For the sake of the analysis, it will considerer that we only have the available data. After the resolution of the case some outlines were given in order to expand the resolution with new datasets.

Preguntas a resolver:

* **MAXIMIZAR LAS POSIBILIDADES DE LOS COMPRADORES DE CERRAR UN TRATO LUEGO DE UN ‘REPLY’ EN UN ANUNCIO**
* **COMO ASEGURARSE QUE COMPRE LUEGO DE HACER UN ‘REPLY’**

REPLY: acto de un comprador contactando al vendedor a través del chat (el chat es el mayor canal de contacto)

DATOS: Están desde el punto de vista del vendedor y las categorías

* Fecha de antigüedad anuncio
* Usuario que vende
* Categoría del anuncio
* Cantidad de replies
* Fecha de primer y último reply

FINDINGS:

* Si el vendedor no contesta, el trato con ese anuncio no se va a cerrar

1. Cuanto tiempo para que no conteste? Cambia en cada categoría?
2. Ofrecer otros vendedores con productos similares?
3. Avisar al SELLER que tiene una pregunta sin contestar
4. Premiar a los SELLERS con mejor tiempo de respuesta.

* Una vez que el BUYER hace el reply, ya sabemos que producto quiere – recomendar otros SELLERS
* Como medir que termino cerrando el trato? Plantear cierta medición
* Que seria una conversación de relevancia? La relevancia cambia por categoría?
* Cuando el SELLER saca el anuncio de la web preguntar si vendio para luego tener datos de cantidad de mensajes y conversión

OBJETIVO: Maximizar las chances de compra luego de un reply

Cuando hace un reply ya sabemos que producto quiere!

NEGOCIO OLX : que haya mas compradores, para que haya mas vendedores, que el comprador encuentre fácilmente lo que busca

CASO REAL: Busque y pregunte por un comic de BATMAN y me recomendó luego botella antiguo, pedal de distorcion, gorra de coca-cola (en categoría Hobbies, arte y libros)

**ESTRATEGIA 1**

Mejorar al vendedor (o anuncio? Vendedores casuales?) Definir un buen vendedor como couchsurfing

* Aviso al vendedor que alguien esta interesado en su producto
* Cuando hay mensajes sin responder luego de X días - push notification
* Cuando el comprado hacer un reply – push notification
* Conectar cliente con mejores anuncios (vendedores?) de la categoría que esta buscando
* Premiar buenos vendedores – minimiza riesgo de defraudarse

**ESTRATEGIA 2**

“Cuando envía el mensaje, sabemos lo que quiere”

* Ofrecer productos dentro de la misma subcategoría
  + Mejores vendedores
  + Mas populares – mas visitas
  + Productos cercanos
  + Basado en el query de búsqueda
* Ofrecer vendedores con ratio de respuesta alto
* String del título del aviso para hacer una recomendación más personalizada
* String del query de búsqueda correlación con la categoría donde manda el mensaje

**MEDICION**

SI mejoramos la calidad de vendedor podemos mejorar la posibilidad de cerrar un trato, entonces vamos buscar que haya mejores vendedores. Un mejor vendedor es aquel que contesta mas mensajes, en menos tiempo y de calidad.

**Product Analyst Relevance – Case Study**

**Part 1 – Dataset Exploration**

*Item\_data.csv*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | date\_live\_nk | category\_sk | Listing\_sk | User\_sk encrypted | replies |
| dtype | object | object | Object |  | Float64 |
| Unique | 81 | 54 | 2000 | 1935 |  |
| Mean |  |  |  |  | 1.917 |
| Std |  |  |  |  | 2.4511 |
| Max |  |  |  |  | 41 |
| Min |  |  |  |  | 1 |
| Nan | 0 | 0 | 0 | 0 | 0 |
| Top | 2017-09-19 | olx|mea|za|362|378 | 1051110968 | d36caa07b89dd70  878ff87e35a8835aa |  |
| freq | 126 | 536 | 1 | 3 |  |

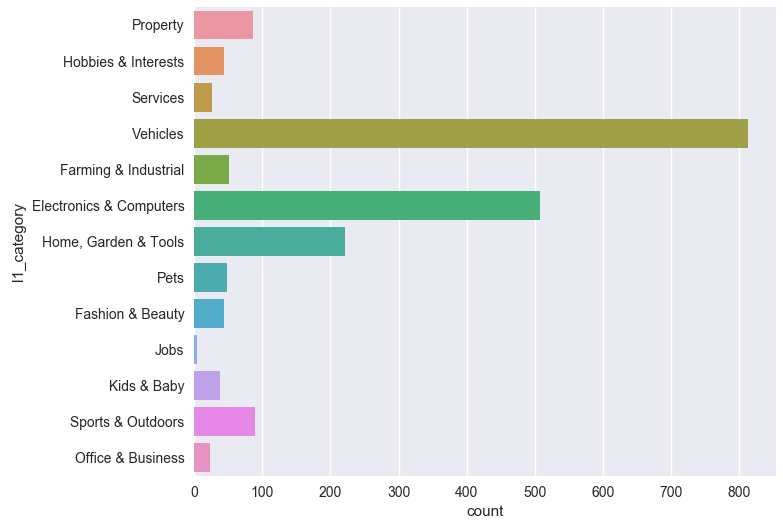
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Successful\_replies | Average\_conversation length | First\_reply date | Last\_reply\_date |
| dtype | Float64 | Float64 | Object | object |
| Unique |  |  | 8 | 8 |
| Mean | 0.981 | 2.522 |  |  |
| Std | 1.639 | 3.874 |  |  |
| Max | 21 | 81 |  |  |
| Min | 0 | 1 |  |  |
| Nan | 0 | 0 | 0 | 0 |
| Top |  |  | 2017-09-14 | 2017-09-21 |
| freq |  |  | 335 | 379 |

*category\_data.csv*

Non NaN values

|  |  |  |
| --- | --- | --- |
| Category l1 code | category\_l1\_name\_en | subcategories |
| 16 | Property | 388,368,367,363,301 |
| 185 | Hobbies & Interests | 911,820,243,214,211 |
| 191 | Services | 647,633,372,324,207,198 |
| 362 | Vehicles | 417,416,379,378,376,377 |
| 5170 | Office & Business | 5171,5172,5173,5174 |
| 600 | Farming & Industrial | 887,604,603,602 |
| 800 | Electronics & Computers | 912,870,805,804,803,802,801 |
| 806 | Home, Garden & Tools | 910,809,807,808 |
| 811 | Pets | 814,813,812 |
| 815 | Fashion & Beauty | 819,817,816 |
| 821 | Jobs | 823,822 |
| 853 | Kids & Baby | 856,855 |
| 881 | Sports & Outdoors | 889,883,882 |

Categories with more items:



**Part 2 – Data preprocessing**

1. Date variables have been transformed to be able of handle them

*item['first\_reply\_date'] = pd.to\_datetime(item['first\_reply\_date'])*

*item['last\_reply\_date'] = pd.to\_datetime(item['last\_reply\_date'])*

*item['date\_live\_nk'] = pd.to\_datetime(item['date\_live\_nk'])*

1. We desegregate the “category\_sk” variable to get the variable in other columns and in numeric values

*categories\_item = item['category\_sk'].str.split('|',expand = True)*

*item = pd.concat([item,categories\_item[3],categories\_item[4]], axis = 1)*

*item = item.rename(columns = {3:'l1\_category',4:'l2\_category'})*

*item['l1\_category'] = pd.to\_numeric(item['l1\_category'])*

*item['l2\_category'] = pd.to\_numeric(item['l2\_category'])*

1. The category table is split in two datasets in order to have the categories and subcategories (called l1\_category and l2\_category) when it is necessary.

*l1\_category = category['category\_sk'].str.split('|',expand = True)[3]*

*l1\_category = pd.to\_numeric(l1\_category)*

*l1\_category = pd.concat([l1\_category, category['category\_l1\_name\_en']], axis = 1).drop\_duplicates([3], keep = 'last')*

*l1\_category = l1\_category.sort\_values([3])*

*l2\_category = category['category\_sk'].str.split('|',expand = True)[4].fillna(0)*

*l2\_category = pd.to\_numeric(l2\_category)*

*l2\_category = pd.concat([l2\_category, category['category\_l2\_name\_en']], axis = 1).drop\_duplicates([4], keep = 'last')*

**Part 3 – Product recommendation model**

**We are going to define an advertisement ranking based in the some indicators within the category of the reply. Once the potential BUYER makes a reply we offer the best advertisement.**

**Lemmas of the strategy**

* Once the BUYER made a reply, we already know which product the BUYER wants.
* A good SELLER improves the chances of closing a deal; we need reliable and better sellers.

**Metric: CTR, we are going to measure if more people “click” the recommendations and then make a “reply” in one of the recommendations.**

**Assumptions:**

* We suppose that the data which was given is the total data or the most reachable
* The dataset is from the SELLER’s point of view, we are going to reference BUYER’s data (history, past buyers, similarities, etc.) as an outline work and not included in the present project. So the data is a proxy to buyer’s chances of closing a deal.

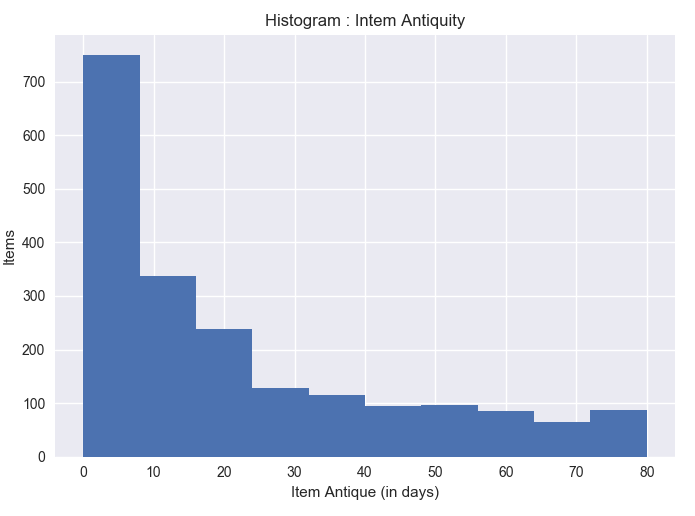
**Step 1: Indicators selection**

1. Item antiquity
2. Conversation relevance
3. Rate of replies given by the seller
4. Number of items of the seller
5. Intensity of the item’s replies
6. Category and subcategory
7. Randomness

**Step 2: Indicator definition and exploratory analysis**

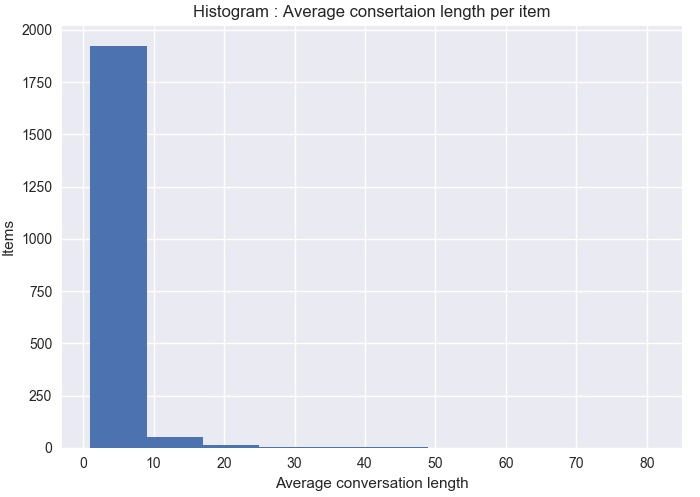
1. Item antiquity

We suppose that newest items are more probably to be sold because the seller recent activity and in order to avoid “forgotten items”.



1. Conversation relevance

We suppose that a greater average of messages refers to a more active seller which is more willing to sell as being more descriptive. Also, it could be that lot of successful transactions are made with few messages.

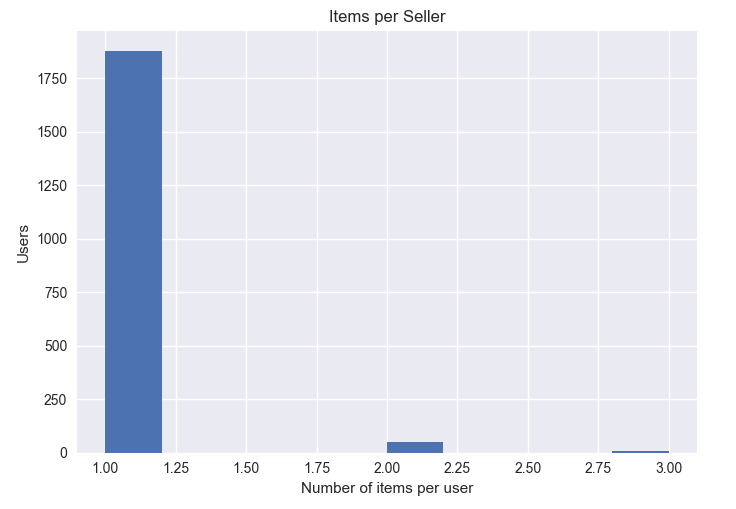


1. Rate of replies given by the seller

The rate is applied only if first *reply < current date (17/09/2017)*, given at least four days to the seller to respond. If still the seller doesn’t respond, he will have a *p = 0*, which acts as a filter not considering the item.

1. Number of items of the seller

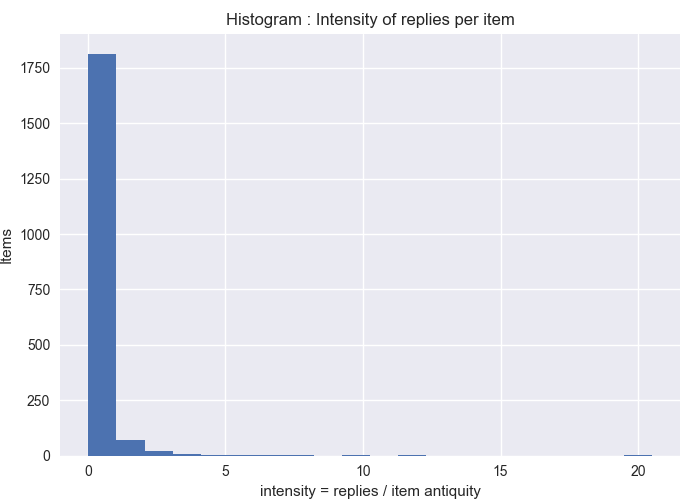
Most of them seem to be “casual sellers” (97%) but we decided to give a favor for those who sell more items in order to be more dedicated to the business.



1. Intensity of the item’s replies

For all items with more than one day since it was publish we take de intensity of replies

With this factor we want to favor those articles which have a strong attention over the time



1. Category and subcategory

We have found that 56 l2\_categories that belongs to 13 l1\_categories. The l2\_category will be the filter to take the items and make the ranking.

1. Randomness *(r)*

Adding some random variable in order to have some variation and not offer always the same thing to each person.

**Step 3: Scoring function and scoring process definition**

1. Weighted product model

**Why?**

* An approach that it still found in sites as “Popular News”, “Most read articles”, “Amazon best sellers”.
* The approach can serve as a foundation of more complex systems



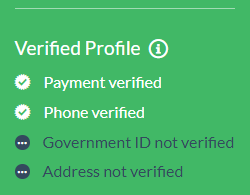
*Amazon best sellers*

**Weaknesses:**

* The method lack of any personalization per client given that the ranking activation occurs with a reply in a subcategory.
* It only works once the buyer makes a reply since it means (under the assumption) we know what he wants.
* Data handling: we rank over the entire number of items in a subcategory which could take lot of computational time. It could be easily solved by adding other filters to diiscard items that we already know that are in the bottom of the rank.

**Outline:**

* Adding more information as user information, this model can feed a SVM rank model as a training set for the algorithm.
* The key of the strategy is to improve the reliability of the SELLER, with this purpose a “couchsourfing approach” can be taken to have more genuine items.

  
*Couchsurfer’s verified profile*