NET-A-PORTER

NAP – UCL Project Team 28 Customer Service Chatbot

Wayne Tsui (Team Leader) Aouss Sbai Jason In Bi-Weekly Report #3 27/2/2017

Project Overview (10/2/2017 - 27/2/2017)

In these two weeks, we have prepared our individual work on the chatbot for its integration next week. We have the components of IBM Conversation workspace, the front-end user interface, and the back-end server. We had planned many use cases for the chatbot, but for now we have implemented one of the use case. We now focus on integrating the components together before finishing off the other use cases. Once the integration is done, we would be able to show it to our client to get their feedback.

Meetings Summary

We had no face-to-face meetings these two weeks as it was reading week and scenario week. However, we gave progress updates to one another through our group chat with images of our work.

Tasks Completed

- Exported "Categories" and product sort information from API call to JSON files
- Wrote scripts to convert JSON files to CSV files
- Uploaded CSV files to populate above mentioned entities efficiently
- Populated synonyms for above mentioned entity values using Regex
- Crafted dialog for product recommendation intent
- Tested IBM Conversation service call from application

Problems to be Resolved Before Next Meeting

While we are able to get user intents and entities from their input, specifying how product information is shown to users is a big design problem. We wanted information output from the chatbot to be easily read and processed by users, and also for them to prefer using the chatbot than to just browse the website or google any queries they might have. Another problem is to make sure that our integration between the 3 components will be smooth.

Plan for Next Two Weeks

The plan is to integrate the 3 components together. Following that, it will continue involving us working on the individual components as working in parallel is more efficient. For the IBM Conversation workspace, it will involve planning and creating several more intents such as Site Redirect, FAQ, Returns Policy, Promotions, Fashion Advisor, etc. The front-end will involve user interface design in terms of representing the output information. The back-end needs work on exposing a simple API that the front-end can use (potentially available for use next time with other NAP teams), which inputs a string message (user query) and outputs the answer in JSON.

Contributions

Wayne Tsui

% Intents Entities Dialog	customer-service-chatbot 💬	\equiv % Intents Entities Dialog		customer-service-chatbot 🤄
y entities System entities		Conversation starts > Welcome		PR - Entity Matching
Create new 🕘 🗘	8 entities Sort by: Newest 🗸	> welcome		A Three O
> @product_category Accessories, Bags, Beauty, Clothing, Gifts, Lingerie, Shoes, Sport, Vacation Shop, What to Wea	, Winter Shop	Product Recommendation Traper #product_recommendation	Jump to の前 PR - Entity Matching	1 Trigger () If @product_category_type () @product_category_type () @product_category_type () @r
@product category Lype All Sortower, Backvalos, Bag Accessores, Bags and Waltets, Beth and Body, Bach Accessor Beakly Catess, Beakly Stearfain, Beakly Steit, Beth, Biock-Heels, Bodyauts, Books, Boo Carrisolea and Dinnies, Candida, Cathmen, Cathmee Glit, Cothing Pathon, Cuth Blags, All Denses, Equentine, Espadielles, Evering, Streinig Bags, Evering Stoes, Exclusives and Off Box Denses, Equentine, Isopathes, Favirra, Terring, Bags, Evering Stoes, Exclusives and Off Box Denses, Equentine, Isopathes, Cathward, Cather Cather, Cather Bags, Denses, Equencing, Levery, Swelly and Wathres, Junganis, Roychars, Knither, PORTER Magas Shapeware, Ninth Sags, Nais, Opticals, Outdoor, Over 500, Partis, Parts and Skrits, Bespresser, Sweeten and Cats, Swell and Sart, Bod Chick, Sciences, Strits, Bespresser, Sweeten and Kats, Swell and Sart, Roodine S, Chick, Sciences, Strits, Bespresser, Sweeten and Kats, Swell and Sart, Roodine S, Colic, Skiences, Strits, Bespresser, Sweeten and Kats, Swell and Sart, Roodines, Walter, Markan Accessores, Wither Baus, Yoba and Chi Malets, Wather, Matchward, Wathers, Walter, Wather, Markan Accessores, Wither Baus, Yoba and Chi Sart, Post Sart, Post Dense, Teophysica, Baus, Sart, Baus, Cathol, Sart, Post Baus, Stris, Benser, Stris, Baus, Markan, Cathol, Stris, Baus, Markan, Stever, Sart, Dense, Teophysica, Baus, Stris, Baus, Walter, Michael, Markan, Wather, Markan, Wathers, Wath	8, Bridal Accessories, Bridal Lingerie, Briefs, ands, Cauts and Juckis, Constry, Comentica Cases, Denim, ss, Fina Jaweity, Fine Watches, Fattorms, Fatt Shoes, Foottwar, at Accessories, Haitmane, Haits, Hornware, Hosidery, In Fight, Arcessories, Haitmane, Haits, Hornware, Hosidery, In Fight, Piez, Lingerie and Sleepware, Longeware, Longeware, Battoreny, Suits, Sun, Sunglasses, Suspender Beits, eakers, Stationery, Suits, Sun, Sunglasse, Suspender Beits, eakers, Stationery, Suits, Sun, Sunglasses, Suspender Beits, eakers, Stationery, Suits, Sun, Sunglasses, Suspender Beits, eakers, Stationery, Suits, Sun, Sunglasse, Stationer, Stationer, Stationer, Sunglasse, Stationer, Stationer, Stationer, Sunglasse, Stationer, St	Water monese DECKJO, Merchar Product Recommendation	 PR - Entity Matching Topor Oproduct, cellegory or oproduct, cellegory or oproduct, series devension Oproduct, series devension Oproduct, series devension Oproduct, series devension DEBUG. Matched less che productors. 	
Oproduct_category_type_specific Accessories, Arkie Boots, Arkier, Bag Charm, Bags, Bag Shrapa, Balconette Bras, Balet Flats, Beach Dresses, Bearies, Berets, Bervark, Bervark, Body Meatarizer, Body Ok, Bodyauka, Bohr, Bornber, E Body Exhibitint and Scrub, Body Jewarhy, Body Meatarizer, Body Ok, Bodyauka, Bohr, Bornber, E Caterrene, Catalande Body, Body Jewarhy, Body Meatarizer, Roby Ok, Bodyauka, Bohr, Bornber, E Caterrene, Catalau Lestet, Cathar Brang, Chernies, Cohener, Caterrare of Cathaire, Cohingo Caterrene, Catalau Lestet, Canhar Brang, Chernies, Cherne, Catalau Lestet, Cathar Hairane, Hairon, Hair Cote, Harris, Reving, E., Ney, Keyler, E., Watesan, Pilot, Bodyau Hairane, Hairon, Hui Cote, Harri, Larring, Serving, C., Ney, Keyler, E., Watesan, Millow, Born, Hairane, Huing, Hui Cote, Harri, C., Hendare, Handhand, Haddhonet, Lendander, Bendard, Hairane, Huing, Hui Cote, Harri, C., Hendare, Handhand, Haddhonet, Lendare, Borney, Holley, Illuminator, Phone S Cases, Phone S Cases, Dhone S Dase, Jacke Dagdonau Liba, Lin Mehok, Cuterwar, Orent Keyle, Lang Balewar, Melaka, Malkewy B Nightdresse, Rowdrig Mith, Poncho, Pouch, Powder, Primer, Printed, Purap, Patets, Patets, Peters, Howel, Ofthis Shouddo, Cuterwar, Orene there, Name, Sharring, Shirri, Shorth, Shouder Bag, Shoudder Shang, Sk, Simpo, Sharpeo, Sharpewar, Sharring, Shirri, Shorth, Shouder Bag, Shoudder Shang, Sk, Simpo, Sharpeo, Sharpe	kazom, Biouses, Bukh, Body, Body Cleanser and Sonp, oos, Bou, Boylford, Bronellas, Bridls, Unreisk, Breita And Shorts, minises, Candida, Capes, Cape, Carchidders, Cardigun, Christog Cam, Cockia, Color Tisseld, Concessione, Contillorent Print, Carlos, Cardia, Cardinal Cardia, Cardiana Santo, Strandard, Fano, Faoro OI, Faihlon Jewelly, Fedoras, Y. Torzy Hair, K. Lingeri, Farard Shearting, Gavons, Hair, Kini, High Heal, High Filea, High Top, High Waisted, Hotali, thi, Lakostar and Yesen, Javerby Cases, Lingenste, Maseu, Mascana, Maxik, Masocae and Traitments, Matemity, mas, Nails, Neck and Dacollen, Recolations, Night Chem, and Noch and Dacollen, Recolations, Night Chem, and Nator, Bardyana, Removers, Filey, Bridg Brites, Dinos, Short, Short, and Min, Short Robes, Shorts, Steved, Sleweikers, Sides, Sitypen, Sites, Sharts, Stevelsmen,	> anything_else	 	DEBUG: Matched at least one product recommendation related entity Add a variation to this response Orable another response Orable another response
Populated entities from CSV		Dialog	flow for Produc	t Recommendation
here is Product Categories fro	III NET-A-PORTER)			size m, from lowest to highest price
<pre>Woynes-MacBook-Pro:nap-chatbot waynetsui\$ node app.js Hello there! What can I do for you? ENTIRE OUTPUT: { intents:], entities:], input: {}, output: {log_messages:], text: ['Hello there! What can I do for you?'], nodes_visited: ['Welcome'] }, context: { conversation_id: '9be5077e-c041-4307-baca-8d9f7df65b8 system: { dialog_turn_counter: 1, _indet_output_map: [Object] } } } </pre>		Detected intent: #product_ree Detected intent - value: #pro Detected entity: eproduct_sor Detected entity: eproduct_cor Detected entity: eproduct_cor Detected entity: eproduct_cor Detected entity: eproduct_sor Detected entity: eproduct_sor ENTIRE OUTPUT: { intents: [{ intents: [{ entity: 'product_sort I coordidnce: 0.93405420 entities:] { entity: 'product_cort I coordidnce: use 'price-desc' }, value: 'price-desc' }, { entity: 'product_colo	<pre>duct_recommendation t t e-desc our k egary_type s eedation mendation', 7801B188 }], , r', location: [Object], value:</pre>	'Black' },
>> Ola wants to see view some black suits by balenciaga in size m, from lowest to highest price Detected intent: #product_recommendation Detected entity: @product_sort Detected entity: @product_colour Detected entity: @product_category_type Detected entity: @product_category_type Detected entity: @product_soits Detected entity: @product_size Detected entity: @product_size Detected entity: @product_size Detected entity: @product_size Detected Product Recommendation ENTIRE OUTPUT: Test IBM Conversation se		{ entity: 'product_size' input: { text: 'Ola wonts t	ion: [Object], value: 'Balenci) location: [Object], value: 'b o see view some black suits by ct Recommendation', set one product recommendation t Recommendation', 'PR - Entity 7e-c041-4307-baca-8d9f7fdf65b8']. 2. p: 2.	<pre>4' }], balenciaga in size m, from lowest to highest price' }, related entity'], / Matching'] },</pre>
г	oct IRM Conversation	vice call from external	application	

I have been working on exporting the NET-A-PORTER product categories into the IBM Conversation workspace. It involves breaking down the categories into 3 different layers, as I have to work with the structure of how it is represented in the JSON response from the API call.

After the intents and entities are populated, I crafted a dialog for how user would query with the intent for product recommendation. The response would clearly provide the necessary information (such as colour, brand, product type, etc if available) for the back-end to call the correctly configured API call.

Lastly, as we were all working in parallel and yet to integrate, I wrote a simple script to test the IBM Conversation service by calling it from an external application. Seen above is a test query and the response from the Conversation service, correctly identifying the product information found in the query.

Aouss Sbai

During the past 2 weeks, I have been working on implementing intent and entity identification from the IBM Watson output file. That is to say that I have tried to polish the parsing process in order to recognize a broader range of queries and make the appropriate API calls.

The latter part has also been a difficult part to handle. In fact, conceiving the structure that builds the API queries from the intent recognition has caused me issues.

The objective is now to standardize the output file so it is usable and exploitable by the front end.

Jason In

For the past 2 weeks, I have been focusing on the front-end design of the chatbot. The languages that I have been using are AJAX, HTML, CSS and JavaScript.