Active goods, “1” for those sold items and “2” for those products which was not sold and expired. The diagram of selling status is shown in Fig 2.

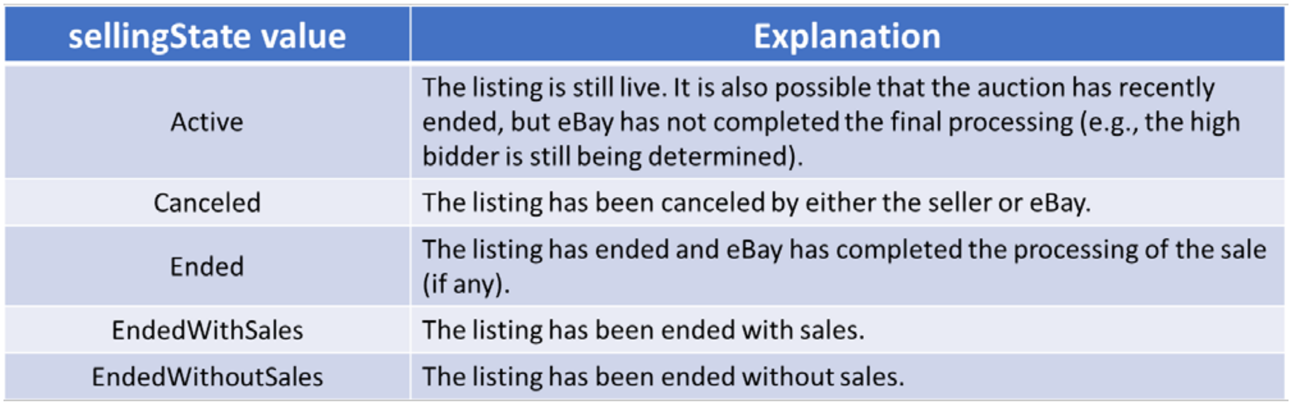
After we successfully fetched the details of items by API, we started to think of where to store these gradually increasing data. Finally, we chose MongoDB as the platform to store the data. In order not to push repeated data to the database, every time before we insert data, we used a unique item number (ID) as the key to make sure one item is in the database or not. If the data we want to push doesn’t exist in the database, we insert it and return the messages with insert time. The diagram of data base is shown in Fig 3.

With enough data in data base, we stared to analyze them. We classify all the items by month and calculate the average monthly price. According the results, the average price of SSD with 128GB capacity has been hovering between 15 to 40 US dollars. The diagram of the average monthly price of 128 GB SSD from 2018 till now is shown in Fig 4, 5 and 6.



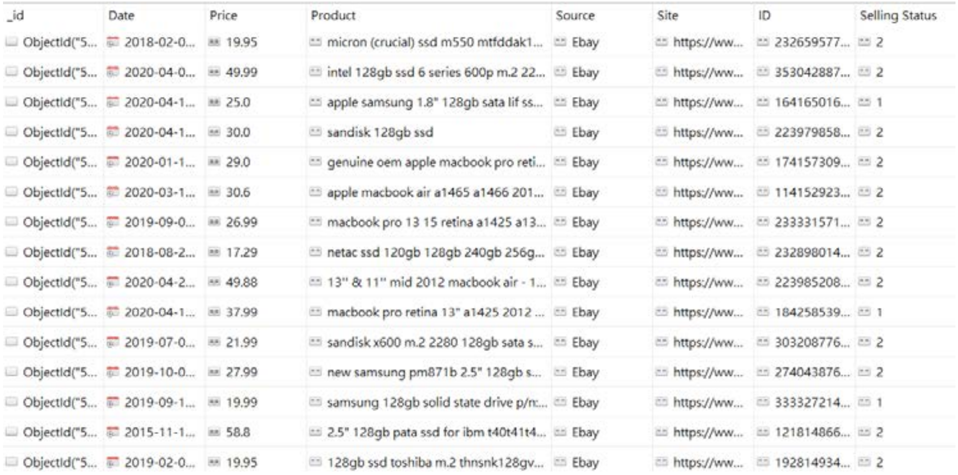
Call	Summary
findCompletedItems	Finds items by a keyword query and/or category and doesn't allow searching within item descriptions.
findItemsAdvanced	Finds items by a keyword query and/or category and allows searching within item descriptions.
findItemsByCategory	Finds items in a specific category. Results can be filtered and sorted.
findItemsByKeywords	Finds items on eBay based upon a keyword query and returns details for matching items.
findItemsByProduct	Finds items based upon a product ID, such as an ISBN, UPC, EAN, or ePID.
findItemsIneBayStores	Finds items in eBay stores. Can search a specific store or can search all stores with a keyword query.
getHistograms	Gets category and/or aspect metadata for the specified category.
getSearchKeywordsRecommendation	Checks specified keywords and returns correctly spelled keywords for best search results.
getVersion	Returns the current service version.

Fig 1. The diagram of API parameters



sellingState value	Explanation
Active	The listing is still live. It is also possible that the auction has recently ended, but eBay has not completed the final processing (e.g., the high bidder is still being determined).
Canceled	The listing has been canceled by either the seller or eBay.
Ended	The listing has ended and eBay has completed the processing of the sale (if any).
EndedWithSales	The listing has been ended with sales.
EndedWithoutSales	The listing has been ended without sales.

Fig 2. The diagram of selling status is shown in Fig x



_id	Date	Price	Product	Source	Site	ID	Selling Status
ObjectId("5...	2018-02-0...	19.95	micron (crucial) ssd m550 mtfddak1...	Ebay	https://ww...	232659577...	2
ObjectId("5...	2020-04-0...	49.99	intel 128gb ssd 6 series 600p m.2 22...	Ebay	https://ww...	353042887...	2
ObjectId("5...	2020-04-1...	25.0	apple samsung 1.8" 128gb sata lif ss...	Ebay	https://ww...	164165016...	1
ObjectId("5...	2020-04-1...	30.0	sandisk 128gb ssd	Ebay	https://ww...	223979858...	2
ObjectId("5...	2020-01-1...	29.0	genuine oem apple macbook pro reti...	Ebay	https://ww...	174157309...	2
ObjectId("5...	2020-03-1...	30.6	apple macbook air a1465 a1466 201...	Ebay	https://ww...	114152923...	2
ObjectId("5...	2019-09-0...	26.99	macbook pro 13 15 retina a1425 a13...	Ebay	https://ww...	233331571...	2
ObjectId("5...	2018-08-2...	17.29	netac ssd 120gb 128gb 240gb 256g...	Ebay	https://ww...	232898014...	2
ObjectId("5...	2020-04-2...	49.88	13" & 11" mid 2012 macbook air - 1...	Ebay	https://ww...	223985208...	2
ObjectId("5...	2020-04-1...	37.99	macbook pro retina 13" a1425 2012 ...	Ebay	https://ww...	184258539...	1
ObjectId("5...	2019-07-0...	21.99	sandisk x600 m.2 2280 128gb sata s...	Ebay	https://ww...	303208776...	2
ObjectId("5...	2019-10-0...	27.99	new samsung pm871b 2.5" 128gb s...	Ebay	https://ww...	274043876...	2
ObjectId("5...	2019-09-1...	19.99	samsung 128gb solid state drive p/n:...	Ebay	https://ww...	333327214...	1
ObjectId("5...	2015-11-1...	58.8	2.5" 128gb pata ssd for ibm t40t41t4...	Ebay	https://ww...	121814866...	2
ObjectId("5...	2019-02-0...	19.95	128gb ssd toshiba m.2 thnsnk128gv...	Ebay	https://ww...	192814934...	2

Fig 3. The diagram of the data base

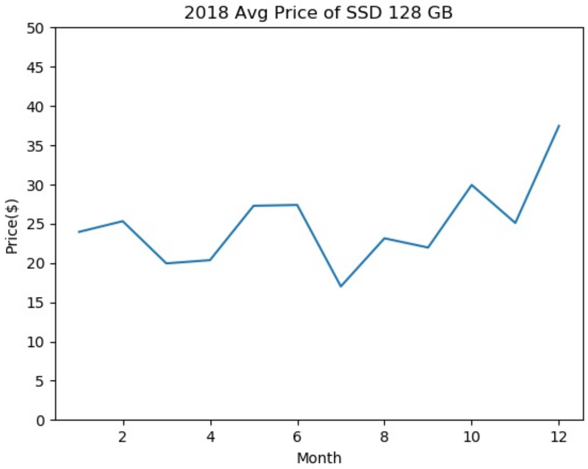


Fig 4. The diagram of the database

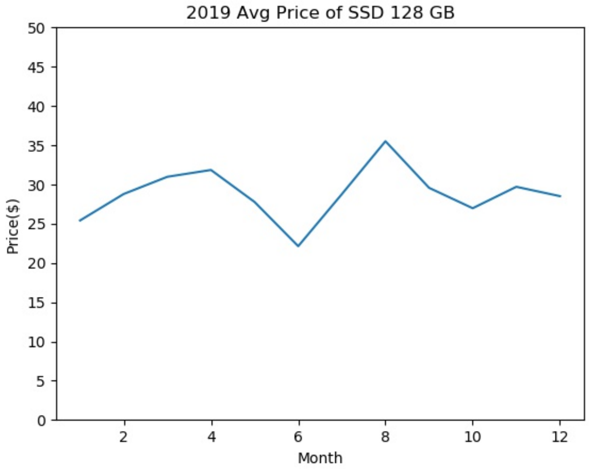


Fig 5. The diagram of the database

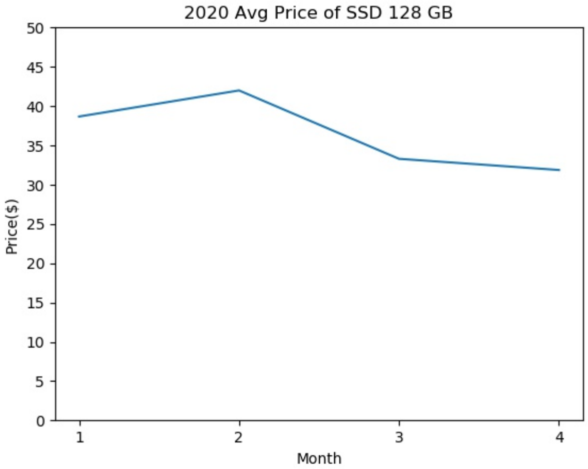


Fig 6. The diagram of the database

3.2. Amazon API

To get a certain product price on amazon, our team tried to request the Amazon API giving us the prices, however, there came out some issue, so we parsed the Amazon website first. After parsing the Amazon website, we chose Samsung SSD 250GB as a certain product and kept collecting its price. The price of Samsung SSD 250GB is at 60 to 70 USD. The diagram of the price of Samsung SSD 250GB is shown in Fig 7.

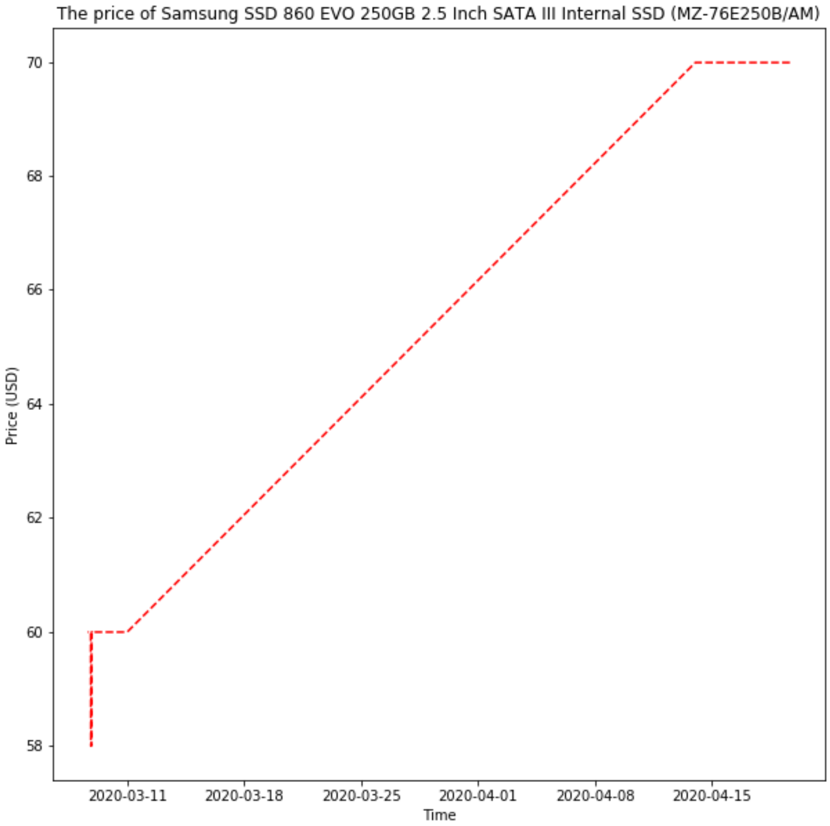


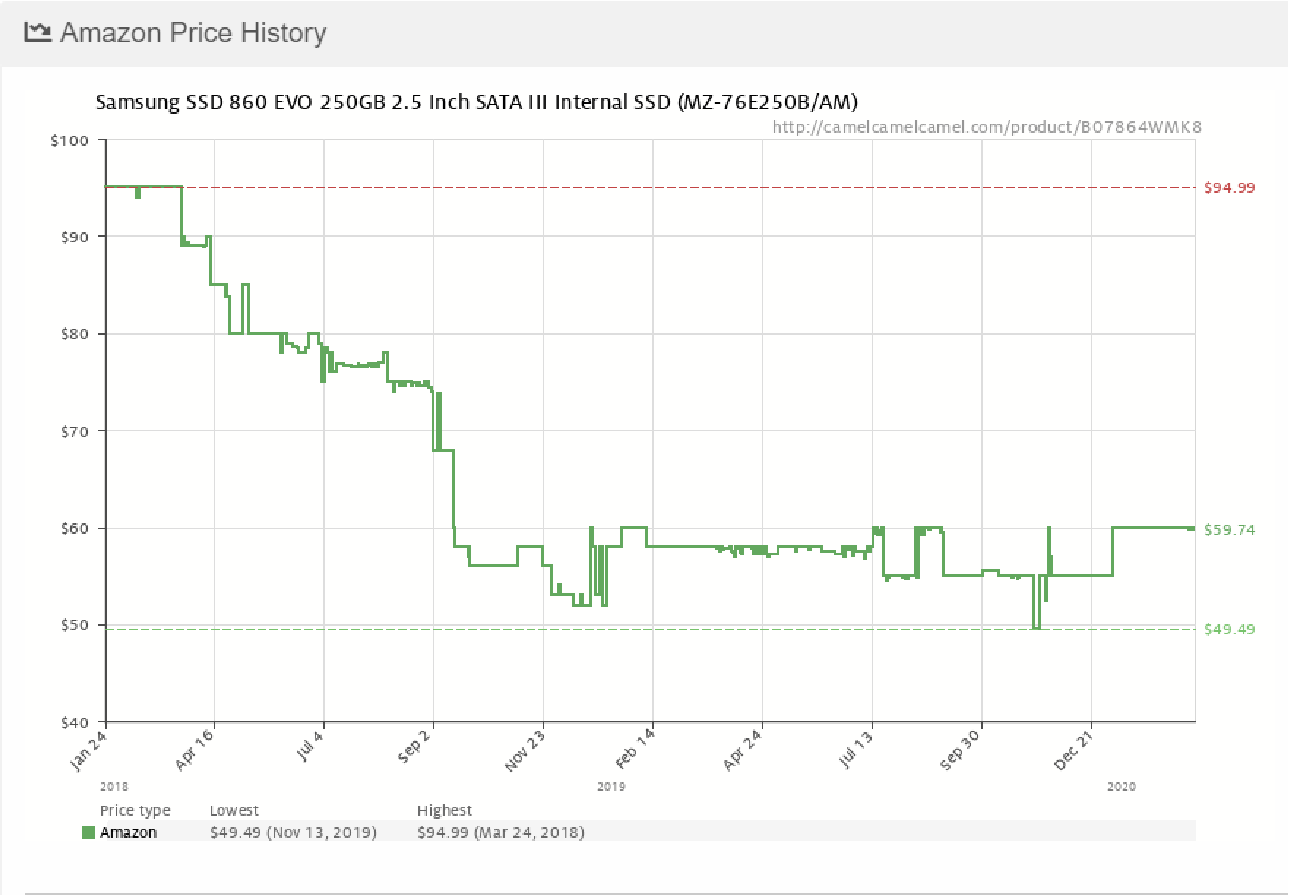
Fig 7. The diagram of the price of Samsung SSD 250GB

4. THE OTHER REFERENCE WEBSITES

4.1. Camelcamelcamel

Camelcamelcamel is a website that keeps tracking the prices of products in Amazon for almost all the products. This website can show history, average, highest, and lowest price of a certain product to users. The diagram of the price history of Samsung SSD 250GB in camelcamelcamel is shown in Fig 8.

We have the historical, average, highest, and lowest price of Samsung SSD 250GB. The average price is 57.77 USD which is almost the same as the price we get from the Amazon website right now. The diagram of each price of Samsung SSD 250GB in camelcamelcamel is shown in Fig 9.



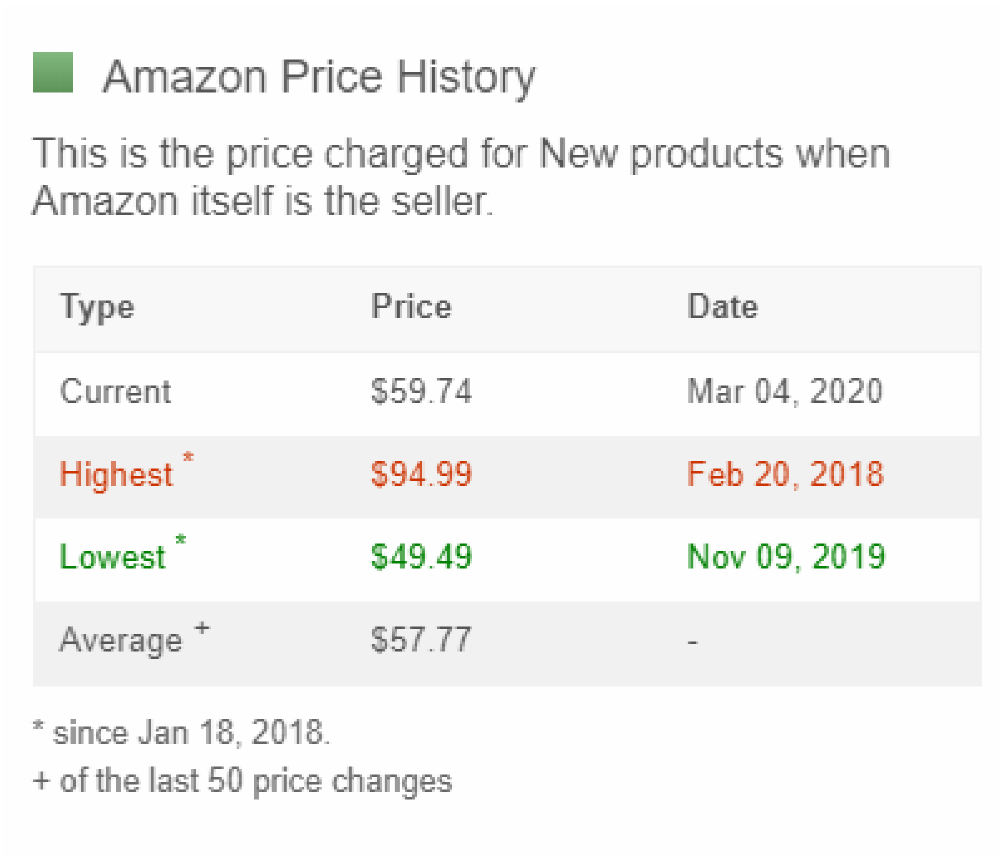
Amazon Price History

Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)

http://camelcamelcamel.com/product/B07864WMK8

Price type	Lowest	Highest
🟩 Amazon	$49.49 (Nov 13, 2019)	$94.99 (Mar 24, 2018)

Fig 8. The diagram of price history of Samsung SSD 250GB in camelcamelcamel



Amazon Price History

This is the price charged for New products when Amazon itself is the seller.

Type	Price	Date
Current	$59.74	Mar 04, 2020
Highest *	$94.99	Feb 20, 2018
Lowest *	$49.49	Nov 09, 2019
Average +	$57.77	-

* since Jan 18, 2018.
+ of the last 50 price changes

Fig 9. The diagram of each price of Samsung SSD 250GB in camelcamelcamel

4.2. Slickdeals

Slickdeals is a website that has lots of great deals from different sources including Amazon. Our team gathered all of the prices of SSD which are 256GB by analyzing the RSS feeds of every page in Slickdeals with the keyword “SSD 256gb” to get the historical price. Then, we plot out the time versus price diagram. The diagram of the historical price of SSD 256GB in Slickdeals is shown in Fig 10.

The mean value of all the data points is 715.68 USD and the standard deviation is 391.19 USD. According to the prices from Amazon and camelcamelcamel, we are sure that the prices of SSD which are 256GB are normally lower than 100 USD. We can discover lots of outliers because the keyword “SSD 256gb” also shows the price of laptops which has the same feature. To solve this problem, we have to drop the outliers. The mean value and standard deviation before dropping outliers are shown in Table 1.

We first drop 70 percent of data that have a higher price as outliers and then calculate the mean value and standard deviation of the data points. The mean value after dropping 70 percent of outliers is 318.7 USD which is still greater than the normal price. The standard deviation is 153.2 USD which represents the prices are a bit dispersed. The diagram of the historical price of SSD 256GB after dropping 70 percent of outliers is shown in Fig 11. The mean value and standard deviation after dropping 70 percent of outliers are shown in Table 2.

Making the mean value much closer to the normal price, we decide to drop 90 percent of data as outliers. This processing makes the mean value drop to 79.52 USD, and the standard deviation decrease to 53.58 USD which is more normal than before. The diagram of the historical price of SSD 256GB after dropping 90 percent of outliers is shown in Fig 12. The mean value and standard deviation after dropping 90 percent of outliers are shown in Table 3.

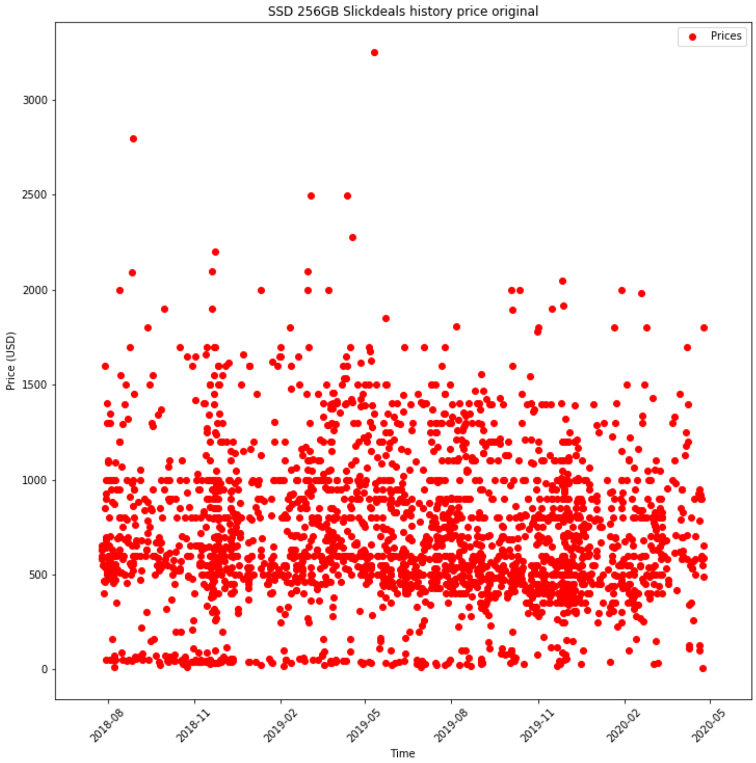


Fig 10. The diagram of the historical price
of SSD 256GB in Slickdeals

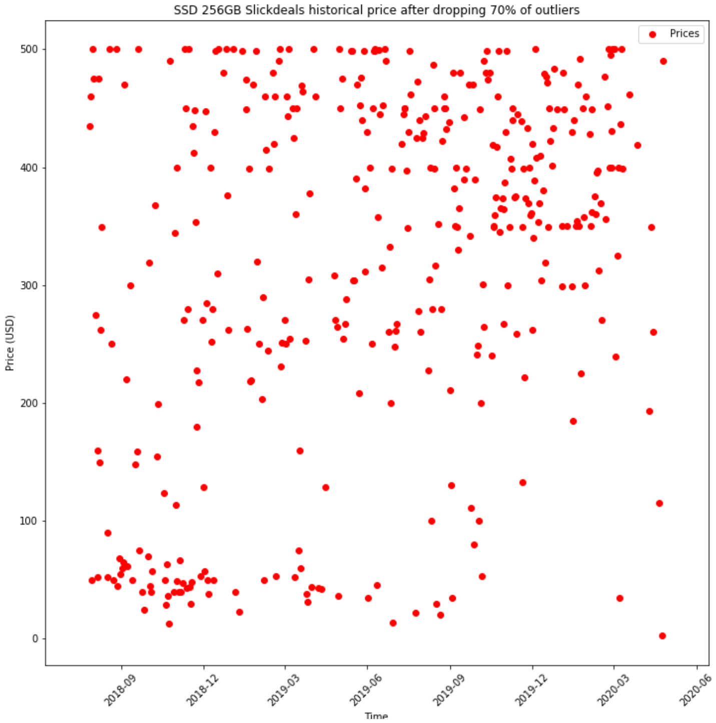


Fig 11. The diagram of the historical price of SSD 256GB after dropping 70 percent of outliers

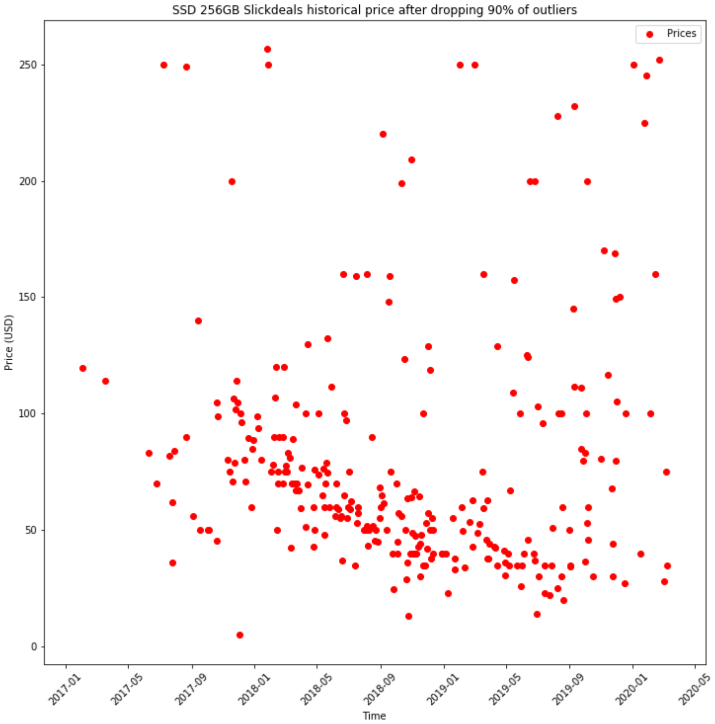


Fig 12. The diagram of the historical price of SSD 256GB after dropping 90 percent of outliers

Before dropping outliers	
Mean	715.68 USD
Standard deviation	391.19 USD

Table 1. The mean value and standard deviation
before dropping outliers

After dropping outliers	
Mean	318.7 USD
Standard deviation	153.2 USD

Table 2. The mean value and standard deviation
after dropping 70 percent of outliers

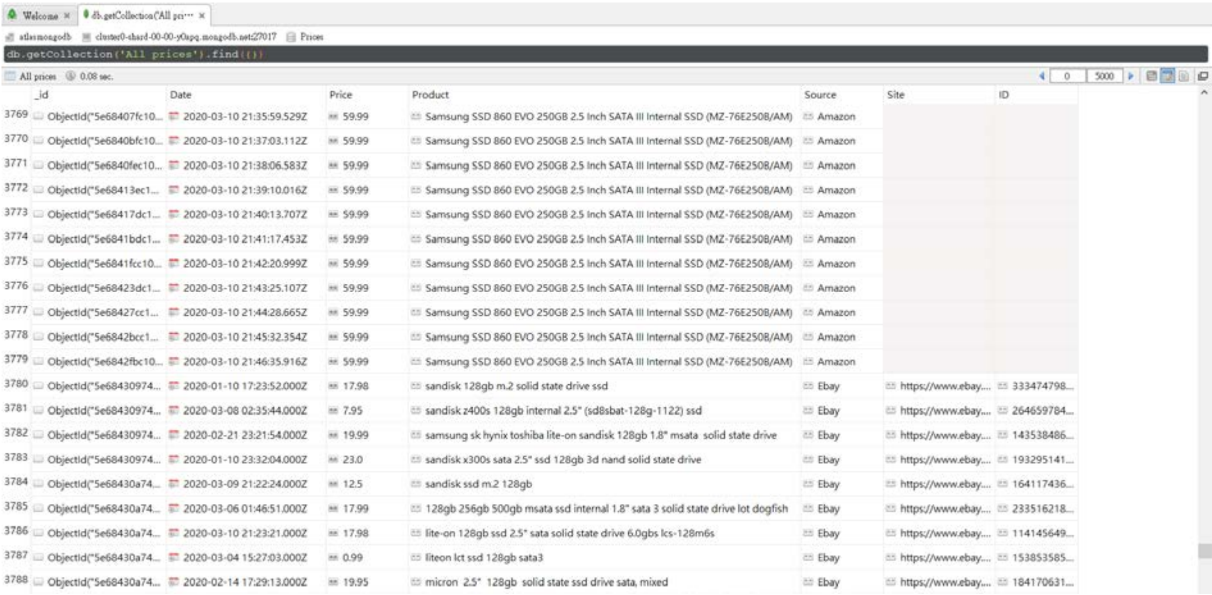
After dropping more outliers	
Mean	79.52 USD
Standard deviation	53.58 USD

Table 3. The mean value and standard deviation
after dropping 90 percent of outliers

5. DATABASE

5.1. Database

We pushed the data we collected from three online shopping websites (Amazon, eBay, and Slickdeal) altogether with the same format for further usage to a free online database called Atlas MongoDB. There are several columns in this dataset which stores the details of each item, which is “Date”, “Price”, “Product”, “Source”, “Site”, and “ID”. Among them, “Date” records the date of each item been post; “Source” indicates the website of each item been post. The column called “ID” is used to avoid collecting redundant data. The diagram of the database is shown in Fig 13.



Welcome
db.getCollection('All pri···
atlasmongodb
cluster0-shard-00-00-y0apq.mongodb.net:27017
Prices
All prices
0.08 sec.
0
5000

```
db.getCollection('All prices').find({})
```

	_id	Date	Price	Product	Source	Site	ID
3769	ObjectId("5e68407fc10...	2020-03-10 21:35:59.529Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3770	ObjectId("5e6840bfc10...	2020-03-10 21:37:03.112Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3771	ObjectId("5e6840fec10...	2020-03-10 21:38:06.583Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3772	ObjectId("5e68413ec1...	2020-03-10 21:39:10.016Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3773	ObjectId("5e68417dc1...	2020-03-10 21:40:13.707Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3774	ObjectId("5e6841bdc1...	2020-03-10 21:41:17.453Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3775	ObjectId("5e6841fcc10...	2020-03-10 21:42:20.999Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3776	ObjectId("5e68423dc1...	2020-03-10 21:43:25.107Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3777	ObjectId("5e68427cc1...	2020-03-10 21:44:28.665Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3778	ObjectId("5e6842bcc1...	2020-03-10 21:45:32.354Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3779	ObjectId("5e6842fbc10...	2020-03-10 21:46:35.916Z	59.99	Samsung SSD 860 EVO 250GB 2.5 Inch SATA III Internal SSD (MZ-76E250B/AM)	Amazon		
3780	ObjectId("5e68430974...	2020-01-10 17:23:52.000Z	17.98	sandisk 128gb m.2 solid state drive ssd	Ebay	https://www.ebay....	333474798...
3781	ObjectId("5e68430974...	2020-03-08 02:35:44.000Z	7.95	sandisk z400s 128gb internal 2.5" (sd8sbat-128g-1122) ssd	Ebay	https://www.ebay....	264659784...
3782	ObjectId("5e68430974...	2020-02-21 23:21:54.000Z	19.99	samsung sk hynix toshiba lite-on sandisk 128gb 1.8" msata solid state drive	Ebay	https://www.ebay....	143538486...
3783	ObjectId("5e68430974...	2020-01-10 23:32:04.000Z	23.0	sandisk x300s sata 2.5" ssd 128gb 3d nand solid state drive	Ebay	https://www.ebay....	193295141...
3784	ObjectId("5e68430a74...	2020-03-09 21:22:24.000Z	12.5	sandisk ssd m.2 128gb	Ebay	https://www.ebay....	164117436...
3785	ObjectId("5e68430a74...	2020-03-06 01:46:51.000Z	17.99	128gb 256gb 500gb msata ssd internal 1.8" sata 3 solid state drive lot dogfish	Ebay	https://www.ebay....	233516218...
3786	ObjectId("5e68430a74...	2020-03-10 21:23:21.000Z	17.98	lite-on 128gb ssd 2.5" sata solid state drive 6.0gbs lcs-128m6s	Ebay	https://www.ebay....	114145649...
3787	ObjectId("5e68430a74...	2020-03-04 15:27:03.000Z	0.99	liteon lct ssd 128gb sata3	Ebay	https://www.ebay....	153853585...
3788	ObjectId("5e68430a74...	2020-02-14 17:29:13.000Z	19.95	micron 2.5" 128gb solid state ssd drive sata, mixed	Ebay	https://www.ebay....	184170631...

Fig 13. The diagram of the database

6. RESULTS

6.1. Analyze by linear regression

Our team predicts the future price of SSD by fitting a linear regression model to the dataset. One is the historical price of SSD 256GB after dropping 70 percent of outliers and the other is the historical price of SSD 256GB after dropping 90 percent of outliers. We can see the result of the previous one shows that the price of SSD will rise in the future while the latter one shows that the price of SSD will drop in the future. The key which makes the result so different is the number of outliers. While there are too many outliers, the linear regression model will be affected and the price trend will be dominated by the outliers. The latter result makes sense because while the technology of SSD is improving, the cost of SSD will drop in the future which also will make the price of SSD decreased. We finally plot the linear regression result with eBay prices and Amazon prices. The result shows that most of the prices of Amazon and eBay are below the price history which means most of the deals on Amazon and eBay are great deals. The diagram of linear regression is shown in Fig 14 and Fig 15. The diagram of linear regression with other sources data points is shown in Fig 16.

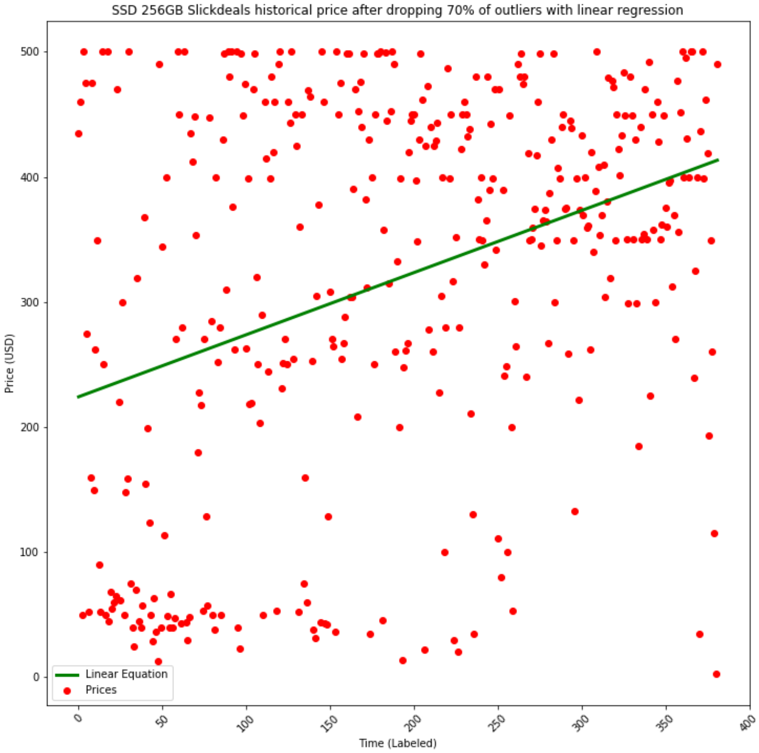


Fig 14. The diagram of linear regression

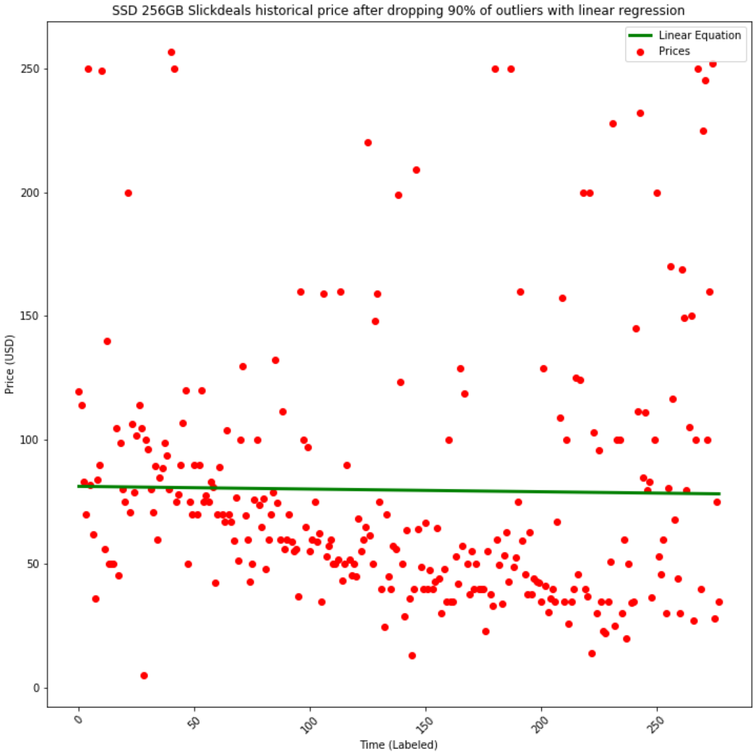


Fig 15. The diagram of linear regression

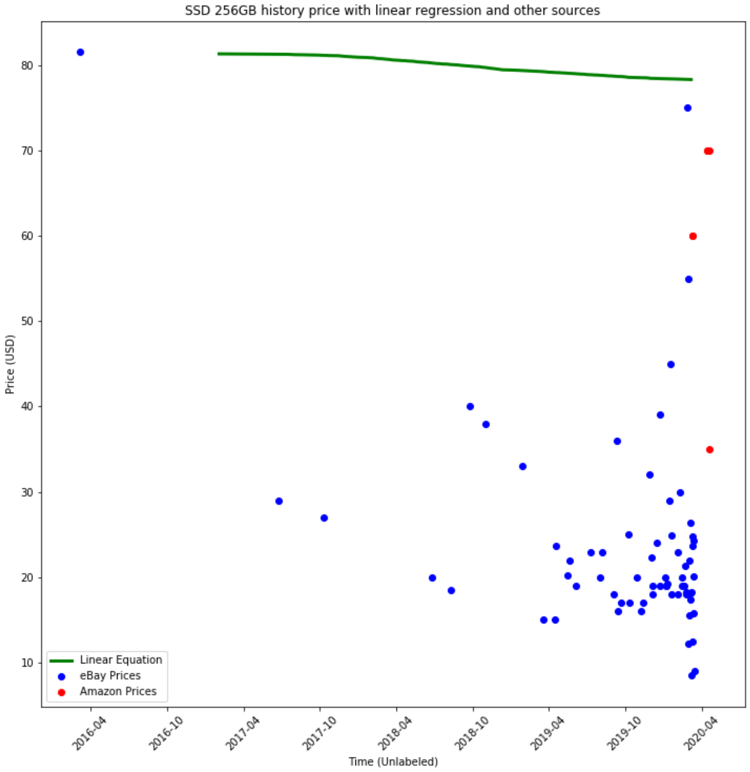


Fig 16. The diagram of linear regression with other sources data points

7. CONCLUSION

According to our results, we discover that the price of SSD will drop in the future. On the other hand, outliers will dominate the result of linear regression. Choosing SSD as our certain product might not be a good decision because while searching SSD, it will also show the result which relative to SSD such as laptops. And those relative result are not our target which will make them become outliers. In the future, we will change the certain product into other things which will not show too much relative results.

8. ACKNOWLEDGMENTS

We appreciate the National Taipei University of Technology and the University of Cincinnati for providing us an opportunity to go to the United States. Fewer people have a chance to go abroad and learn what they are interested in because it will cost lots of money. Fortunately, Taipei Tech gives all of us funding so that we do not need to pay for the ticket, tuition, and housing fee. The only thing we need to pay is living expenses, so we are very grateful for our family funding us to stay in America. We also thank our advisor's open-minded that he allowed us to go abroad while we still have one project is going to finish. We promise that we will do our best to finish the project. Finally, we are grateful that we have a chance of being taught by professor Mehdi. Mehdi is an excellent advisor that always pushing us to face the difficulties and try to explore a different kind of method to solve problems. It is our pleasure to be instructed by Mehdi. We hope we have other opportunities to cooperate in the future.

REFERENCES

[1] eBay, *Call Index*, eBay, 2013 – 2019. Accessed on: Jan. 29, 2020. [Online]. Available: https://developer.ebay.com/DevZone/finding/CallRef/index.html#CallIndex

[2] eBay, *Call Index*, eBay, 2013 – 2019. Accessed on: Jan. 29, 2020. [Online]. Available: https://developer.ebay.com/DevZone/finding/CallRef/types/SellingStatus.html

[3] python-amazon-product-api, *python-amazon-product-api*, python-amazon-product-api, Feb. 5, 2017. Accessed on: Jan. 29, 2020. [Online]. Available: https://python-amazon-product-api.readthedocs.io/en/latest/

[4] PYPI, *python-amazon-simple-product-api*, PYPI, Feb 25, 2017. Accessed on: Jan. 29, 2020. [Online]. Available: https://pypi.org/project/python-amazon-simple-product-api/

[5] Sorlas, *Amazon-Price-Check*, Github, Sep 4, 2017. Accessed on: Jan. 29, 2020. [Online]. Available: https://github.com/Sorlas/Amazon-Price-Check/blob/master/price_check.py

[6] Ethan Jarrell, *Building a REST API using Mongo DB*, Medium, July 13, 2017. Accessed on: Feb. 12, 2020. [Online]. Available: https://codeburst.io/building-a-rest-api-using-mongo-db-75cac3403fab

[7] Camelcamelcamel, *Popular Products*, Camelcamelcamel, 2008 – 2020. Accessed on: Mar. 4, 2020. [Online]. Available: https://camelcamelcamel.com/popular

[8] Camelcamelcamel, *Top Amazon Price Dropss*, Camelcamelcamel, 2008 – 2020. Accessed on: Mar. 4, 2020. [Online]. Available: https://camelcamelcamel.com/top_drops

[9] Denis Sureau, *RSS - Really Simple Syndication*, XUL. Accessed on: Feb. 19, 2020. [Online]. Available: https://www.xul.fr/en-xml-rss.html

[10]w3schools, *XML RSS*, w3schools. Accessed on: Feb. 19, 2020. [Online]. Available: https://www.w3schools.com/xml/xml_rss.asp