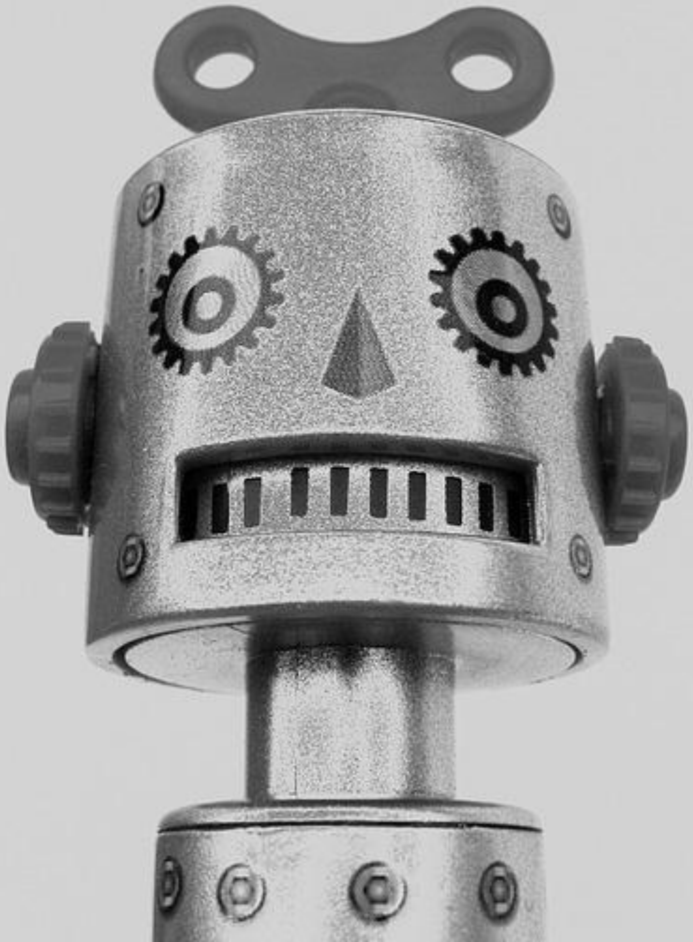
UDD - Universidad del Desarrollo_MDS - Master in Data Science

PRODUCT RECOMMENDER CHATBOT



Students

Rovai, Marcelo_ Sacasa, Manuel

Professors

Reinoso, Pablo_ Seguel, Rodrigo

May, 2019

PARTS & PIECES

Steps for the developing, designing and building a recommender chatbot

1_DATA SET AND MODEL

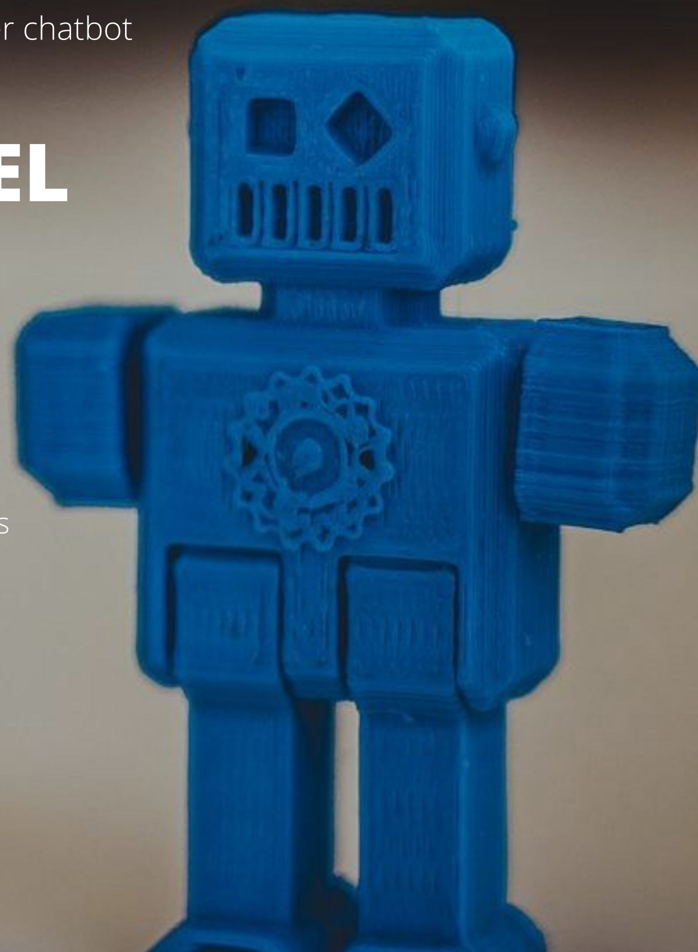
Data engineering and filtering to built a Bot Dataframe with users and recommendations

2_BOT DESIGN

Architecture, personality and Q&A tree for different recommendations

3_BUIDING A BOT

Cloud tools, intents, entities and dialogues



COLLABORATIVE FILTERING

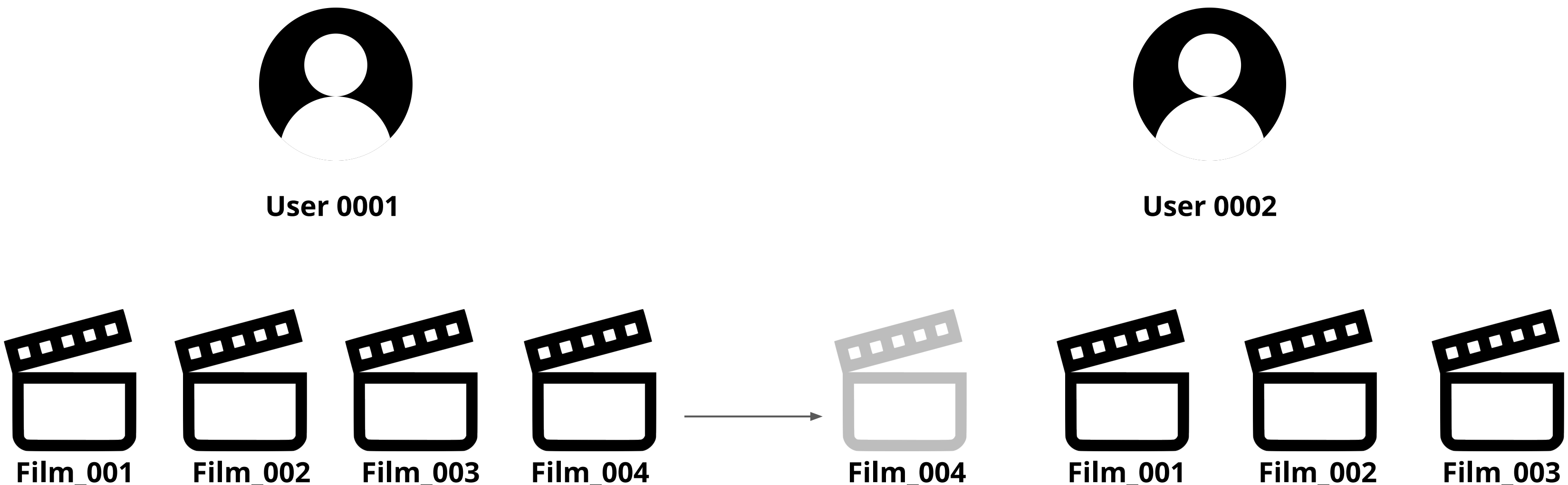
As stated by Yifan et al, a common task of recommender systems is to improve customer experience through personalized recommendations based on prior implicit feedback. These systems passively track different sorts of user behavior, such as purchase history, watching habits and browsing activity, in order to model user preferences. Unlike the much more extensively researched explicit feedback, we do not have any direct input from the users regarding their preferences. In particular, we lack substantial evidence on which products consumer dislike.

Collaborative filtering refers to the process of identifying patterns among the objects in a dataset in order to make a decision about a new object. In the context of recommendation engines, we use collaborative filtering to provide recommendations by looking at similar users in the dataset.

The assumption here is that if two people have similar ratings for a particular set of movies or series, then their choices in a set of new unknown movies would be similar too. By identifying patterns in those common content, we make predictions about new movies or series.

Collaborative filtering is typically used when we have huge datasets. These methods can be used for various verticals like finance, online shopping, marketing, customer studies, and so on.

RECOMMENDER APPROACH



Similar user without a film. Recommendation film/serie to complete perfect match

Similar user without a film. Recommendation film/serie to complete perfect match

DATA SET

•All data are from Chile

•6 months of data, collected from 09/01/2018 to 03/10/2029

•187K observations:

•Series 123,016

•Movies 58,845

•Live TV   5,402

•41,442 unique customers

•12,845 different titles on dataset (*id_grupo*)

•Each Serie episode has a unique *id_grupo*

•4,373 unique movie/series titles

•There are 2 types of operation: Subscription (~94%) and Rent

•Majority of streaming happen in mobile phones, followed by TV and web browsers

•30% of "MAX_VIS", that is the maximum time spent on streaming are “null”

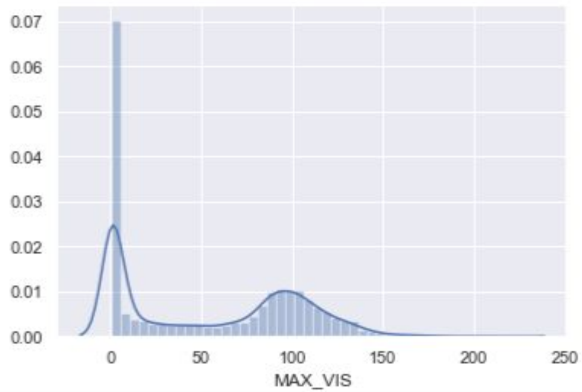
•5,016 ID_CLIENTE are NULL - Make no sense customers with Null ID, should be deleted

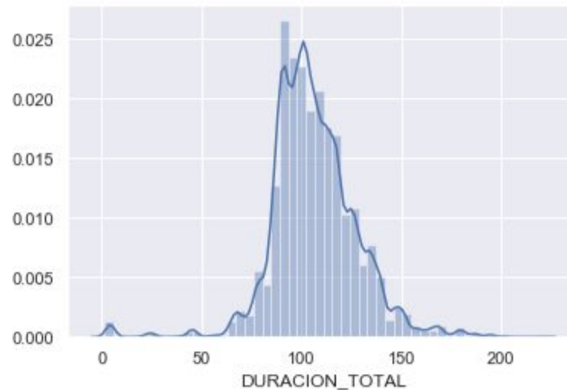
•ID_GRUPO and ID_CLIENTE will be converted to integer to reduce memory space

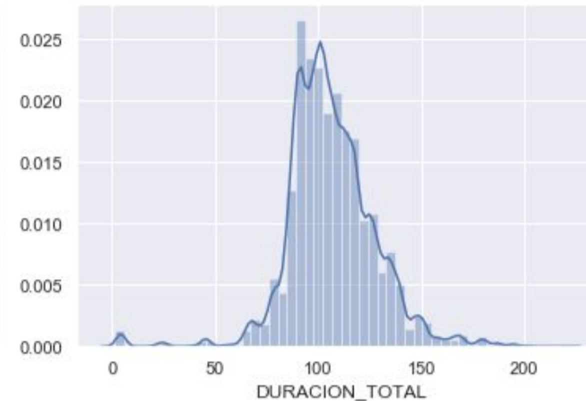
•“DURACION_TOTAL” has 3,655 observations invalid. From that, 3,653 are related with “Live TV” and 2 for movies. It is OK for Live TV to have no data regarding total duration time, but not for movies. Those 2 observations are related with ID_GRUPO = 779381, that has a null total duration time and was not seen by users (o or also null time). Those data should be deleted.

•FECHA_PAIS should be converted to *datatime format*

1_DATA SET CLEANING



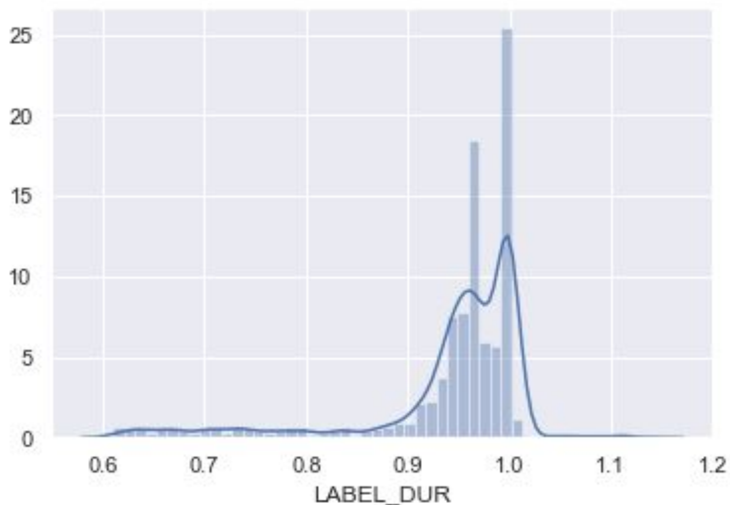


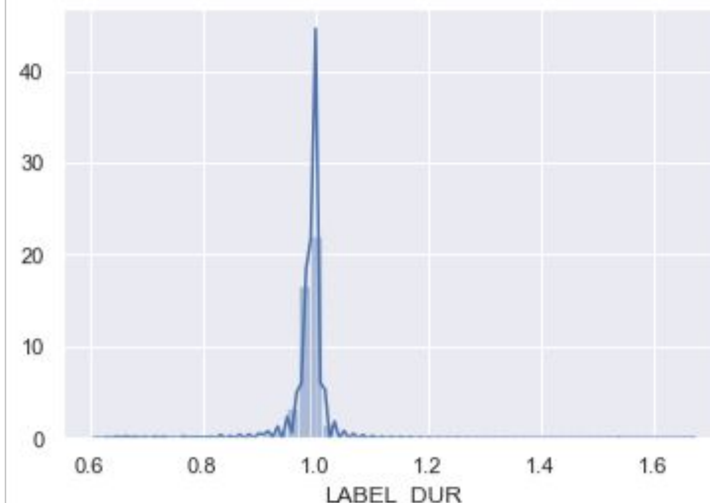


- MAX_VIS being Null make no sense for series and movies (it is OK for Live TV), but should be kept because shows that spite that user could not "connect" (or data is wrong), he is in principle interested on that content. Null value should be converted to '1' (1 minute of view).
- MAX_VIS negative values (1 observation) must be deleted.

- DURACION_TOTAL , that is the total duration time of content must be converted to minutes format (for example: 01:20:30 converted to 80.5)
- There are 26 items, Series episodes with total duration equal to zero. We must change it to median that should be 42 minutes

1_DATA SET CLEANING





- LABEL_REC: A new column is created with the simple (and clean) title, been series (w/0 "s2e7" for example) or movie.
- LABEL_DUR: A new column with percentage of max view over total duration is created
- Dataset is filtered for LABEL_DUR > 60% and duplicates eliminated.

1_THE MOST SEEN

•Once we have 2 filtered and cleaned datasets, one for movies and one for series, where only content that was really saw was kept (more than 60%), it is interesting to know what were the most seen movies and series on current/past month. This answer will be important to recommend content to users that have not enough data to be used with trained model.

ID_GRUPO	TITULO
531775	League of Extraordinary Gentlemen. The
536902	Mean Girls (2004)
542014	Admiral, The (aka Isoroku Yamamoto)
605493	Megafactories: Extreme Roller Coaster
680953	Recién casados
711992	Curious George 3: Back to the Jungle
716084	Pequeños privilegios, Los
756410	Inframundo
757049	Eragon (2006)
778548	Karate Kid

ID_GRUPO	TITULO
580933	Ultimate Spider-Man
685392	Saving Grace
694736	Family Guy
730798	Gran Hotel
750951	muñecas de la mafia, Las
755020	Esmeraldas
755022	Esmeraldas
771970	Hijos de su madre
777539	Reign
778192	Señora Acero

1_DATA PRE_PROCESSING

•The data to be used with Implicit Library, should be grouped in two variables, ID_CLIENT (user) and ID_GRUPO (content), having a count column created with number of times that a specific content were seen (in our case will be 1).

•Those variables must be converted to “category”

•To make sense, a recommender model based on content seen by a user must have a minimum of items seen by an individual customer in order to use it for recommendation. 4 is an empirical value based on practice. But for this project, we will keep it at “1”, because only 10% of users has a decent number of content history.

1_THE MOST SEEN

•Once we have 2 filtered and cleaned datasets, one for movies and one for series, where only content that was really saw was kept (more than 60%), it is interesting to know what were the most seen movies and series on current/past month. This answer will be important to recommend content to users that have not enough data to be used with trained model.

ID_GRUPO	TITULO
531775	League of Extraordinary Gentlemen. The
536902	Mean Girls (2004)
542014	Admiral, The (aka Isoroku Yamamoto)
605493	Megafactories: Extreme Roller Coaster
680953	Recién casados
711992	Curious George 3: Back to the Jungle
716084	Pequeños privilegios, Los
756410	Inframundo
757049	Eragon (2006)
778548	Karate Kid

ID_GRUPO	TITULO
580933	Ultimate Spider-Man
685392	Saving Grace
694736	Family Guy
730798	Gran Hotel
750951	muñecas de la mafia, Las
755020	Esmeraldas
755022	Esmeraldas
771970	Hijos de su madre
777539	Reign
778192	Señora Acero

1_RUNNING THE MODEL

- The date will be split in Train (80%) and Test (20%)
- The model will be run separated, once for movies dataset and once for series dataset.
- The model from Implicit library to be used is: “NMSLibAlternatingLeastSquares”, with parameters:
    - factors=35,
    - regularization=0.5,
    - iterations=25,
    - calculate_training_loss=True
- From Train part of data, a sparse matrix is created
- The model is applied over the sparce matrix
- Alpha parameter of 5 (empiric)

```
m_model = NMSLibAlternatingLeastSquares(
    factors=35,
    regularization=0.5,
    iterations=25,
    calculate_training_loss=True)

print("Starting Training movies model")
m_model.fit(df_movies_train_csr * 5.0)
print("Trained movies model")
```

```
s_model = NMSLibAlternatingLeastSquares(
    factors=35,
    regularization=0.5,
    iterations=25,
    calculate_training_loss=True)

print("Starting Training Series model")
s_model.fit(df_series_train_csr * 5.0)
print("Trained Series model")
```

1_TESTING THE MODEL

Selecting IDs from top users

```
1 id_client = 164059254
2 recs = show_movies_recomendation(df, id_client)
```

executed in 71ms, finished 19:23:04 2019-05-15

```
Movies saw by user 164059254:

      ID_GRUPO        TITULO
976     526499  Brother Bear
1312    526591   Tinker Bell
4467    528329        Cars 2
6243    529168     I, Robot
6774    530047          Bean

Movies recomended to user: 164059254:

    ID_GRUPO                 TITULO  confidence
0     561960    Resident, The (2011)   0.378643
43    775496  Zapatero a tus zapatos   0.256077
53    777220           Querido John    0.256077
```

```
1 id_client = 85528236
2 recs = show_series_recomendation(df, id_client)
```

executed in 76ms, finished 19:22:53 2019-05-15

```
Series saw by user 85528236:

       ID_GRUPO              TITULO
21346    552506          Nanny, The
23013    555540    Creature Comforts
27416    569962   Castle: Nikki Heat
28352    572244          Niñas mal 1
30964    577360         Drake & Josh

Series recomended to user: 85528236:

    ID_GRUPO                                   TITULO  confidence
0     771612                 Simuladores, Los (México)   0.247737
2     593877                                     Bones   0.247737
32    601287  Avatar: The Legend of Aang: The Beach      0.247737
```

1_TESTING THE MODEL/API

•Selecting ID from user with few content. The Model will not return a recommendation.

•When a user has very few views, for example (id_client = 151650306), with only 2 movies seen and no series, the final API should return for example the "the most seen content of the month".

```
1 get_user_watched_content(df, id_client, content = 'Serie')
```

executed in 25ms, finished 14:39:58 2019-05-17

ID_CLIENTE	ID_GRUPO	TITULO	LABEL_REC	CATEGORIA

```
1 recs = recommend_series_2(df, id_client)
```

executed in 20ms, finished 14:39:59 2019-05-17

```
[INFO] Not enough content ==> Recommending best series of the month:

        ID_GRUPO               TITULO
32236     580933  Ultimate Spider-Man
103261    685392         Saving Grace
107847    694736           Family Guy
```

1_CREATING A RECOMMENDATION DATA BASE

```
rec_mov.sample(2)
```

executed in 14ms, finished 17:34:58 2019-05-18

	id_client	id_group_1	id_group_2	id_group_3	title_1	title_2	title_3	confiability_1	confiability_2	confiability_3
23105	214377036	531775	536902	542014	League of Extraordinary Gentlemen. The	Mean Girls (2004)	Admiral, The (aka Isoroku Yamamoto)	9.0	9.0	9.0
26666	136662810	531775	536902	542014	League of Extraordinary Gentlemen. The	Mean Girls (2004)	Admiral, The (aka Isoroku Yamamoto)	9.0	9.0	9.0

```
rec_mov.to_csv('recomm_movies_may_19.csv', sep=';', index=False)
```

executed in 387ms, finished 17:31:20 2019-05-18

•To the original list of all users will be applied the model and the 3 recommendations with its respectively data: ID_GROUP, Title and Confiability, will be saved on a CSV file (one file for movies and one file for series.

•In case of the user returned no data (few content saw by user), the monthly recommendation will be add to that specific user. An arbitrary confiability of "9" was plugged. This number is to help the easy identification of a user that was recommended with the monthly content.

month".

1_OPTIONAl: ART COVER RECOVERY

```
id_client = 164059254
covers = movies[movies.id_client == id_client]
create_cover_display(covers)
```

executed in 17.1s, finished 18:41:04 2019-05-16

```
[INFO] Wait, recovering possible covers..... [DONE]
Recommendation 1: Resident, The (2011)
Recommendation 2: Zapatero a tus zapatos
Recommendation 3: Querido John
```


```
id_client = 164059254
covers = series[series.id_client == id_client]
create_cover_display(covers)
```

executed in 19.0s, finished 17:47:36 2019-05-16

```
[INFO] Wait, recovering possible covers..... [DONE]
Recommendation 1: Bienvenida realidad
Recommendation 2: Rubirosa
Recommendation 3: En la boca del lobo
```


```
id_client = 189425022
covers = movies[movies.id_client == id_client]
create_cover_display(covers)
```

executed in 19.8s, finished 18:42:14 2019-05-16

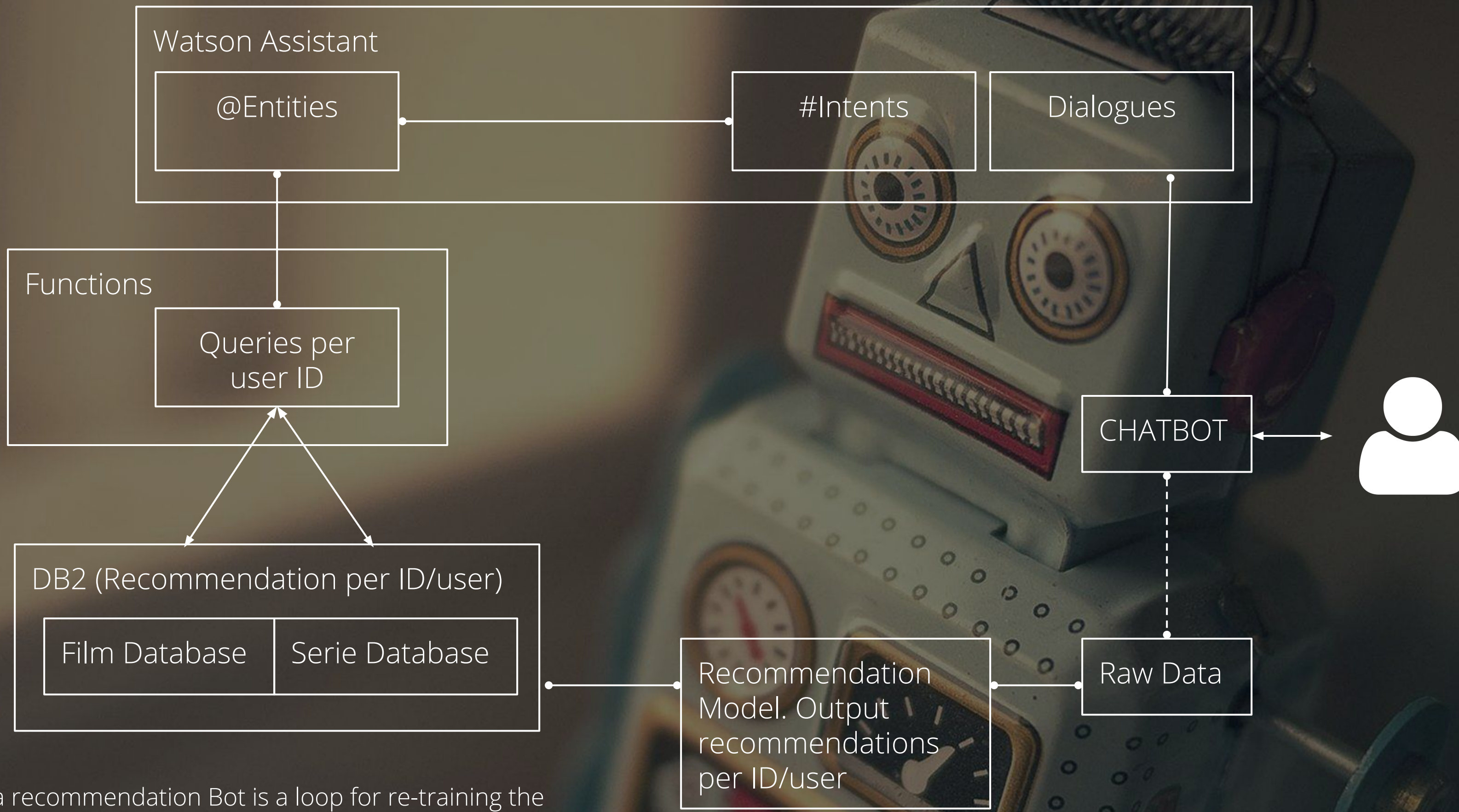
```
[INFO] Wait, recovering possible covers..... [DONE]
Recommendation 1: Evan Almighty
Recommendation 2: Quartet
Recommendation 3: Pretty Woman
```


```
id_client = 184928670
covers = series[series.id_client == id_client]
create_cover_display(covers)
```

executed in 19.3s, finished 18:43:29 2019-05-16

```
[INFO] Wait, recovering possible covers..... [DONE]
Recommendation 1: 24
Recommendation 2: Isabel
Recommendation 3: Tudors, The
```

2_ARCHITECTURE



Technically a recommendation Bot is a loop for re-training the recommendation model with new users preference. In this first exercise we disable the connection between chatbot and raw data

2_PERSONALITY

FORMALITY

The answer should have the same tone, formal style and grammar. For example: Tutear, usted, etc. This will profiled the bot as an expert, friend, etc.

LANGUAGE

The answer should be written in the same language of the user.

EXPRESSIONS

The style, saids and ways should be coherent with the formality tome of the dialogue. Not= a friend of 80 years, an expert of 15 years

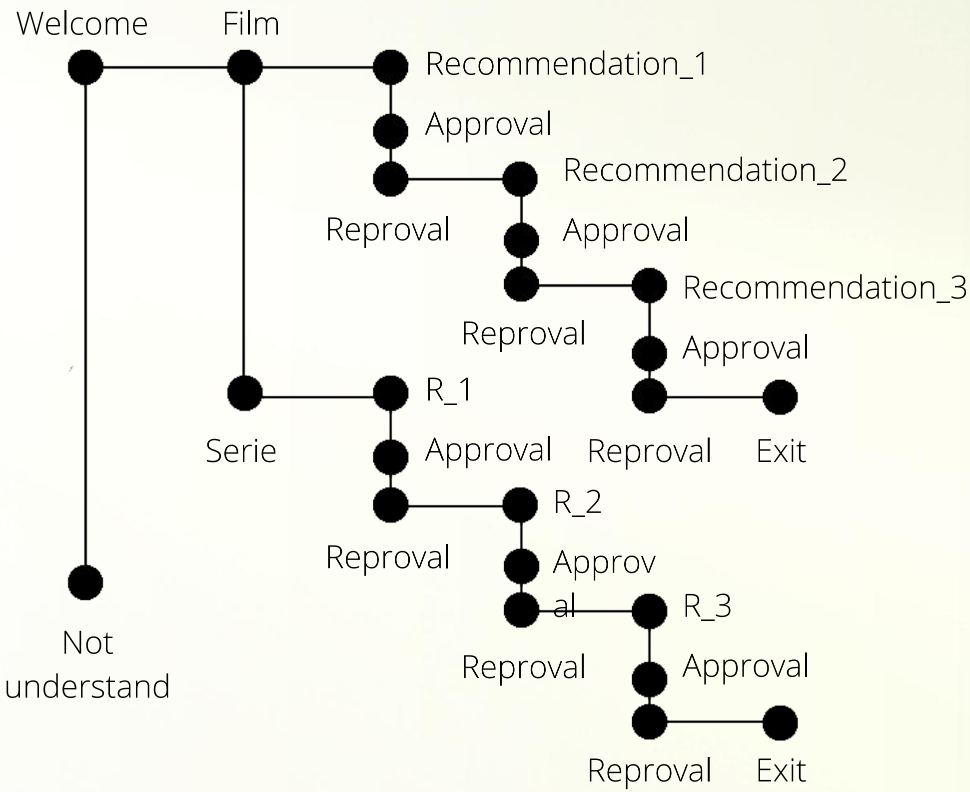
VALUE TIME

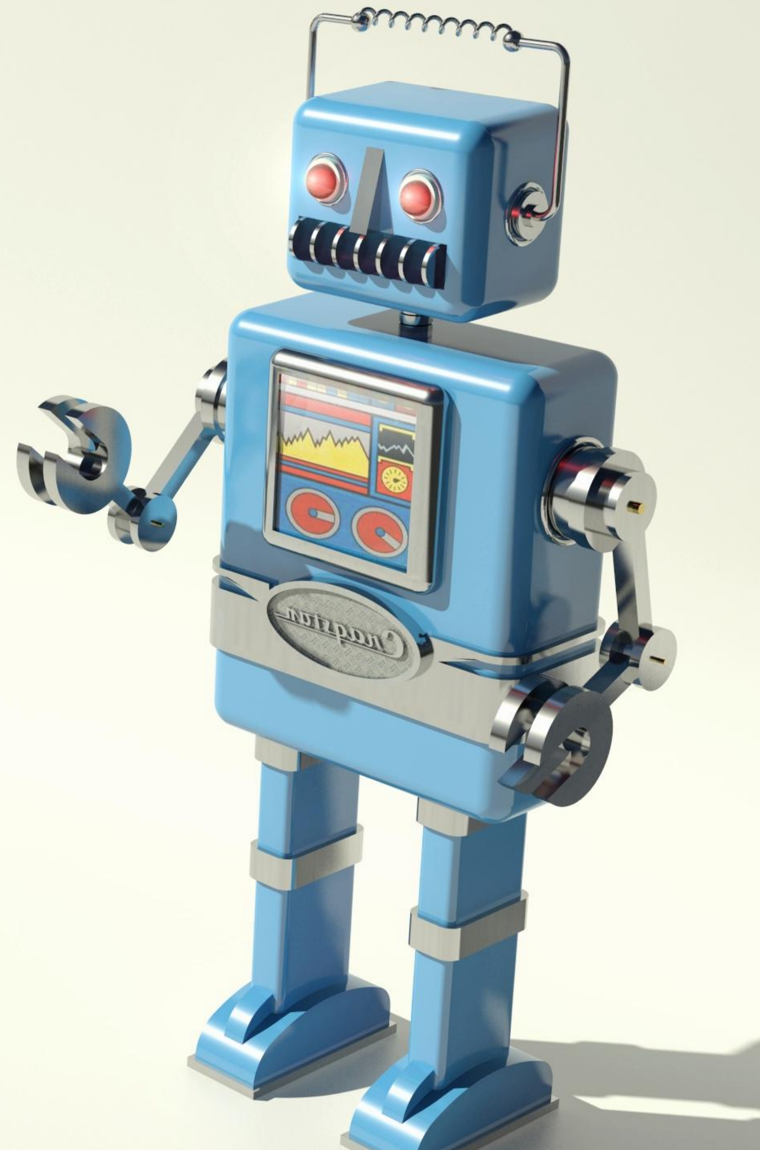
User should have a recommendation as an answer next after a film/serie request. Not over Chat.

COHERENCE

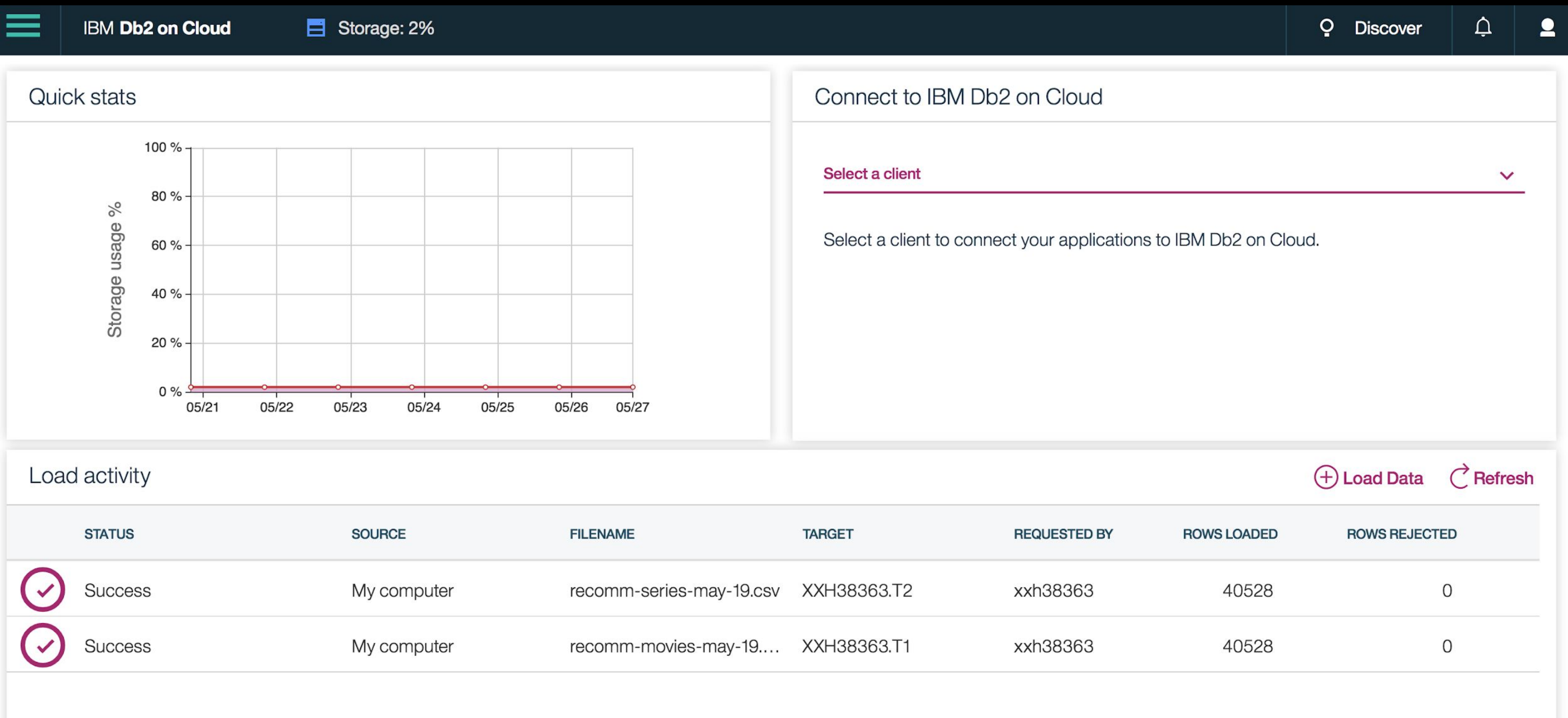
All the personality design decisions should be in all the dialogues. No answers with “ud” and other with “tu”. Or singular and plural.

2_Q&A TREE LOGIC





3_FILM/SERIE DATABASE



3_FILM DATABASE STRUCTURE

IBM **Db2 on Cloud** Storage: 2% Discover

Back

XXH38363.T1

Delete Table Export to CSV

	ID_CLIENT INTEGER	ID_GROUP_1 INTEGER	ID_GROUP_2 INTEGER	ID_GROUP_3 INTEGER	TITLE_1 VARCHAR(50)	TITLE_2 VARCHAR(58)	TITLE_3 VARCHAR(89)	CONFIABILI... DECFLOAT(34)	CONFIABILI... DECFLOAT(34)	CONFIABILI... DECFLOAT(34)
1	151650306	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
2	151257090	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
3	183500820	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
4	202637340	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
5	216662058	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
6	174456876	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
7	210239532	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
8	178520112	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
9	141426738	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
10	149946420	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
11	132382776	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0
12	205914174	531775	536902	542014	League of Extraordi	Mean Girls (2004)	Admiral, The (aka Is	9.0	9.0	9.0

3_INTENTS & ENTITIES

IBM Watson **Assistant** Preferencias para cookies

Skills /

Try it

Recomender

Save new version

Intents **Entities** Dialog Analytics Options Versions Content Catalog

My entities System entities

Create entity

Entity (3)	Values	Modified
@film	película	15 days ago
@new_film	película nueva	15 days ago
@serie	serie	15 days ago

IBM Watson **Assistant** Preferencias para cookies

Skills /

Try it

Recomender

Save new version

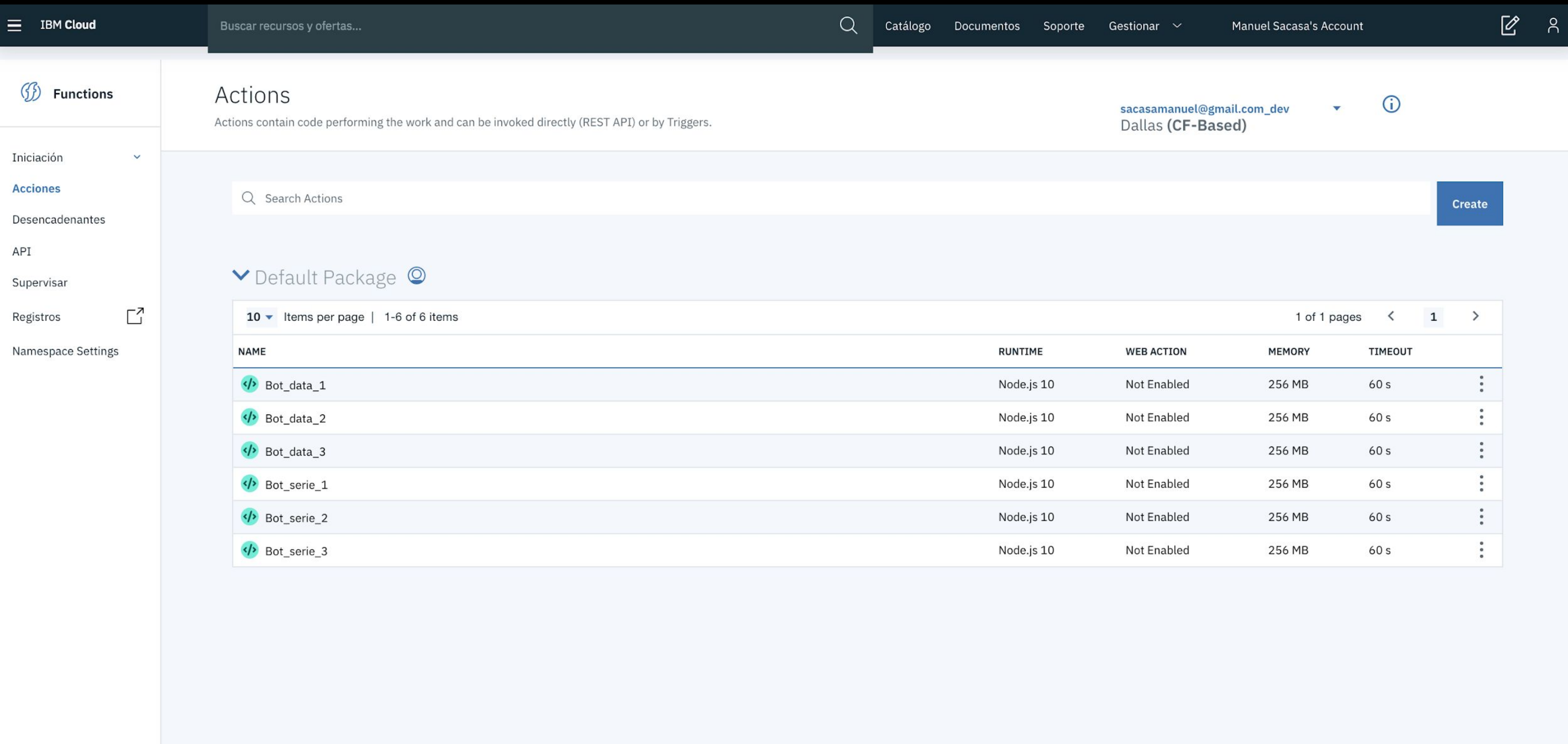
Intents Entities Dialog Analytics Options Versions Content Catalog

Create intent

Show only conflicts

Intent (6)	Description (optional)	Modified	In Conflict	Examples
#aproval	be ok with the recomendation	8 days ago		8
#film_recomendation	a film recomendation for an user	15 days ago		7
#ID_Client	Client or user number	9 days ago		0
#new_film	recomendación de película nueva	15 days ago		6
#reproval	not ok with de recomendation	4 days ago		14
#serie_recomendation	a TV serie recomendation	15 days ago		7

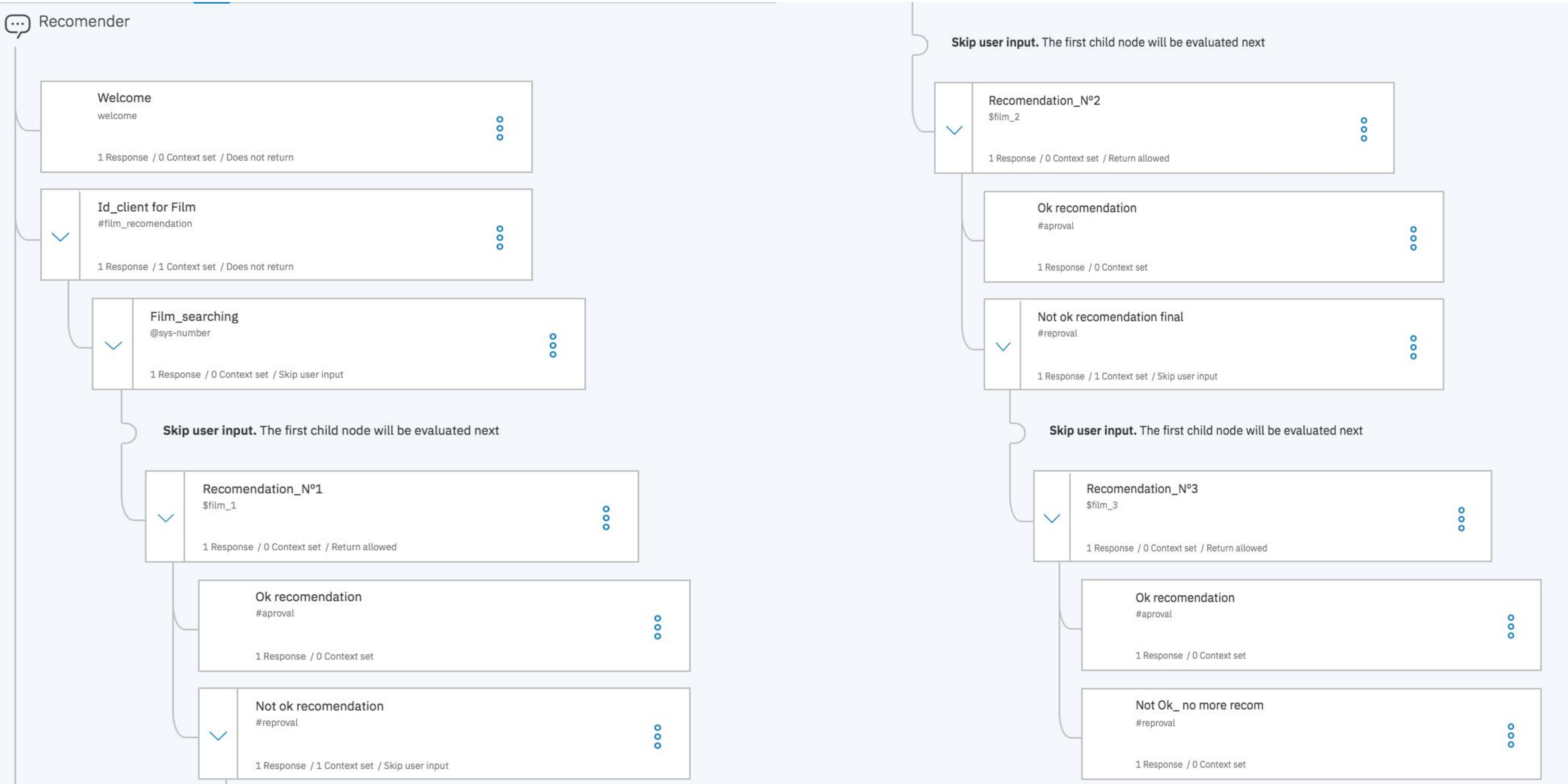
3_QUERIES


IBM Cloud
Buscar recursos y ofertas...
Catálogo
Documentos
Soporte
Gestionar
Manuel Sacasa's Account
Functions
Iniciación
Acciones
Desencadenantes
API
Supervisar
Registros
Namespace Settings
Actions
Actions contain code performing the work and can be invoked directly (REST API) or by Triggers.
sacasamanuel@gmail.com_dev
Dallas (CF-Based)
Search Actions
Create
Default Package
10 Items per page | 1-6 of 6 items
1 of 1 pages
1
NAME RUNTIME WEB ACTION MEMORY TIMEOUT
Bot_data_1 Node.js 10 Not Enabled 256 MB 60 s
Bot_data_2 Node.js 10 Not Enabled 256 MB 60 s
Bot_data_3 Node.js 10 Not Enabled 256 MB 60 s
Bot_serie_1 Node.js 10 Not Enabled 256 MB 60 s
Bot_serie_2 Node.js 10 Not Enabled 256 MB 60 s
Bot_serie_3 Node.js 10 Not Enabled 256 MB 60 s

3_DIALOGUES

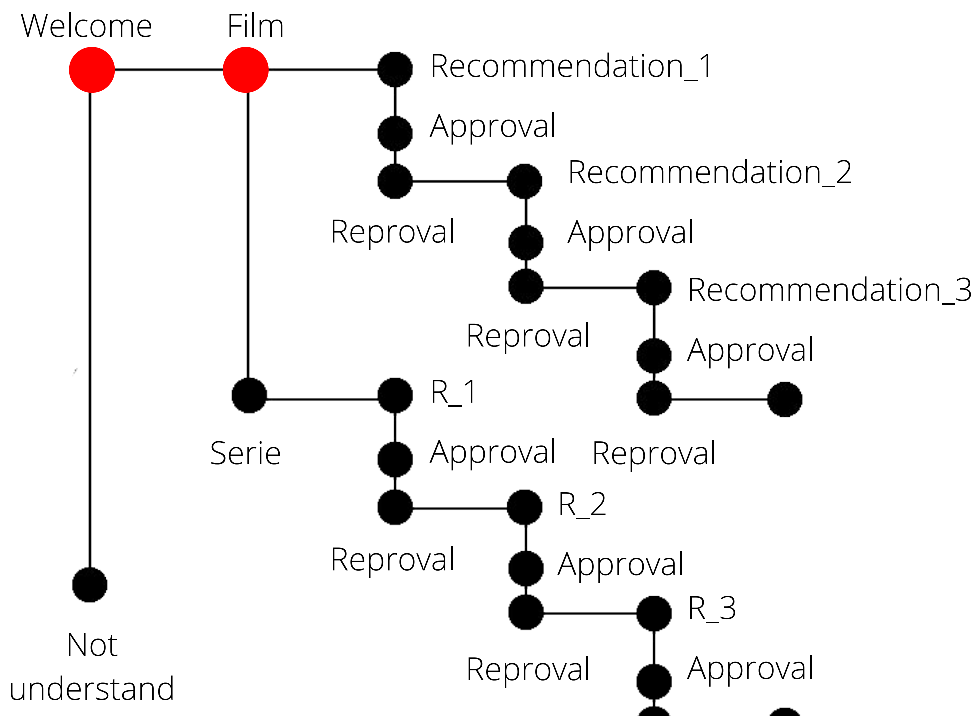
Real Film recommendation dialogue zoom in

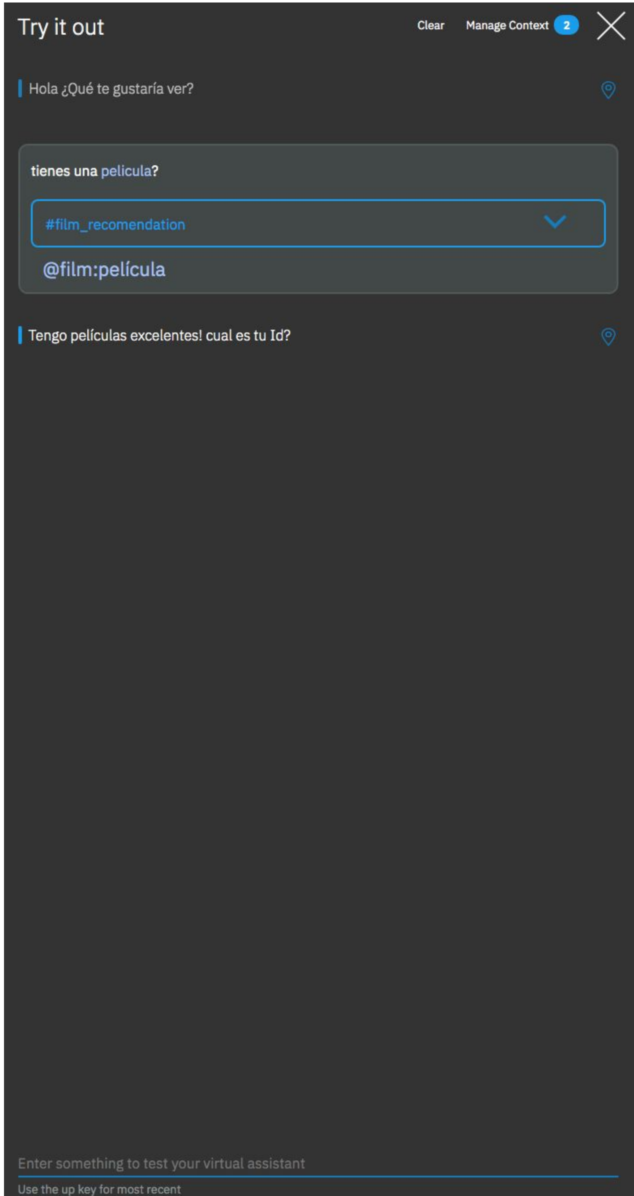
Zoom from welcome dialogue to OK/Not OK recommendation dialogue. Each dialogue is link with a context variable, entity or intent to select the next path. In this way the chatbot reacts to recommend, recommend again, apologize for not understand, and close de conversation



3_CHATBOT

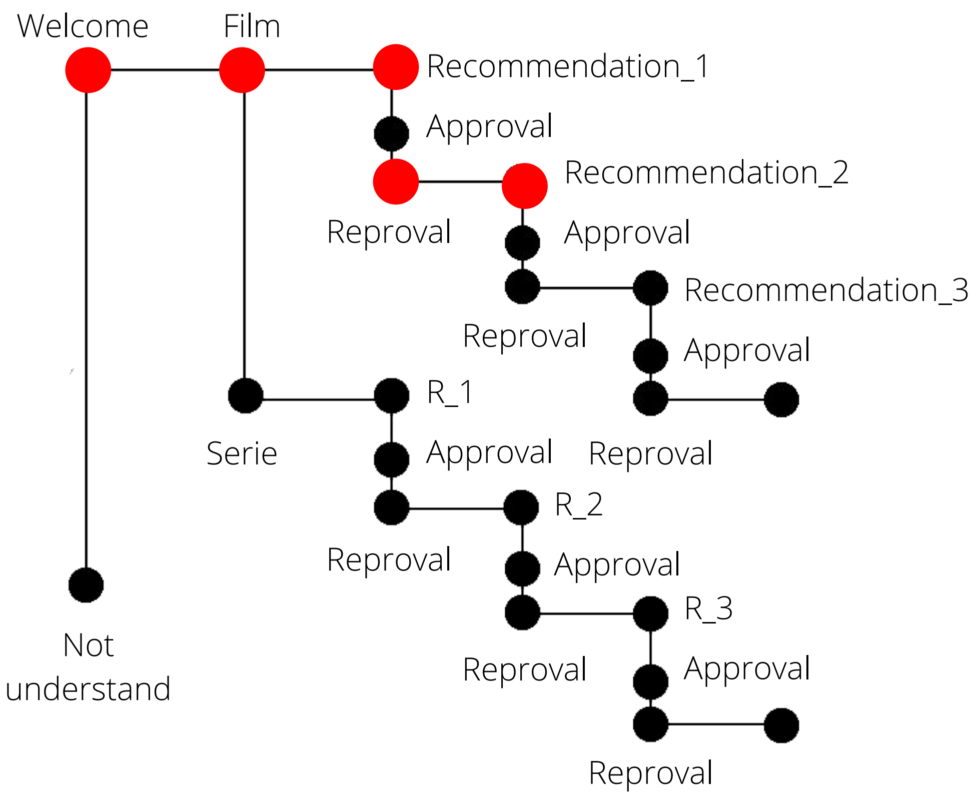
Image _ Watson assistant testing interface

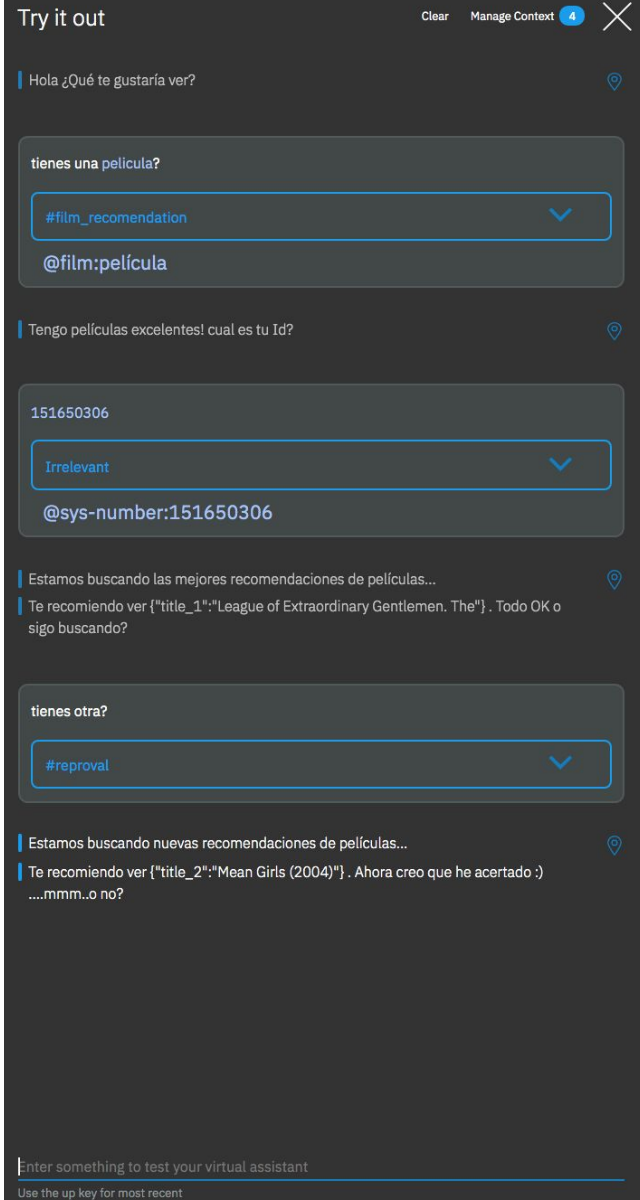




3_CHATBOT

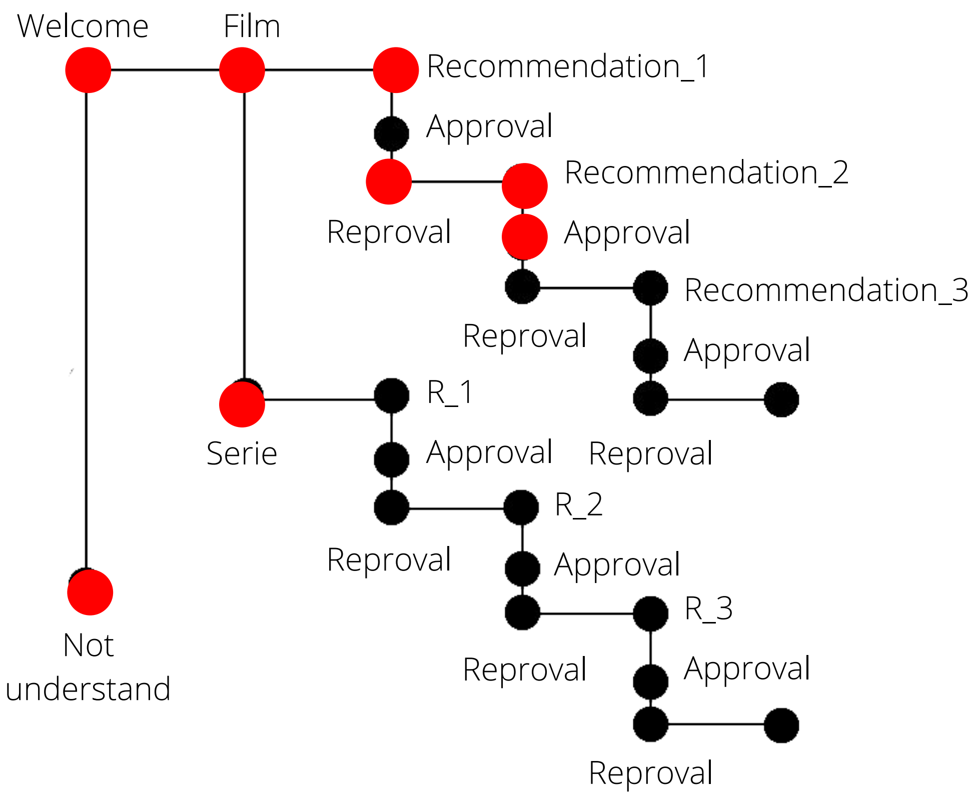
Image _ Watson assistant testing interface

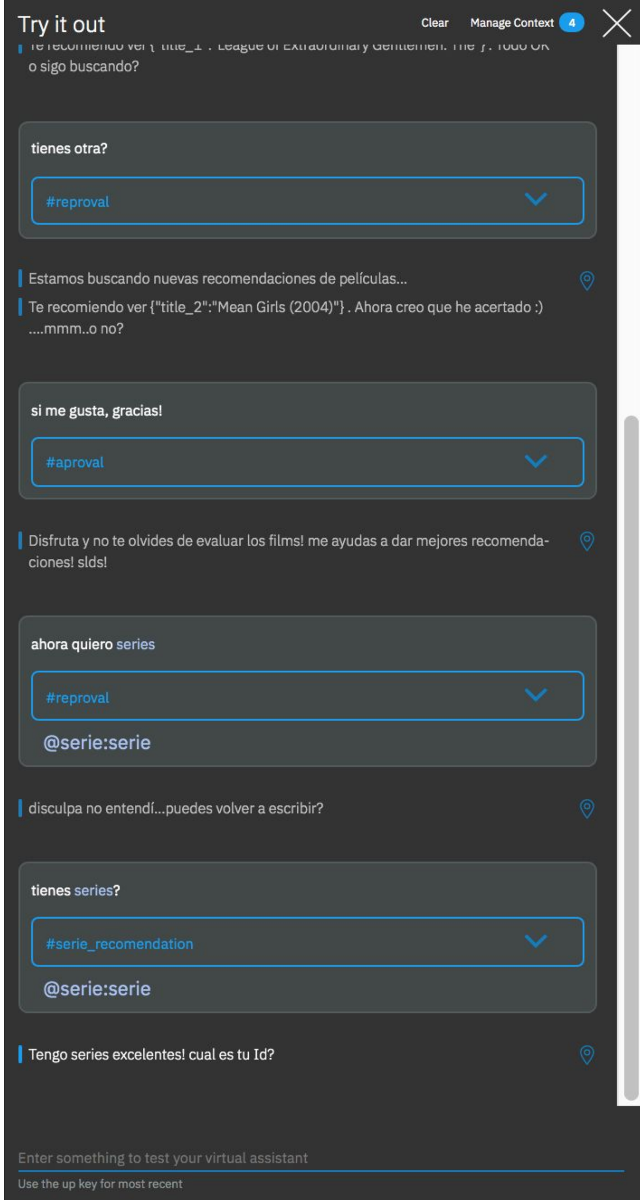




3_CHATBOT

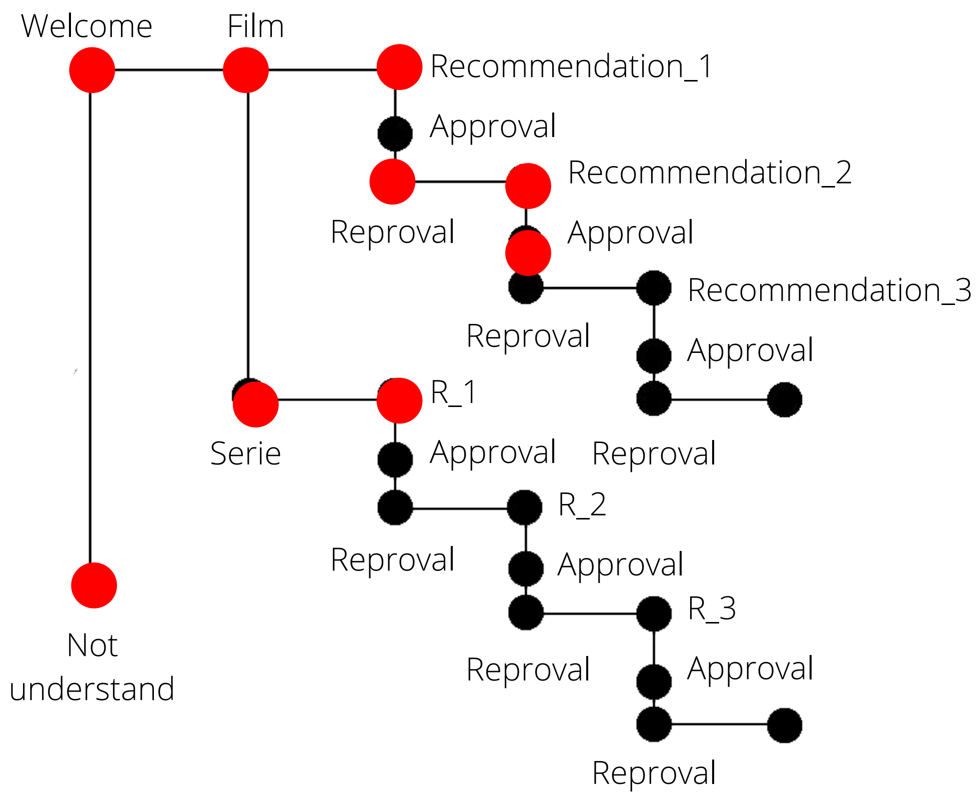
Image _ Watson assistant testing interface

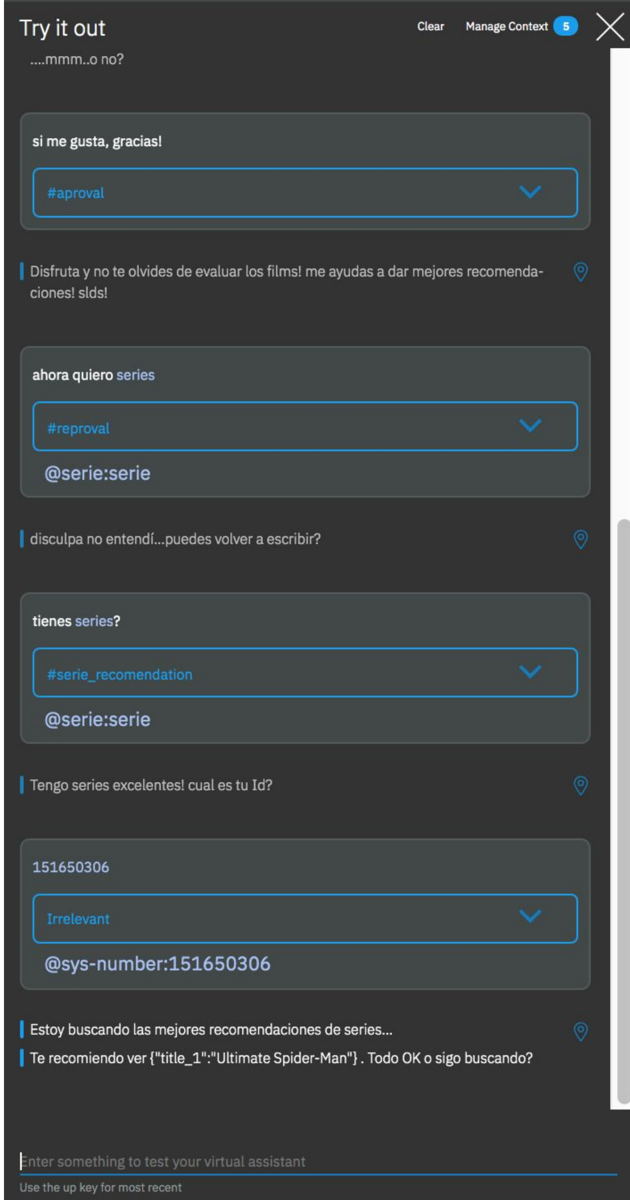




3_CHATBOT

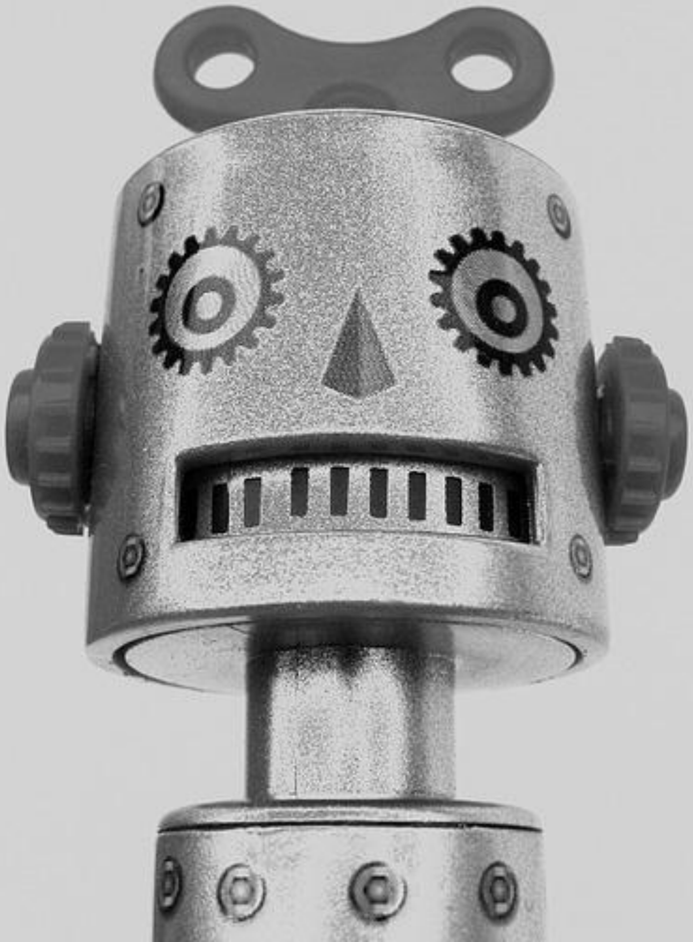
Image _ Watson assistant testing interface





UDD - Universidad del Desarrollo_MDS - Master in Data Science

PRODUCT RECOMMENDER CHATBOT



Students

Rovai, Marcelo_ Sacasa, Manuel

Professors

Reinoso, Pablo_ Seguel, Rodrigo

May, 2019