Report for Data Science Coding Challenge
at Radius Intelligence

1. Brief Introduction of Data

The data recorded 10 fields about contacting information, location, company operating status features of a list of businesses. They were business name, address, city, state, zip, time in business, phone number, headcount, revenue and category code. All of the fields were filled with string data and totally information of 1 million businesses were recorded on file.

Among all of the 10 fields, they contained several types of abnormal value including None, 0, ‘’, ‘ ’, ‘null’, ‘none’, ‘0’, which didn’t follow the format of every field in the dataset. So before using the dataset for further analysis, these abnormal values should be detected and processed at first.

2. Fill Rate Calculation

2.1 Definition of filled recordings

First of all, it was necessary to check the data quality, for example, how many percent of missing values exist. Here, the recordings having values were considered as filled, so the recordings having missing values should be considered as non-filled. Then, the definition of missing data in this specific case should be clarified ahead of calculation.

In the dataset, each column was filled with categorical values to record corresponding businesses information. For categorical values, whatever numbers or text data, they are in the format of string data with quotation marks to refer to. Therefore, when considering valid filled recordings, they should be aligned with string values format. However, it was existed that values of None, ‘’ and 0 were not in the string data format, which all should be classified as missing values. Hence, the definitions of missing values and filled recordings were clarified.

2.2 Fill rate calculation formula

Then the fill rate for each field was defined as:
(Total number of recordings excluding values of None, 0 and ‘’) / (Total length of the field)

2.3 Fill rate results and root causes analysis

The result of fill rate of each column is shown as follows in the table:

	features	total_row_num	filled_num	not_filled_num	fill_rate
0	address	1000000	999957	43	0.999957
1	category_code	1000000	999962	38	0.999962
2	city	1000000	999950	50	0.99995
3	headcount	1000000	962334	37666	0.962334
4	name	1000000	999964	36	0.999964
5	phone	1000000	590859	409141	0.590859
6	revenue	1000000	943066	56934	0.943066
7	state	1000000	999961	39	0.999961
8	time_in_business	1000000	916107	83893	0.916107
9	zip	1000000	999955	45	0.999955

Figure1: result of fill rate of each column

The column 'fill_rate' in the above table represents the fill rate of each column. The 'not_filled_num' column stands for the total number of None, '' or 0 values occurred in each field, which should be considered as non-filled recordings.

So it is observable that the overall fill rate is quite high, especially for fields like 'address', 'category_code', 'city', 'name', 'state', and 'zip'. So it can be referred that the location related information for businesses were greatly collected and recorded. It made sense since most of the business location information can be googled or searched online. Location could be a relatively open source data for business information.

And fields of 'headcount', 'revenue' and 'time_in_business' have fill rate on another level. Although their fill rate results are also high with values more than 90%. But for absolute values, they have orders of magnitude for tens of thousands, which is a relatively huge amount compared with dozens of missing values. For these fields, the information should change over time, so it was reasonable if companies didn't provide accurate recordings at the time of data collection or it was harder to get the accurate number.

At last, the field 'phone' has the lowest fill rate, which is only 0.5908 out of the total recordings. So almost half of the business phone numbers were missed. And it could be deduced that most of the businesses didn't have an official phone number for public contact or they didn't publish their official phone number for public awareness but only for inside using. Actually, based on practical experience, companies may provide official email contacts more than phone numbers, which could be a potential aspect to add into the database and to complete the business information.

For better understanding, a bar plot was generated for more clear result visualization:

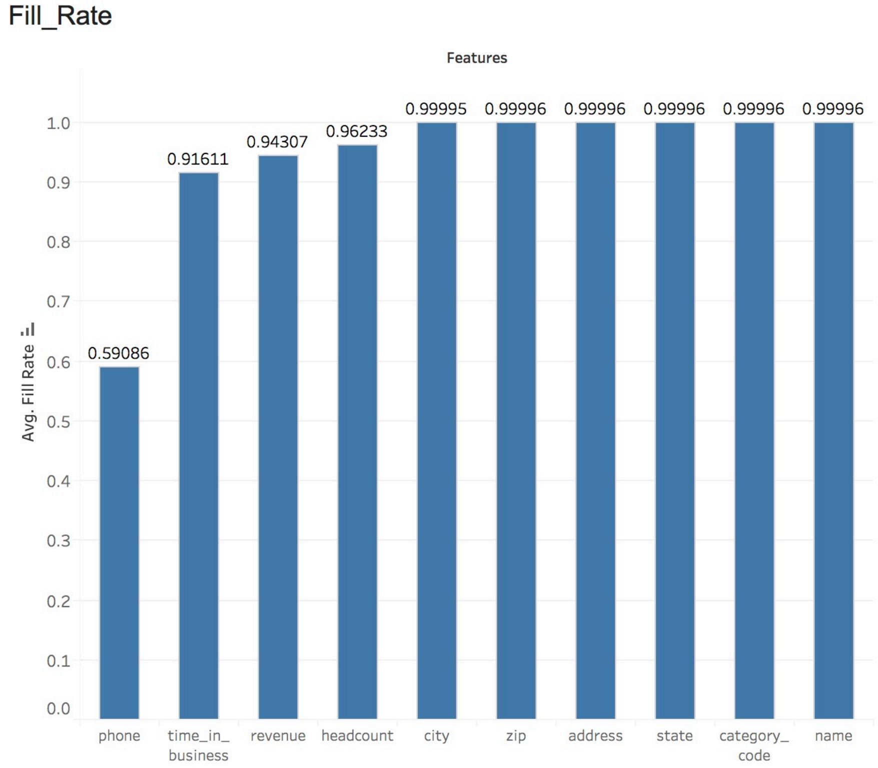


Figure2: bar plot of fill rate for each field

It can be noticed that the fill rate of field ‘phone’ is much smaller than other 9 fields.

3. True-valued Fill Rate Calculation

3.1 Definition of true-valued fill recordings

As for true-valued fill rate, its difference with the fill rate was that the cell was considered as filled with valid data format but meaningless values for the field. So for the dataset, not true-valued filled cells should have valid string values recorded but the value itself didn’t contain the same kind of information as other recordings in the same field.

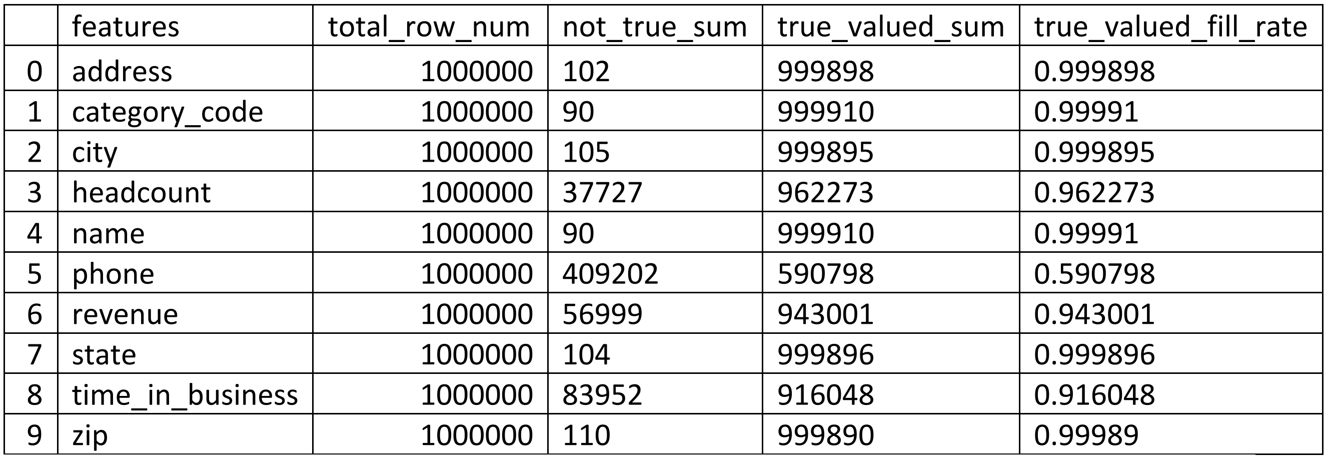
So it was determined to mark values of ‘null’, ‘none’, ‘0’ and ‘ ’ as not true-valued recordings. They should be excluded when calculating true-valued fill rate.

3.2 True-valued fill rate calculation formula

The true-valued fill rate was defined as:
(Number of filled recordings excluding values of ‘null’, ‘none’, ‘0’ and ‘ ’) / (Total length of the field)

3.3 True-valued fill rate results analysis

The result of fill rate of each column is shown as follows in the table:



	features	total_row_num	not_true_sum	true_valued_sum	true_valued_fill_rate
0	address	1000000	102	999898	0.999898
1	category_code	1000000	90	999910	0.99991
2	city	1000000	105	999895	0.999895
3	headcount	1000000	37727	962273	0.962273
4	name	1000000	90	999910	0.99991
5	phone	1000000	409202	590798	0.590798
6	revenue	1000000	56999	943001	0.943001
7	state	1000000	104	999896	0.999896
8	time_in_business	1000000	83952	916048	0.916048
9	zip	1000000	110	999890	0.99989

Figure3: result of true-valued fill rate of each column

After considering recordings of 'null', 'none', '0' and ' ' as irrelevant data in each field, the total number of true-valued filled recordings are shown in the 'true_valued_sum' column above. Compared with 'filled_num' in *figure1,* the total number of true-valued recordings decreased in all the fields. So, valid and true-valued data were summarized and the true-valued fill rate for each field was calculated with the results in the 'true_valued_fill_rate' column above.

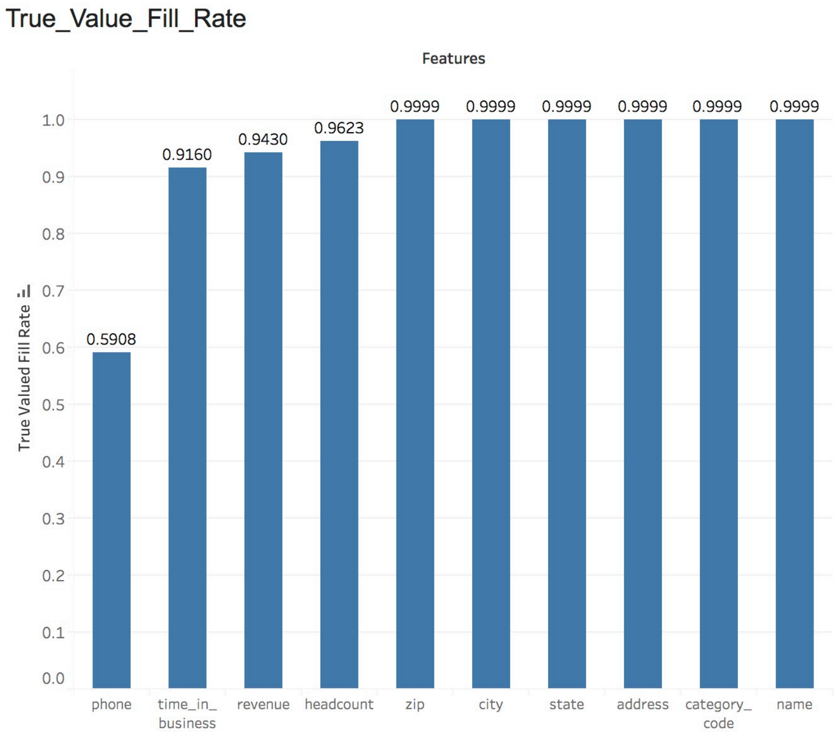


Figure4: bar plot of true-valued fill rate for each field

The bar plot displayed above is the visualization of true-valued fill rate. It can be observed that true-valued fill rate decreased a little bit compared with fill rate. But the overall magnitude didn't change. So it can be included that valid but not true-valued recordings were not too much existed in the data.

For more distinct comparison of results, the following table assembles the fill rate and true-valued fill rate of each field together.

	features	fill_rate	true_valued_fill_rate
0	address	0.999957	0.999898
1	category_code	0.999962	0.99991
2	city	0.99995	0.999895
3	headcount	0.962334	0.962273
4	name	0.999964	0.99991
5	phone	0.590859	0.590798
6	revenue	0.943066	0.943001
7	state	0.999961	0.999896
8	time_in_business	0.916107	0.916048
9	zip	0.999955	0.99989

Figure5: comparison of results of fill rate and true_valued fill rate for each field

The above table displays a clearer comparison between fill rate and true-valued fill rate of the data. So it can be noticed that every column had meaningless values recorded as categorical levels. They should be discovered and marked as missing values instead of categories.

4. Cardinality Calculation

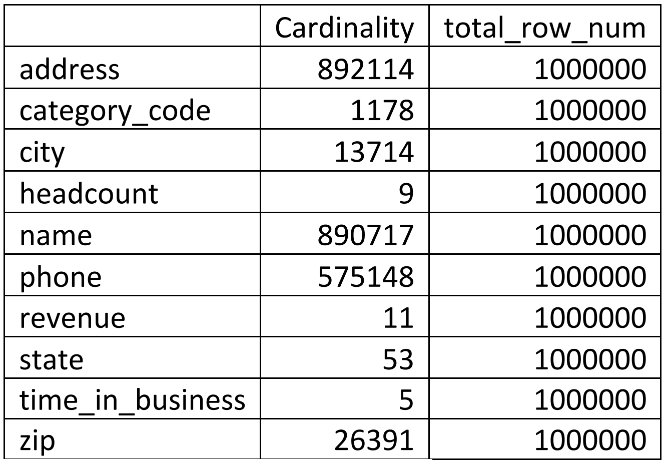
4.1 Cardinality understanding in data table

In a set, cardinality measures the number of elements of the set. And a set allows only unique elements included. So for data frames having rows and columns with duplicate values, cardinality of each column should measure the uniqueness of data values contained in this particular column.

In order to calculate the number of unique values in each column, the abnormal values were processed at the first step. All of the 7 types of abnormal values None, 0, ‘’, ‘ ’, ‘null’, ‘none’, and ‘0’ actually represented the same meaning, that they were missing data without true value. So, they were processed to be None value uniformly and were not considered as a data level in each field.

4.2 Cardinality calculation and results

For cardinality in each field, it was calculated as the total number of unique categories in each field. The result is shown as follows:



	Cardinality	total_row_num
address	892114	1000000
category_code	1178	1000000
city	13714	1000000
headcount	9	1000000
name	890717	1000000
phone	575148	1000000
revenue	11	1000000
state	53	1000000
time_in_business	5	1000000
zip	26391	1000000

Figure6: cardinality of each field

From the above table, it can be clearly noticed that cardinalities of all the fields varied a lot. So the type of information contained in each fields varied accordingly.

4.3 Cardinality results analysis

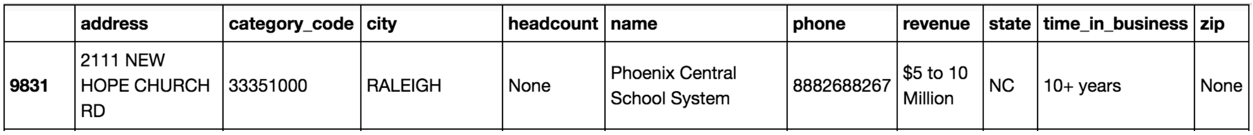
For fields like 'time_in_business', 'headcount', 'revenue' and 'state', they were low-cardinality fields with only several unique data levels. They can be flags to classify the whole data set into several major parts and to find similarities inside each part. And like 'category_code', 'city' and 'zip', they were normal-cardinality fields representing somewhat uncommon data values but they still had somewhat generality among all of the dataset. For fields of 'address', 'name' and 'phone', they represented high-cardinality fields, which contained high level of uniqueness of values in each field. So values in these fields were very uncommon. They contained unique and identification style information for each business. So somehow, they could be identifiers for businesses in the data set.

5. Interesting Discoveries

5.1 Analysis of relationship among location related fields

There were 4 fields out of the total 10 showing information related to a business's location. They were 'address', 'city', 'state' and 'zip'. Combining all of them together, there would be a complete location information of a business. And after processing all of the abnormal values into None uniformly, it was discovered that these 4 fields had similar amount of missing values of 102, 105, 104 and 110. So intuitively thinking about it, there was a possibility that all of the missing values of these 4 features came from the same businesses. So it meant that these businesses lost the entire location information recorded so that it brought missing values in these 4 fields at the same time.

But when tracing back to the data set, it was intended to get the index of missing values in each field. And the index results showed that the missing information of each field came from different businesses. So it meant that each business only lost part of the location information to be recorded, which was just not as expected.



	address	category_code	city	headcount	name	phone	revenue	state	time_in_business	zip
9831	2111 NEW HOPE CHURCH RD	33351000	RALEIGH	None	Phoenix Central School System	8882688267	$5 to 10 Million	NC	10+ years	None

Figure7: example of missing values in 'zip'

Then, after browsing the missing data, an idea came. Excepting for address, the missing values from one of the 3 fields ('city', 'state', 'zip') can be accurately imputed by the remaining two. For example, in the above business information, the zip code of the business was missed. But it is feasible to impute the zip code by 'city' and 'state' information. As searched in Google, the zip code should be '27604'.

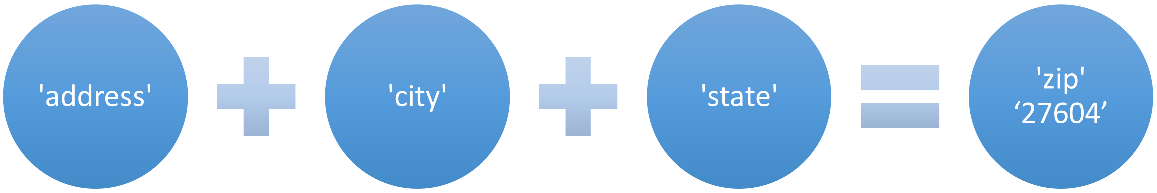


Figure8: example of using known location information to impute missing zip code

And None values in 'state' and 'city' can be applied the same way for imputation. For 'state', it can be determined by combining information of 'city' and 'zip'. For 'city', it can be deduced with information of 'state' and 'zip'. This will be feasible to go further for the company if it is possible to connect current data set with databases having complete street, city, state and zip information.

5.2 Analysis of business distribution based on location

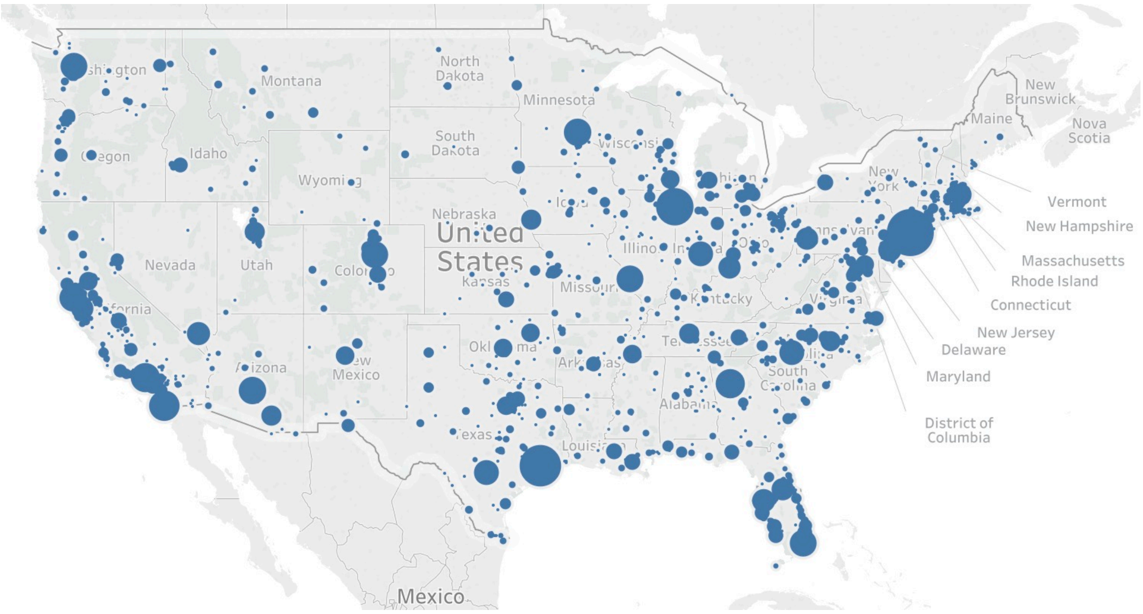


Figure9: frequency distribution of businesses on map

The map with blue circles marked above shows the location distribution of businesses in the States. The larger the circles are, the higher density of businesses at the location is. So it can be observed that big circles are mostly located along the coastline, especially east coast and west coast. It makes sense because there are many big cities with flourishing industries developing along the coastline. So business's locations is a really important features for business to consider. And for Radius, tracing or connecting companies by location and geo information should be helpful to quickly get the industry or company information.

5.3 Analysis of how businesses grow over time

In this part, business's headcount changing over time is analyzed.

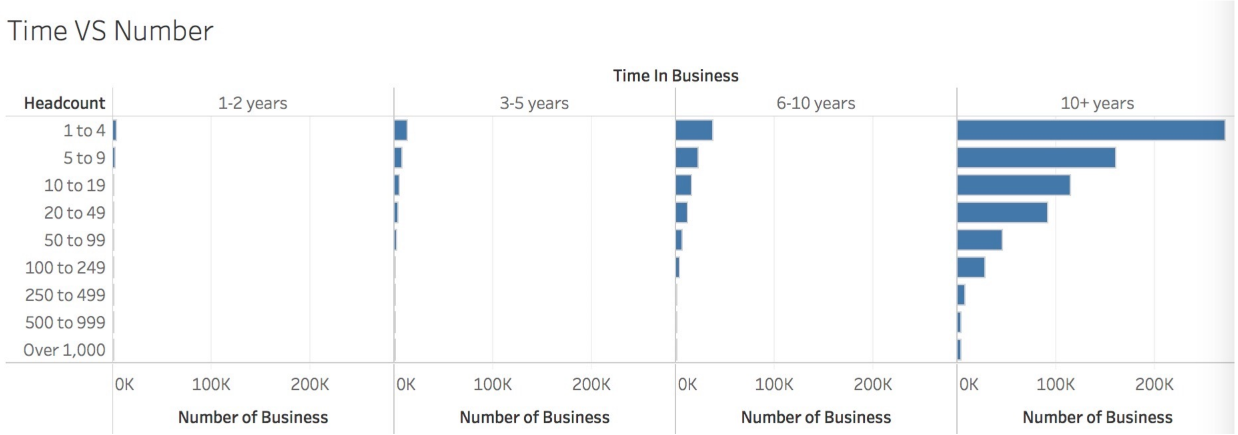
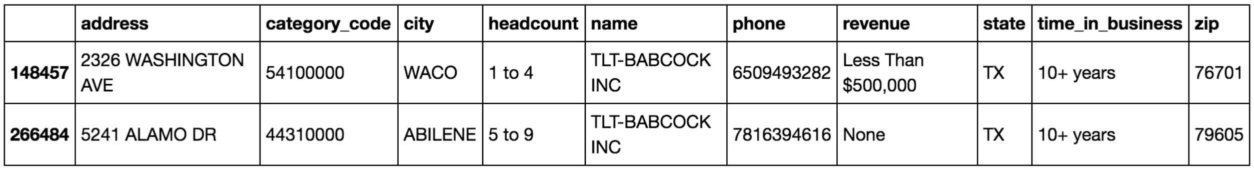


Figure10: how businesses' headcount changes over time

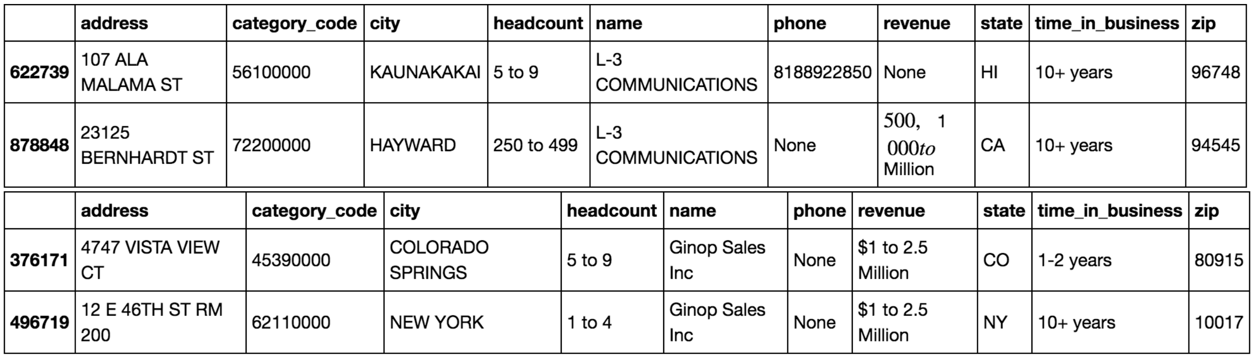
For the above plot, every block represents a category of time in business, the y-axis represents the categories of headcount, and length of the bars means the number of businesses for each headcount category under specific time blocks. So it can be observed that the shorter the time in business, the less the business amount is. And as time in business increases, the amount of every size of businesses increases, especially small size businesses with headcount less than 50. This disclosed the information that small businesses can also exist for a relative long period of time, which is not like the common sense that the longer the time in business, the greater the amount of large businesses.

5.4 Analysis of duplicated business name

For the field of 'name', it had totally 1 million records and 999910 as valid. But there were only 890717 names as unique for the business. Generally, name could be the identifier for each business since their names shouldn't be the same. But here in the dataset, there must be duplicated values of name recorded. So analysis was conducted in the field of business name. The following charts display information of business with duplicated name as examples.



	address	category_code	city	headcount	name	phone	revenue	state	time_in_business	zip
148457	2326 WASHINGTON AVE	54100000	WACO	1 to 4	TLT-BABCOCK INC	6509493282	Less Than $500,000	TX	10+ years	76701
266484	5241 ALAMO DR	44310000	ABILENE	5 to 9	TLT-BABCOCK INC	7816394616	None	TX	10+ years	79605



	address	category_code	city	headcount	name	phone	revenue	state	time_in_business	zip
622739	107 ALA MALAMA ST	56100000	KAUNAKAKAI	5 to 9	L-3 COMMUNICATIONS	8188922850	None	HI	10+ years	96748
878848	23125 BERNHARDT ST	72200000	HAYWARD	250 to 499	L-3 COMMUNICATIONS	None	500, 1 000*to* Million	CA	10+ years	94545

	address	category_code	city	headcount	name	phone	revenue	state	time_in_business	zip
376171	4747 VISTA VIEW CT	45390000	COLORADO SPRINGS	5 to 9	Ginop Sales Inc	None	$1 to 2.5 Million	CO	1-2 years	80915
496719	12 E 46TH ST RM 200	62110000	NEW YORK	1 to 4	Ginop Sales Inc	None	$1 to 2.5 Million	NY	10+ years	10017

Figure11: information about business with duplicate name

So it can be discovered that the condition of duplicated name was complicated. For example, 'TLT-BABCOCK INC'have two companies under the name, both are located at TX but only one of the company has revenues recorded. And for 'L-3 COMMUNICATIONS', there are two businesses located at CA and HI separately, which might be relations of branch and HQ. And for 'Ginop Sales Inc', the one located at CO only has 1-2 years in business, so the two might be two different companies.

The problem comes from this condition is that there is no primary key in the dataset as unique identifier for each row. So it increases the difficulty to accurately check information for a specific company. It would be better to fix this problem and refine the data accuracy.