ITEMMASTER - NATURAL LANGUAGE PROCESSING FOR MATCHING CUSTOMER COMPANY NAMES

By
Manqing Sun and William L. Guzmán

Supervisor: Nick Kadochnikov

A Capstone Project

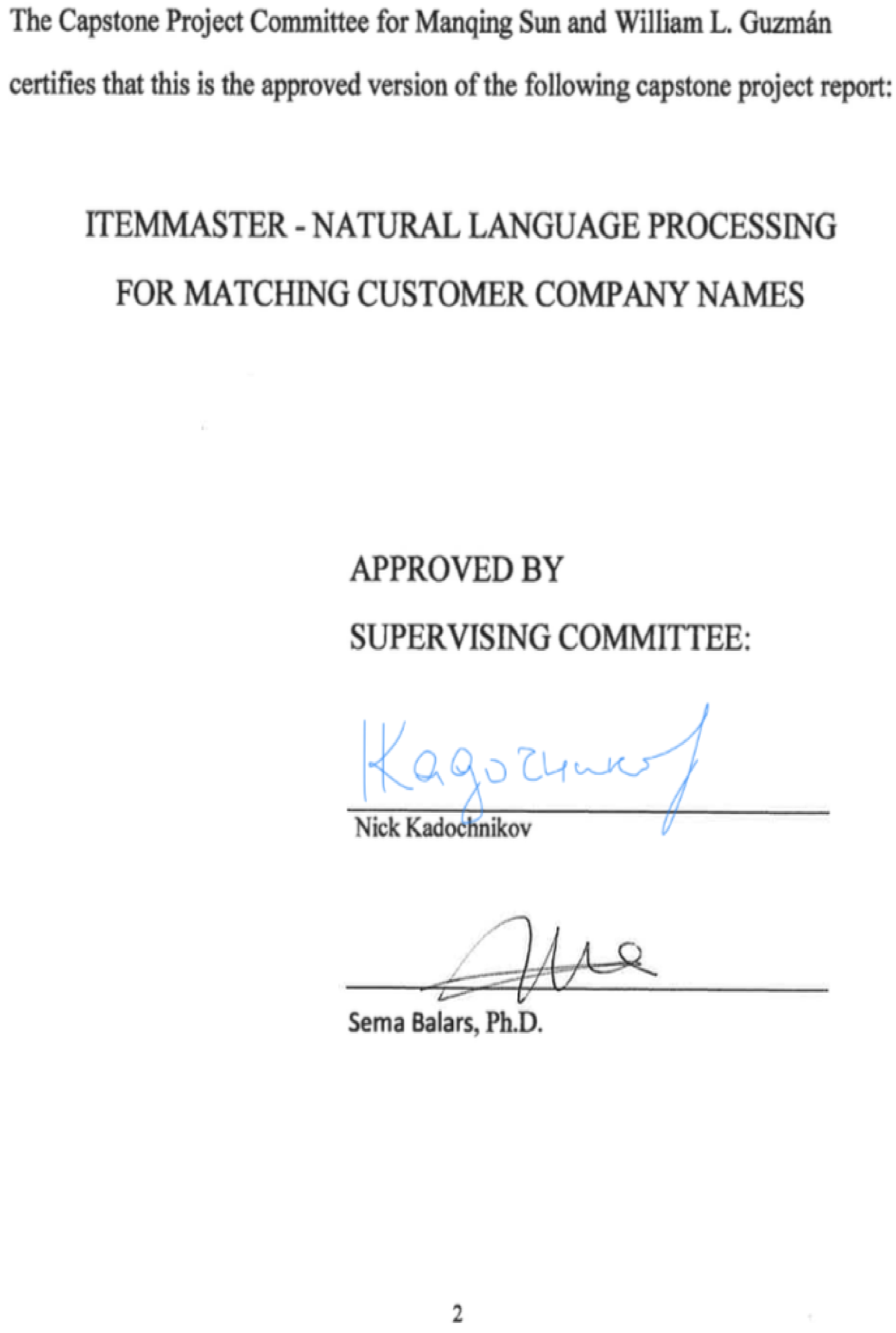
Submitted to the University of Chicago in partial fulfillment of

the requirements for the degree of

Master of Science in Analytics

Graham School of Continuing Liberal and Professional Studies

(May, 2018)



The Capstone Project Committee for Manqing Sun and William L. Guzmán certifies that this is the approved version of the following capstone project report:

ITEMMASTER - NATURAL LANGUAGE PROCESSING

FOR MATCHING CUSTOMER COMPANY NAMES

APPROVED BY

SUPERVISING COMMITTEE:

Кадочников

Nick Kadochnikov

Sema Balars, Ph.D.

ABSTRACT

Itemmaster® is a **spinoff** of **Peapod**.com established in 2009. It provides product content solutions for brands, retailers, and consumers alike. Our algorithm uses natural language processing (nlp) techniques to assist in finding the matching account in Itemmaster's salesforce database for every product in a retailer's assortment sheet. Our algorithm is not only able to automate the manual work, but also able to find additional accounts and unique items that manual work cannot. An assessment performed by Itemmaster concluded that, based on the four sets of data they provided for our development purposes, our algorithm could add more than $2,000,000 dollars to Itemmaster's sales pipeline.

KEY WORDS

Natural Language Processing (NLP), ItemMaster, Salesforce, retailer's assortment sheet, matching account.

EXECUTIVE SUMMARY

OBJECTIVE: The objective of this project is to develop a Natural Language Processing (NLP) algorithm that can map the unstandardized products from retailer's assortment sheet to ItemMaster's standardized Salesforce account. The deliverable of this project is to predict the matching account name, account ID, and parent account ID and give a similarity index value for this prediction in every unique combination of manufacturer and brand names.

METHODS: The Jaccard index, also known as "Intersection over Union" and the Jaccard similarity coefficient, is chosen for measuring the similarity of a retailer's product name with the ItemMaster's product account name. Before calculating the Jaccard similarity, the product names are pre-processed by using the NLP Toolkit (NLTK), Pandas frame, NumPy array, autocorrect (for spelling correction), re (regular expression), and ngrams, etc.

Our algorithm finds all the matching accounts with a Jaccard index larger than zero for brand name first, and then it takes the matching account with the largest similarity value to compare with the brand similarity threshold value set by Business Analyst. If this largest similarity is larger than the threshold the algorithm will proceed to find the matching account's parent account's name and compare it with unstandardized product's manufacturer name; based on the matching scenario of the manufacturer name, different matching account name, account ID, parent account ID and Jaccard similarity value will be returned (more detailed information can be found in later section of this report).

RESULTS: Our algorithm is not only able to automate the manual work, but also able to find additional accounts and unique items that manual work cannot. An assessment performed by ItemMaster concluded that our algorithm is able to identify 3,057 additional accounts which translates to saving ItemMaster 305 hours of manual work, plus 41,734 additional unique items, which will add $2,086,600 to ItemMaster's sales pipeline. This assessment is based on four sets of data they provided for our development purposes.

CONCLUSIONS: This NLP algorithm we have developed for ItemMaster Inc. has reached or even exceeded our project client company's expectation.

ACKNOWLEDGEMENT

Our sincere thanks to Dr. Sema Barlas for her support, her advice along the way, and for making the NLP class available when we really needed it.

Thanks to Nick Kadochnikov for teaching us these NLP techniques and all the advice he's given.

Thanks to ItemMaster's analytics team for always being available whenever we had questions .

TABLE OF CONTENTS

1. INTRODUCTION

- Company Background

ItemMaster®, a **spinoff** of **Peapod**.com established in 2009, is headquartered in downtown Chicago. It provides product content solutions for brands, retailers, and consumers alike. ItemMaster enables partners to plan and merchandise, market across channels, and build online eCommerce and media experiences. ItemMaster's flexible product content management system is available to Consumer-Packaged Goods (**CPG**) brands, CPG retailers, and ecosystem partners or mobile applications that can benefit from the 100,000+ manufacturer products active in the rapidly growing ItemMaster platform. ItemMaster structures data for today's consumer trends and tomorrow's technologies.

- PROBLEM STATEMENT

In a data-powered market, product digitization is key for delivering any product to different clients across the globe. No matter whether the company is a traditional retailer or a modern e-commerce store, data is crucial for creating a profitable business model. ItemMaster is the leader in creating and delivering comprehensive, certified content for major brands both online and offline. Its cloud-based Brand ActivationTM Platform enables manufactures to share, verify and manage their product portfolio and custom branded content for distribution across all channels. However, Grocery retailers sell products from thousands of manufacturers, and this data is rarely standardized. There are often variations of spellings, punctuation, Unicode characters, etc. in the manufacturer and brand names across retailers and even within one retailer's data file. Mapping a product to ItemMaster's Salesforce database has to be done manually.

- RESEARCH PURPOSE

The research purpose of this project is to develop a Natural Language Processing (NLP) algorithm that can map the unstandardized products from a retailer's assortment sheet to ItemMaster's standardized Salesforce database account.

2. DATA

For this project there are two types of data files. One is ItemMaster's Salesforce database Account Names file, which contains three variables: SFAccountName, SFAccountID, and SFParentAccountID. The other is Retailer's Assortment Sheet that has ten variables, among which we only need two: Manufacturer and Brand for finding the matching salesforce account.

- ITEMMASTER'S DATA

The standardized ItemMaster's Salesforce database Account Names format:

Variable	**Definition**	**Other attributes (Format)**
SFAccountName	Item Salesforce Account Name	Unique Key
SFAccountID	Item Salesforce Account ID	
SFParentAccountID	Item Salesforce Parent Account ID	

Example records:

SFAccountName	**SFAccountID**	**SFParentAccountID**
Gildan	001G000001H841ZIAR	001G000001XciJiIAJ
Doskocil Manufacturing Company	001G000001H841aIAB	
…	…	…

- THE RETAILER'S ASSORTMENT SHEET

The unstandardized Retailer's Assortment Sheet format:

Variable	Definition	Other attributes (Format)
Manufacturer	Manufacture's name	
Brand	Brand	
Category A	Category A that the item belongs to	
Category B	Category B that the item belongs to	
Category C	Category C that the item belongs to	
ItemDescription	Description of the item	
ItemUPC	Item Universal Product Code (UPC), or barcode.	
ShipItemID	Item Shipment ID	
VerificationStatus	Item Verification Status	
PublishedStatus	Item Published Status	

Example records:

Manufacturer	Brand	Description	ItemUPC	Category A	Category B	Category C	Retailer	Private Label
1 800 FLOWERS.COM INC	FANNIE MAY	FMAY MINT 7Z	5.2746E+10	POS DEPT GROCERY	CANDY	NON SEASONAL CANDY	Peapod	N
1 800 FLOWERS.COM INC	FANNIE MAY	FMAY PIXIES 7Z	5.2746E+10	POS DEPT GROCERY	CANDY	NON SEASONAL CANDY	Peapod	N
21ST AMENDMENT	21ST AMENDMENT BREWERY	21ST AMEND SSNL 6 12Z	8.5961E+11	POS DEPT GROCERY	DSD BEER N WINE	BEER	Peapod	N
34 DEGREES - FOODS WITH LATITUDE	34 DEGREES	GFI 34 DGRS CRSPBRD4.5Z	8.9477E+11	POS DEPT DELI	DELI CHS SHOP	DELI DRY GOODS	Peapod	N
34 DEGREES - FOODS WITH LATITUDE	34 DEGREES	GFI 34 DGRS RSMRY 4.5Z	8.9477E+11	POS DEPT DELI	DELI CHS SHOP	DELI DRY GOODS	Peapod	N

3. EXPLORATORY DATA ANALYSIS

NATURAL LANGUAGE PROCESSING FOR MATCHING CUSTOMER COMPANY NAMES

Based on ItemMaster's data analytics team's suggestion, we can assume the ItemMaster's Saleforce account names are created either based on the product's manufacturer name or its brand name.

Since the combinations of Manufacturer name and Brand name in the retailer's Assortment Sheet are not unique, we need to create an additional file that contains the unique combination of the manufacturer and brand names only. We then use this file to look for the matching standardized ItemMaster's Salesforce Account Name, Account ID, and Parent Account ID.

The Retailer's Unique Account Names file format:

Variable	**Definition**	**Other attributes (Format)**
Manufacturer	Manufacture's name	Unique Key (combined w/Brand)
Brand	Brand	Unique Key (combined w/Manufacturer)

Example records:

Manufacturer	**Brand**
IMPORT - PURCELL INTERNATIONA	
LOREAL USA	SOFT SHEEN CARSON LETS JAM
SOLO CUP CO	SOLO SQUARED
Jeg & Sons Inc	Apple
	Apple
Fisk Industries	.

Observations from example records above:

1) Manufacturer is not null, but Brand is null or Brand will become null after filtering, i.e. the last record.
2) Manufacturer is null, but Brand is not null

3) Both Manufacturer and Brand are not null.

4) Mis-spelling of Manufacturer name (i.e. INTERNATIONA, LOREAL USA).

5) Name combines number and word together (i.e. 1HARBOR, 1NORTH)

6) Mixture of uppercase and lowercase names, etc.

Example records of products per account from retailer:

This following table shows that one unique Manufacturer and Brand combination has 30 products in retailer's assortment sheet.

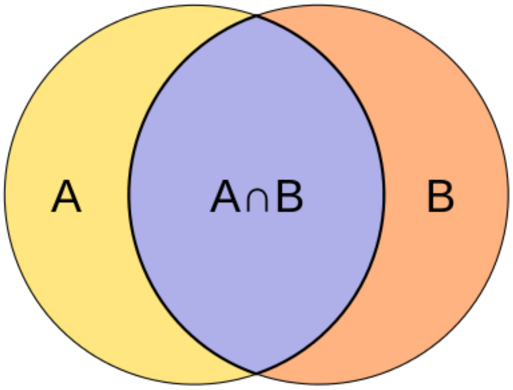
Manufacturer	Brand	Description	ItemUPC	Category A	Category B	Category C
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS STEMS/PIECE	21140216830	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD RED GRPFRUIT SECTIONS	21140218346	GROCERY	FRUIT CANNED	SPECIALTY FRUIT
IMPORT - PURCELL INTERNATIONA		WD ARTICHOKE HEARTS	21140218650	GROCERY	VEGETABLES - CANNED	PIMIENTOS/ONIONS/ARTICHOKES
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS BUTTONS	21140216854	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH SLICED MUSHROOMS	6.0788E+11	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH MUSHROOM STEMS & PCS	6.0788E+11	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH MUSHROOMS PCS&STEMS	6.0788E+11	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS SLICED/GLASS	21140216861	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS WHL /GLASS	21140216878	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD P/A TIDBITS IN JUICE	21140218421	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS SLICED	21140216847	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD SLICED PINEAPP N JUICE	21140218414	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED

IMPORT - PURCELL INTERNATIONA		WD SLICED PINEAPP IN SYRP	21140218407	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD CRUSHED P/A IN JUICE	21140218438	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS SLICED	21140216892	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD PINEAPPCHNK /HVY SYRP	21140218391	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD CRUSHED P/A IN SYRUP	21140218445	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD CRUSHED P/A IN JUICE	21140218377	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD CHUNK P/A IN JUICE	21140218452	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		SH CHUNK PINEAPPLE IN JUICE	6.0788E+11	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD SLICED P/A IN JUICE	21140218360	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		WD MUSHROOMS STEMS/PIECES	21140216885	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH CRUSHED PINEAPL IN JUICE	6.0788E+11	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		SH SLICED PINEAPPLE IN JUICE	6.0788E+11	GROCERY	FRUIT CANNED	PINEAPPLE- CANNED
IMPORT - PURCELL INTERNATIONA		SEG MUSHROOMS SLICED	38259107607	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH ARTICHOKE QUARTERS CAN	6.0788E+11	GROCERY	VEGETABLES - CANNED	PIMIENTOS/ONIONS/ARTICHOKES
IMPORT - PURCELL INTERNATIONA		SEG MUSHROOMS STEMS/PIECE	38259107591	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		WD ARTICHOKE QUARTERS	21140016492	GROCERY	VEGETABLES - CANNED	PIMIENTOS/ONIONS/ARTICHOKES
IMPORT - PURCELL INTERNATIONA		WD MUSHROOM WH BUTTON	21140216908	GROCERY	VEGETABLES - CANNED	MUSHROOMS CAN & GLASS
IMPORT - PURCELL INTERNATIONA		SH ARTICHOKE HEARTS	6.0788E+11	GROCERY	VEGETABLES - CANNED	PIMIENTOS/ONIONS/ARTICHOKES

4. METHODOLOGY

- JACCARD SIMILARITY VERSUS MINIMUM EDIT DISTANCE

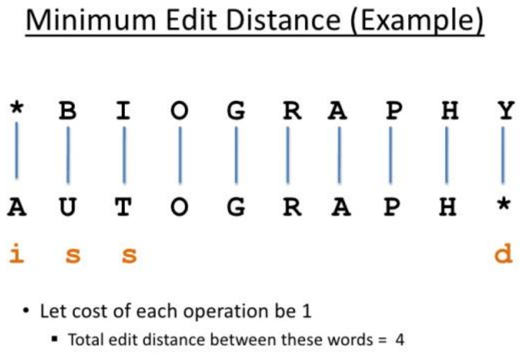
Jaccard similarity is chosen in this project for comparing the similarity of unstandardized names from retailer's assortment sheet to standardized names from ItemMaster's Salesforce database. The Jaccard coefficient measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:



$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

The best matching account is obtained by using the Python **'Map a Lambda function to a list'**.

The minimum edit distance between two strings can be an alternative method. We chose Jaccard because its index value is easier to interpret and it gives satisfying results.



Minimum Edit Distance (Example)

- Let cost of each operation be 1
  - Total edit distance between these words = 4

- The Natural Language Toolkit (NLTK)

NLTK is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language.

- Regular expression (RE)

A ***regular expression*** is a special sequence of characters that helps you match or find other strings or sets of strings, using a specialized syntax held in a pattern.

- Spelling autocorrect

We found **autocorrect spell** from **GitHub**. **GitHub** is a site where people build software. More than 27 million people use **GitHub** to discover, fork, and contribute to over 80 million projects.

Python 3 Spelling Corrector, https://github.com/phatpiglet/autocorrect.

To install, “pip install autocorrect”

- Steps taken before computing Jaccard similarity

In order to find the match, the following steps were taken before computing the Jaccard similarity coefficient:

1. To lower case both unstandardized and standardized names.
2. Using nltk library function nltk.word_tokenize() to tokenize both lower cased unstandardized and standardized names.
3. Filtering out non-meaningful tokens (words) from both sets of words
4. Using re.match() (regular expression match) to identify tokens consisting of both number and character (i.e. 4C), then splitting these types of tokens into number string and character string (i.e. "4", "C").
5. Using nltk library function nltk.WordNetLemmatizer() to lemmatize each token in both sets of words.
6. Using autocorrect function spell() to autocorrect spelling errors of every token
7. Using nltk library function nltk.ngrams() to get the unigram of both sets of words.
8. Computing the intersection and union of the two sets of unigrams
9. Computing the Jaccard similarity = intersection / union

Example records after data preprocessing:

Manufacturer	Brand	SFAccountName	SFAccountID	SFParent AccountID	AcctName_Similarity
IMPORT - PURCELL INTERNATIONA		Purcell International	001G000001H84CLIAZ		1
4 C FOODS CORP BR		4C Foods Corp.	001G000001H83vLIAR		0.75
1HARBOR SEAFOOD,INC		Harbor Seafood	001G000001Mj3AeIAJ		0.66667

Observations from example records above:

- The misspelled upper-case word 'INTERNATIONA' matched to 'International'

- The num-char compositive word ‘4C’ matched to ‘4’ ‘C’
- The non-meaningful words ‘IMPORT’, ‘Corp’, ‘.’, ‘,’ and ‘INC’ were filtered out
- The num-char compositive word ‘1HARBOR’ splitted into ‘1’ and ‘HARBOR’ and then matched to ‘Harbor’

- Rules for determining the match being found

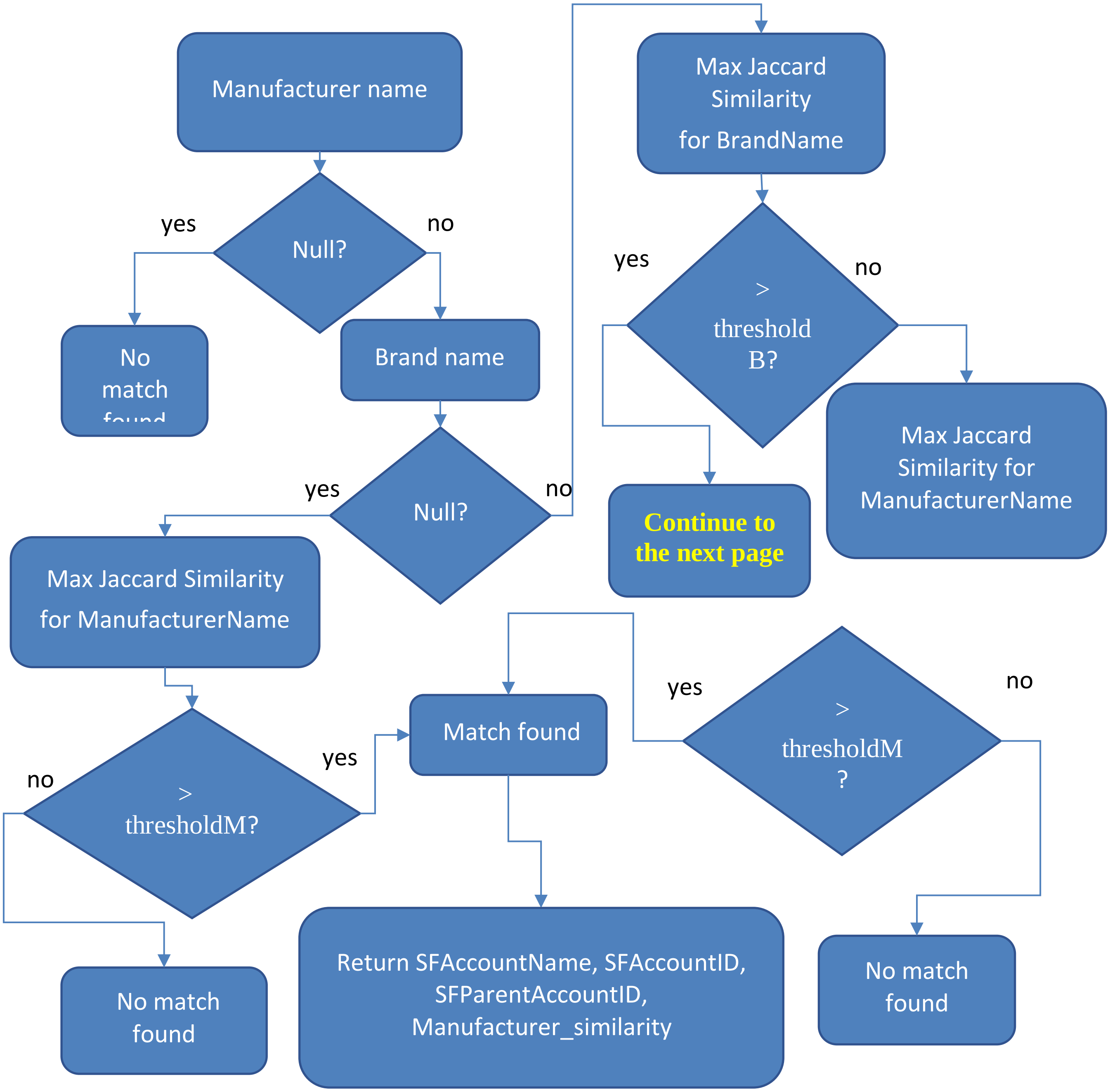
We use the following measures to decide whether a match is found:

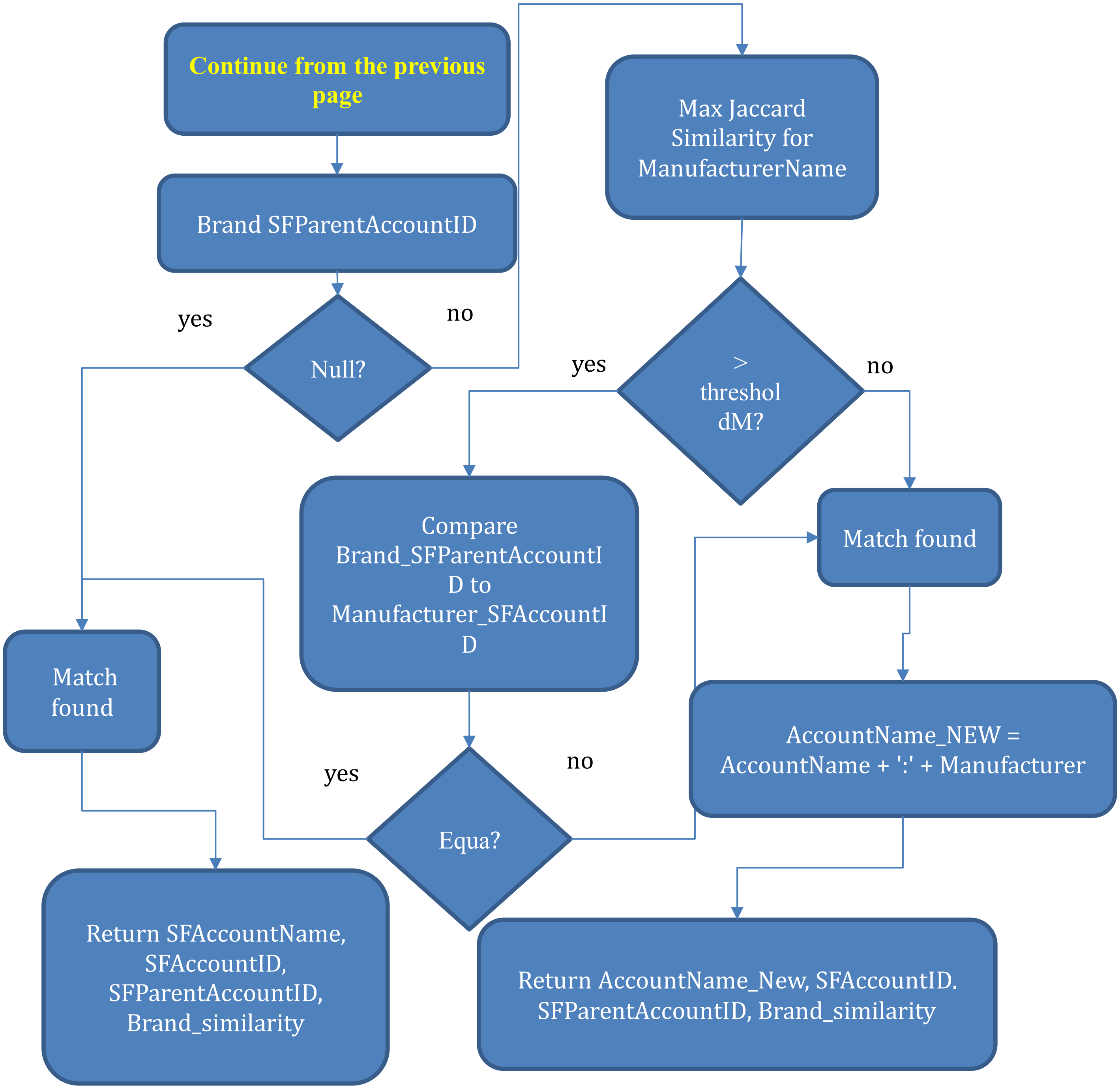
1. When a product’s manufacturer name is null in retailer's file, we say “no match found”. The function call will give a return_values = ["No Match","", "", ""]
2. When the largest Jaccard similarity value among all matches found for a brand name in a retailer's file is larger than thresholdB, and this matching account’s parent account name matches the retailer's manufacturer name of the brand in the retail file, we say that the match is found! The function call will give a return_values = [AccountName_Brand, Brand_Account _SFAccountID, Brand_Account_SFParentAccountID, brand_similarity].
3. When the largest Jaccard similarity value among all matches found for a brand name in a retailer's file is larger than thresholdB, and this matching account’s parent account ID is 'nan', we say that the match is found! The function call will give a return_values = [AccountName_Brand, Brand_Account _SFAccountID, Brand_Account_SFParentAccountID, brand_similarity].
4. When the largest Jaccard similarity value among all matches found for a brand name in a retailer's file is larger than thresholdB, and this matching account’s parent account name doesn't match the retailer's manufacture's name, then according to Item Master’s analytics

team, this should also be considered as a match. Additionally a new record created with AccountName_NEW = parent_SFAccountName:retailer_manufacture. The function call will give a return_values = [AccountName_NEW, Brand_Account _SFAccountID, Brand_Account_SFParentAccountID, brand_similarity].

5. When the largest Jaccard similarity value among all matches found for a brand name in a retailer's file is less than or equal to thresholdB, but the largest Jaccard similarity value among all matches found for the manufacturer name of the product is larger than thresholdM, we say that the match is found! The function call will give a return_values = [AccountName_Manufacturer, Manufacturer_Account _SFAccountID, Manufacturer_Account_SFParentAccountID, Manufacturer_similarity].

- PSEUDOCODE - RULES TO DETERMINE THE MATCH BEING FOUND (DIAGRAM)


Manufacturer name
Null?
yes
no
No match found
Brand name
Null?
yes
no
Max Jaccard Similarity for ManufacturerName
> thresholdM?
no
yes
No match found
Match found
Return SFAccountName, SFAccountID, SFParentAccountID, Manufacturer_similarity
Max Jaccard Similarity for BrandName
> threshold B?
yes
no
Continue to the next page
Max Jaccard Similarity for ManufacturerName
> thresholdM ?
yes
no
No match found


Continue from the previous page
Brand SFParentAccountID
yes
no
Null?
Max Jaccard Similarity for ManufacturerName
yes
> threshol dM?
no
Compare Brand_SFParentAccountID to Manufacturer_SFAccountID
Match found
Match found
AccountName_NEW = AccountName + ':' + Manufacturer
yes
no
Equa?
Return SFAccountName, SFAccountID, SFParentAccountID, Brand_similarity
Return AccountName_New, SFAccountID. SFParentAccountID, Brand_similarity

Example records for account matching rules:

Manufacturer	Brand	SFAccountName	SFAccountID	SFParentAccountID	AcctName Similarity
BEAVER STREET FISHERIES INC	SEA BEST	Sea's Best	001G000002BUvsXIAT	001G000001ffUhIIAU	1
PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA	LIPTON PURE LEAF	Pepsico:PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA	0014A00002Hxs88QAB	001G000001H84BNIAZ	0.66667
3M COMPANY	POST-IT	3M Homecare Division:3M COMPANY	001G0000025ZxY9IAK	001G000001IcYy4IAF	1

Observations from example records above:

- First record, the match is based on 'Brand', and the parent account ID matches Manufacturer's account ID. The Jarccard similarity is 1.
- Second record, the matching account is also based on Brand; however, its parent account ID matched an account name as 'Pepsico', not the manufacturer name of 'PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA'. This tells us that our algorithm can find the match for newly merged customer company such as 'PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA'.

5. RESULTS

The deliverable of this project is to predict the matching account name, account ID, parent account ID, and give a similarity index value for this prediction in every unique combination of manufacturer and brand names.

1) EXAMPLE RECORDS OF THE OUTCOME RESULT FILE

Manufacturer	**Brand**	**SFAccountName**	**SFAccountID**	**SFParentAccountID**	**AcctName Similarity**
IMPORT - PURCELL INTERNATIONA		Purcell International	001G000001H84CLIAZ		1
BEAVER STREET FISHERIES INC	SEA BEST	Sea's Best	001G000002BUvsXIAT	001G000001ffUhIIAU	1
PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA	LIPTON PURE LEAF	Pepsico:PEPSI LIPTON TEA PARTNERSHIP NORTH AMRCA	0014A00002Hxs88QAB	001G000001H84BNIAZ	0.666666667
4 C FOODS CORP BR		4C Foods Corp.	001G000001H83vLIAR		0.75
1HARBOR SEAFOOD,INC		Harbor Seafood	001G000001Mj3AeIAJ		0.666666667
3M COMPANY	POST-IT	3M Homecare Division:3M COMPANY	001G0000025ZxY9IAK	001G000001IcYy4IAF	1
LOREAL USA	SOFT SHEEN CARSON OPTIMUM OIL THERAPY	L'Oreal	001G000001H847OIAR		0.5

BON SECOUR FISHERIES INC	NELSONS	No Match			
1PESCANOVA INC, DBA PES USA		Pescanova USA	001G000001TYjpgIAD		0.4
GREENWOOD PACKING PLANT	CAROLINA PRIDE	Carolina Pride Foods Inc	0014A00002HzBSZQA3		0.666666667
2CAROLINA PRIDE		Carolina Pride Foods Inc	0014A00002HzBSZQA3		0.5

2) Assessment from ItemMaster

The following table shows the results before spelling auto-correction:

Note: In the table below, New Algorithm is our NLP algorithm, and Old "Algorithm" is ItemMaster’s manual work

Retailers	SEG	Walmart	Peapod	Ahold	Similarity	
Total records (products)	18,804	14,688	3,064	8,031		
Matched - New Algorithm	9,277	6,749	2,581	5,343		
Matched - Old "Algorithm"	8,407	5,719	2,640	5,155		
Missing - New Algorithm	9,527	7,939	483	2,688		
% Missing - New Algorithm	51%	54%	16%	33%		
Total Matched (Old + New)	11,228	7,649	2,862	6,177		
%Total Matched (Old + New)	60%	52%	93%	77%		
% Improvement	34%	34%	8%	20%		

Total Missing (Old + New)	7,576	7,039	202	1,854		
% Missing (Old + New)	40%	48%	7%	23%		
Matched - In New, not in Old	**2,818**	**1,919**	**222**	**1,020**	Unique bolded Brand-MFRs	3057
Missed Match - In Old, not in New	1,951	900	281	834	# of Items for those Unique Brand-MFRs	41734

From the table above, we see that the Total Matched (Old + New) for SEG is 11,228, and the number of matches our algorithm found is 9,527. There is still room for our algorithm to improve

The following tables show the results after using spelling auto-correction:

Retailers	SEG - improved	SEG - old	Walmart - improved	Walmart - old
Total records	18,804	18,804	14,688	14,688
Number of Matches	SEG - improved	SEG - old	Walmart - improved	Walmart - old
Similarity=1	8,312	7,678	6,963	6,029
Similarity>=0.8	8,335	7,696	6,970	6,041
Similarity>=0.75	8,507	7,880	7,057	6,144
Similarity>=0.66	9,788	9,277	7,597	6,749
Similarity>=0.6	10,015		7,648	
Similarity>=0.5	12,704		9,371	
Similarity>=0.4	13,198		9,619	
Similarity>=0.33	16,225		12,297	
Time to run	73 min	50 hours	57 min.	42 hours

Retailers	Peapod - improved	Peapod - old	Ahold - improved	Ahold - old

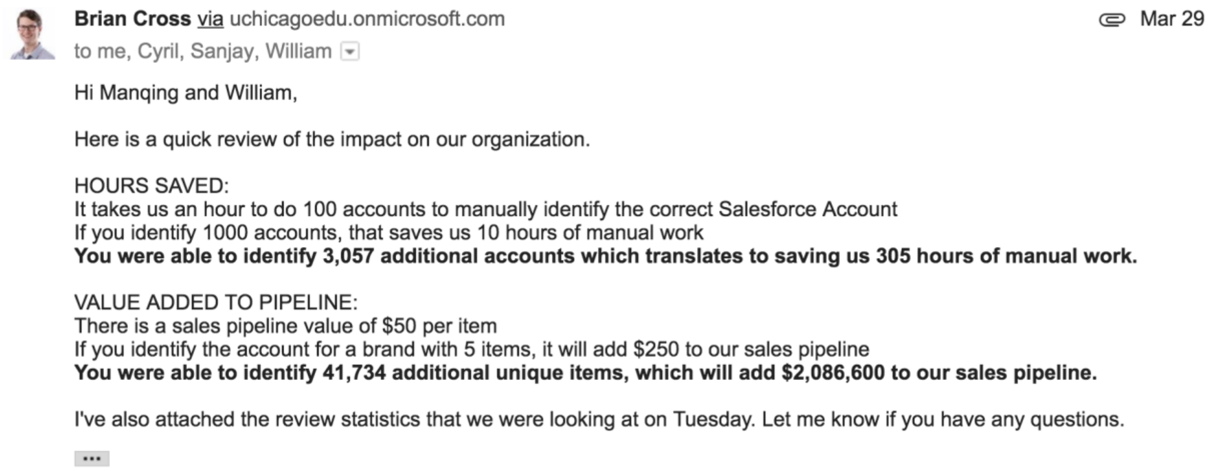
Total records	3,064	3,064	8,031	8,031
Number of Matches	Peapod - improved	Peapod - old	Ahold - improved	Ahold - old
Similarity=1	2,321	1,947	4,962	4632
Similarity>=0.8	2,329	2,022	4,969	4646
Similarity>=0.75	2,368	2,130	5,040	4744
Similarity>=0.66	2,596	2,579	5,556	5343
Similarity>=0.6	2,612	2,581	5,602	
Similarity>=0.5	2,799		6,271	
Similarity>=0.4			6,377	
Similarity>=0.33	2,963		7,361	
Time to run	12 min	10 hours	31 min	23 hours

As we can see, the "Time to run" has been improved 40 to 50 times faster! This is because the old code was running too slowly with the auto spelling correction; as a result, we had to optimize the code. We rewrote many parts of the code using map() and lambda() function, and it served our purpose very well! The improved code not only finds up to 15% more matching accounts, but also runs 40 to 50 times faster!

Considerations in running the old code using the Big Data platform to improve its performance have now becoming totally unnecessary with this improved code.

6. IMPACT ON ITEMMASTER'S BUSINESS

I am attaching the screen shot of the feedback email from ItemMaster's analytics team to show the impact our project made on their business.



Brian Cross via uchicagoedu.onmicrosoft.com Mar 29

to me, Cyril, Sanjay, William

Hi Manqing and William,

Here is a quick review of the impact on our organization.

HOURS SAVED:
It takes us an hour to do 100 accounts to manually identify the correct Salesforce Account
If you identify 1000 accounts, that saves us 10 hours of manual work
You were able to identify 3,057 additional accounts which translates to saving us 305 hours of manual work.

VALUE ADDED TO PIPELINE:
There is a sales pipeline value of $50 per item
If you identify the account for a brand with 5 items, it will add $250 to our sales pipeline
You were able to identify 41,734 additional unique items, which will add $2,086,600 to our sales pipeline.

I've also attached the review statistics that we were looking at on Tuesday. Let me know if you have any questions.

This evaluation was made based on our old code.

7. CONCLUSION

- Our algorithm is not only able to automate the manual work, but also able to find additional accounts and unique items that manual work cannot find. An assessment performed by Itemmaster based on four sets of data provided for our development purposes concluded that our algorithm is able to identify 3,057 additional accounts, which translates to saving ItemMaster 305 hours of manual work plus 41,734 additional unique items, which will add $2,086,600 to ItemMaster’s sales pipeline.
- This NLP algorithm we have developed for ItemMaster Inc. has reached or even exceeded our project client company’s expectation.