DS4A FINAL PROJECT | WRITTEN REPORT

Evaluation of the attitudes of target consumers
towards CI HERMECO S.A (OFFCORSS)

TEAM 25

1. Introduction.

Colombian kids'-clothing brand OFFCORSS by C.I. Hermeco is a company dedicated to the design, production and marketing of complete clothing and accessories proposals for babies, boys and girls. OFFCORSS efforts are directed towards the promotion and commercialization of Colombian products in foreign markets

In numbers, OFFCORSS is the top tier retailer leading colombian childrenswear market. In their growing process, OFFCORSS planned to reach a chain of 130 shops and sales of COP$700bil for 2020. The targets set for 2020 should be attainable through investment of COP$50bil. One-fifth of this sum has been already poured into technological modernisation, notes Muñoz. E-commerce managed around 1% contribution but provided 6% of sales in 2016. The brand leads its local segment with a market share of 12%, ahead of EPK and Baby Fresh, which apply different business models [1]

The textile-and-garment-sector performed well during 2019, growing an annual growth rate (CAGR) of 2.8%[2]. Nevertheless, in 2020 this sector has been one of the most affected by the measures that have been taken to control the expansion of the coronavirus. This has led to a contraction in the industrial production as well as household spending on the products of the fashion system has fallen, and imports and exports have decreased[3].

Business context: OFFCORSS understood the importance of data to know better their customers. By understanding first what the public wants, they will be able to implement the acquired knowledge in further processes of innovation , optimization of management techniques, new product development and market analysis. Therefore, through different processes of customer and brand positioning surveys, OFFCORSS has acquired data to understand what the customers think about the brand and their products. However, taking full advantage of opportunities generated by the data may require improvements in data management and visualization, event processing techniques and trend analysis, all requiring

[1] OffCorss tiene el plan para crecer casi cuatro veces. Gutiérrez
https://www.portafolio.co/negocios/empresas/planes-offcorss-crecer-cuatro-veces-494207

[2] La industria textil, un sector importante en la economía de Colombia. Rengifo
https://cecane3.com/la-industria-textil-un-sector-importante-en-la-economia-de-colombia/

[3] Textile and garment supply chains in times of COVID-19: challenges for developing countries
https://unctad.org/news/textile-and-garment-supply-chains-times-covid-19-challenges-developing-countries

specialist expertise. In this sense, OFFCORSS does not have a structured pipeline for the data acquisition and interpretation of all this information.

Business problem: OFFCORSS by C.I. Hermeco requires a pipeline that allows it to collect, process and analyse all information of natural language, so they can understand what their customers think of their products and services.

Analytical context: OFFCORSS provided a series of data from surveys continuously applied to the customers through client-service and marketing surveys (Net Promoter Scores, NPS). Additionally, we have public access to comments in social networks to access information about brand sentiment.

To take full advantage of available data, different levels of analysis were considered:

> **L1.** Analyse the perception of customers from surveys responses, and how they change in the time
>
> **L2.** Analyse the brand perception of the company form comments in the main social networks of the company, and how these are compared to the brand perception of OFFCORSS' main competitors
>
> **L3.** Analyse the relation of the company incomes and their relationship with brand perception.

2. Executive summary of results

Together with OFFCORSS, we defined that the most interesting social networks for the company were instagram and facebook.

2.1 Data Acquisition

There are several data analysis software packages for data acquisition containing functionalities for viewing, retrieving and analysing data from social media APIs. There are both proprietary tools and free and open-source solutions, which serve the purpose of collecting content from different social media platforms.

There are different approaches to acquire social media data ranging from manual searches to programmatic access to data (Batrinca and Treleaven, 2015; Lomborg and Bechmann, 2014). The purchase of data from an authorized data vendor has several advantages such as little to no manual work and programming effort, and the availability to access time series of historical data. Costs may limit the practical usefulness of this approach in conservation science. Web scraping, or web crawling, is an approach for downloading and extracting data from web pages using an automated script. In comparison to APIs, web crawlers can only access the public web, while APIs may provide access to content that requires authentication (Lomborg and Bechmann, 2014).

For querying the data in this project from instagram platform Instaloader was used. Instaloader is a tool to download pictures (or videos) along with their captions and other metadata from Instagram. This tool allowed extracting information OffCorss, Polito and EPK profiles; as the hashtags, user stories, feeds and saved media, comments, geotags and captions of each post, and then we processed it in a python script using Pandas. To manipulate this spreadsheet, we used a python library called nltk y sckitlearn (view notebook "preprocess_instagram_comments.ipynb") for extraction of lexical roots, stop words to accept important words in our context (mostly for bigrams and trigrams), definition of features for classification, spell correction and use lower letters to the comment column.

Also, two independent robots were implemented for accessing each social network (instagram and facebook). These robots were implemented under javaScript and integrated by python scripts. The data was converted to json files and then, the target information was converted to csv files.

The elements implemented for the scrapping were:

Robot Framework: https://robotframework.org/
Facebook-Scraper: https://pypi.org/project/facebook-scraper/
Instagram Scraper: https://github.com/arc298/instagram-scraper

2.2 Social networks scraping:

Depth: It was possible to obtain the posts and comments with all the related data: Users, responses and number of likes (It was extracted from January 1, 2019). We used the python library *instaloader* [4] to access instagram post information. This library allows to put any instagram user and download descriptions in .txt, but it also allows to bring the instagram metadata capturing all the calls that contain all the data of the post

To improve processing time, we implemented the script *instagrammer*[5] developed in Java to update the scrapping. Using this package it is possible to extract the latest posts and the data of the comments.

Inventory scrapping
The specified fields that returned for each individual post were: message, link, created_time, type, name, ID, number of likes, number of comments, shares, summary for likes, summary for comments, etc.

Example of web scraping files: Post: ID, text, # likes, #comments, video, imagen, date, post link, hashtags, brand (Offcorss, EPK, Polito)

[4] https://instaloader.github.io/
[5] https://developer.aliyun.com/mirror/npm/package/instagrammer

Example of tables: Comments /Comment responses: ID, related post ID , text, #likes, date

	Unnamed: 0	rootPost_id	parentPost_id	reponsePost_id	brand_username	text	time	likes	username
0	0	1947257691818809835	1.947258e+18	1.794235e+16	offcorss	Así es un día para disfrutar sana mente y con...	2019-01-01	1	rosaliafaneite
1	1	1947952244502703605	1.947952e+18	1.801764e+16	offcorss	Buenos días que precio tienen?	2019-01-02	1	aleja.vasquez911
2	2	1948073194162672935	1.948073e+18	1.791861e+16	offcorss	🔥	2019-01-02	1	pactrii188
3	3	1948284380053077836	1.948284e+18	1.798609e+16	offcorss	Hermosa mi sheshe	2019-01-03	1	dalekeyboutique
4	4	1948284380053077836	1.948284e+18	1.790035e+16	offcorss	Cuanto cuesta el vestido de manga larga talla 14	2019-01-03	3	delgadoa953
...	...	...	...	...	...	...	...	...	...
33483	33483	2404696945357653802	1.791184e+16	1.793289e+16	politokids	@tina.benjumea 🌟 Todos los tapabocas tienen un...	2020-09-23	0	politokids
33484	33484	2404696945357653802	1.787722e+16	1.796018e+16	politokids	@anil_rizo 💛 todos los tapabocas tienen un val...	2020-09-23	0	politokids
33485	33485	2404696945357653802	1.790429e+16	1.786213e+16	politokids	💛💛 @paopao.martinezfigueroa todos los tapaboc...	2020-09-23	0	politokids
33486	33486	2404696945357653802	1.785560e+16	1.788603e+16	politokids	@contreraszarateyamileth 🌻🌻 todos los tapabo...	2020-09-23	0	politokids
33487	33487	2404696945357653802	1.785799e+16	1.793346e+16	politokids	@laer1217 hola!!! Todos los tapabocas tienen ...	2020-09-23	1	nataliazabala

	rootPost_id	parentPost_id	reponsePost_id	brand_username	text	time	likes	username	score	class
0	1947257691818809835	1.947258e+18	1.794235e+16	offcorss	así es un día para disfrutar sana mente y con...	2019-01-01	1	rosaliafaneite	0.614808	neutral
1	1947952244502703605	1.947952e+18	1.801764e+16	offcorss	buenos días que precio tienen?	2019-01-02	1	aleja.vasquez911	0.619522	neutral
2	1948073194162672935	1.948073e+18	1.791861e+16	offcorss	🐥	2019-01-02	1	pactrii188	0.448265	neutral
3	1948284380053077836	1.948284e+18	1.798609e+16	offcorss	hermosa mi sheshe	2019-01-03	1	dalekeyboutique	0.992207	good
4	1948284380053077836	1.948284e+18	1.790035e+16	offcorss	cuanto cuesta el vestido de manga larga talla 14	2019-01-03	3	delgadoa953	0.262286	bad
...	...	...	...	...	...	...	...	...	...	...
33483	2404696945357653802	1.791184e+16	1.793289e+16	politokids	@tina.benjumea 🌟 todos los tapabocas tienen un...	2020-09-23	0	politokids	0.560458	neutral
33484	2404696945357653802	1.787722e+16	1.796018e+16	politokids	@anil_rizo 💛 todos los tapabocas tienen un val...	2020-09-23	0	politokids	0.486015	neutral
33485	2404696945357653802	1.790429e+16	1.786213e+16	politokids	💛💛 @paopao.martinezfigueroa todos los tapaboc...	2020-09-23	0	politokids	0.486015	neutral
33486	2404696945357653802	1.785560e+16	1.788603e+16	politokids	@contreraszarateyamileth 🌞🌞 todos los tapabo...	2020-09-23	0	politokids	0.486015	neutral
33487	2404696945357653802	1.785799e+16	1.793346e+16	politokids	@laer1217 hola!!! todos los tapabocas tienen ...	2020-09-23	1	nataliazabala	0.477935	neutral

2.3 OFFCORSS surveys processing:

We have three initial datasets available to perform the analysis:

NPS_Responses This dataset contains the responses of a Net Promoter Score (NPS) survey that the company sent to their clients.

1. *Survey Date* (Date): The date when the company sent the survey to the client. Currently, there is only the results for a survey sent on 10/08/2020.
2. *Name* (Str): the name of the user assigned to the client responded to. Some of them are the client's name, and others the name of the process that they have done (like returning a product).
3. *User ID* (Int): an unique identificator of the client that responded to the survey.
4. *Email* (Str): the email address that has sent the response.
5. *Rating* (Int): an identifier in the range 0 to 10 that the likelihood to recommend the company.
6. *Classification* (Str): a categorical variable that clasificate the likelihood to recommend the company into three categories: 7. Promoter (9-10 rating), passive (8-7 rating) and detractor (0-6 rating).
7. *Comment* (Str): the comment that the client made with the survey. Most of the registers don't have a comment associated and some of them have useless values like dots or blank spaces.
8. *Response Date* (DateTime): the date and hour when the client sent the response of the survey.

Though this data contains a Classification column based on Rating, it was found that this might be miss-classified based on customer comments; negative comments classified as promoter, or the opposite. Additionally, there are significant null values in the Comments column. A new classification system based on comments from customers might be more beneficial for the sentiment analysis integrated with data from social networks.

Satisfaction_Ratings This dataset contains the level of satisfaction of the client when they have been attended to in any request.

1. *Requester* (Str): the name of the user who responded to the satisfaction survey. Some of them are the client's name, and others the name of the process that they have done (like returning a product).
2. *User ID* (Int): an unique identificator of the client that responded to the satisfaction survey.
3. *Email* (Str): the email address that has sent the response.
4. *Ticket ID* (Int): an unique identificator of the ticket generated.
5. *Brand* (Str): the name of the brand associated with the support process. The only brand available in the dataset is "OFFCORSS".
6. *Group* (Str): a categorical variable that represents the type of the support that the client needs. There are four groups in the dataset: "Call Center", "Soporte OFFCORSS", "Tienda Virtual" and "Venta Directa".
7. *Assignee* (Str): a categorical variable that represents the area of the company that was in charge of response to the support. There are 13 different areas in the dataset.
8. *Satisfaction* (Str): a categorical variable that represents the classification that the client made based on the satisfaction of the solution for the request made. There are only two categories, "good" and "bad".
9. *Comment* (Str): the comment that the client made with the survey. Some of the registers don't have a comment associated and some of them have useless values like dots or blank spaces.
10. *Survey Date* (DateTime): the date an hour when the ticket was generated.

We'll first use satisfaction and NPL ranking for exploring the satisfaction of OFFCORSS services. Therefore, for the satisfaction dataset, two groups were previously specified according to the valence of the comments:

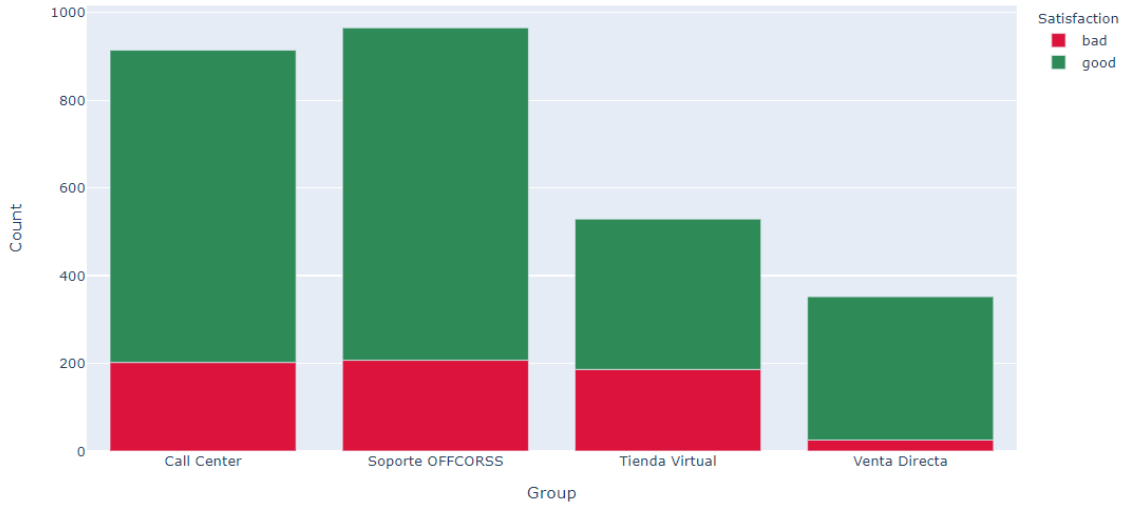
data_satisfaction
- bad
- good

And for the NPS there are three identified groups:

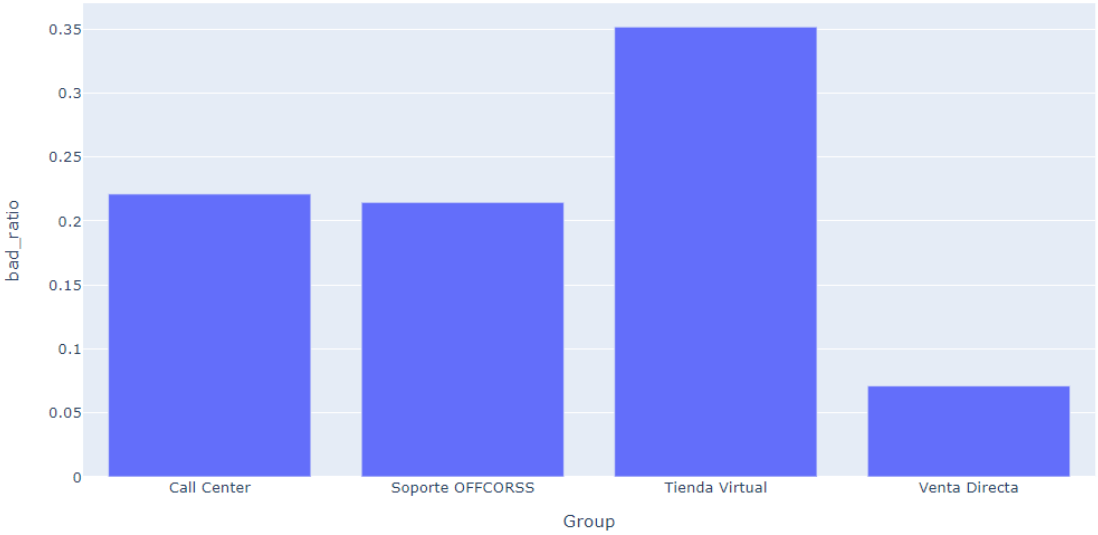
data_responses:
- promoter
- passive
- detractor

For both datasets, we found that there were more good/promoter comments than bad/detractor comments by a ratio of 3:1. This was a pivoting point to understand future data.

Also, we analysed satisfaction levels by organizacional groups:

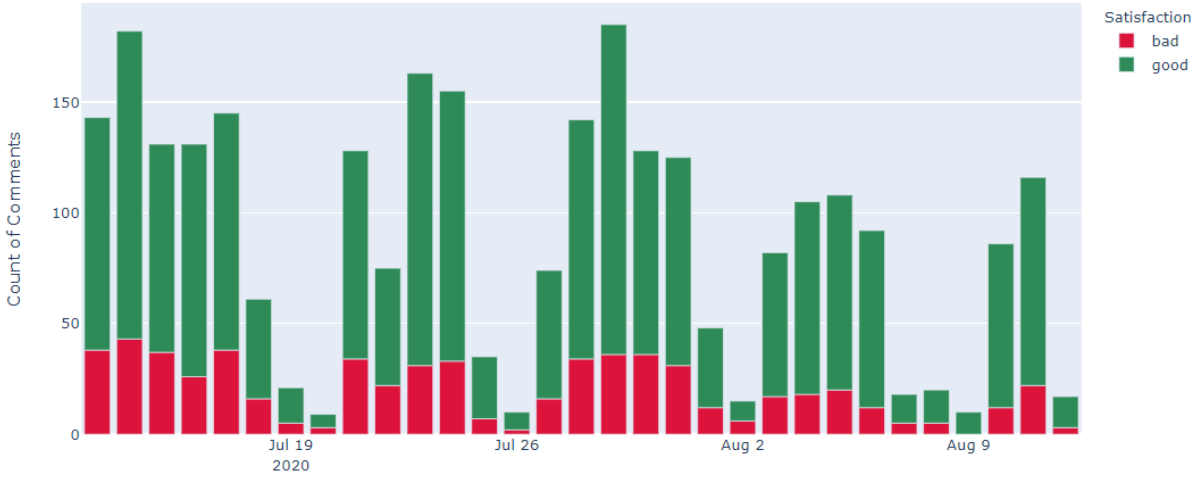


And the ratio of bad comments by each groups:



Suggesting that most of the bad comments are caused by the online-store (Tienda Virtual).

And we analysed the temporality of the data finding a weekly trend determined by the comments



2.4 Text data processing:

For text data including web-scraping and survey information, we used a unified process to clean, preprocess and organize textual data before analysis.

The process is characterized by applying spell correction, stopwords identification and word standardization in lowercases. Furthermore, the information from each data source was organized in tables for further handling and analysis.

During the data processing of OFFCORSS surveys, we identified unigrams, bigrams and trigrams related to each sentiment (good, bad) and they were retained to be used in the development of the sentiment classifier:

Here some examples for top unigrams, bigrams and trigrams rated with a good sentiment and further used for classification:

```
Top unigrams_good:
['excelente', 'bueno', 'buena', 'buen', 'bien']
Top bigrams_good:
['excelente servicio', 'buen servicio', 'buena atención', 'ex
celente atención', 'mil gracias']
Top trigrams_good:
['servicio muchas gracias', 'excelente muchas gracias', 'exce
lente servicio gracias', 'buen servicio gracias', 'excelente
servicio cliente']
```

Here some examples for top unigrams, bigrams and trigrams rated with a bad sentiment and further used for classification:

```
Top unigrams_bad:
['mal', 'dinero', 'malo', 'dirección', 'pésimo']
Top bigrams_bad:
['pésimo servicio', 'mal servicio', 'dado respuesta', 'falta
respeto', 'hice pedido']
Top trigrams_bad:
['parece falta respeto', 'nunca recibí respuesta', 'pésimo se
rvicio cliente', 'nunca dieron respuesta', 'mal servicio clie
nte']
```

Sentiment classification:

The sentiment analysis implemented was based on lexical dictionaries. . Based on an original work *Sentec* [6]using ecuadorian lexicon. Specifically, we used the information in the sentiment dataset from customer-support surveys provided by OFFCORSS. The initial corpus was then updated with specific bag/good expressions (top unigrams, bigrams and trigrams)

[6] by Julio Vasconez Yulan in https://github.com/jvas28/sentec

determined by OFFCORSS surveys. Likewise, lexicon was next complemented using the englist-to-spanish translation of the SentiWordNet v3.0.0 dataset[7] updated the June 1st, 2010.

- For the SentiWordNet dataset, the sentiment score is determined by computing PosScore - NegScore. (This number is determined between -1 and 1, where -1 is categorized as mostly bad and 1 mostly good)

- For the custom lexicon, the sentiment score has been given in a 1-5 scale, where 1 is categorized as mostly bad and 5 mostly good. Then, the score is standardized to be in the interval [-1, 1] to be consistent with the SentiWordNet dataset

- A working lexicon is created from these two datasets. Only one instance is kept from any repeated word maintaining the last instance in the dataset.

The comments are tokenized, and stop-words removed from the lists. For each comment, the stemmed unigrams, bigrams and trigrams are then scored based on the joined lexicon. For words not located in the lexicon, a score of 0 is assigned. The word lexicon also included values for emoticons and jargon expressions

Examples of ratings in the SentiWordNet dataset:

word	score
parturiente	0.250
comparativo	-0.250
sorbefaciente	0.375
asimilado	-0.750
quimiorresistente	-0.250

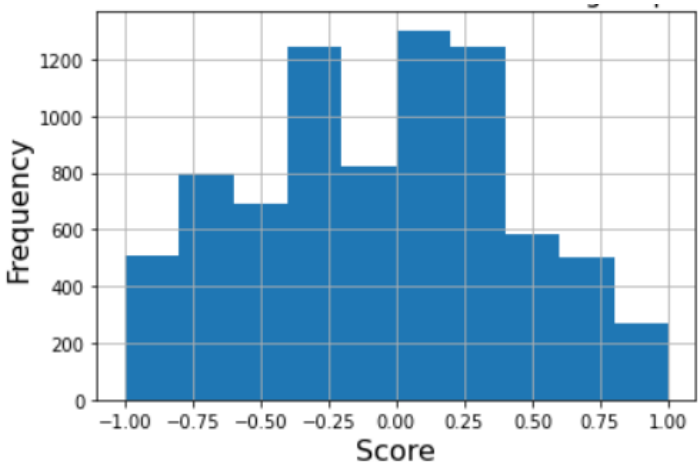
Examples of ratings for emoticons:

word	score
🤩	1.0
😭	-0.5
😘	1.0
🔥	0.5
👏	0.5

In total, the lexicon dictionary includes more that 9000 words, expressions and emoticons used to rank the text of each comment.

[7] https://github.com/aesuli/SentiWordNet

Distribution of scores in the lexical dictionary:



For the training of the classifier, we used the sentiment dataset provided by OFFCORSS. Nevertheless, this had an imbalance of classes detrimental for training. Briefly, the Total number of good comments was 6980 and the total number of bad comments was 3397. Therefore, we can create a random subsampling of the good comments to finally have a even number of instances for classification (good = 3397, bad = 3397, Total =6794)

Before any further step, we randomly divided the selected data in training (66.66%) and testing (33.33%) sets.

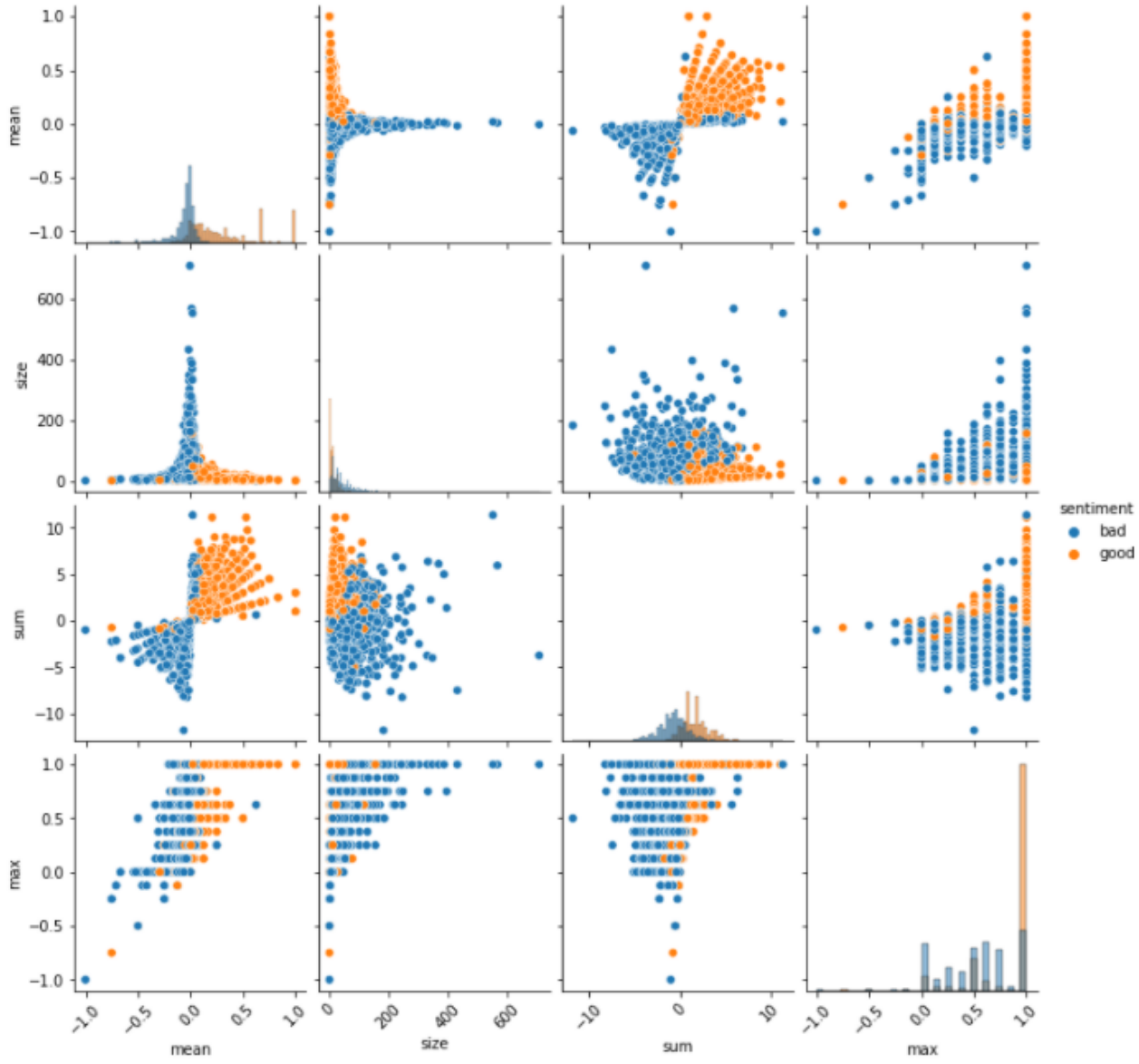
Then, we trained four classifiers to predict the probability to be in a specific class (bad/good). The average ROC curves after 10 folds cross validation were finally compared.

The features computed for classification were computed: the mean score of tokens, the total sum of tokens, the total length of the comment, the maximum score for the tokens in the comment. Nevertheless, after computing the correlation of features we discarded the minimum score because of its high correlation with the mean score:

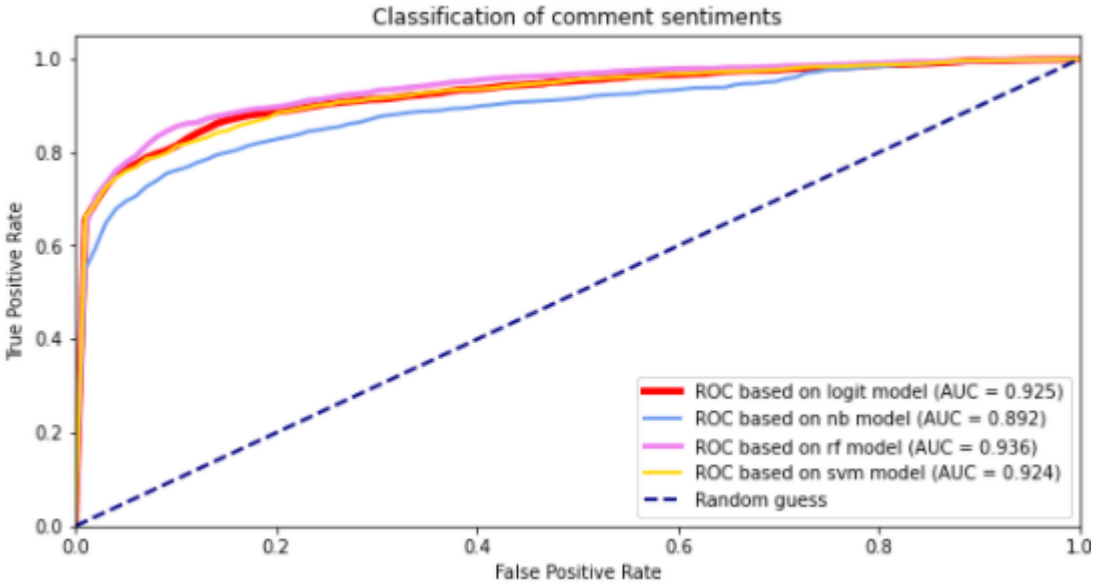
Correlation of computed features in the training set:

	mean	sum	size	min	max
mean	1.000000	0.503985	-0.279305	0.826399	0.571648
sum	0.503985	1.000000	-0.167437	0.498699	0.554877
size	-0.279305	-0.167437	1.000000	-0.444535	0.127881
min	0.826399	0.498699	-0.444535	1.000000	0.315735
max	0.571648	0.554877	0.127881	0.315735	1.000000

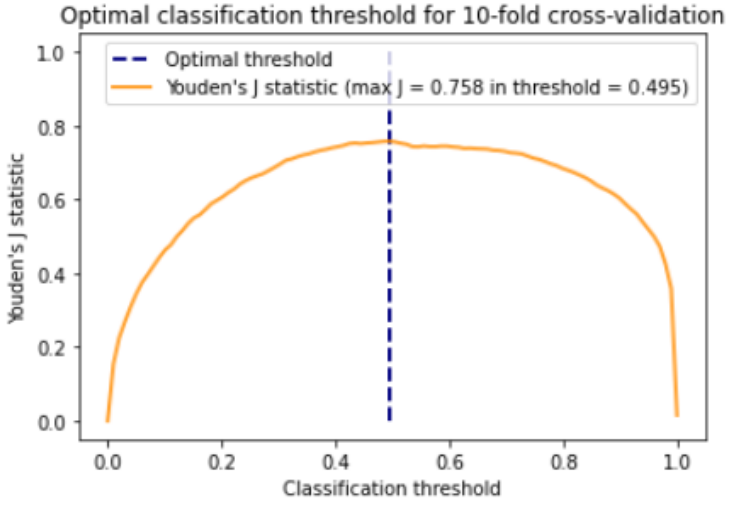
The pair plots of the selected features indicate a good chance for the model classification:



We implemented thus four different classifiers in the training set: a logit model, a naive-bayes model, a random Forest model and a SVM model. We found that based on the average AUROC of the cross validation all the models had similar performance in the training set. Nevertheless, the AUC model had the highest AUC value and therefore, this was selected for further analysis:



Then, we evaluated the Youden's J statistic to identify the best threshold for binary classification and was identified as Threshold = 0.495:



Then, after applying the selected model in the test set we got a good performance of the classification with an accuracy of 87%:

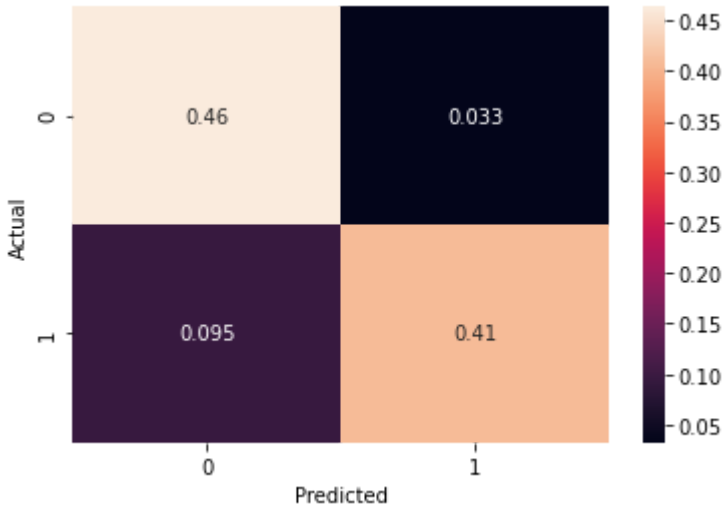
Summary of the classifier performance on the test set:

	precision	recall	f1-score	support
0	0.83	0.93	0.88	1112
1	0.93	0.81	0.86	1131
accuracy			0.87	2243
macro avg	0.88	0.87	0.87	2243
weighted avg	0.88	0.87	0.87	2243

MCC score: 0.750

Where *bad* comments were coded as 0 and *good* comments coded as 1.

Likewise, ,the confusion matrix indicates a low rate of false positives (FP = 3.3%) and false negatives (FN = 9.5%)



We evaluated examples of false positives:

FP example 1:

"buenas tardes mi nombre es e manifiesto no haber recibido el pedido de enviado por offcorss el dia 17 de julio del prensa año y que según reporte lo entregaron el 20 de julio a otra persona diferente a la de toda mi familia es decir que lo entregaron en una direccion distinta . cabe recordar desde el 19 de junio hasta la fecha he realizado 5 pedidos como es posible que este lo entregaron a otra dirección exijo una respuesta favor esclarecer la entrega del pedido"

FP example 2:

"la respuesta no fue rápida"

FP example 3:

"no he podido aplicar el vale"

And examples of false negatives:

FN example 1:

"recibí atención por el chat , estoy a la espera que mi pedido # 1034112375883-01 por fin llegue . lo único malo del servicio es el tiempo de espera para ser atendido , por lo demás bien ."

FN example 2:

"el día de ayer estuve de viaje y quise que mi pedido fuese entregado otro día pero me fue negado ."

FN example 3:

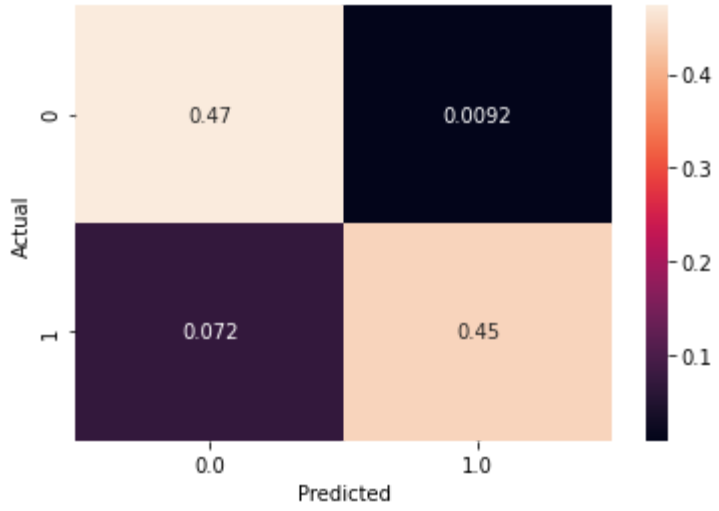
"se demora un tiempo , pero entiendo que fue por temporada de diciembre . muchas gracias"

And realized that some comments have both good components and bad components, therefore, we determined a third class established as neutral based on the same score predicted by the logit classifier. In that way, bad comments were determined for logit scores < 0.33, neutral comments for scores in the interval 0.33 < score < 0.66, and good comments in the range where score > 0.66. Hence, evaluating only instances not in the neutral range (about 393 comments in the test set), we found an increase of the performance, with an accuracy of 92% as follows:

	precision	recall	f1-score	support
0	0.87	0.98	0.92	892
1	0.98	0.86	0.92	958
accuracy			0.92	1850
macro avg	0.92	0.92	0.92	1850
weighted avg	0.93	0.92	0.92	1850

MCC score: 0.844

And confusion matrix:

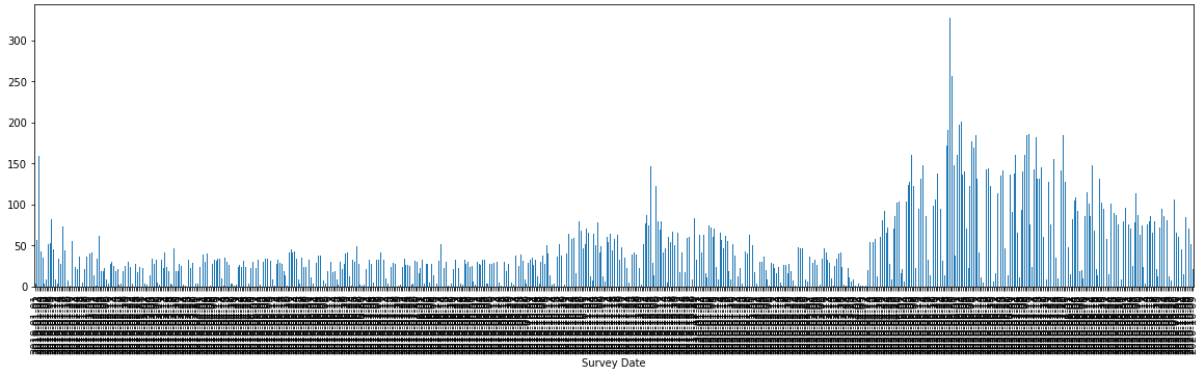


Based on this new strategy, we applied the scoring of comments to all text datasets, keeping information of the score and the assigned score.

2.5 Sentiment vs incomes analysis

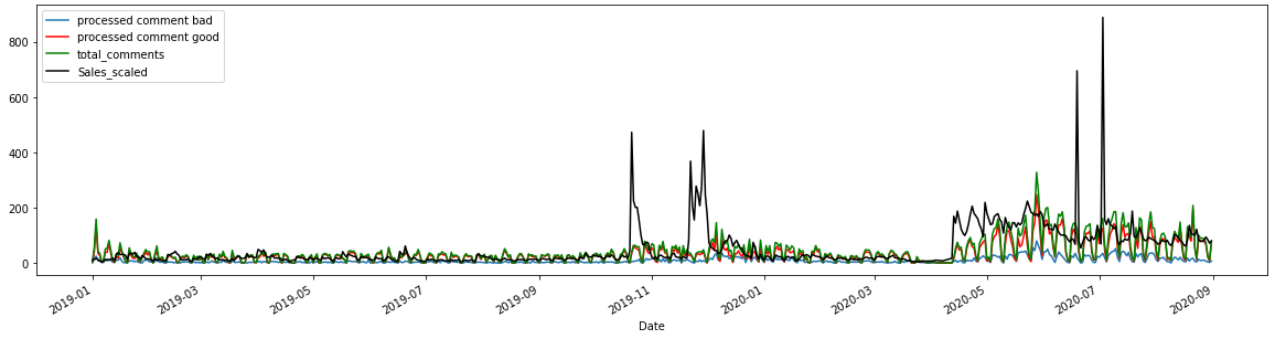
As a additional plus, we also analysed of the sales in the virtual store were related to variability of comment sentiments across the time (Since Jan 1st, 2019)

We found an increasing of comments after the COVID pandemic



2019-01 2020-10

We evaluated the trends in the sentiments of the comments and the sales:

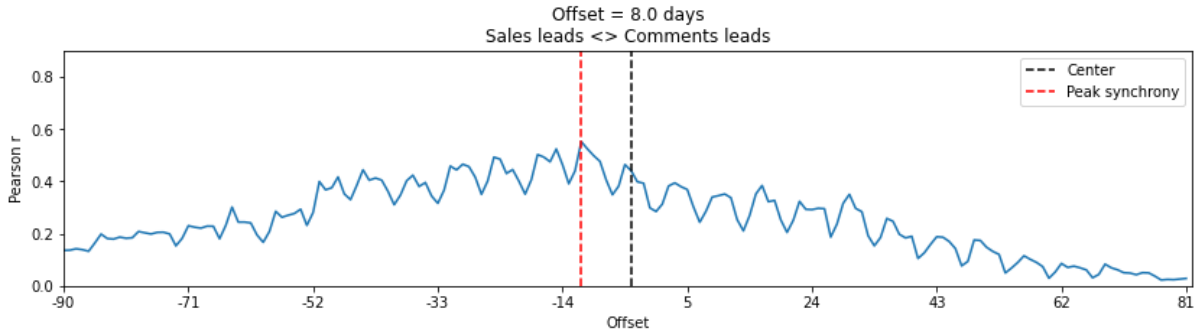


We computed correlations of this data:

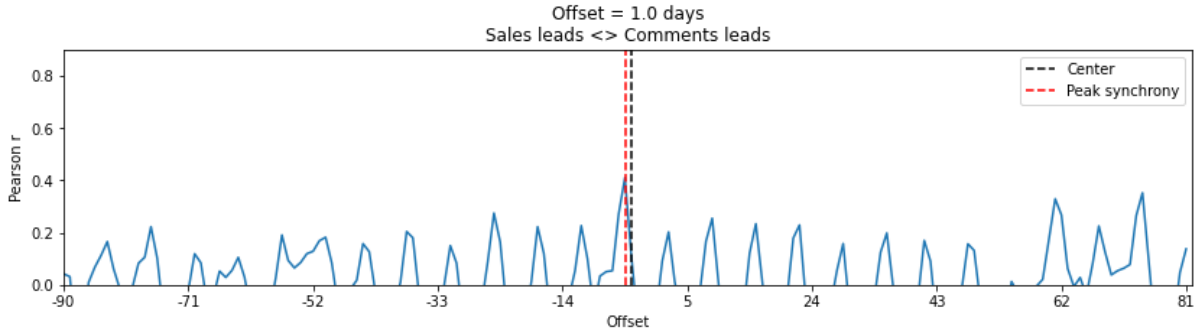
	processed comment bad	processed comment good	sum	count	Sales_scaled	total_comments
processed comment bad	1.000000	0.854514	0.379337	0.444218	0.444218	0.913250
processed comment good	0.854514	1.000000	0.388146	0.453659	0.453659	0.992000
sum	0.379337	0.388146	1.000000	0.978021	0.978021	0.396624
count	0.444218	0.453659	0.978021	1.000000	1.000000	0.463776
Sales_scaled	0.444218	0.453659	0.978021	1.000000	1.000000	0.463776
total_comments	0.913250	0.992000	0.396624	0.463776	0.463776	1.000000

This information tells us that there is not a very strong correlation between the sales and the comments. However, the comments could be delayed some days after the sales.

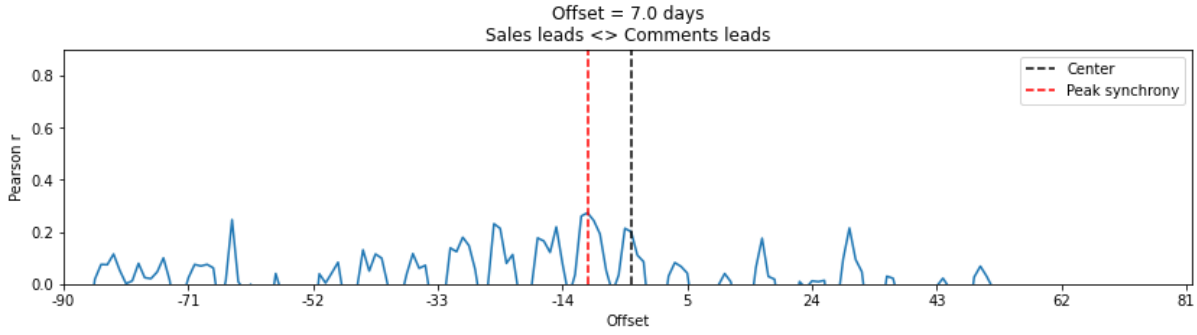
Therefore, we checked the **Time Lagged Cross Correlation** to evaluate the temporal dynamics of volume of sales and volume of comments. Hence, without further knowledge, a Time Lagged Cross Correlation suggest that the number of comments has peak lag of ~8 days following the volume of sales in the virtual store:



Likewise, we compared the same dynamics before Covd (2019-04 to 2019-10):



And the dynamics during covid (2020-04 to 2020-10):



Indicating a possible delay in the correlation of sales and comments of about 6 more days

3. Technical Details

3.1 Interactive Front-end

The solution proposed by the members of Team 25, consist in the deployment of a series of dashboards, designed to capitalize the important amount of publicly available data generated daily by the OFFCORSS online community, also extending the capabilities of the tools developed, to also capture the feedback of the followers and the interactions of the

company's main competitors, EPK and POLITO. The non structured data was processed by the team, in order to extract key insights to facilitate the understanding of the public image of OFFCORSS, EPK and POLITO, their level of activity on the main social media platforms and in general, to perform a sentiment analysis behind all the available interactions in the public platforms, prioritizing the interactions available in Facebook and Instagram where the great majority of users, leave their good or bad feedback to the public.

Considering the sources of data and repositories built for the analysis, the team identified a set of KPIs and proceeded to organize and deploy these metrics on a group of interactive, user friendly and automated dashboards published through the internet using Power BI.

This is the initial design of the online solution:

- **Password Protected Interface**:

URL:
https://app.powerbi.com/view?r=eyJrIjoiNjA3ZjA5ZmEtNmRmYi00MGVlLWExZWMtODc0ZWY1MDQyZDAzIiwidCI6IjMzZDI3NTU4LTIwM2ItNDEzNC05ZWEwLTMyNmExMDI5YzAwZCJ9&pageName=ReportSection9a8d65b06bb1e94819cd
(Access code in slack channel: lease keep private because of sensible information)

The team ideated this first interface, as a way to restrict/control the access to the database and its processed dashboards.

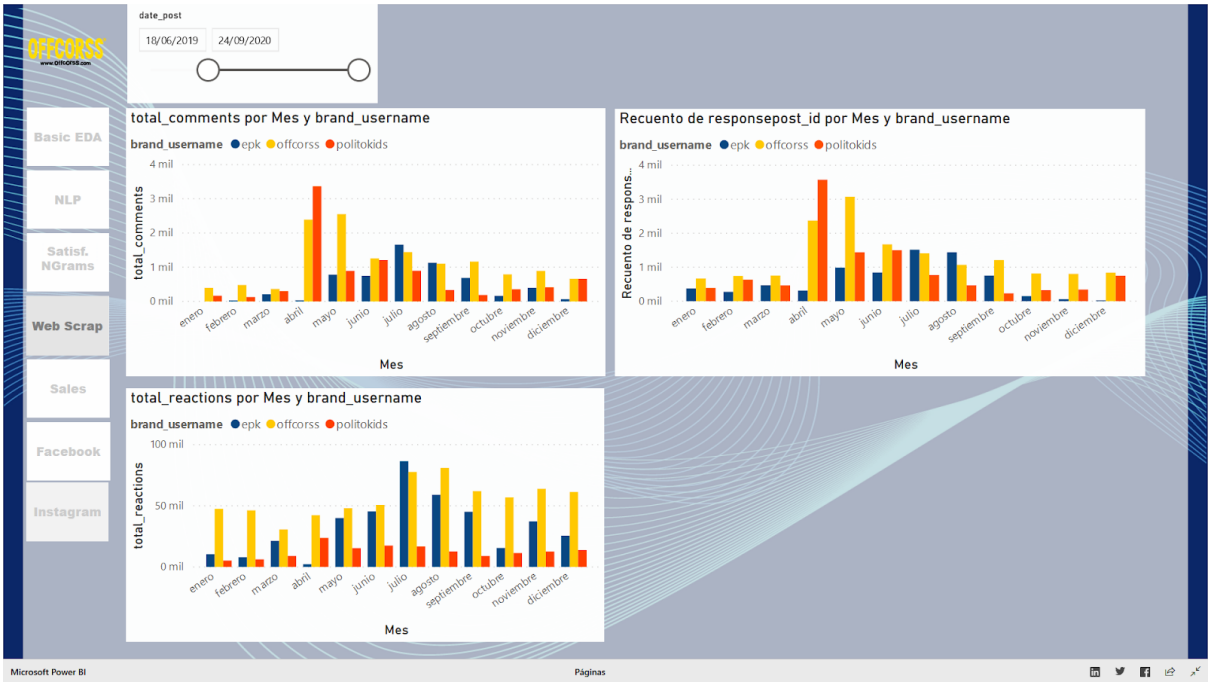
Once the analyst enters the provided password, the tool will allow the access to the calculations and metrics deployed in different interactive dashboards, designed to streamline the analysis of the information compiled in the tool, the user will first encounter a "General" dashboard, summarizing some of the most important metrics that helps putting together the "big picture" of the level of activity and appreciation of the company, according to the interactions in social media.

It's important to capitalize the fact that each of the dashboards, metrics and visuals selected in the tool, were carefully selected, looking to facilitate the interaction, zoom, filtering and/or grouping of the data.

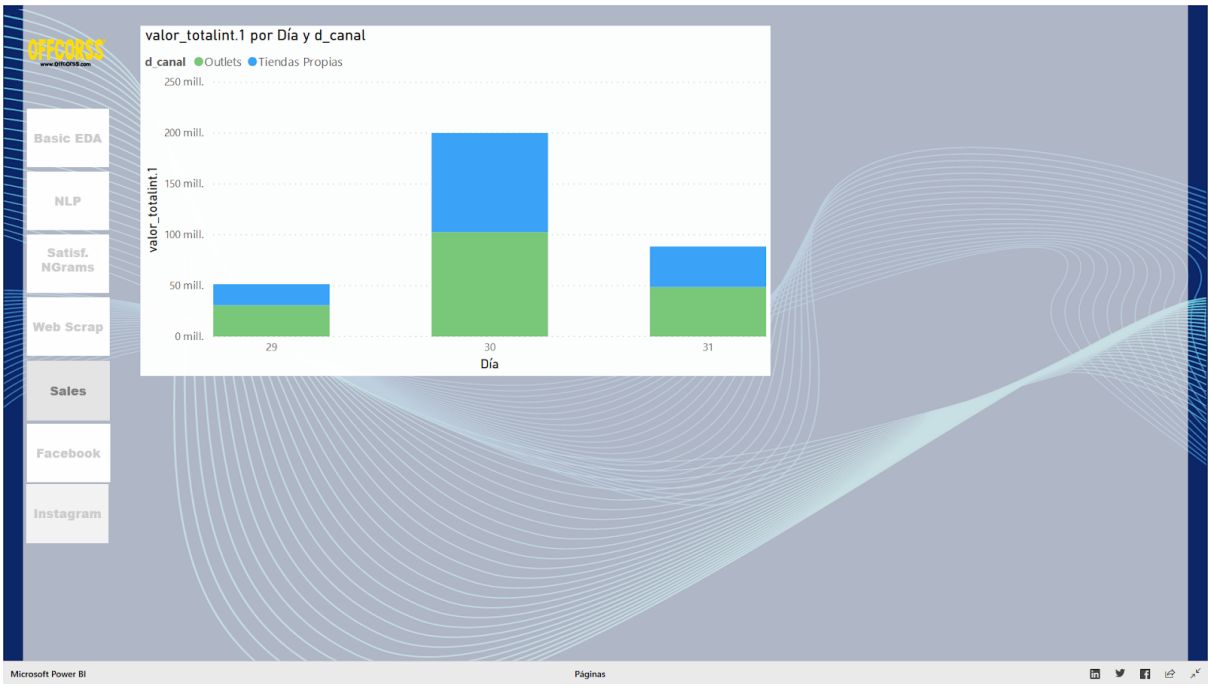
Currently, front-end solution powered by PowerBi has been deployed considering only the data from the interactions of social networks (Facebook and Instagram), however, the solution is nurtured by a set of processes that allows the integrated database to continually and systematically capture and integrate new interactions that will then be available for the analyst through the curtailed dashboards.

Once the “Summary”dashboard has been analyzed, the user can direct to the “NLP” and “Ngrams” dashboard; the intention of this screen is to display major results of the NLP processes contained in the tool.

After this, the user can view a brief summary of the engagement and level of activity for the online communities of OFFCORSS and its competitors in the “Web Scrap” section, it is important to mention that the team deliberately chose not only to look at the most recent available data, but also to capture a significant amount of information of the past two years; the intention behind this, was to get a general idea of the panorama if the industry in Colombia in times of “pre” and “during” COVID-19 in terms of social media interactions.

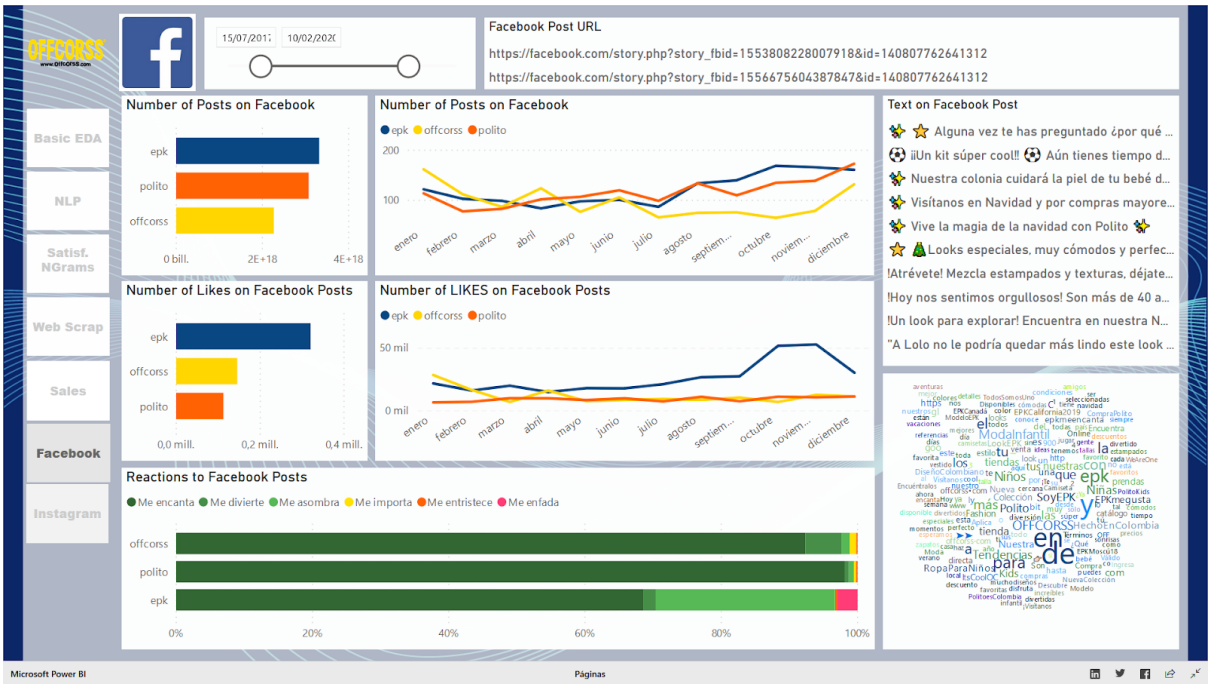


Structured, curated and highly confidential sales metrics are captured in the “Sales” dashboard. The intention of this section is to provide a tool to extend the findings of the social media platforms and see their repercussions in the sales department.



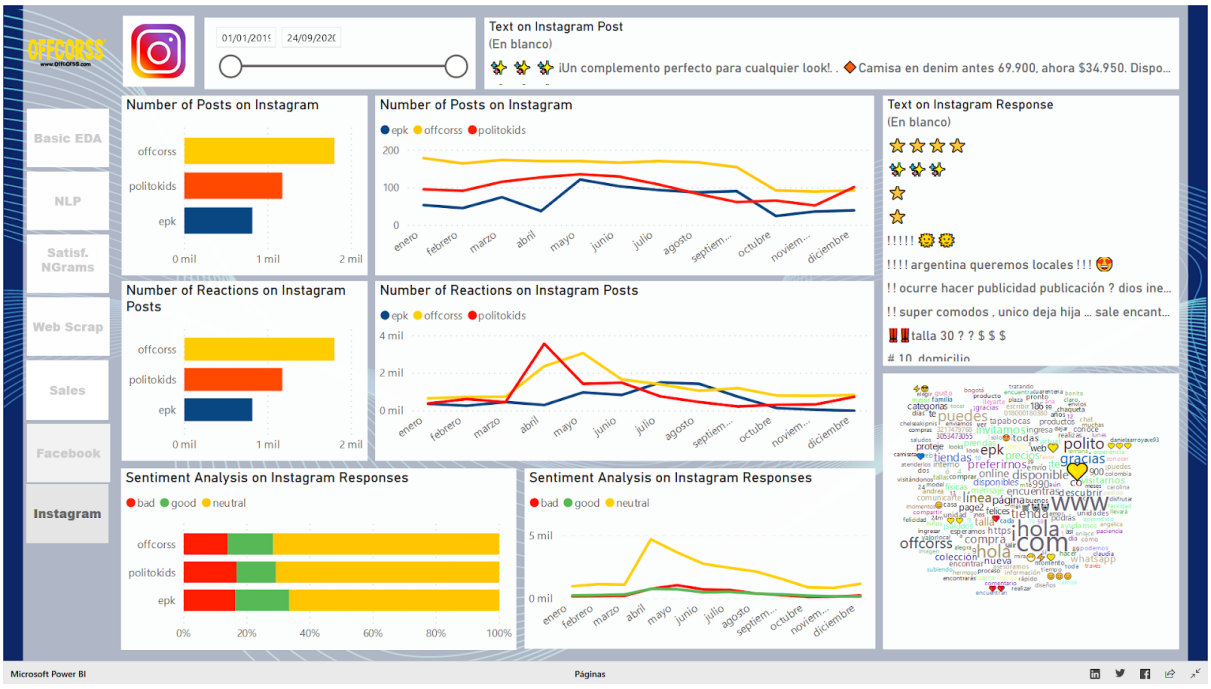
The Facebook page includes a filter to select the date range to visualize the information from this social network. Based on this selection, the visualization is updated showing the following information:

- Number of posts: Total count and count along time for each company.
- Number of likes: Total count and count along time for each company.
- Reactions to Facebook posts discretizing to show positive and negative reactions for each company.
- Text and Word-Cloud from the text in the post.
- URL to the Facebook post(s).



The instagram page also includes a filter to select the date range and visualize the information from this social network. The visualization is updated based on this selection with this information:

- Number of posts: Total count and count along time for each company.
- Number of reactions (likes): Total count and count along time for each company.
- Sentiment analysis: Results of the ML model predicting the sentiment associated with the comments on Instagram for each company. The total count for good, bad and neutral sentiment is shown along with this information through time.
- Text and Word-Cloud from the text in the comments.
- Text of the original post on Instagram.



The visualization is aimed to process and showcase the crucial information that is continuously gathered from the social networks along with the power of the machine learning model to predict the sentiment associated with the comments on Instagram. The visualization facilitates the understanding of the perception of each company and how the perception for OFFCORSS compares against the major competitors (EPK and Polito Kids), which is one of the major objectives.

3.2 Back-end structure

The back-end is AWS-hosted for data analysis and computation. Briefly the structure design has the following elements:

S3 Datasets:

NPS_Responses.csv: NPS responses from ZS platform. The data is from one survey but more surveys will be requested. It contains the NPS rating that each user gave and the NPS classification. The comments from each user can be evaluated to better understand the client satisfaction level.

Satisfaction_Ratings.csv: Customer satisfaction level when they interact with the support team to solve their issues. It has a level of satisfaction and client comments.

Ventas.csv: Information of sales by date and products

S3 Classification instance:

wordsLexicon.csv: lexicon for words rating needed for the classifier

classifier.pickle: Sentiment rating classifier

EC2 instance:

This instance is run automatically to get data for *web scraping* and updated datasets from OFFCORSS.

Text preprocessing, includes data cleaning, word tokenization and spelling checking from new data.

Data classification ranks new comments feeded into the datasets

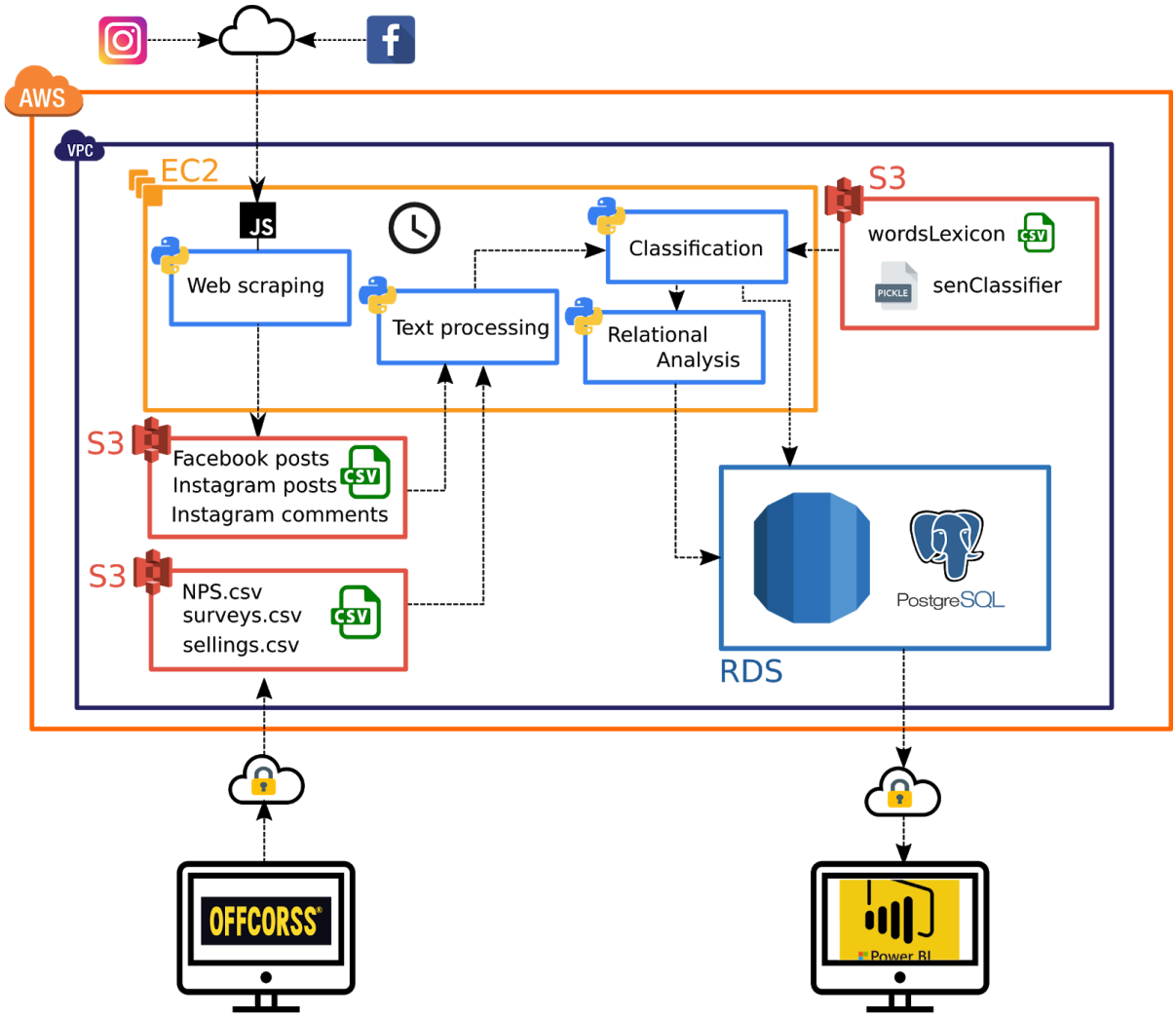
A final layer of relational analysis is added to processes information of sales and OFFCORSS comments

- AWS databases

The RDS dataset keeps the processed data and tables to visualize. This information is accessed by the user using postgresQL for visualization of results

- Security layers

We understand that information is a valuable part of the company. Therefore, we established security procedures for accessing sensible data in the platform. Data writing and data reading is protected by password access.



4. Discussion and final remarks

To complete in the next week