Bazeley, Mikiko
Springboard 2018

Report: Data Wrangling

1. What kind of cleaning steps did you perform?

In order to collect the data I first needed to access my company’s internal Redshift data warehouse. Using the python libraries sqlalchemy and psycopg2 (a postgreSQL driver) I queried five main tables representing Salesforce objects (Leads, Opportunities, Demos, Accounts, and Products). Demos are our main objects of interest with Leads, Accounts, Opportunities and Products providing further clarity into customer demographics and demo outcomes. Providing my user credentials and database information, I opened a connection to pass queries (which is later shutdown after the queries are completed”.

An important concept to understand in analyzing sales data is the business process from Lead to Opportunity and how the different entities are represented. The stage of the sales cycle we are interested in examining will determine how the different tables are joined. The typical sales process is depicted below:

```
[Lead] → MQL→ Sales Accepted Lead → Sales Qualified Lead → [Opportunity] → [Customer]
```

Along the way from Prospect to Customer, at least 3+ entities can be created when the individual enters the system as a Prospect, engages with our sales team, is converted to an Opportunity which is connected to an Account and the individual (now represented as a Contact). Each object will have a number of standard and custom editable fields that can be easily created to enrich a company’s insight into the individual, deal, or company.

Given how frequently the metadata and schemas in Salesforce change in the start-up, I needed to query for all the relevant columns and fields and export csv samples for a visual inspection of the available data names, types and quality. Using the summaries, I manually constructed a data catalog showing the objects, related fields, the data warehouse names, the new names, necessary data transformations as well as possible data quality issues. After completing the first round of checks and evaluations and labeling fields which could be used for prediction or labeling, I created strings of candidate fields to subset the queried tables.

Original API Name	New Name	Type	select	new_names	Transformation	Transformation 2	Transformation 3	Use for model?	Use for EDA?	Data Issues	Type
email	email	Lead_PersonalInformation	'email',	'email___Lead_PersonalInformation',	Create new column - filled in non gmail/hotmail/etc email domain		String	N	Y	Can be fake "xyz","123" | Can be n	Lead: Personal Information
firstname	firstname	Lead_PersonalInformation	'firstname',	'firstname___Lead_PersonalInformation',	Create new column - Filled in First Name		String	N	Y	Can be fake "xyz","123" | Can be n	Lead: Personal Information
lastname	lastname	Lead_PersonalInformation	'lastname',	'lastname___Lead_PersonalInformation',	Create new column - Filled in last name		String	N	Y	Can be fake "xyz","123"	Lead: Personal Information
title	title	Lead_PersonalInformation	'title',	'title___Lead_PersonalInformation',	Create new column - filled in title	Create new column - title contains "manager/director/etc"	String	Y	Y	Not always filled	Lead: Personal Information
customer_type__c	customerType	Lead_LeadCompanyInformation	'customer_type__c',	'customerType___Lead_LeadCompanyInformation',			String	Y	Y	Can be null	Lead: Lead Company Information
company	company	Lead_LeadCompanyInformation	'company',	'company___Lead_LeadCompanyInformation',	Create new column - filled in company	Create new column - filled in non-dummy	String	N	Y	Not filled or fake	Lead: Lead Company Information
street	street	Lead_LeadCompanyInformation	'street',	'street___Lead_LeadCompanyInformation',			Address	N	N		Lead: Lead Company Information
city	city	Lead_LeadCompanyInformation	'city',	'city___Lead_LeadCompanyInformation',			Address	N	Y		Lead: Lead Company Information
state	state	Lead_LeadCompanyInformation	'state',	'state___Lead_LeadCompanyInformation',			Address	N	Y		Lead: Lead Company Information
country	country	Lead_LeadCompanyInformation	'country',	'country___Lead_LeadCompanyInformation',			Address	Y	Y		Lead: Lead Company Information
linkedin_page__c	linkedinPage	Lead_MarketingInformation	'linkedin_page__c',	'linkedinPage___Lead_MarketingInformation',			String	Y	Y		Lead: Marketing Information
traffic_channels__c	trafficChannel	Lead_MarketingInformation	'traffic_channels__c',	'trafficChannel___Lead_MarketingInformation',			String	Y	Y	Picklist value, can be null	Lead: Marketing Information
marketing_channel_camp	mktChannelcampaign	Lead_MarketingInformation	'marketing_channel_campa	'mktChannelcampaign___Lead_MarketingInformation',			String	Y	Y	Can be null	Lead: Marketing Information
landing_page__c	landingPage	Lead_MarketingInformation	'landing_page__c',	'landingPage___Lead_MarketingInformation',	Create new column - Has a linkedin page		String	Y	Y	Can be null & not always lined up	Lead: Marketing Information
landing_page_url__c	landingPageUrl	Lead_MarketingInformation	'landing_page_url__c',	'landingPageUrl___Lead_MarketingInformation',			String	Y	Y	Can be null & not always lined up	Lead: Marketing Information
google_campaign__c	googleCampaign	Lead_MarketingInformation	'google_campaign__c',	'googleCampaign___Lead_MarketingInformation',			String	Y	Y	Optional	Lead: Marketing Information
leadsource	leadsource	Lead_MarketingInformation	'leadsource',	'leadsource___Lead_MarketingInformation',			String	Y	Y	Can have nulls	Lead: Marketing Information
converteddate	converteddate	Lead_ConversionInformation	'converteddate',	'converteddate___Lead_ConversionInformation',	Standardize date		Date	N	Y	3/19/2015	Lead: Conversion Information
status_reason__c	statusReason	Lead_ConversionInformation	'status_reason__c',	'statusReason___Lead_ConversionInformation',	Filter out duplicates		Text	N	Y	Can have duplicates – but is there	Lead: Conversion Information
status	status	Lead_ConversionInformation	'status',	'status___Lead_ConversionInformation',			String	Y - Target	Y	Can have leads that are open	Lead: Conversion Information
id	PK_LeadID	Lead_ImportantJoinKey	'id',	'PK_LeadID___Lead_ImportantJoinKey',			String	N	Y		Lead: Important ID Info
convertedaccountid	FK_LeadtoAccount	Lead_ImportantJoinKey	'convertedaccountid',	'FK_LeadtoAccount___Lead_ImportantJoinKey',			String	N	Y		Lead: Important Join Key
convertedcontactid	FK_LeadtoContact	Lead_ImportantJoinKey	'convertedcontactid',	'FK_LeadtoContact___Lead_ImportantJoinKey',			String	N	Y		Lead: Important Join Key
convertedopportunityid	FK_LeadtoOpportunity	Lead_ImportantJoinKey	'convertedopportunityid',	'FK_LeadtoOpportunity___Lead_ImportantJoinKey',			String	N	Y		Lead: Important Join Key
ownerid	FK_LeadtoUser	Lead_ImportantJoinKey	'ownerid',	'FK_LeadtoUser___Lead_ImportantJoinKey',			String	N			Lead: Important Join Key
createdbyid	createdbyid	Lead_ImportantSystemInfo	'createdbyid',	'createdbyid___Lead_ImportantSystemInfo',			String	N			Lead: Important System Info
createddate	createddate	Lead_ImportantSystemInfo	'createddate',	'createddate___Lead_ImportantSystemInfo',	Need to Standardize to date		Date	Y			Lead: Important System Info
duplicate_lead__c	duplicateLead	Lead_ImportantSystemInfo	'duplicate_lead__c',	'duplicateLead___Lead_ImportantSystemInfo',	Need to filter out duplicate leads		Boolean	N		Useless field	Lead: Important System Info
isconverted	isconverted	Lead_ImportantSystemInfo	'isconverted',	'isconverted___Lead_ImportantSystemInfo',			Boolean	N - Target Check			Lead: Important System Info
isdeleted	isdeleted	Lead_ImportantSystemInfo	'isdeleted',	'isdeleted___Lead_ImportantSystemInfo',	Filter out		Boolean	N			Lead: Important System Info

Lead | Intro Call | Oppty | Oppty Line Items | Account | InitialSelect_Lists | Rename MasterDataSet

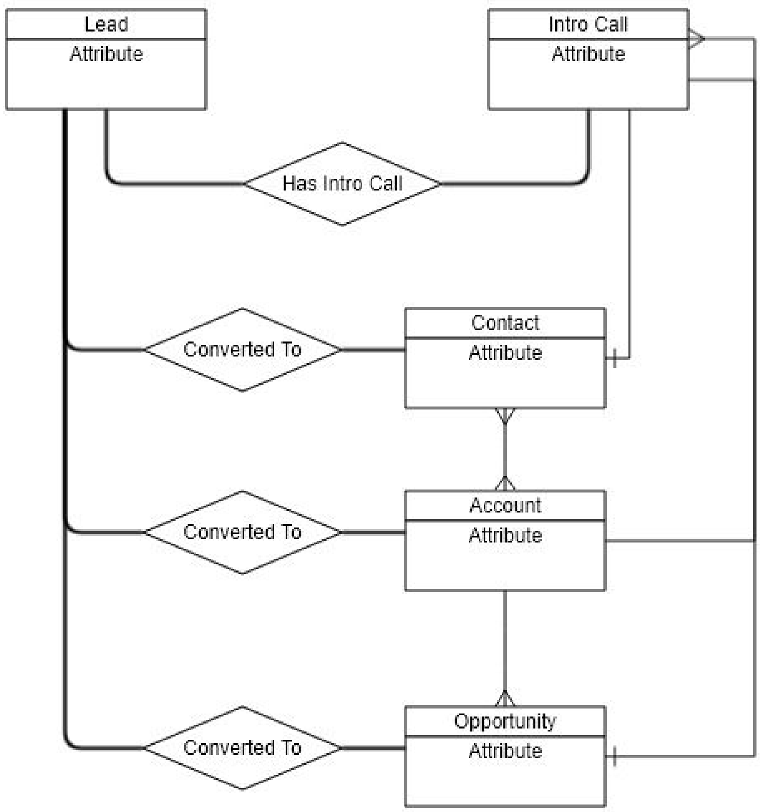
Once the data frames were subset, the columns were renamed, indicating the attribute, originating object, and type of attribute.

For example: “email___Lead_PersonalInformation” indicates the field describes Personal Information about a lead (in this case the email address) and came from the Lead dataframe (which was derived from the Lead table in the data warehouse). Another example: the field ‘PK_OpptyID___Oppty_ImportantJoinKey' indicates that the field is the Opportunity ID, is meant to be used in a join, and is the primary key that describes an opportunity instance.

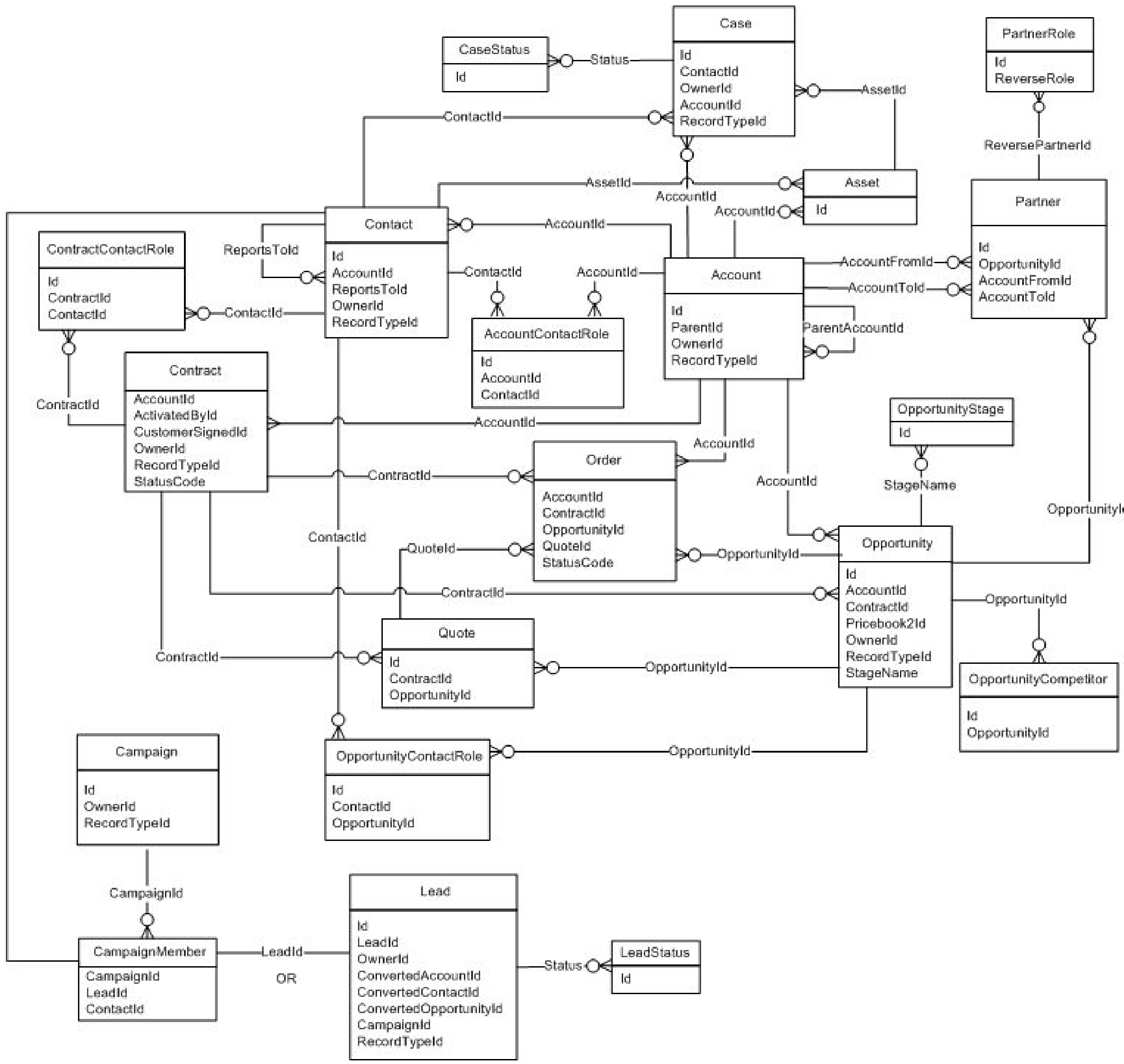
The naming convention and detailed data catalog is crucial for a few reasons: (1) similarly named fields could be duplicated across Salesforce objects without necessarily being related or exact - especially in the case of field mismatches; (2) real-time feedback was being given back to the data engineering team about data quality issues and solutions; (3) multiple data sets for exploration were being created and ideally we’d need clarity for the order of joins for 1:M and M:M relationships.

The subsetted data frames were then joined via the table/object keys identified during the data cataloging using merge. Two master data sets were created, masterDataSet where each demo call is a unique row (left joined by Leads, Opportunities, Accounts) and a masterDataSet_product where each product line item from the opportunity was left joined and enriched by the masterDataSet. The rationale for creating split data sets was to be able to accurately classify demo call outcomes but have the product data set available in order to facilitate exploratory analysis around product purchases, SLA's, and support add-ons.

In sales analytics and sales operations, time stamping is crucial to: (1) estimating velocity of opportunity pipeline, (2) triaging individual opportunities for attention, (3)



ensuring sales teams are meeting the agreed upon standard of performance.



https://developer.salesforce.com/docs/atlas.en-us.api.meta/api/sforce_api_erd_majors.htm

One complication was that all the time fields were different data types and had different patterns. For example:

‘2018-11-08T20:12:05.000Z’, ‘Q1-2015’, ‘10/17/2014 17:09’, ‘10/28/2014’ are just four examples of the 10+ data columns that needed to be parsed and converted to a datetime object. I wrote a function that would take a dataframe, the target column to be parsed, name of a new column, and the date time pattern to pass in. The function clean_dates takes the specified columns, parses the timedate string, returns a datetime object as the new column and deletes the old column.

The next import step was re-grouping the categorical features. After inspecting the different grouped columns and values, I remapped the values using dictionaries containing the new group values and deleted the old columns.

	A	B	E	F	J	K	O	P
1	Target___IntroCall_Outcome		rejectedReason___IntroCall_Outcome		status___Lead_ConversionInformation		trafficChannel___Lead_MarketingInformation	
2	**Original Value**	**New Label**	**Original Value**	**New Label**	***Original Value***	***New Value***	***Original Value***	***New Value***
3	Attributed	Qualified	Company Too Small	Wrong_Demographic	Cancelled	Not Qualified	Affiliate	Affiliate
4	Cancelled	Not Qualified	Does Not See Benefit of WalkMe	Not_Interested	Conference Rejuvenated	Open	Banner	Other
5	No Show	Not Qualified	Existing Opportunity	Duplicate	Contacted	Open	Bing	Bing
6	Qualified	Qualified	No Budget/Price Too High	Price_Too_High	Converted	Qualified	Biz Dev	Other
7	Rejected	Rejected	No Commercial influence	Not_Right_Person	Engaged	Open	BizoLi	Other
8	Rescheduling	Open	Not A Use Case Fit	Not_Interested	Finished Sequence	Open	Brand	Brand
9	Scheduled	Open	Not Decision Maker	Not_Right_Person	Junk	Not Qualified	Bulk upload - R&D's bug	Other
0			Other (please specify)	Other	Moved to SE	Open	Conference Emails	Email
1			Project Fully Outsourced	Other	No Show	Not Qualified	Conferences	Event/Conference
2			Startup - Too Expensive	Price_Too_High	No longer with company	Not Qualified	Conferences Lead Swaps	Event/Conference
3			Too Few Users	Wrong_Demographic	Not Relevant	Not Qualified	Customer Engagement Event	Event/Conference
4	"Attributed":"Qualified",		Wrong Source Version	Other	Nurture	Open	Email Nurturing	Email
5	"Cancelled":"Not Qualified",		Wrong Timing	Not_Interested	Nurture (Outbound)	Open	Email Nurturing Conferences	Email
6	"No Show":"Not Qualified",		__	Other	Open	Open	EmailMarketing	Email
7	"Qualified":"Qualified",				Prospecting	Open	EmailNurturing	Email

Fields with long text data were also dropped due to inexperience with NLP techniques.

Additional columns were created in order to measure the duration between various stages of leads, demos, and opportunities.

After creating all the necessary columns for exploratory analysis, additional id and demographic columns were deleted.

2. How did you deal with missing values, if any?

There are a couple categories of missing values:

	Value Missing? (Yes)(No)	
Value important? (Yes) (No)	Didn’t include Ex:	Included
	Treat as additional enrichment	

Essnetially, we would expect there to be missing values for some fields (given they wouldn’t be populated until the user had progressed past a stage). For other fields the columns had to be excluded because of poor practice and field enforcement within Salesforce.

For date columns I initially included a dummy date "1-1-1800'" which required a fair amount of processing to produce reasonable charts and statistics on the avg cycle of a lead to an opportunity. On further advice I left the date fields blank. Given the data came from Salesforce, a majority of the date fields are created automatically by the CRM.

For demographic data like employee size I could impune the values based on the categorization of the account as accounts are stratified by employee size (i.e. an Account labeled Mid-Market vs Enterprise is supposed to fall within a certain employee size range, etc so even if the employee size is missing we can impune the value by choosing either the median of the range or the actual employee counts within the data).

3. Were there outliers, and how did you handle them?

The types of outliers that occurred and why were:

- Deal Age - Some opportunities were incredibly old because an opportunity had been created, dropped for some time, and then picked up by a different sales rep. Those outliers were kept given how that occur frequently due to sales rep turnover and should be reflected in efficiency metrics.