A Work Project presented as part of the requirements for the Award of a Master's degree in Business Analyst from the Nova School of Business and Economics.

HyperML and Deep Interest Network to build a Recommender System for Modatta:

Measuring the effectiveness of targeting users with new offers

Carolina Carvalho Lucas

Work project carried out under the supervision of:

Qiwei Han

and in collaboration with

Rodrigo Moretti and Eduardo Pinto (Modatta)

14/02/2021

Abstract: Modatta aims at giving users control over their data and marketers the opportunity to target users with the right campaigns. Therefore, this paper proposes the use of Deep Learning to find users' representations, so their data is kept private, but they can still learn about their interests. A Recommender System for predicting new interests is built to create a more complete representation of the users, which allows Modatta to find the offers that are of their most interest. After implementing the targeting strategy, this study shows how to evaluate the effectiveness of a campaign that could be conducted by Modatta.

Keywords: Machine Learning; Deep Learning; Hyperbolic Embeddings; Recommender System; Data Monetization; Targeted Marketing; Campaign Effectiveness

Acknowledgements: This work would not have been possible without all the support and orientation of my advisor, Qiwei Han, as well as the contribution of Modatta founders, Rodrigo Moretti and Eduardo Pinto Basto, who gave me the opportunity to be part of this project and provided all the necessary data and knowledge to develop this work. Lastly, I have to thank the members of my group who were always committed to the project and made this work possible.

This work used infrastructure and resources funded by Fundação para a Ciência e a Tecnologia (UID/ECO/00124/2013, UID/ECO/00124/2019 and Social Sciences DataLab, Project 22209), POR Lisboa (LISBOA-01-0145-FEDER-007722 and Social Sciences DataLab, Project 22209) and POR Norte (Social Sciences DataLab, Project 22209).

1. Introduction

This study was conducted in collaboration with Modatta, a Lisbon (Portugal) based startup company that aims to help European Union (EU) citizens to better manage and monetize their data under EU's digital laws.

Modatta was founded in 2019 by Rodrigo Moretti and Eduardo Pinto Basto who saw benefit in creating an app under the EU General Data Protection Regulation (GDPR) to give its users more control over how their data is collected, used, and protected online. This is accomplished with the creation of a community of potential clients willing to share their information for a fair compensation and companies wanting to reduce the cost of attracting customers while promoting a positive relationship based on respect with their audience.

Today, our data and the valuable insights that comes from it are stored and used by big technological companies to gain a competitive advantage over others, and where consumers lose influence to make decisions about their own personal information. Modatta's goal is to allow everyone – both companies and citizens – to benefit from sharing their own data, meanwhile promoting privacy, transparency and a fair share of value.

Moving forward, this study intends to first, investigate how to incorporate modatta users' historical interests, as well as their personal data, to find meaningful representations of the users in a fully privacy-preserving approach. For that, two different Deep Learning approaches will be followed – Hyperbolic Embeddings and Deep Interest Network. Furthermore, a Recommender System is built to recommend users new interests that they are likely to have interest in.

The implementation of these two representation learning approaches is the first step of the whole process to be developed, where the final goal is to allow the deployment of an utterly privacy-based system that will help Modatta to bring the list of potential customers to marketers while giving users the opportunity to learn about their data, always keeping their privacy.

1.1 Problem definition and Objectives

Identifying the problems that need to be addressed and the goals to be achieved is the first step of any project. With this project, our main goal is to help Modatta to build a marketplace for both users and marketers, where both have the opportunity to get valuable insights from data. While users have total control over their own data and can still learn more about it, companies who want to target users with particular attributes will be able to do it in a much easier and less costly way, and never accessing users' data.

As we need to keep data privacy, as mentioned above, the first problem we face throughout this study is the access to real data. To comply with the GDPR, we don't have access to users' data, meaning that the data cannot be inspected.

To overcome this problem, we need to use techniques that allow us to work with the data we have access to. For this reason, we propose dividing our project into 2 Deep Learning approaches: Hyperbolic Embedding and Deep Interest Network (DIN). While the first is using the hierarchical information from Facebook pages categories to get a representation of the user on the Hyperbolic Space, the second is using synthetic users' data and trying to find a way of capturing the similarities between users by learning a latent and abstract representation of them.

The rest of the first part of this paper is structured as follows:

- Section 2 describes the Hyperbolic Embedding and Deep Interest Network approaches.
- Section 3 presents the implementation process of both approaches.
- Section 4 describes the evaluation of both methods.
- Section 5 defines the next steps to be taken.

In the second part, it will be shown how Modatta could evaluate a campaign that a company wants to release in their app, considering that the targeting requirements and the users to be targeted have already been appropriately defined.

2. Deep Learning Approaches

Recently, learning embeddings from data such as text, graphs and multi-relational data has become a central topic in Machine Learning (ML), as well as Artificial Intelligence (AI), since most ML models can´t read and understand linguistic associations in the human sense. Therefore, the learning representations algorithms that characterize semantically meaningful dense vectors shows a response to an increased necessity to the challenges of Natural Language Processing (NLP).

Moreover, Deep Learning based methods have been recently proposed for tackling click-through-rate (CTR) prediction tasks in online advertising, where embedding vectors are used to represent large scale sparse input features, and then are transformed into fixed-length vectors that will work together to learn the nonlinear relations among the features. (Zhou, et al. 2018)

In this project, instead of clicking in an advertisement that is displayed on a website, the prediction task can be defined as whether a Modatta user is interested in a particular insight or not.

To understand how these methods could be used to tackle the problem of this study, two different approaches will be followed to train models that are then used to find meaningful representations of the users' interests and predict new recommendations of the insights and categories that the users are most likely to be interested in.

2.1 Hyperbolic Embedding

In the context of extracting knowledge from symbolic objects, such as words, entities and concepts, hyperbolic embeddings present properties able to capture excellent quality hierarchy information in a few dimensions with arbitrarily low distortion. Even though, the Euclidean geometry is widely used, it is shown that linear embeddings of graphs require higher dimensionality that normally is outperformed by lower dimensionality of hyperbolic embeddings, in terms of both the similarity between objects (distances), and their relative depths in the hierarchy (in their norms). (Maximilian e Kiela 2017)

The constant negative curvature of the hyperbolic space in the area of a circle or volume of a sphere, which may be expanded exponentially with its radius, can be used to model complex networks with hierarchical data structures. Meaning that the distance between two objects is well preserved, giving a measure of their similarity, thus, reflecting their semantic or functional relationship.

In this sense, the Poincaré Ball model is the most adequate in learning hierarchical representations, as it offers to conform mapping between hyperbolic and Euclidean space, i.e., the angles between adjacent vectors indicate to be similar to angles in Euclidean geometry, although the length of their vector is not, but can be illustrated by the hyperbolic distance [Appendix A1 – Equation 1].

This equation exemplifies the utility of implementing Poincaré embeddings, due to their ability to preserve original graph distances to capture the hierarchy of objects, as a result of their norm. Additionally, the dataset used in this study comprises multiple latent hierarchies, which the Poincaré ball is better suited as a larger embedding is needed to model more complex structures such that the difficulty for an optimization method is decreased. (Maximilian e Kiela 2017)

In this sense, the Riemannian SGD optimization involves computing the Riemannian gradient (a scaled version of the Euclidean gradient) with respect to the loss, performing a gradient-descent step and projecting any embeddings that move outwards within its boundary. (Bhuwan, et al. 2018)

To evaluate the hierarchical embeddings, the same approaches were used as described by (Maximilian e Kiela 2017): **Reconstruction** error in relation to the embedding dimension, that is, reconstruction of a hierarchy from the embedding to evaluate the representation capacity. **Link prediction** by splitting the data into a train, validation and a test set to evaluate the generalization performance.

PoincareKMeans

Clustering is one of the most common approaches to identify subgroups in the data showing similar characteristics (Jin X. 2011). To create some groups from the performed embeddings over Poincaré ball, the group resorted to a different version of the K-Means algorithm because of the Euclidean based distance as a similarity measure. The context of the hyperbolic space prohibits the use of linear spaces to determine in its most accurate sense comparable features without exhausting the original graph distances. Thus, the PoincareKMeans will be used. (AlexPof 2021)

Recommendation system

Across the literature, we can find a variety of machine learning models applying Recommender Systems based on metric learning models concerned with designing matching functions between users and items. However, the gross of these examples only encompasses user-item representations in the Euclidean space.

To explore the notion of learning user-item representations with more complex relationships and still preserving their distances when performing embeddings in the hyperbolic space, it was proposed a HyperML model that is able to maintain valuable representations of user-item pairs in the hyperbolic space. (Vinh Tran, et al. 2020) A Recommender System was developed to overcome the limitations of the Poincaré model that unable the option to obtain the relations between categories and users. The Recommender System will then get new insights based on the similarity of the categories and also, on the similarity of the users.

2.2 Deep Interest Network

As referred, our goal is to predict whether users would be interested in the insights that we recommend to them. Many models could do similar work to what we intend to, such as Wide & Deep Learning (Cheng, et al. 2016) or Youtube Recommendation (Covington, Adams e Sargin 2016) for the CTR model. These models share a similar structure [Appendix A2 – Figure

2]: to learn the non-linear relations among the users' features, first projecting extensive sparse features – which are the historical behaviors of users in this study – into lower-dimensional embedding vectors, which are subsequently modified into fixed-length vectors, and lastly concatenated together to be fed into fully connected layers. However, although these models allow reducing the engineering workload significantly, there are still some drawbacks that cannot be compensated, mainly due to the diversity of users' interests. The above-mentioned models are not capable enough to express these users' diverse interests, as they learn the representation of all the interests by compressing the user historical behaviors into embedding vectors, and then obtaining a fixed-length vector by pooling them using Equation 3 in Appendix A2. Essentially, to make these models able to get a representation sufficient enough to express all the users' interests, the fixed-length vectors would need to be broadly enlarged. Consequently, it would significantly expand the size of the learning parameters, increasing the risk of overfitting under limited data. (Zhou, et al. 2018)

Thus, to overcome this problem, a model called Deep Interest Network for Click Through Rate (DIN-CTR) was created by Alibaba Group in 2018 (Zhou, et al. 2018), improving the defects of the models mentioned above. Considering that users may have a great range of vast interests among Facebook pages, to express this interests' diversity, DIN uses a multimodal distribution in which each peak represents a user's interest. Also, users may show interest in the insights that we recommend to them only depending on a small portion of the historical interests instead of the entire historical behavior. For example, imagine a person who loves animals and has liked a dog photo, Trump's scandal, and a post of Marvel's movies. Now we would like to recommend a cat-related insight to him/her, and whether he/she will be interested in this insight or not is uncorrelated to whether he/she has liked Trump's scandal or Marvel's movies before – it is related to his/her previous like of a dog's photo. In other words, in this CTR estimation, some historical data played a decisive role, while the others did not contribute to the prediction.

For this reason, DIN introduced an innovative designed local activation unit adapted from Neural Machine Translation task (Bahdanau, Cho e Bengio 2015) to the structures of models mentioned above [Appendix A2 – Figure 3], maintaining the representative vector of users the same. Activation units are applied to the user behavior features, paying more attention to the related user interests by soft-searching for proper insights of historical behaviors, then taking weight sum pooling to adaptively calculate user representation given a candidate recommendation [Appendix A2 – Equation 4]. Also, as the input of DIN is highly sparse, which is a vector consisting of many zeros, resulting in many parameters, a significant amount of them appears several times, adding extra noises to training. Thus, the model can be easily overfitted. DIN proposed a productive mini-batch aware regularizer to mitigate the overfitting. In every mini-batch, it calculates the L2-norm over the parameters of sparse features [Appendix A2 – Equation 5]. Finally, a last introduced innovative technique was the Dice activation function [Appendix A2 – Equation 6], instead of applying PReLU activation function. The PReLU takes a hard rectified point with a value of 0, which may be inappropriate when the distributions of inputs of each layer are different. The idea of Dice is to adaptively adjust the rectified point according to the distribution of input data, whose value is set to be the mean of input.

3. Implementation

3.1 Hyperbolic Embeddings

The process of implementation of the Hyperbolic Embeddings approach was structured by the following steps: Preprocessing and curation of the data, Generation of a synthetic data of users, implementation of the Poincaré model for learning hierarchical representation, execution of the PoincareKMeans model to obtain clusters, represent the users in the space, and the last part, the deployment of a Recommender System. For more information, check Appendix A1 – Exhibit 1.

The preprocessing of the data was the curation of Facebook categories of users' preferences into tuples fit to serve as input on the Poincaré model [Appendix B1 – Figure 6], as well as an additional dataset to be used as a test set containing information from one of the members of the group to evaluate the model along the whole process. Furthermore, it was necessary to generate a synthetic dataset, which can be read about in Appendix B1.

The Poincaré model takes the hierarchical data coming from Facebook's categories and performs embeddings into the hyperbolic space. The Poincaré is represented as a circle, where the categories that are on the high-level of the hierarchy are laid closer to the center and lower-level categories (child-nodes) shifted to the end of the circumference. The end result illustrates a tree-like structure where similar categories are put nearer each other and the less similar ones farther away [Appendix B1 – Figure 7].

The outcome of the Poincaré model is a set of vectors or coordinates of each category on the hyperbolic space, which is useful to calculate the distances between categories to get similar (closer) categories or else, check the ranks (relationship) between two variables [Appendix B1 – Figures 8 & 9]. To note that from the distances between the categories we can also get new insights for the users, as can be seen in Appendix B1 – New Insights for Users.

Since the Poincaré model shows to be incredibly useful for learning the latent hierarchical embeddings, it enabled to represent the user into the hyperbolic space. Firstly, each user was represented into the space by calculating an average of their interests and characteristics, creating one unique vector. Additionally, to represent the embeddings of the users into the space, the creation of target groups through the PoincareKMeans, would group users that have similar characteristics together. That way, the characteristics of the targeted group create an avatar of the users that were targeted.

By representing the embeddings of the users into the space, the closeness between them would posteriorly, create target groups of users, in which the users that have similar characteristics will be grouped into a cluster. For more details, check Appendix B1 – Representation of the User into the Space. The implementation of the PoincareKMeans allowed to create 6 target groups [Appendix B1 - Figure 13], enabling the suggestion to a specific user, certain categories that he/she might like, because the users of their cluster (that are similar to he/she) like them too.

It is worth highlighting, the Poincaré model is not able to get relations between users and categories, so until this instant, the focus was on the underlying relationship between categories. Nevertheless, the necessity for a Recommender System able to perform in the hyperbolic space as well as, exploring the notion of relationships between categories and users let us to the next, and final step. Its implementation. We used the aforementioned model, HyperML, by means of a tensorflow package, called CurvLearn, providing the framework for training deep learning models in non-Euclidean spaces (Alibaba 2021). The Recommender System will outcome the recommendations of insights to users.

3.2 Deep Interest Network

Input Data

As this project is done on a data privacy basis, there is no access to Modatta users' data. However, to train a Deep Interest Network, data about users' profile and previous interests needs to be inputted to the model. So before starting the training process, synthetic data had to be created to train and test the models. The generated data is divided into three main sections: **Insight features**, which contains a list of insights that the users may be interested in (originated by the users' likes in Facebook pages), associated with the correspondent categories. **User profile features**, which corresponds to synthetic data that has the profile information about each user. **User historical behavior features**, which has the historical behavior of each user, containing data about whether the user has interest in a particular insight.

Besides creating the synthetic data, we are also using one of our members' consented data to validate the recommendations with real data. To a better understanding of all the data we have generated in each section, further details are given in Appendix B2 – Generated Data section.

Implementation

To implement the models under this approach, a Kaggle example of Ad CTR Prediction using the DIN model (Sanagapati 2019) was followed to understand how the data should be handled. Likewise, the DeepCTR documentation was studied for a better understanding of how to approach the model training (Shen 2021). To start the training process, several models were tried and the model that produced the best results was the already explained DIN model. To choose which model should be used, several parameters were changed, and overfitting tried to be reduced as much as we could [Appendix C2].

After having the best model trained, two different approaches were followed to build the recommendation system and get insights recommendations for Modatta users. While the first consists of directly using the *model.predict* method to recommend some of the unknown insights to the users as individuals, the second approach lies in getting the embedding weights for each user in the dataset and using them to find similarities between users.

Individual Recommendation

In this approach, the predict method was applied to the *x_unlabeled* data and the probabilities of recommendation were assigned to each *user_id – insight_id* observation. Then, to get the insights that a particular user might like, one should just get the ones with the highest probability of recommendation. The results of applying this approach to the real user's (Emila's) data can be found in Appendix B2 –Table 16.

Besides recommending particular insights, another recommendation system was built to get the insight categories that the users are more likely to have an interest in [Appendix B2 – Table

17]. This is because the insights are very specific, and we can get more accurate results by aggregating them by their categories.

Finding Users Representation

The second approach consists of finding users' representations based on their embedding weights (which represent a mapping of the categorical variables to a vector of continuous numbers). These vectors will then be used to identify which users are more similar to each other, through a clustering technique: K-Means. This information can then be used to target groups of users with new insights or new offers, based on characteristics of the target audience.

Step 1 – Retrain the model

To develop this approach, there was a need for a different split of the data on the user side. Only some of the users were used to train the model and then a function was defined to obtain the embedding weights for the new users. For this reason, the previous model needs to be fitted again, now with the new training and test split.

Step 2 – Find representations for the different users

In this step, the embedding weights for the different users were obtained and the profile data of each user was merged with the embedding weights, so that we have a more complete representation of the users. As the users' profile data is categorical, one hot encoder was used to transform them, so that later on, when analyzing the cluster centroids, valuable information can be extracted from the values that are shown for each variable.

Step 3 – K-Means clustering

In this section, the clustering technique K-Means is used to get the clusters for each of the users present in the dataset, based on the previously created representations. The optimal number of clusters was found with *KElbowVisualizer* and the users were assigned to each cluster. *Umap* technique was used to reduce the dimension of the data so that it could be easily visualized in the 2-D space [Appendix B2 – Figure 16]. The profile distribution of the users within the clusters was also plotted so that one could get a clear picture of how the clusters are defined [Appendix B2 – Figures 17, 18 and 19]. Different functions were created to get: **1.** the cluster assigned to each user; **2.** the centroids for each cluster; **3.** the users and correspondent liked insights and categories for each cluster.

By analyzing the centroids for each cluster, one could check the average profile information characterizing each group of users. The results comparing Emila's data and the cluster he/she is inserted in can be found in Appendix B2 –Table 18 and Figure 20. Getting the users for each cluster allows us to find the insights and insight categories that are the most liked inside each cluster. By doing that, one could then get recommendations for a new user.

Lastly, a final recommendation system was built for this approach, and the results of applying it to Emila's data can be found in Appendix B2 –Tables 19 and 20.

4. Evaluation

Throughout the process of implementation of the techniques previously stated, some evaluation procedures were undertaken to assure the outcomes succeeded to translate meaningful conclusions. Conclusions that can afterwards be successfully implemented by Modatta.

4.1 Poincaré model and HyperML

The first step in the evaluation process occurred in the Poincaré model aimed at choosing the best model for a set of hyperparameters displaying the most adequate learning hierarchical representation. In order to find the best model, it was done a series of experiments to fine tune the model's hyperparameters [Appendix C1 – Figure 21], to which the evaluation assumed two approaches, mentioned before: **Reconstruction** and **Link Prediction**. These two approaches can be determined by two methods:

- **Mean Rank**: the average of the ranks for all observations within each sample.
- **MAP**: refers to Mean Average Precision, which is a metric that captures how well each vertex's neighborhoods are preserved.

Table 1 - Poincaré Ball Evaluation

	Reconstruction				**Link Prediction**			
	25D	50D	100D	200D	25D	50D	100D	200D
Mean Rank	2.59	2.55	**2.50**	2.53	3.19	2.98	3.07	**2.76**
MAP	0.534	0.534	0.536	**0.54**	0.413	0.418	0.416	**0.538**

The results of the Implementation of the Poincaré Ball can be seen on the table above. Looking at the Reconstruction, one can check that the values vary slightly, specially the MAP values that only increase from 0.53 to 0.54. However, looking at the Link Prediction values, the Mean Rank had 3.2 on 25 dimensions and decreased to 2.76 as well as the MAP value got from 0.41 to 0.538, showing a substantial increase on the Link Prediction. Therefore, the best one had the following characteristics: **20 negatives** sample to use, **0 number of epochs** to use in burn-in initialization, no regularization, **200 epochs**, in a **200-dimensional space**.

Moreover, the PoincareKMeans model is in its essence an unsupervised learning algorithm, where the evaluation metrics for clustering analysis is still very shy. The choice for the number of clusters came down mostly to domain knowledge and intuition, so the number of clusters chosen after a series of attempts was 12.

The last step regarded the evaluation of the Recommender System with the Hit Rate (HR) metric, measuring the fraction of users for which the right answer is in fact included in the suggestion by the recommendation system. The final **HR@10** was of **0.77**, which is a good result, as compared with the results from HyperML experiments (Vinh Tran, et al. 2020). The parameters are represented in Appendix C1 – Figure 23.

To understand if the Recommender System was deploying the intended results, we experimented with a test set containing information from one of the members. The final results were presented as insights as well as master categories [Appendix C1 – Tables 22 & 23].

4.2 DIN

During the training process of the DIN model, some problems of overfitting emerged, meaning that the model was able to get very accurate predictions for the training set but when applied to new data the evaluation was not that good. Thus, to overcome the overfitting the model was suffering from, L2 regularization and Dropout parameters were changed, having the final values for these parameters been defined as follows:

- l2_reg_dnn = 0.01
- dnn_dropout = 0.03

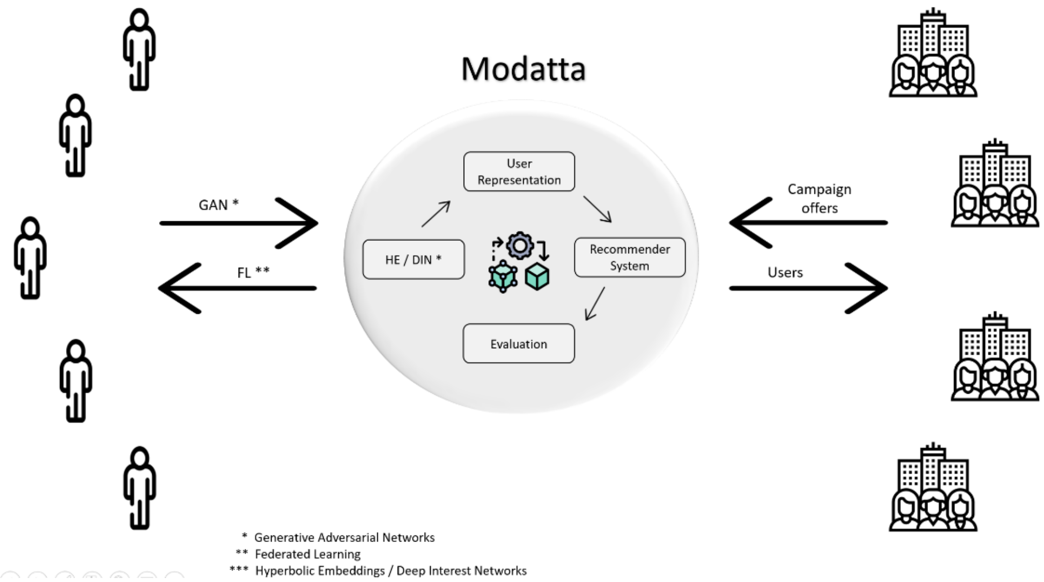
To evaluate the model as a regression problem, the two metrics used were log Loss and AUC, however, to get more interpretable results, we have also transformed it into a classification problem and calculated accuracy, recall and precision metrics. In this case, recall is our most important metric, as we want to minimize the number of False Negatives (insights that users would be interested in, but we predict as non-liked insights, and so don't recommend them to the user), even if we have some False Positives (insights that users would not be interested in, but we wrongly recommend them because we predicted as liked insights). This way, when recommending a wrong insight, the user can tell us if it is not true, but if we don't recommend it at all we are not getting any information from the user side. Thus, the model was chosen mainly based on the recall metric. As we can see in Table 2 below, the model is not overfitting and the recall score is pretty good, meaning that the model is correctly predicting the majority (88%) of insights the users are interested in. More details concerning the best model evaluation can be found in Appendix C2.

Table 2 - Best DIN model Evaluation

Metric	Training set	Test set
Log loss	0.619	0.620
AUC	0.547	0.545
Recall	0.878	0.876

5. Next Steps

Figure 1 - Diagram representation of the system to be developed



Above, the figure shows a diagram of the overall system intended to be developed after applying the models before described in this paper so far. To do so, in the next phases, there is a need to implement the following steps:

1. Identify a user-base for a campaign offer, making sure that marketers target the users that are more prompt to accept the offer, as well as users get the campaigns they want and need.
2. Generative Adversarial Networks implementation for generating synthetic data to feed to the model, preventing users' privacy from the potential data leakages.
3. Federated Learning implementation to demonstrate an alternate approach to the traditional centralized learning, aimed at leveraging communication efficacy and defense techniques to ensure data privacy.
4. Evaluate the effectiveness of the campaign, accounting for both the profile and interests' distribution of the targeted users and use the results to improve the targeting strategy.

The implementation of these four steps together will then allow the deployment of an utterly privacy-based system that will help Modatta achieve its goal of guaranteeing privacy-preserving for the users and bringing the list of potential customers to marketers.

6. Measuring the effectiveness of targeting users with new offers

The main purpose of the work presented in this study so far was to use DL models to infer new insights that Modatta users would be likely to have interest in, based on their profile data and/or their already known interests. Then, the users are recommended with new insights and can access a more complete representation of themselves in Modatta's app. As Modatta is a two-sided marketplace for both companies that want to target the right users and users that want to receive the right offers, it is essential to have a good representation of the users so that Modatta can find the offers that fit them the best.

In the second part of this study, it will be shown how Modatta could evaluate a campaign that a company wants to release in their app. There are several types of offers the users can be targeted with (e.g. to make a purchase online, to visit a website, to watch an advertisement…), so a particular campaign will be defined and evaluated, again based on synthetic results that will be generated.

It is important to note that the user only receives the money when completing the offer until the end, and this completion must be validated. So, when a user is targeted, he/she may accept or discard the offer; if the offer is accepted, the user may complete the offer or not; and finally, the offer needs to be validated so that the user receives the 1€ reward.

Similarly, the marketer only pays Modatta when the offer is validated, so there is no risk associated for the company carrying a campaign using Modatta's app. This is a great advantage when it comes to measure the costs and defining the budget of a company for digital advertising, as it is necessary that the users complete the action until the end to have the company spending money with them (which will probably be compensated by the action performed by the user).

7. Campaign definition

In order to evaluate how well an offer is being received by Modatta users, a particular campaign is defined, as well as its target audience, and the users are targeted based on their profile and interests.

So, let's consider a campaign that Vertbaudet, an online marketplace for babies and children's clothing items, wants to conduct with the help of Modatta, in which the goal is to increase the company's online sales. For that, the offer that the users are asked to complete is to make a purchase in Vertbaudet online store. The company incurs in a total cost of 10% cashback of the value spent by each user that completes the offer (i.e. makes a purchase). 50% of the value paid by the company goes to Modatta and the other half is the reward the users receive in the app by completing the action.

The targeting process is meant to consider both users and companies preferences, and in this part of the study it is assumed that this process was already conducted, having the targeting strategy and the users to target been appropriately defined [Appendix D].

8. Evaluation

To evaluate how well the offer was received by the users, it is important to check, from the users that were targeted, which ones have accepted the offer, and which have completed it until the end. Then, the distributions of the three groups of users are analyzed and compared with each other to see if the targeting criteria could still be improved.

From the companies' perspective, it is also important to evaluate the effectiveness of the campaign in financial terms, thus some KPIs will be defined to measure how the campaign is affecting the company's results.

8.1 Users' distribution

As mentioned above, the first evaluation step will be to check for both the profile distributions and the interests of the following three groups of users:

1. Users that were **targeted** with the offer
2. Users that have **accepted** the offer (not necessarily completing it)
3. Users that have accepted and **completed** the offer until the end

The detailed results of this analysis can be found in Appendix E, and there are some key points that should be considered, regarding both profile and interests' distributions:

Table 3 – Users' profile distribution

	Targeted users	Users who have accepted the offer	Users who have completed the offer
Age: above 20 years	82.9 %	92.1 %	95.3 %
Gender: Female	45.6 %	76.2 %	83.1 %
Civil status: Single	24.9 %	9.8 %	7.4 %

The profile conditions were added, separately, to the original requirements, to measure the impact of each of the features individually. As such, they are not exclusive to each other, and the values represent the rate of users that fit each of the conditions on the left. By looking at Table 3, some differences in the profile distribution of the users that were targeted and the ones who have accepted and completed the offer can be highlighted:

First, regarding the users' age, it is notable the percentage of users **above 20 years** old that have accepted and completed the offer. Although this was not one of the criteria used to target the users, it may be a key insight to consider for next campaigns, as users under 20 years old are not as likely as the older ones to have interest in babies and children's clothing.

Another factor that could be pointed out is the fact that from the users who have accepted the offer, around 76% are **women**, and roughly 83% of the users who have completed the offer until the end are female as well. This might show a slightly higher interest from the women's side in kids' clothing items when comparing to men.

Finally, only less than 10% of the users who have accepted the offer are **single** (and for the users who have completed the offer only less than 7.5%), meaning that users who are not single are more likely to respond positively to this kind of offers.

The results of redoing the targeting process and considering these factors show that it is possible to improve the targeting strategy and obtain better results [Appendix F].

Table 4 - Users' most liked categories of insights

Targeted users	Users who have accepted the offer	Users who have completed the offer
FASHION	**FASHION**	**FASHION**
TV	**SHOPPING AND DEALS**	**SHOPPING AND DEALS**
DESIGN	AGRICULTURE	**DESIGN**
REAL ESTATE	MUSIC	TV
ARTS	**DESIGN**	AGRICULTURE
SHOPPING AND DEALS	MOTOR VEHICLES	SCIENCE AND TECHNOLOGY

From the above, one could also look at the categories of insights that each of the group of users like the most and analyze some of them. It is important to note that the targeted users must had an interest in the following categories: Babies and Children, Fashion, and Shopping and Deals. However, "showing interest in" doesn't mean that it is one of the categories they like the most. So, although "Babies and Children" was one of the categories used for targeting users, it is not part of the most liked categories of any group, and this is probably because these categories are originated from Facebook insights, and there are many other categories in which the users show more interest than babies and children when scrolling in social media platforms.

Regarding the other categories that were part of the requirement, both Fashion and Shopping and Deals are two of the 10 most liked categories of the users targeted, being the 2 most liked for both the users who have accepted and those who have completed the offer successfully.

Then, we could see that the targeted users also had a great interest in Design, which is still part of the most liked categories for the users who accepted the offer, and for the users who have completed it Design is the 3rd most liked category, so it could be beneficial to consider this interest when targeting users with similar campaigns. In fact, the experiment was retaken including users who also like Design and the results proved to be slightly better in terms of the users' conversion rate. [Appendix G]

The results of evaluating the campaign with the users' distributions could also be used to inform the company about the profile and interests of the users that are responding more positively to the offer. The company could then use this to improve the definition of its target audience, not only for future offers that it would like to have in Modatta's app, but in their other targeted marketing campaigns.

Also, this evaluation could serve as a mechanism to get explicit feedback from the users' side, as Modatta could ask the users that were targeted with the offer (because of their profile and interest in particular categories of insights) and didn't accept or complete it if they are still interested in the insights' categories that were determinant in choosing them to be targeted with the offer. This way, users' data is constantly updated with their current interests and the learning process is improved.

8.2 Offer effectiveness

Measuring a campaign effectiveness is essential to guarantee that Modatta is playing a great role in helping companies targeting the right users with the right offers. There are several ways of evaluating the effectiveness of a campaign, but in this case the following KPIs will be used: **CvR** (Conversion Rate), **CPA** (Cost Per Action) and **ROAS** (Return on Ad Spend). [Appendix H]

CvR (Conversion rate)

Conversion rate measures the rate of offers completion, which means it is obtained by dividing the number of users who completed the offer by the number of users targeted with that same offer. As expected, only a percentage of the targeted users will accept and complete the offer until the end. So, CvR is a very important metric as it will help Modatta to determine the number of users to be targeted with similar offers in the future.

Analyzing the data that has been generated, one can see that from the 2833 users that were targeted, 551 have completed the offer with success, so the Conversion rate is equal to **19.45%** [Appendix I].

By looking at the Appendix F, it is possible to check the increase in the Conversion Rate when profile-based retargeting strategies are adopted. The following table shows the CvR when each of the profile features additional conditions are considered (one by one, ceteris paribus):

Table 5 - Increase in CvR with profile-based retargeting

	Age: above 20 years	**Gender: Female**	**Civil Status: not single**
Conversion Rate	23.33 %	30.42 %	25.28 %

It is important to note, though, that one should look at the values that originated these rates. For example, when retargeting based on Gender, only 1292 users were targeted, from which 393 completed the offer, so although there was an increase in the CvR, the company's goal with this campaign would probably not be achieved. As such, Gender could be used as an additional requirement if many users were to be targeted or, on the other hand, it could be used to get more users besides the ones selected based on the original requirements, if needed. This way, female users would be considered because of their higher likelihood to positively respond to the offer.

Appendix G shows the increase in the CvR (to 24.64%) when considering the users' interests to retarget. In this case, 798 users end up completing the offer until the end, so it is proved to be beneficial to add more users based on their interest in Design.

CPA (Cost per action)

CPA is a metric that measures the amount marketers pay per a specified action on their offer. This metric is used instead of the CPC (Cost per click) because the marketers only pay Modatta when the user completes the offer until the end, so there is no risk associated for the marketers. In order to calculate the cost per action of this campaign, one needs to know how much Vertbaudet is paying for each user that completes the offer. As the cost per user is depending on the value spent by the user when making a purchase at Vertbaudet online store, new data was generated with hypothetical buying actions for each of the users who have completed the offer [Appendix J].

The average order value resulting from the various purchases was 35.8€. With the company's cost being 10% cashback of the value spent, the cost per action of this campaign is equal to **3.58€**.

ROAS (Return on Ad Spend)

Return on Ad Spend is considered by many the most important KPI to evaluate the effectiveness of online advertising (Greene 2019), and represents how much Vertbaudet is earning with this campaign for every euro invested. It is obtained by dividing the earnings resulting from the campaign by the cost of the campaign for the company, measuring the revenue earned per euro spent on the campaign. (Return on Ad Spend s.d.)

As the total cost is depending on the total revenue in this campaign, it turns out that ROAS only depends on the defined cashback rate [Appendix K]:

ROAS = 1 / cashback

So, the value obtained for this metric is equal to **10**.

9. Comparison with other ad platforms

As many other online platforms that have their own purpose of existence, Modatta's revenue comes essentially from the amount companies pay for having campaigns in the app, so Modatta can be seen and generally described as an ad platform. However, it is not just one more ad platform, and it has some key points that differentiate it from the others.

Let's consider two well-known entities, Facebook and Google. Facebook is a social media that aims at connecting people all over the world, while Google is a search engine which was created with the goal of making information accessible and useful to everyone. Although the above-described are the purposes of these two companies, they are both ad platforms when it comes to evaluate the business revenues. (Fitzgibbon 2021)

As such, many companies make use of platforms like the ones mentioned above to advertise their business and to target the users they want to. A market study conducted by the Competition and Markets Authority, in July 2020, reveals that “Google has more than a 90% share of the £7.3 billion search advertising market in UK, while Facebook has over 50% of the £5.5 billion display” (Competition and Markets Authority 2020).

Many businesses advertise their products in these platforms because they have the ability to target the right users based on their search history or their interactions in social media, which comprise very unique data for each single user (Curran 2018). However, here arises one of the major problems of nowadays online advertising: the invasion of users’ private data. (Amnesty International 2019) This is one of the most important advantages of Modatta over the referred ad platforms. While other ad platforms access and clearly invade users’ data (such as google searches or Facebook interactions) to be able to target them with the right ads, by using Modatta the users are the ones who have control over their data, with neither Modatta nor the companies having access to it.

Another great advantages of Modatta’s app over these platforms is that companies only pay per action taken until the end. So, in the campaign analyzed in this study, Vertbaudet would only pay per user that made a purchase online, which would result in revenue coming from that performed action. This is not what usually happens in other ad platforms, where companies usually pay per click on their ads, which means that they are paying for every person who clicks on their ad, even if that click doesn’t translate in the expected action (e.g. making a purchase) (What Is PPC? Learn the Basics of Pay-Per-Click (PPC) Marketing s.d.). Or even when the companies pay per action, the advertisement comes to quite expensive values when comparing to Modatta [Appendix L]. (Irvine, Facebook Ad Benchmarks for Your Industry 2021) (Irvine, Google Ads Benchmarks for Your Industry 2021)

When it comes to measuring the users' satisfaction, besides the companies' financial metrics, Modatta is also playing a fundamental role in giving users what they want: to have control over their data but still learning about it, letting it be anonymously used to find the offers they would like the most, and being rewarded for doing so. Also, with the possibility of interacting with Modatta's app, users can explicitly express their interests (and changes of interests) and help the learning process to be constantly improved. This means that users can influence the kind of campaigns they will be targeted with and stop receiving annoying advertisement of companies or products they don't care about at all.

10. Conclusion

To sum up, all the work presented in this study aimed at first, use Deep Learning models to find a way of learning a meaningful representation of Modatta users while keeping their data private, considering both their profile data and interests, and then using that representation to discover new insights and recommend them to the users. By discovering new insights, an even more complete representation of the users was achieved, which then allowed to target users with offers they would like to receive and to help marketers finding the users that would be most likely to complete their offers until the end.

In a second moment, this paper showed how a particular campaign that could be released in Modatta app should be evaluated, and how it could help marketers improving their targeting strategy. Essentially, the work developed in this study aims to help Modatta to achieve a fundamental role in the digital advertising world. By doing so, both companies and users can benefit from completely privacy-based interactions and get the most of it to be active players in the improvement of data-driven marketing strategies.

References

AlexPof. 2021. *PoincareKMeans: K-Means algorithm in the Poincare Disk Model.* 11. https://github.com/AlexPof/PoincareKMeans.

Alibaba. 2021. *Curvlearn.* November 15. Accessed November 2021. https://github.com/alibaba/Curvature-Learning-Framework.

Amnesty International. 2019. *Surveillance Giants: How the business model of Google and Facebook threatens Human Rights.* London: Amnesty International.

Bahdanau, Dzmitry, KyungHyun Cho, and Yoshua Bengio. 2015. "Neural Machine Translation by Jointly Learning to Align and Translate." *3rd International Conference on Learning Representations, ICLR 2015.* San Diego, USA.

Bhuwan, Dhingra, Norouzi Mohammad, M. Dai Andrew, Shallue Christopher J., and E. Dahl George. 2018. "Embedding Text in Hyperbolic Spaces." *Carnegie Mellon University; Google Brain* 11. https://arxiv.org/pdf/1806.04313.pdf.

Bonnabel, Silvere. 2011. "Stochastic gradient descent on Riemannian manifolds." 29. https://arxiv.org/abs/1111.5280.

Chander, Shivani. 2017. *Visualisation of High Dimensional Data using tSNE – An Overview.* Accessed November 2021. https://github.com/shivanichander/tSNE.

Cheng, Heng-Tze, Cheng Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, et al. 2016. "Wide & Deep Learning for Recommender Systems." *DLRS 2016: Workshop on Deep Learning for Recommender Systems.* Boston MA, USA: Association for Computing Machinery. 7 - 10.

Competition and Markets Authority. 2020. *Online platforms and digital advertising.* Market study, Competition and Markets Authority.

Covington, Paul, Jay Adams, and Emre Sargin. 2016. "Deep Neural Networks for YouTube Recommendations." *RecSys '16: Tenth ACM Conference on Recommender Systems.* Boston Massachusetts, USA: Association for Computing Machinery. 191 - 198.

Curran, Dylan. 2018. "Are you ready? Here is all the data Facebook and Google have on you." *The Guardian.*

Erdem, Kemal. 2021. *t-SNE clearly explained.* April 13. Accessed 2021 November. https://towardsdatascience.com/t-sne-clearly-explained-d84c537f53a.

Fitzgibbon, Allie. 2021. "Digital Dominance: the rise and rise of Google and Facebook." *London Connection.*

Greene, Jessica. 2019. *The 28 Advertising KPIs That Every Marketer Should Be Tracking.* May 28. Accessed December 8, 2021. https://databox.com/most-important-advertising-kpis.

Gunel, Beliz, Fred Sala, Albert Gu, and Christopher Ré. n.d. *HyperE: Hyperbolic Embeddings for Entities.* Accessed October 2021. https://hazyresearch.stanford.edu/hyperE/.

Irvine, Mark. 2021. "Facebook Ad Benchmarks for Your Industry." *The WordStream Blog.* October 9. Accessed 12 8, 2021. https://www.wordstream.com/blog/ws/2017/02/28/facebook-advertising-benchmarks.

—. 2021. "Google Ads Benchmarks for Your Industry." *The WordStream Blog.* October 23. Accessed 12 8, 2021. https://www.wordstream.com/blog/ws/2016/02/29/google-adwords-industry-benchmarks.

Jain, Jayant. 2017. *Implementing Poincaré Embeddings.* 12 September. Accessed October 2021. https://rare-technologies.com/implementing-poincare-embeddings/.

Jin X., Han J. 2011. "K-Means Clustering." *Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning.*

Keng, Brian. 2018. *Hyperbolic Geometry and Poincaré Embeddings.* June 17. Accessed October 2021. https://bjlkeng.github.io/posts/hyperbolic-geometry-and-poincare-embeddings/.

Maaten, Laurens van der, and Geoffrey Hinton. 2008. "Visualizing Data using t-SNE." *Journal of Machine Learning Research 9* 27. https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf.

Maximilian, Nickel, and Douwe Kiela. 2017. "Poincaré Embeddings for Learning Hierarchical Representations."

Řehůřek, Radim. 2021. *Train and use Poincare embeddings.* August 30. Accessed October 2021. https://radimrehurek.com/gensim/models/poincare.html.

n.d. "Return on Ad Spend." *Singular Glossare.* Accessed 12 10, 2021. https://www.singular.net/glossary/return-on-ad-spend-roas/.

Sanagapati, Pavan. 2019. "Ad CTR Prediction - DIN Model." *Kaggle.* Accessed 10 2021. https://www.kaggle.com/pavansanagapati/ad-ctr-prediction-din-model/notebook.

Shen, Weichen. 2021. *DeepCTR Documentation.* September 13.

n.d. *Tutorial on Poincaré Embeddings.* Accessed October 2021. https://notebook.community/gojomo/gensim/docs/notebooks/Poincare%20Tutorial.

Vinh Tran, Lucas, Yi Tay, Shuai Zhang, Gao Cong, and Xiaoli Li. 2020. "HyperML: A Boosting Metric Learning Approach in Hyperbolic Space for Recommender Systems." *The Thirteenth ACM International Conference on Web Search and Data Mining (WSDM'20)* 9. https://arxiv.org/pdf/1809.01703.pdf.

Violante, Andre. 2018. *An Introduction to t-SNE with Python Example.* August 29. Accessed October 2021. https://towardsdatascience.com/an-introduction-to-t-sne-with-python-example-5a3a293108d1.

n.d. "What Is PPC? Learn the Basics of Pay-Per-Click (PPC) Marketing." *The WordStream Blog.* Accessed December 10, 2021. https://www.wordstream.com/ppc.

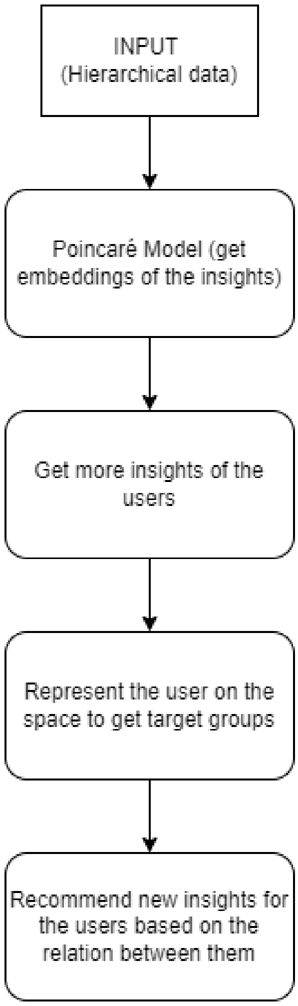
Zhou, Guorui, Chengru Song, Xiaoqiang Zhu, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. "Deep Interest Network for Click-Through Rate Prediction." *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* London, UK: Association for Computing Machinery. 1059–1068.

Appendix

A – Introduction to Hyperbolic Embeddings and Deep Interest Network

A1 – Hyperbolic Embeddings Approach

Exhibit 1 – Diagram of the Hyperbolic Embeddings Approach



Above, the diagram shows a flow of the overall system of the Hyperbolic Embedding Approach. It starts with the implementation of the Poincaré Model and ends up with the deployment of a Recommender System.

Formula of distance on Hyperbolic Space

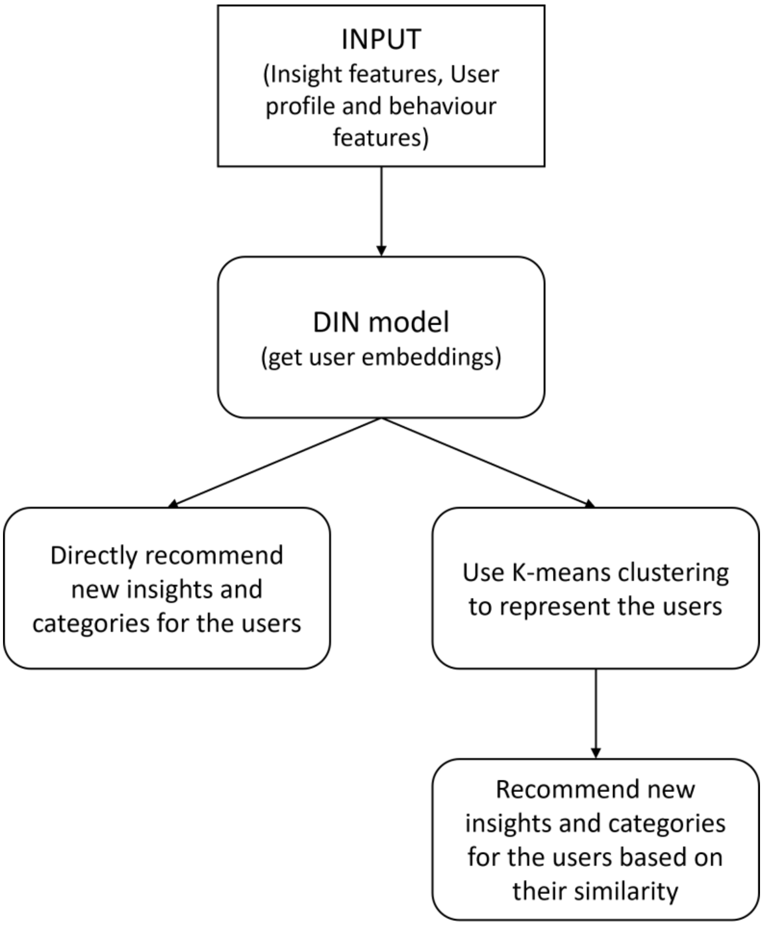
$$d_H(x, y) = acosh\left(1 + 2\frac{\|x - y\|^2}{(1 - \|x\|^2\ )(1 -\ \|y\|^2)}\right)$$

Equation 1 – Formula of distance on Hyperbolic Space

The formula above calculates the distance between 2 points, x and y, on the hyperbolic space. The hyperbolic distance within the Poincaré ball varies smoothly with the position of *x* and *y*. Suppose three points, x and y, are the children of a parent z, placed at the origin 0. When the points move towards the edge of the disk, i.e. x -> 1, in the Euclidean space, the ratio $\frac{d_E(x,y)}{d_E(x,0)+d_E(0,y)}$ remains a constant. By contrast, the ratio $\frac{d_H(x,y)}{d_H(x,0)+d_H(0,y)}$ in the hyperbolic space gets closer to 1, giving the most approximate distance to the original graph distance ratio. This experiment exemplifies the utility for implementing Poincaré embeddings, due to their ability to preserve original graph distances to capture hierarchy of objects, as a result of their norm.

A2 – Deep Interest Network Approach

Exhibit 2 – Diagram of the Deep Neural Networks Approach

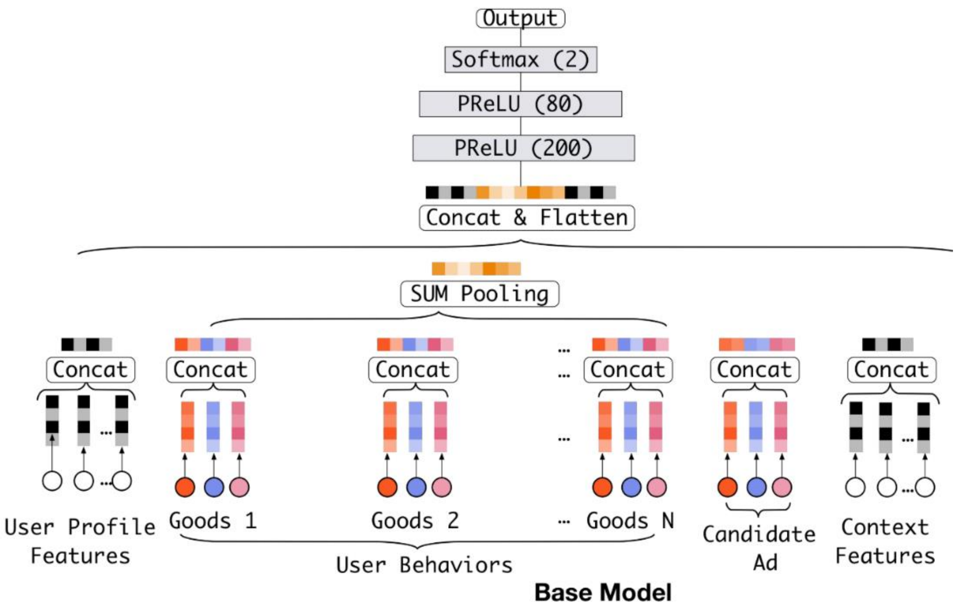


Above, the diagram shows the process of implementing the Deep Neural Networks Approach. It starts with the deployment of the DIN Model, splits into two different recommendation methods, individual and clustering.

Deep Neural Networks Models

Base model

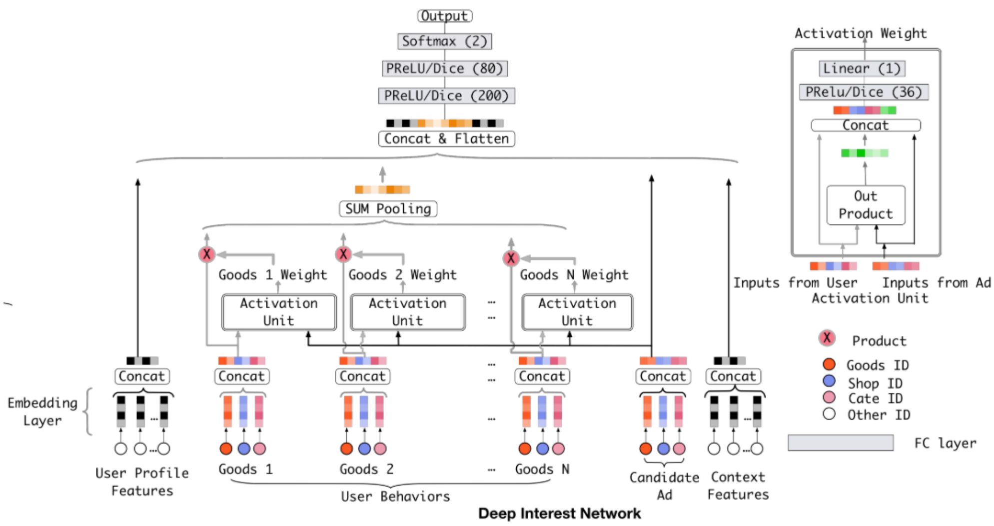
Figure 2 - The architecture of Base models



The architecture of Base models consists of several parts: Embedding layer, Pooling layer, Concat layer and multi-layer perceptro.

DIN Model:

Figure 3 - The architecture of DIN



The architecture of DIN has almost the same architecture as the base model. Introduced an innovative local activation unit function.

Loss function for both models:

$$L = -\frac{1}{N} \sum_{(x,y)\in S}^{n} (\text{y}\ log_{p(x)} + (1-y) log_{(1-p(x))}$$

Equation 2 – Formula of Loss function

S denotes the training set of size N, with x as the input of the network and y$\in\{\mathbf{0},\mathbf{1}\}$ is the label of training set, p(x) is the output of the model, which is the predicted probability of sample x being clicked.

Base models Pooling:

$$\mathrm{e_i} = \mathrm{pooling}(\mathrm{e_{i_1}}, \mathrm{e_{i_2,\ldots,}}\mathrm{e_{i_k}})$$

Equation 3 - Pooling layer for base models

Where $\mathrm{e_i}$ represents a single embedding vector.

Local activation unit:

$$u_U = f(u_A, e_1, e_2, \ldots, e_H) = \sum_{j=1}^{H} a(e_j, u_A)\, e_j = \sum_{j=1}^{H} w_j\, e_j$$

Equation 4 - Local activation unit for DIN model

Where $\{e_1, e_2, \ldots, e_H\}$ is a list of embedding vectors of behaviors of user U with a length of H, u_A is the embedding vector of ad A, a(·) is a feed-forward network with output as the activation weight. With constrains: $\sum_{j=1} Wi$ = 1.

Mini-batch aware regularizer:

Let $\mathbf{W} \in \mathbb{R}^{DxK}$ denote parameters of the whole embedding dictionary, with D as the dimensionality of the embedding vector and K as the dimensionality of feature space.

$$w_j \leftarrow w_j - \eta \left[\frac{1}{|ß_{\boldsymbol{m}}|} \sum_{(x,y) \in ß_{\boldsymbol{m}}} \frac{\partial L(p(x), y)}{\partial w_j} + \lambda \frac{\alpha_{mj}}{nj} w_j \right]$$

Equation 5 - Mini-batch aware regularization formula

Where $w_j \in \mathbb{R}^D$ denotes the j-th embedding vector, nj denotes the number of occurrence for feature id j in all samples, $ß_{\boldsymbol{m}}$ denotes the m-th mini-batch and $\alpha_{mj} = max_{(x,y)} \in ß_{\boldsymbol{m}}\ I(x_j \neq 0)$ if there is at least one instance having the feature id j in mini-batch $ß_{\boldsymbol{m}}$.

Dice activation function:

$$fx{=}ps \cdot s + 1 - psf(x) = p(s) \cdot s + (1 - p(s)) \cdot \alpha s\,,$$

$$p(s) = \frac{1}{1 + e^{-\frac{s - E[s]}{\sqrt{Var[s] + \epsilon}}}}$$

Equation 6 - Dice activation function

Where in the training phase, E[s] and Var[s] is the mean and variance of input in each mini batch. In the testing phase, E[s] andVar[s] is calculated by moving averages E[s] andVar[s] over data. ϵ is a small constant which is set to be 10^{-8}.

B – Implementation

B1 – Hyperbolic Embeddings Approach

<u>Generated Data</u>

As there is no access to data from users, it was necessary to generate a synthetic dataset, with 10.000 users, with specific interests and profiles, selected at random, to replicate the information Modatta's app receives from each user without breaching any privacy regulation set by the GDPR. The following figure shows the generated dataset of the interests of users.

Figure 4 - Sample of the Insights Dataframe

<table>
<tr><th></th><th>Interest</th><th>Interest | Literary arts</th><th>Interest | Performance art</th><th>Interest | Performing arts</th><th>Interest | Science</th><th>Interest | Sports</th><th>Interest | Visual arts</th><th>Community organisation</th><th>Community organisation | Armed forces</th><th>Community organisation | Charity organisation</th><th>...</th><th>Other | TypeAhead | Pet Tricks</th><th>Other | TypeAhead | Public figure type</th><th>Other | TypeAhead | Concentration or Major</th><th>Other | TypeAhead | Degree</th><th>Other | TypeAhead | Work position type</th><th>Other | TypeAhead | Year</th><th>Ur</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td></td></tr>
<tr><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>4</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>8</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>...</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
<tr><td>9</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>...</td><td>-1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td></tr>
</table>

10 rows × 1563 columns

In order to get new insights, we need to have some users with specific interests. In addition, we need to take into consideration the interests of the users:

- The user has a real interest in a category: **1**
- The user does not have an interest (explicit): **-1**

The other categories that aren't filled with 1 or -1, are considered an unknown interest (value = 0), which means that we do not know if the user does have an interest or not.

Data Dictionary for *Insights'Dataframe*

The names of the columns are the Facebook' categories, that englobes the 3 levels of hierarchy. There is a total of 1563 categories. Below, one can find an example of 5 of them.

Table 6 - Example of Facebook Insights

Facebook Insights
Interest
Interest | Performance art
Community organisation | Charity organisation
Other | TypeAhead | Pet Tricks
Other | TypeAhead | Public figure type

Below, Figure 5 represents the dataset that was generated with some profile data for users, containing especially demographic characteristics.

Figure 5 - Sample of the Customers' Dataframe

	user_id	Birth year	Gender	Home country	Civil status	Number of children	Degree	Professional status	Annual income	Practice sports	Smokes
0	0	1936	male	AE	married	1	2	2	45	yes	no
1	1	1919	male	NP	single	2	3	5	70	yes	yes
2	2	1939	male	AD	widow	10000	1	3	70	yes	yes
3	3	1984	female	IS	single	1	0	3	100	yes	yes
4	4	1995	male	MQ	married	10000	0	3	70	no	yes
...	...	...	...	...	...	...	...	...	...	...	...
9995	9995	1948	male	BL	married	1	4	1	70	no	no
9996	9996	1950	non_binary	IO	single	0	2	6	15	yes	yes
9997	9997	1947	male	CN	married	0	2	1	15	no	no
9998	9998	1942	male	AM	single	0	2	2	10000	yes	yes
9999	9999	2017	male	TK	widow	0	4	2	15	yes	yes

Table 7 - Data Dictionary for Customers'Dataframe

Column Name	Description	Values
user_id	Unique identifier of a Modatta's user	0 to 9999
Birth_year	Birth year of the user	1900 to 2019
Gender	Gender of the user	-
Home country	Home country of the user	-
Civil_status	Civil status of the user	-
Number of children	Number of children of the user	-
Degree	Degree of the user	0 to 6
Professional_status	Professional status of the user	0 to 6
Annual_income	Annual income of the user	-
Practice_sports	Binary variable saying if the user practices sports	Yes or No
Smokes	Binary variable saying if the user smokes	Yes or No

Features Data Dictionary

Gender

Value
Male
Female
Non-binary

Civil_status

Value
Divorced
Married
Single
Widow

Degree

Value	Label
0	Other
1	High school diploma
2	Technical degree
3	Bachelor degree
4	Major degree
5	Master degree
6	Doctor degree

Professional_status

Value	Label
0	Unemployed
1	Student
2	Retired
3	Freelancer
4	Fixed term contract
5	Permanent contract
6	Business owner

Annual_income

Value	Label
15	Less than 15000€
30	Between 15000€ and 30000€
45	Between 30000€ and 45000€
70	Between 45000€ and 70000€
100	Between 70000€ and 100000€
10000	More than 100000€

Practice_sports

Value
No
Yes

Number of children

Value	Label
0	No children
1	Has 1 children
2	Has 2 children
1000	Has 3+ children

Smokes

Value
No
Yes

For simplicity purposes, the Home Country is not referred, as it has 249 values. Those values are country names. Below one can find an example of 4 of them.

Value	Label
PT	Portugal
ES	Spain
UK	United Kingdom
US	United States

Poincaré Model: Implementation

Table 8 - Tuples of Categories' Dataframe

Child	Parent
Literary arts	Interest
Performance art	Interest
Performing arts	Interest
Science	Interest
Sports	Interest
Visual arts	Interest
Armed forces	Community organisation
Charity organisation	Community organisation
Country club/Clubhouse	Community organisation
Community group	Community organisation
Environmental conservation organisation	Community organisation
Government organisation	Community organisation
Intergovernmental organisation	Community organisation
Trade union	Community organisation
Political organisation	Community organisation
Political party	Community organisation
Private members club	Community organisation
Religious organisation	Community organisation
Social club	Community organisation
Sorority & fraternity	Community organisation

Table 8 represents the data frame created to be an input for Poincaré Model. Later, it will be transformed into tuples.

Figure 6 - Input for the Poincaré Model

```
tuples

[('Literary arts', 'Interest'),
 ('Performance art', 'Interest'),
 ('Performing arts', 'Interest'),
 ('Science', 'Interest'),
 ('Sports', 'Interest'),
 ('Visual arts', 'Interest'),
 ('Armed forces', 'Community organisation'),
 ('Charity organisation', 'Community organisation'),
 ('Country club/Clubhouse', 'Community organisation'),
 ('Community group', 'Community organisation'),
 ('Environmental conservation organisation', 'Community organisation'),
 ('Government organisation', 'Community organisation'),
 ('Intergovernmental organisation', 'Community organisation'),
 ('Trade union', 'Community organisation'),
 ('Political organisation', 'Community organisation'),
 ('Political party', 'Community organisation'),
 ('Private members club', 'Community organisation'),
 ('Religious organisation', 'Community organisation'),
 ('Social club', 'Community organisation'),
 ('Sorority & fraternity', 'Community organisation'),
 ('Sports club', 'Community organisation'),
 ('Youth organisation', 'Community organisation'),
 ('Art', 'Media'),
```

To implement the Poincaré Model, there was a need to transform the Facebook' categories into tuples, in order to represent the hierarchical structure of the categories. The tuple represented on Figure 6 is composed by (child, parent) nodes.

Figure 7 - Representation of the Poincaré Embeddings in 2D

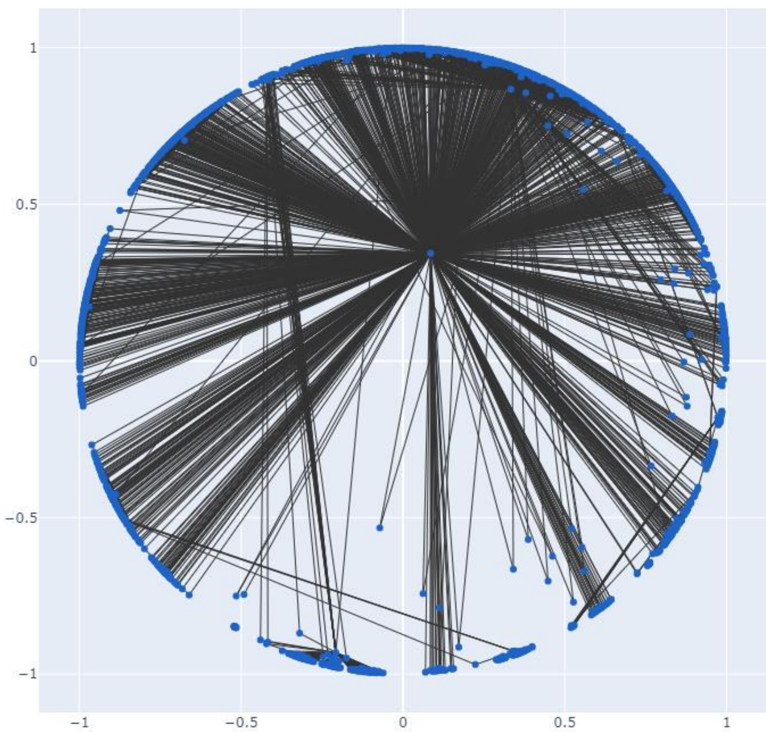


Figure 7 represents all the embeddings resulting from the implementation of the Poincaré Model. Each blue dot represents a category that has a unique representation into the space. The straight black lines represent the relations between the levels of the categories.

The categories that are on the high-level of the hierarchy are laid closer to the center and lower-level categories (child-nodes) shifted to the end of the circumference. Finally, similar categories are put nearer each other and the less similar ones farther away. This representation was created using the parameters of the best model, changing the dimensions from 200 to 2.

Figure 8 - Example of Closer Categories

```
]: print("The category Music is most similar to:")
   model.most_similar('Music')

   The category Music is most similar to:

]: [('Playlist', 1.8499834460025835),
    ('Music award', 1.8929040891332125),
    ('Music chart', 2.446066311528132),
    ('Podcast', 2.5070578937722567),
    ('Musical genre', 2.5715183839839804),
    ('Album', 2.742832353196168),
    ('Record label', 3.3631610546738644),
    ('Music video', 3.461160005308079),
    ('Choir', 3.856084689279786),
    ('Song', 3.873946372133142)]
```

From the Poincaré model, a set of vectors or coordinates of each category on the hyperbolic space, let us check which are the coordinates closer to a specific category. On Figure 8, we used the category *Music* as an example. From the results, one can see that *Playlist* and *Music Award* are the insights closer to this category. This tuple also returns the distance between the category chosen and another category. As we can see, the closest categories mentioned before, have a hyperbolic distance below 2.

Figure 9 - Example of Rank between 2 Categories

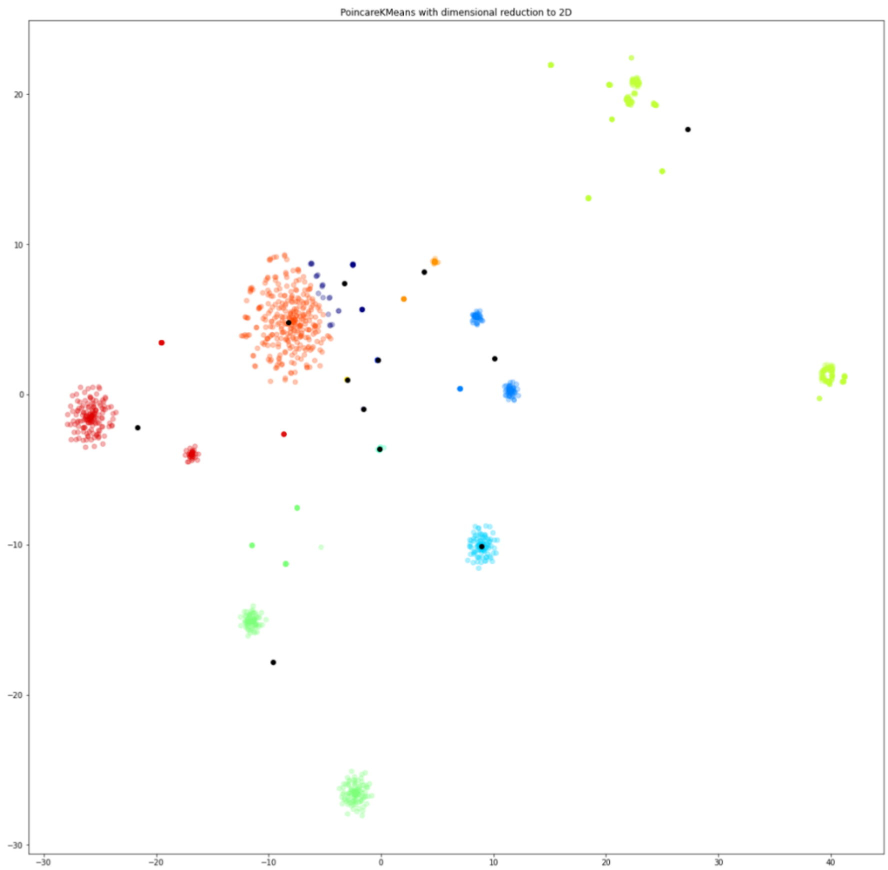
```
[31]: # Rank of distance of node 2 from node 1 in relation to distances of all nodes from node 1
      print("The category Music is at the second position of Song (most similar)")
      model.rank('Song', 'Music')
      # In Song, the Music is at 2st place on the most similar

      The category Music is at the second position of Song (most similar)

[31]: 2
```

Figure 9 shows an example of a rank of distance between 2 categories, that demonstrates their relationship. As an example, we have *Song* and *Music*. The rank function calculates the distance from all categories to the category *Song* and then it will see at each position is *Music*. We can see that it is at the second position, but that doesn't mean that vice versa the same number will be 2. When calculating the rank between *Music* and *Song*, meaning that were calculating the distance from all categories to the category *Music*, *Song* is in 10th.

Figure 10 - PoincareKMeans in 2D



The implementation of the PoincareKMeans, even though, doesn't have a clear influence on our final goal, allowed us to analyze the categories in each cluster. Figure 10 illustrates the grouping of the vectors of each category that was created by implementing the Poincaré Model. Therefore, it shows the different centroids (black dots) and their clusters, that are assigned with a different color. Because the best model has 200D, we used it on our project, but as a mean of visualization, the dimension was reduced to 2D.

Looking at the figure, you will notice some clusters without any category. This may be due to the context of performing computations in the hyperbolic space, which may hold back some centroids to find any category.

New Insights for Users

From the Poincaré Model, it's possible to use the most similar function, aforementioned, that will give us the categories that are most similar to the interests of the user. So, for each category, one can check which are the categories closer, and recommend those for the user. Once again, we're using insights of Emila' data. Below we can find the categories of insights that our real user has showed interest or disinterest in:

Figure 11 - Interests and disinterests of Emila

[50]:

	POSITIVE INSIGHTS	NEGATIVE INSIGHTS
0	Businesses | Education | State school	Community organisation | Armed forces
1	Community organisation	
2	Non-business places	
3	Businesses	
4	Other | Brand | Health/Beauty	
5	Other | Community	
6	Other | Brand	
7	Other | Brand | Electronics	
8	Businesses | Beauty, cosmetic & personal care	
9	Businesses | Shopping & retail | Swimwear shop	
10	Non-business places | Campus building	
11	Businesses | Beauty, cosmetic & personal care ...	
12	Businesses | Arts & entertainment	
13	Businesses | Food & drink | Family-style resta...	
14	Other | Brand | Entertainment website	
15	Businesses | Media/news company	
16	Media | TV & film | TV Programme	
17	Businesses | Sport & recreation	
18	Businesses | Education | University	
19	Businesses | Non-profit organisation	
20	Media | TV & film | TV	
21	Businesses | Shopping & retail | Mobile phone ...	
22	Media | Books & magazines	
23	Other | Brand | Product/Service	
24	Businesses | Education | Higher education	
25	Businesses | Food & drink | Cocktail bar	
26	Community organisation | Social club	
27	Public figure | Comedian	

Table 9 - Predicting new insights

New Insights	Distances
Social club	0.305306
Sorority & fraternity	0.403680
Fashion model	0.552897
Political organisation	0.603168
Gaming video creator	0.684306
Orchestra	0.769347
Sportsperson	0.848823
Public figure	0.881704
Books & magazines	0.883995
Newspaper	0.883995
Visual arts	0.933503
Article	0.959758
Science	0.999553
TV network	1.770001
Other	1.974406
Brand	1.974406
Art	2.040909
Theatrical play	2.158405
Book series	2.192080
Language	2.589027

For getting the new insights, we used a test set from a member of our group, Emila, that contains some positive and negative interests. Table 9 shows the new insights for this user as well as their distances between the category that the user is interested in and the new insight recommended. As we want the categories that are most similar, we got 20 new insights sorted by ascending distances.

Explicit Negative Interests

Notice that, if the user does have a negative interest in some category, the model makes sure that there is an impact on the final new insights, meaning that the negative interests and their similar ones, will not be on the new insights derived from the model.

Imagine we have a user that doesn't like Soccer. We are going to make sure that the insights derived from the model will not get categories that are related with Soccer, but also, be aware that even though this user doesn't like soccer, he can like other type of sports. Not limiting all the choices of Sports (the parent node).

Representation of the User into the Space

Doing an average of the coordinates of each user, allowed to represented each user into the space. This means that, based on the interests of the user, the embeddings of each interest is retrieved and then it’s applied an average on those embeddings, creating one unique vector that will represent each user.

Below one can see the representation of 7000 users in a 2D visualization. For reducing from 200D to 2D, UMap was used as a dimensional reduction technique.

Figure 12 - Representation of the users into the space

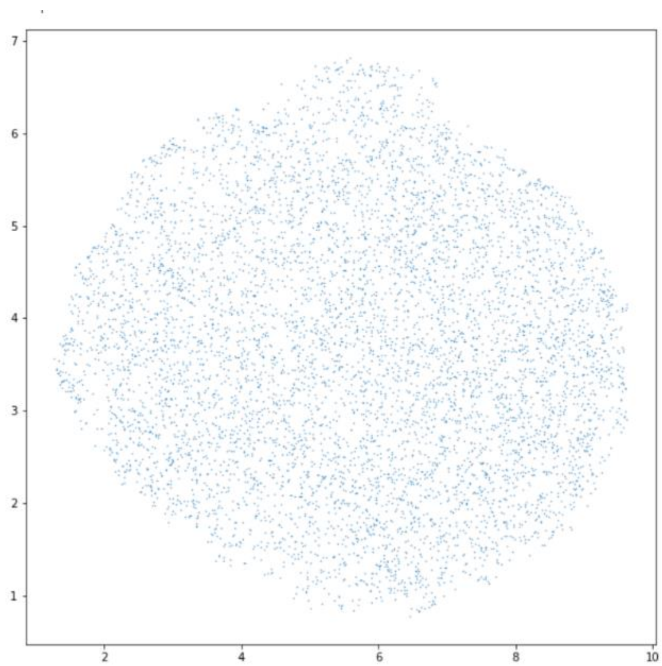


Figure 13 - KMeansPoincare for Representing users

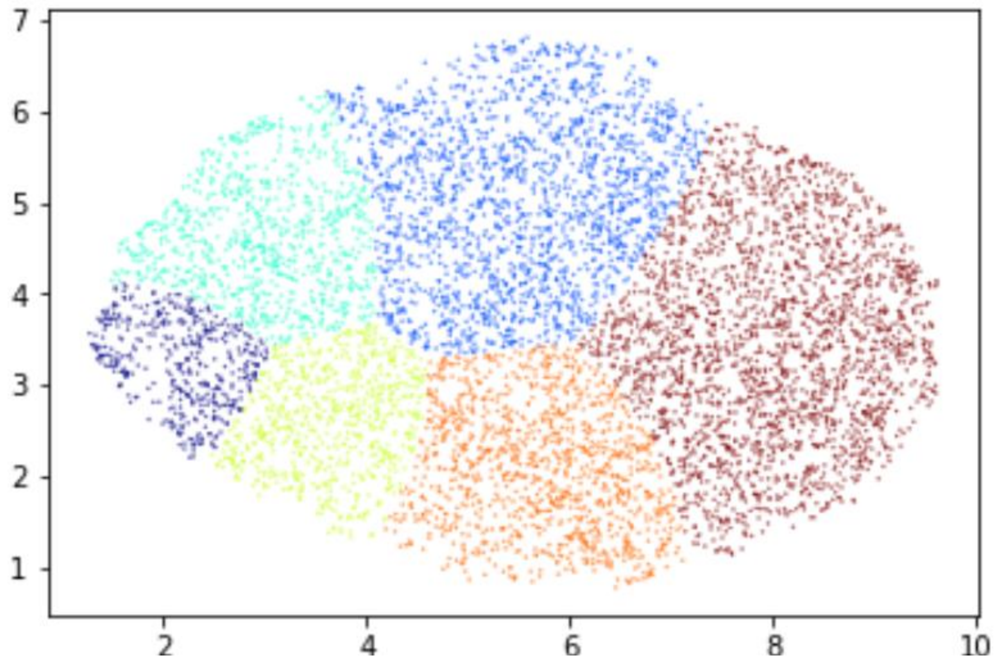


Figure 13 shows 6 target groups created by implementing PoincareKMeans on the representation of the users.

Having target groups, will allow us to suggest to a specific user a certain category that he/she might like, because of the cluster he's in.

In order to understand the target groups, we had to analyze each cluster and understand their characteristics. This was done by going to each user, in a specific cluster, and calculate the insights that appear the most, meaning that the users that are on that cluster are characterized by those attributes.

The following tables identify the main characteristics of each cluster. The target groups are characterized as follows:

Cluster 1

Users in cluster 1 tend to like Real State and Media.

Businesses – Property
Businesses – Media/news company
Businesses – Arts & Entertainment

Cluster 2

Users in cluster 2 are interested in commerce, sports centers, cinema, food and drink and religious places.

Businesses – Commercial & industrial
Businesses – Medical & health
Non-business places - religious
Businesses - Sports & Recreation
Businesses - Food & Drink

Cluster 3

Users in cluster 3 are interested in Education, finance services and telecommunication companies.

Businesses - Education
Businesses - Finance
Businesses - Science, technology & engineering

Cluster 4

Users in cluster 4 like to go to Restaurants, all types of cars, motorcycles and boats. They also like sports.

Businesses - Food & Drink
Businesses - Vehicle, aircraft and boat
Businesses - Sport & recreation
Shopping & Retail

Cluster 5

Users like all types of brands, like to buy all kind of things and travel. Like books and music.

Other - Brand
Businesses - Food & Drink
Businesses - Travel & Transport
Media

Cluster 6

Users in cluster 6 are into Medical and health; Like to shop, and go to restaurants.

Businesses - Medical & health
Shopping & Retail
Businesses - Food & Drink
Sports

In order to evaluate the representation of the user, we predicted the cluster of Emila, as we have been doing before. From the prediction, Emila belongs to cluster 1. We believe that the results indicate to be good, as she is interested on Real State, Media and Entertainment.

Having groups of users together, we are able to identify people more similar to each other. Therefore, it will allow us to suggest to a specific user a certain category that he/she might like, because the users of their cluster (that are similar to he/she), like them too. Below, we have an example of user 1000, where it outputs their closest / similar users.

Figure 14 - Closest Users to user 1000

```
x                1000
[[('User 2598', 0.08168382446355738)],
 [('User 6644', 0.08659737440963623)],
 [('User 5454', 0.08699786909174022)],
 [('User 1414', 0.08776293177995817)],
 [('User 1476', 0.08814171036714673)],
 [('User 4763', 0.09036007364312187)],
 [('User 1497', 0.09059016391621268)],
 [('User 5262', 0.09076536525640845)],
 [('User 6983', 0.09091308487799786)],
 [('User 6160', 0.09126500366380853)],
 [('User 2093', 0.09129561814139267)]]
```

B2 – Deep Interest Network Approach

Generated Data

Insight features

This section comprises data that is related to the insights information that we have access to. **Insights** are particular interests that the users are able to get when connecting their Facebook data with their Modatta app. These insights are particular categorizations of the Facebook pages the users like. Then each insight is assigned with a more general **category**, which will be used later on to find more meaningful recommendations. It is important to note that only each user has control over their own data, and neither Modatta nor the marketeers can access it.

Table 10 - Insight features datasets description

Data	Description
insight_dic	dictionary for the insights data, where each *insight_id* is assigned to its correspondent label
insight_cat_dic	dictionary for the insights' categories data, where each cat_id is assigned to its correspondent label
insight_features_df	dataset with each *insight_id* assigned to its correspondent *cat_id*

User profile features

In this section, synthetic user profile data was created. The following variables were chosen to represent a user's profile: *Birth year*, *Gender*, *Civil status*, *Degree*, *Professional status*, *Annual income*, and *Practice sports*.

Table 11 - User profile features datasets description

Data	Description
labeled_user_profiles	synthetic data created to represent 10 000 users with random profile data. This data frame contains the labeled data
user_profile_df:	*labeled_user_profiles* with the labels transformed into codes, so that it can be easily read by the model later on
labeled_new_profiles	real data that comes from a new profile (in this case, we are using Emila's consented data). This data frame contains the labelled data
new_profiles_df	*labeled_new_profiles* with the labels transformed into codes, so that it can be easily read by the model later on
final_profile_df	*user_profile_df* merged with the *new_profiles_df*

Table 12 - Data Dictionary for final_profile_df

Column Name	Description	Values
user_id	Unique identifier of a Modatta's user	0 to 10000
Birth_year	Birth year of the user coded into bins	0 to 8
Gender	Code for identifying the gender of the user	0 to 2
Civil_status	Code for identifying the civil status of the user	0 to 3
Degree	Code for identifying the degree of the user	0 to 6
Professional_status	Code for identifying the professional status of the user	0 to 6
Annual_income	Code for identifying the annual income of the user	0 to 5
Practice_sports	Binary variable saying if the user practices sports	0, 1

Features Data Dictionary

Birth_year

Value	Label
0	Birth year between 1900 and 1920
1	Birth year between 1921 and 1940
2	Birth year between 1941 and 1960
3	Birth year between 1961 and 1970
4	Birth year between 1971 and 1980
5	Birth year between 1981 and 1990
6	Birth year between 1991 and 2000
7	Birth year between 2001 and 2010
8	Birth year between 2011 and 2021

Gender

Value	Label
0	Male
1	Female
2	Non-binary

Civil_status

Value	Label
0	Divorced
1	Married
2	Single
3	Widow

Degree

Value	Label
0	Other
1	High school diploma
2	Technical degree
3	Bachelor’s degree
4	Major degree
5	Master’s degree
6	Doctor degree

Professional_status

Value	Label
0	Unemployed
1	Student
2	Retired
3	Freelancer
4	Fixed term contract
5	Permanent contract
6	Business owner

Annual_income

Value	Label
0	Less than 15000€
1	Between 15000€ and 30000€
2	Between 30000€ and 45000€
3	Between 45000€ and 70000€
4	Between 70000€ and 100000€
5	More than 100000€

Practice_sports

Value	Label
0	No
1	Yes

User historical behavior features

In this section, the user behaviors dataset, which represents the historical association between the **users** and the **insights**, has been created. When building this data frame, each user (previously created and stored with a unique *user_id*) has some insights associated and a value saying if:

1. the user shows interest in the insight: **True = 1 & False = 0**
2. the user is not interested in the insight: **True = 0 & False = 1**
3. we don't have information about whether the user is interested: **True = 0 & False = 0**

Table 13 - User historical behavior datasets description

Data	Description
user_behaviors_df	synthetic data created to represent 1000000 "behaviors" for the 10000 created users – 1000000 interactions between a user and an insight
new_users_behavior	real data that comes from a new profile (in this case, Emila's data is used). This data frame has information about which insights the new user is interested in, is not interested at all and the insights we don't any have information about for the users
final_behaviors_df	*user_behaviors_df* merged with the *new_users_behavior* data frame

Table 14 - Data Dictionary for final_behaviors_df

Column Name	Description	Values
user_id	Unique identifier of a Modatta's user	0 to 10000
insight_id	Unique identifier of an insight that may be of the user's interest	0 to 1563
True	Binary variable saying if the user is interested in the insight	0, 1
False	Binary variable saying if the user is not interested in the insight	0, 1

Detailed explanation of True / False values

True

Value	Label
0	We may not know if the user is interested in the insight; or the user is not interested at all
1	We know explicitly that the user is really interested in the insight

False

Value	Label
0	We may not know if the user is interested in the insight; or the user is really interested
1	We know explicitly that the user is not interested in the insight at all

Final dataset

Finally, the three datasets – *insight_features_df*, *final_profile_df* and *final_behaviors_df* – are merged into one final data frame – *final_df*. This is how the data that will be inputted to the model looks like:

Figure 15 - Final dataset (all the previous generated final datasets merged)

```
final_df
```

	user_id	insight_id	True	False	Birth_year	Gender	Civil_status	Degree	Professional_status	Annual_income	Practice_sports	c
0	235	1153	1	0	1	1	0	4	2	0	1	
1	6285	1153	1	0	3	0	3	5	4	3	0	
2	431	1153	1	0	5	0	3	5	4	1	0	
3	2244	1153	0	1	1	2	0	1	2	2	0	
4	1214	1153	1	0	0	1	3	4	0	1	0	
...	...	...	...	...	...	...	...	...	...	...	...	
1001559	6690	1444	0	0	2	0	0	5	4	4	1	
1001560	5193	1444	1	0	1	0	3	5	0	5	1	
1001561	4211	1444	1	0	7	0	1	6	1	2	0	
1001562	2878	1444	1	0	8	2	3	0	4	4	1	
1001563	10000	1444	0	0	6	1	2	5	1	0	1	

1001564 rows × 14 columns

Emila’s TRUE and FALSE insight categories

Below we can find the categories of insights that our real user has showed interest or disinterest in:

Table 15 - Interests and disinterests of Emila

TRUE	FALSE
ARTS	ARMED FORCES
BEAUTY AND PERSONAL CARE	
COMEDY	
COMMUNITY SERVICES	
ENTERTAINMENT	
FASHION	
FOOD AND DRINK	
GADGETS	
HEALTH AND FITNESS	
JOURNALISM	
LITERATURE	
NON-PROFIT ORGANIZATION	
Non-business places	
PHOTOGRAPHY	
SCIENCE AND TECHNOLOGY	
SHOPPING AND DEALS	
SPORTS	
TEACHING AND EDUCATION	
TV	

DIN Model Implementation

Individual Recommendation

Table 16 - Recommending Insights

Insight	Category
Businesses | Sport & recreation | Baseball field	SPORTS
Businesses | Local service | Veterinarian	PETS AND ANIMALS
Businesses | Beauty, cosmetic & personal care | Threading service	BEAUTY AND PERSONAL CARE
Businesses | Beauty, cosmetic & personal care | Image consultant	BEAUTY AND PERSONAL CARE
Businesses | Medical & health | STD testing centre	HEALTH AND FITNESS
Public figure | Author	LITERATURE
Businesses | Travel & transport | Private bus service	TRAVEL
Businesses | Shopping & retail | Outdoor equipment shop	SHOPPING AND DEALS
Businesses | Medical & health | Medical device company	HEALTH AND FITNESS
Other | Brand | Website	COMPUTER AND TECHNOLOGY
Businesses | Travel & transport | Tour guide	TRAVEL
Businesses | Food & drink | Hot pot restaurant	FOOD AND DRINK
Businesses | Food & drink | Rajasthani restaurant	FOOD AND DRINK
Businesses | Shopping & retail | Women's clothes shop	SHOPPING AND DEALS
Businesses | Shopping & retail | Rent-to-own shop	SHOPPING AND DEALS
Businesses | Food & drink | Padangnese restaurant	FOOD AND DRINK
Businesses | Sport & recreation | Football pitch	SPORTS
Businesses | Food & drink | Goan restaurant	FOOD AND DRINK

As we are first recommending particular insights that the users may be interested in, when looking at the categories assigned to each of the recommended insights for our real user, one can see that the majority of them fall into categories that the user has already shown interest in (SPORTS, BEAUTY AND PERSONAL CARE, HEALTH AND FITNESS, LITERATURE, SHOPPING AND DEALS, FOOD AND DRINK), while only four insights are assigned with new categories (PETS AND ANIMALS, TRAVEL and COMPUTER AND TECHNOLOGY). This makes sense, as our goal is to find new interests that the user may have and thus it is realistic that these new interests are similar to the ones the user likes.

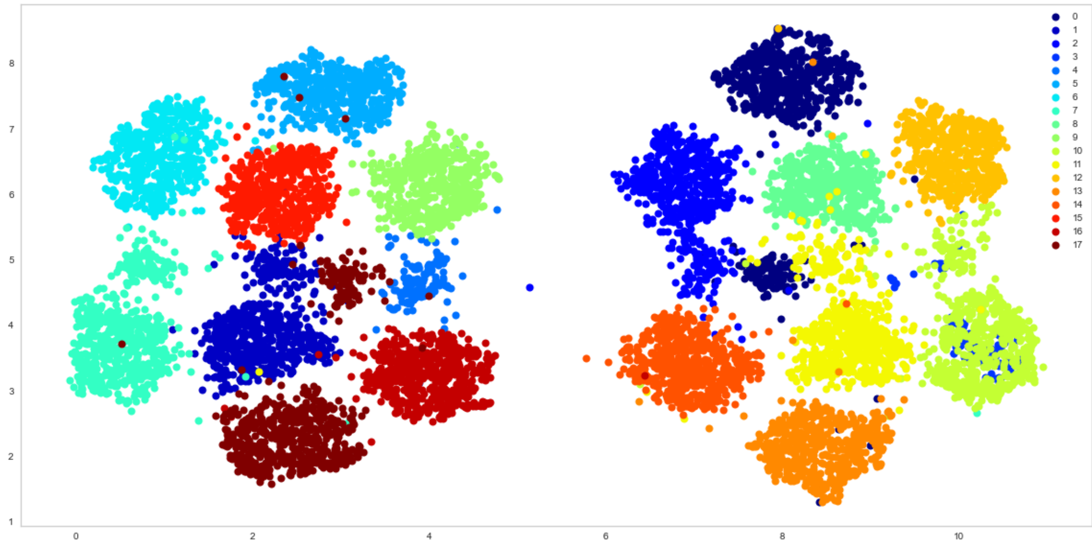
Table 17 - Recommending Categories Insights

Category
COMPUTER AND TECHNOLOGY
POLITICS
HISTORY
PETS AND ANIMALS
FINANCE
GAMING
MOTOR VEHICLES
HOSPITALITY AND TOURISM
MUSIC
ENVIRONMENT
RELIGION
DESIGN
TRAVEL
REAL ESTATE

By grouping the insights per category and trying to find the right categories of insights to recommend, besides the ones the user has already shown interest in, these are the results of applying the model to Emila's data. As expected from the previous recommendations, COMPUTER AND TECHNOLOGY and PETS AND ANIMALS are part of the first categories to be recommended. From the remaining ones, we know that at least FINANCE, MUSIC and TRAVEL would be positively validated by the user concerned, while the others, despite their probable non direct validation, could be hidden interests that the user might explore in more detail when being shown these recommendations by the app.

Clustering Recommendation

Figure 16 - Kmeans Clusters Visualization



These are the clusters resulting from applying the K-Means clustering and UMAP dimensionality reduction techniques. As we can see, the users are grouped into well-defined clusters, with a low dispersion of users between clusters.

An interesting analysis one could do is to look at the distribution of users by their profile features. By analyzing some of the variables, it is possible to note a clear distribution based on three determinant features: users' gender, users' civil status and whether they practice sports or not. Let's plot the clusters distributed by these three features, one by one:

Figure 17 - Distribution by Gender

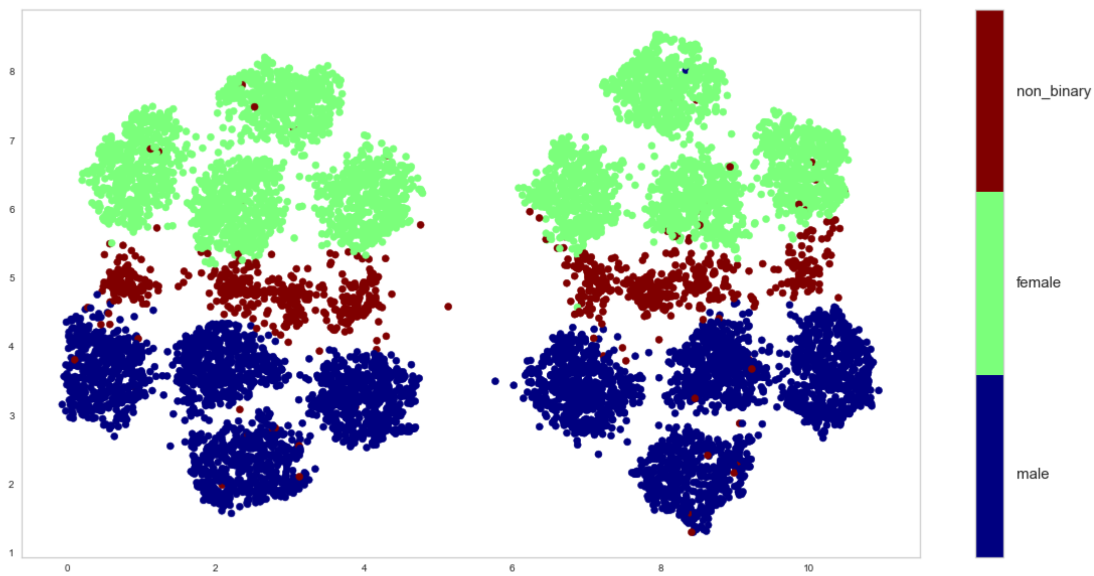


Figure 18 - Distribution by Civil Status

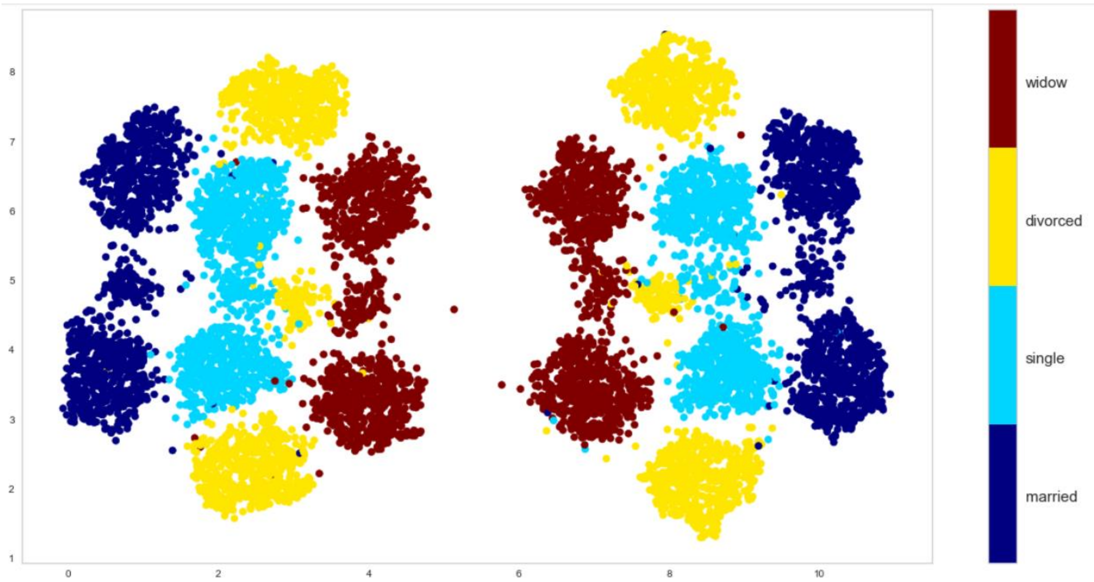
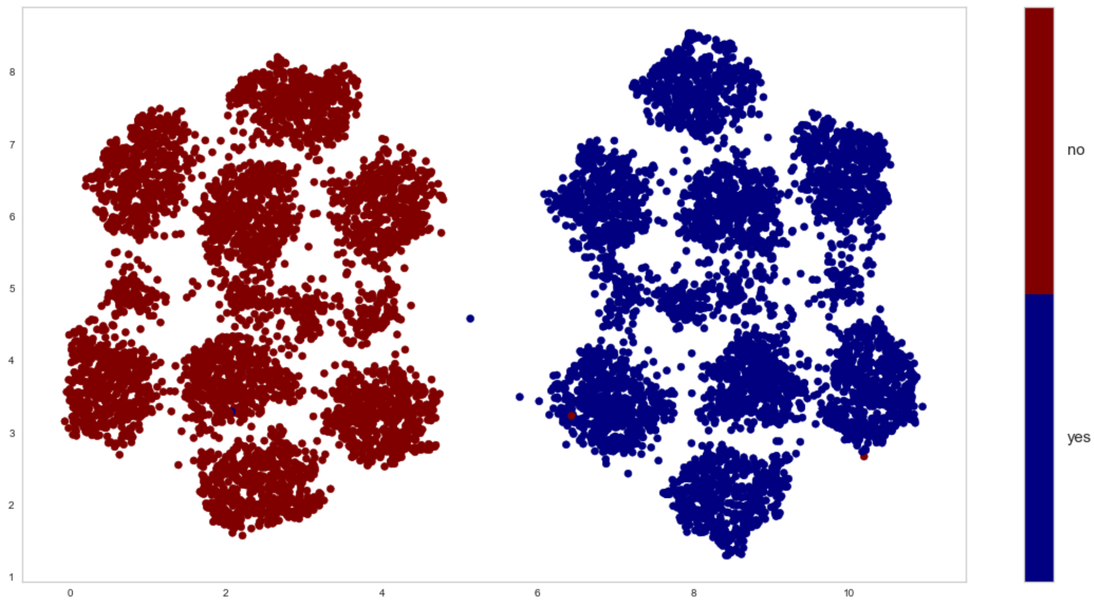


Figure 19 - Distribution by Sports Practice



Concluding, this clusters' division is being highly determined by these three variables, while the others play a secondary role in the groups' definition.

Validating cluster distribution with Emila’s data:

Let’s now check a particular example by comparing our real user’s data with the distribution of the cluster she is inserted in.

Table 18 – Emila’s profile data

Value	Label
Birth_year_6	1991 – 2000
Gender_1	Female
Civil_status_2	Single
Degree_5	Master’s Degree
Professional_status_1	Student
Annual_income_0	Less than 15000€
Practice_sports_1	Yes

Figure 20 - Emila’s cluster distribution

	distribution
Birth_year_2	0.1775
Birth_year_1	0.1618
Birth_year_0	0.1596
Birth_year_3	0.0899
Birth_year_5	0.0876
Birth_year_7	0.0854
Birth_year_8	0.0809
Birth_year_4	0.0787
Birth_year_6	0.0787

	distribution
Civil_status_2	1.0
Civil_status_0	-0.0
Civil_status_1	-0.0
Civil_status_3	0.0

	distribution
Gender_1	1.0
Gender_0	0.0
Gender_2	-0.0

	distribution
Degree_0	0.1820
Degree_3	0.1461
Degree_2	0.1438
Degree_5	0.1393
Degree_6	0.1371
Degree_4	0.1303
Degree_1	0.1213

	distribution
Practice_sports_1	1.0
Practice_sports_0	0.0

	distribution
Professional_status_4	0.1551
Professional_status_5	0.1528
Professional_status_2	0.1483
Professional_status_6	0.1438
Professional_status_0	0.1371
Professional_status_1	0.1348
Professional_status_3	0.1281

	distribution
Annual_income_3	0.1865
Annual_income_2	0.1820
Annual_income_1	0.1730
Annual_income_0	0.1596
Annual_income_4	0.1551
Annual_income_5	0.1438

As we had already concluded, Gender, Civil status and Practice sports are the determinant factors in the cluster definition. For the other variables the distribution is quite disperse, and we can even see that Emila is in the group with the lowest % for birth year.

Final Recommender System

A final Recommender System was built based on the second approach, so that we could take advantage of the information that is gained by considering the similarity between users to represent them into clusters. Thus, as for the first approach, we have recommendations for both particular insights and categories of insights:

Table 19 - Recommending Insights

Insight	Category
Businesses | Vehicle, aircraft and boat | Aviation repair station	AVIATION
Other | TypeAhead | Coming of age	Other
Businesses | Shopping & retail | Rent-to-own shop	SHOPPING AND DEALS
Businesses | Food & drink | Xinjiang restaurant	FOOD AND DRINK
Non-business places | Outdoor recreation | Pond	Non-business places
Other | TypeAhead | Holiday	TRAVEL AND VACATIONS
Businesses | Local service | Personal assistant	PERSONAL IMPROVEMENT
Other | Brand | Musical instrument	MUSIC
Businesses | Hotel & B&B	HOSPITALITY AND TOURISM
Other | Brand | Society & culture website	CULTURE
Businesses | Food & drink | Asian restaurant	FOOD AND DRINK
Businesses | Vehicle, aircraft and boat | Marine supply shop	MOTOR VEHICLES
Businesses | Local service | Animal shelter	PETS AND ANIMALS
Businesses | Arts & entertainment | Decorative arts museum	DESIGN
Businesses | Shopping & retail | Fishing shop	SHOPPING AND DEALS

By comparing the insights obtained with the individual recommendation with these insights, it is notable that the categories of the latest are way more diverse and different from the ones that Emila has already shown interest in. This is because we are now trying to recommend insights not only based on the user's personal interests but on his/her peers' interests, with peers being the most similar users that are inserted in the same cluster. This is an interesting result since our goal is exactly to try to predict possible hidden insights that the users may like, so even though at first the users might find it intriguing to be recommended with insights they had never considered to like, they might end up discovering new insights that they are really interested in.

Table 20 - Recommending Categories of Insights

Category
AGRICULTURE
MUSIC
REAL ESTATE
HISTORY
MOTOR VEHICLES
DESIGN
HOSPITALITY AND TOURISM
FINANCE
AVIATION
RELIGION
ENVIRONMENT
COMPUTER AND TECHNOLOGY
TRAVEL
PETS AND ANIMALS
MOVIES

Regarding the recommendation of categories of insights, the results are very similar to the ones obtained with the individual recommendation. The main difference is the order of recommendations – for instance, COMPUTER AND TECHNOLOGY was the first category to be recommended individually, however it is now the 12th; the same for MUSIC, which was previously the 9th and now it’s the second most likely to be of the user’s interest.

C – Evaluation

C1 – Hyperbolic Embeddings Approach

<u>Poincaré Model: Evaluation</u>

Figure 21 - Fine-tuning of parameters on Poincaré Model

```
negatives_array=[10, 20]
dimensions = [5, 25, 50, 100, 200]
epochs_array = [50, 100, 150, 200]
burn_in_array = [0, 10]
regularization_coeff = [0, 1]

# Try the different values of the parameters;

for neg in negatives_array:
    for di in dimensions:
        for epo in epochs_array:
            for burn in burn_in_array:
                for reg in regularization_coeff:
                    model = Model(negative = neg, size = di, epochs = epo, burn_in=burn, regularization_coeff=reg )
```

On our code, we created a function called *Model* implementing the Poincaré Model, based on some parameters that will be evaluated in order to get the best model possible.

In this case, we are going to fine tuning the following:

- **<u>Negative:</u>** Number of negative samples to use.
- **<u>Size</u>**: Number of dimensions of the trained model
- **<u>Burn</u>-<u>in</u>**: Number of epochs to use for burn-in initialization (0 means no burn-in)
- **<u>Regularization_coeff</u>**: Coefficient used for l2-regularization while training (0 effectively disables regularization)
- **<u>Epochs:</u>** hyperparameter that controls the number of complete passes through the training dataset.

To get the best model, we are going to fine tuning the parameters described above. The values for the parameters are the following:

- **Negatives:** 10, 20;
- **Dimensions:** 5, 25, 50, 100, 200;
- **Epochs:** 50, 100, 150, 200;
- **Burn-in:** 0, 10;
- **Regularization coeff:** 0, 1;

Figure 21 illustrates the code that was built to apply these parameters. In Figure 22, it is shown the best model hyperparameters.

Figure 22 - Best Model Parameters

After analyzing, the choosen model was `200D_20N_0B_0R_200Ep` which has:

- `20 negatives`
- `0 burn-in` ;
- `0 regularization` ;
- `200 Epochs` ;
- `200 Dimensions` ;

For evaluation on Link Prediction, it was required a test set. This test set was transformed into tuples, as explained before.

To check the results on real data, we used data from one of our members, Emila. Table 21 shows the tuples of Emila's data.

This data allowed us to evaluate the generalization performance of the Poincaré Model.

Table 21 - Tuples of Emila's data

Child	Parent
Books & magazines	Media
TV & film	Media
Comedian	Public figure
Brand	Other
Media/news company	Businesses
Education	Businesses
Non-profit organisation	Businesses
Community	Other
Just for fun	Other
Arts & entertainment	Businesses
Beauty, cosmetic & personal care	Businesses
Youth organisation	Community organisation
Local service	Businesses
Sport & recreation	Businesses
Food & drink	Businesses
Shopping & retail	Businesses
Social club	Community organisation
Campus building	Non-business places
Video creator	Public figure
Magazine	Books & magazines

Recommender System: Implementation and Evaluation

Figure 23 - Parameters used on HyperML

```
training_config = {
    "manifold"            : PoincareBall(),  # The manifold to be used in the model
    "init_curvature"      : -1.0,  # Default value of the curvature of the manifold
    "trainable_curvature" : True,  # Whether the curvatures are trainable

    "training_steps"      : 1000,  # Steps for training
    "log_steps"           : 100,  # Log intervals during training
    "eval_steps"          : 100,  # Evaluation intervals during training

    "batch_size"          : 60,  # Batch size used in training
    "learning_rate"       : 0.001,  # Learning rate of training
    "optimizer"           : RAdam,  # The optimizer to be used in training

    "dim"                 : 25, # Dimension of the embedding tables
    "margin"              : 1.0,
    "candidate_num"       : 10,  # Sampled negative candidates for evaluation
    "K"                   : 10,  # Metric of HR@K

    "gamma"               : 0.1,  # Multi-task learning rate between the two types of loss
    "epsilon"             : 1e-9,  # Small value for avoiding dividing zeros in distortion optimization (Loss)

    "embedding_initializer": tf.initializers.glorot_uniform(),  # Initializer for the embedding tables
    "shuffle_buffer"      : 10000,  # Buffer size for TF's shuffling over the dataset
}
```

Figure 23 illustrates the parameters used on the implementation of HyperML.

The final HR@10 result was of 0.77, which demonstrated to be more desirable given previous results.

Table 22 - Top 10 Recommendations of Insights

Top 10 item Recommendations
Businesses | Food & drink | Scandinavian restaurant
Businesses | Food & drink | Café
Businesses | Arts & entertainment | Museum
Businesses | Shopping & retail | Beauty supply shop
Businesses | Food & drink | Ice cream shop
Businesses | Arts & entertainment | Amusement & theme park
Public figure | Talent agent
Businesses | Sport & recreation | Volleyball court
Businesses | Vehicle, aircraft and boat | Wheel & rim repair service
Businesses | Travel & transport | Private bus service

Table 23 - Recommendations of Master Categories

Master Categories Recommendations
FOOD AND DRINK
TEACHING AND EDUCATION
BEAUTY AND PERSONAL CARE
TRAVEL
AVIATION
GAMING
MUSIC
FASHION
WINE
ENTERTAINMENT
THEATRE
POLITICS
TV
PUBLIC FIGURE
JOURNALISM
ARTS
PETS AND ANIMALS
LITERATURE
MOVIES
MOTOR VEHICLES
SPORTS

Tables 22 and 23 shows the results of the Recommender System for Emila's test set. The final results were presented as insights, as well as master categories.

The use of master categories made sense as a way to obtain new insights of users in broader terms, which can later benefit organizations when looking for new users that have some specific but wide-ranging characteristics.

C2 – Deep Interest Network Approach

DIN model parameters and evaluation

After trying to overcome the overfitting the model was suffering from, we have decided to keep the following parameters for the best model:

- dnn_use_bn = True
- dnn_hidden_units = (200, 80)
- dnn_activation = "relu"
- att_hidden_size = (40, 20)
- att_activation = "dice"
- att_weight_normalization = False
- l2_reg_dnn = 0.01
- l2_reg_embedding = 1e-6
- dnn_dropout = 0.03
- seed = 1024
- task = "binary"

Table 24 - Best DIN model Evaluation

Metric	Training set	Test set
Log loss	0.619	0.620
AUC	0.547	0.545
Recall	0.878	0.876

As we can see from the above, the model is not overfitting and the recall score is pretty good, meaning that the model is correctly predicting the majority (88%) of insights the users are interested in.

When retraining the model for the clustering recommendation, the results were very similar, showing a Log loss of 0.543 on the training set and 0.569 on the test set, and a recall of 0.852 and 0.849 on the training and test sets, respectively.

D – Targeting requirements and outcomes from the targeting process

For an effective campaign, Vertbaudet would like to target users with the following characteristics:

- Profile distribution
  - Number of children higher than 0
- Interests
  - Babies and children
  - Fashion
  - Shopping and deals

Considering these requirements, from a sample of 10000 users, 2833 were selected to be targeted, from which 798 accepted the offer and 551 completed it until the end (248 accepted but then didn't complete the offer).

Figure 24 – Users' distribution outcomes from the targeting process

	targeted	accepted	completed
0	7168	9203	9450
1	2833	798	551

From here, the evaluation process starts by assessing the distributions of each group of users and taking some key points from the analysis.

E – Users' profile distribution detailed results

By analyzing the users' profile distributions, there are three main profile features that deserve show considerable differences between the three groups of users: Birth year, Gender and Civil Status. Therefore, the following tables show the distribution of each of these profile features for each group of users – targeted, accepted and completed.

<u>Targeted users</u>

Figure 25 - Profile distribution for the targeted users

	distribution
Birth_year_<=6	0.829156
Birth_year_0	0.175785
Birth_year_>6	0.170844
Birth_year_1	0.169432
Birth_year_2	0.156018
Birth_year_8	0.095658
Birth_year_4	0.087540
Birth_year_3	0.086128
Birth_year_5	0.080127
Birth_year_7	0.075185
Birth_year_6	0.074126

	distribution
Gender_1	0.456054
Gender_0	0.451112
Gender_2	0.092834

	distribution
Civil_status_0	0.255912
Civil_status_3	0.254501
Civil_status_2	0.248853
Civil_status_1	0.240734

Users who have accepted the offer

Figure 26 - Profile distribution of users who accepted the offer

	distribution
Birth_year_<=6	0.921053
Birth_year_0	0.234336
Birth_year_1	0.162907
Birth_year_2	0.155388
Birth_year_3	0.107769
Birth_year_5	0.090226
Birth_year_4	0.088972
Birth_year_6	0.081454
Birth_year_>6	0.078947
Birth_year_8	0.041353
Birth_year_7	0.037594

	distribution
Gender_1	0.761905
Gender_0	0.195489
Gender_2	0.042607

	distribution
Civil_status_3	0.324561
Civil_status_0	0.304511
Civil_status_1	0.273183
Civil_status_2	0.097744

Users who have completed the offer

Figure 27 - Profile distribution of users who completed the offer

	distribution
Birth_year_<=6	0.952813
Birth_year_0	0.241379
Birth_year_1	0.181488
Birth_year_2	0.161525
Birth_year_3	0.107078
Birth_year_6	0.092559
Birth_year_4	0.085299
Birth_year_5	0.083485
Birth_year_>6	0.047187
Birth_year_7	0.023593
Birth_year_8	0.023593

	distribution
Gender_1	0.831216
Gender_0	0.147005
Gender_2	0.021779

	distribution
Civil_status_3	0.341198
Civil_status_0	0.304900
Civil_status_1	0.279492
Civil_status_2	0.074410

For a better understanding of the results presented above, check the data labelling of the values of each profile feature:

Birth year

- Birth_year_0: born between 1900 and 1920
- Birth_year_1: born between 1921 and 1940
- Birth_year_2: born between 1941 and 1960
- Birth_year_3: born between 1961 and 1970
- Birth_year_4: born between 1971 and 1980
- Birth_year_5: born between 1981 and 1990
- Birth_year_6: born between 1991 and 2000
- Birth_year_7: born between 2001 and 2010
- Birth_year_8: born between 2011 and 2021
- Aggregate measures:
  - Birth_year_>6: born after 2000 (aged less than 21)
  - Birth_year_<=6: born in 2000 or before (aged 21 or more)

Gender

- Gender_0: male
- Gender_1: female
- Gender_2: non-binary

Civil status

- Civil_status_0: divorced
- Civil_status_1: married
- Civil_status_2: single
- Civil_status_3: widow

F – Retargeting with users' profile distribution improvements

The first retargeting process was done based on the insights taken from the users' profile distribution. The three features Birth year, Gender and Civil status will be added to the target requirements, separately, and the results will be evaluated.

Birth year

Figure 28 - Users' distribution outcomes from the retargeting process with Birth year

	targeted	accepted	completed
0	7652	9226	9453
1	2349	775	548

By adding the condition of users being born before 2001 (aged above 20), the following outcomes are obtained: From the 10000 users in the sample, 2349 were targeted, 775 accepted the offer and 548 completed it until the end (227 accepted the offer but didn't complete it). As such, the Conversion Rate is increased to:

CvR = 100 * 548 / 2349 = **23.33%**

Gender

Figure 29 - Users' distribution outcomes from the retargeting process with Gender

	targeted	accepted	completed
0	8709	9471	9608
1	1292	530	393

Now let's add to the original target requirements the condition of users being female. From the 10000 users in the sample, 1292 were targeted, 530 accepted the offer and 393 completed it until the end. Although the Conversion Rate is increased, as presented below, only a small subset of users is targeted and probably the campaign budget is not achieved, so it is not that beneficial to target users based on gender in this case. The gender condition could be used to restrict the target audience if many users were being targeted inadequately.

CvR = 100 * 393 / 1292 = **30.42%**

Civil status

Figure 30 - Users' distribution outcomes from the retargeting process with Civil status

	targeted	accepted	completed
0	7873	9245	9463
1	2128	756	538

When the condition of users not being single is added to the original target requirements, the following outcomes are obtained: From the 10000 users in the sample, 2128 were targeted, 756 accepted the offer and 538 completed it until the end. The Conversion Rate is increased, as follows:

CvR = 100 * 538 / 2128 = **25.28%**

Overall, all these conditions based on the profile distribution could be used to improve the targeting strategy, perhaps even combined in such a way that the target audience didn't become too restrictive (for example, choosing a base set of users to target considering the original requirements of both profile and interests, and then adding some more users that fitted the extra profile requirements).

G – Retargeting with users' interests improvements

Figure 31 - Users' distribution outcomes from the retargeting process with interest Design

	targeted	accepted	completed
0	6762	9026	9203
1	3239	975	798

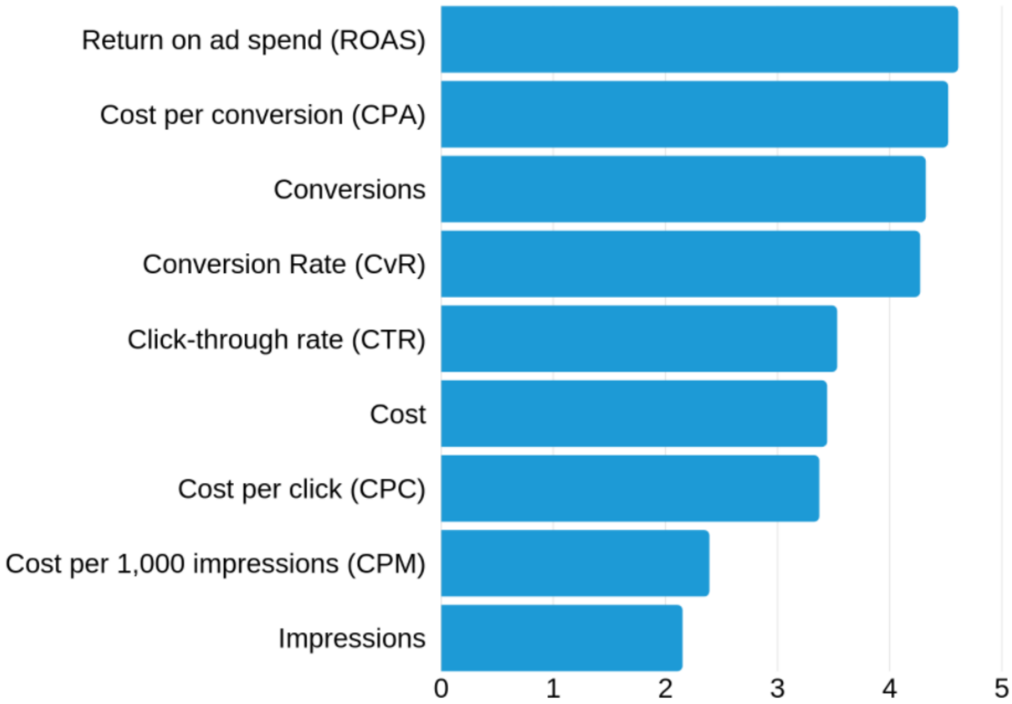
Now considering the original requirements for targeting, and adding more users based on their interest in Design, Figure X shows the outcomes from this new strategy. From a sample of 10000 users, 3239 were selected to be targeted, from which 975 accepted the offer (but not necessary completed it) and 798 completed the offer until the end. As such, the Conversion Rate is increased to:

CvR = 100 * 798 / 3239 = **24.64%**

H – Most important KPIs for evaluating ad effectiveness

According to a study conducted by databox, the following were considered by 71 marketers the most important metrics to evaluate the effectiveness of paid advertising:

Figure 32 - Most important KPIs for evaluating paid ad effectiveness



As such, and considering the context of this project, ROAS, CPA and CvR are the chosen KPIs to proceed with the evaluation of campaign effectiveness.

I – CvR (Conversion Rate)

CvR is very important to determine the number of users to be targeted with the offer. So, in our case study, if Vertbaudet has a budget of 3000€, for example, to spend with this campaign, and is paying an average value of 3€ for every user that completes the offer, the goal is that in the end 1000 users complete the offer successfully. However, as only a some of the targeted users will complete the offer, one should try to find out this rate so that the company's goal for this campaign is successfully achieved.

With the Conversion Rate being equal to 551 / 2833 = 19.45%, for a future similar campaign that Vertbaudet wants to conduct with Modatta, it is expected that 19.45% of the targeted users complete the offer. So, if 1000 users are intended to complete the action, around 5140 (1000 / 0.1945) users should be targeted.

J – Average value spent

In order to calculate the Cost Per Action, the following data was generated to know how much users are spending, on average, on Vertbaudet online store:

Figure 33 - Sample of purchases made by users who completed the offer at Vertbaudet store

	user_id	purchase
1	1	28
12	12	58
14	14	20
26	26	26
28	28	48
...	...	...
9930	9930	15
9944	9944	42
9971	9971	53
9974	9974	16
9983	9983	55

551 rows × 2 columns

```
average_value_spent = np.mean(purchases["purchase"])
average_value_spent
```

```
35.82940108892922
```

The average value spent resulting from this data was 35.8€, meaning that the CPA is equal to 0.10 * 35.8 = **3.58€**.

K – ROAS (Return on Ad Spend)

As referred, ROAS is obtained by dividing the total revenue resulting from the campaign by the total cost the company incurs in during the campaign:

ROAS = total_revenue / total_cost

The total revenue is obtained by multiplying the average order value by the number of users who have completed the offer, and the total cost corresponds to 10% (cashback defined by the company) of that value, which results in the following:

ROAS = 35.8*551 / 0.1*35.8*551 = **10**

Actually, if the total cost is depending on the total revenue like in this campaign, being a cashback of the value spent by the users, ROAS is not influenced by neither the number of users who completed the offer nor the average order value, as it only depends on the defined cashback rate:

ROAS = total_revenue / (cashback * total_revenue)

ROAS = 1 / cashback

L – Comparison with Facebook and Google KPIs

Conversion Rate (CvR)

As it can be seen below, Facebook's average CvR is 9.21%, and Google's average CvR is 3.75% on the search network and 0.77% on the display network. For the particular case of Vertbaudet, which is inserted in the Apparel industry, the expected average CvR would be 4.11% for Facebook, 1.96% for Google search, and 1% for Google display advertising. All these values are considerably lower than the obtained Conversion Rate of the campaign under analysis in this study.

Figure 34 - Facebook's Average Conversion Rate

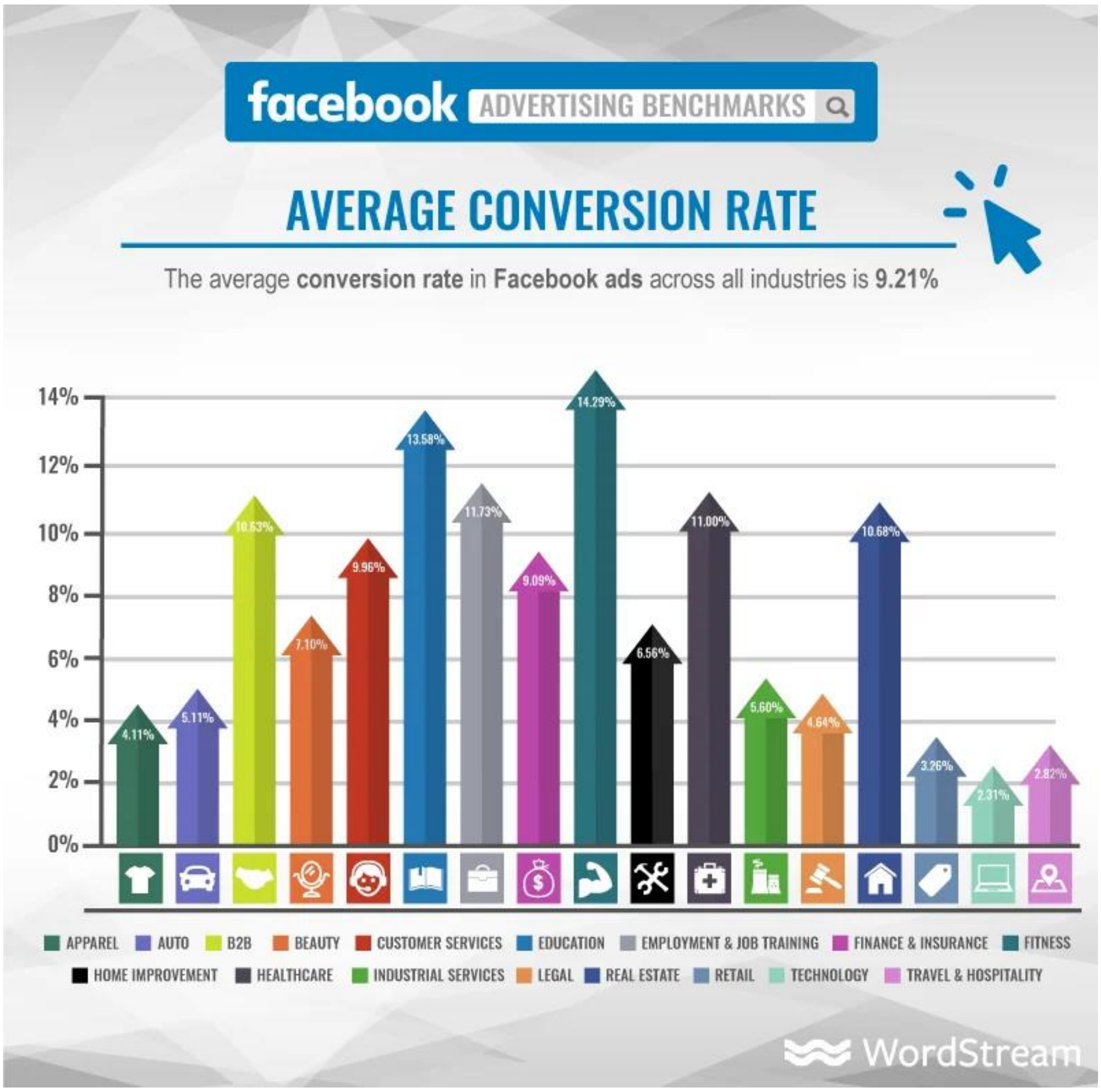
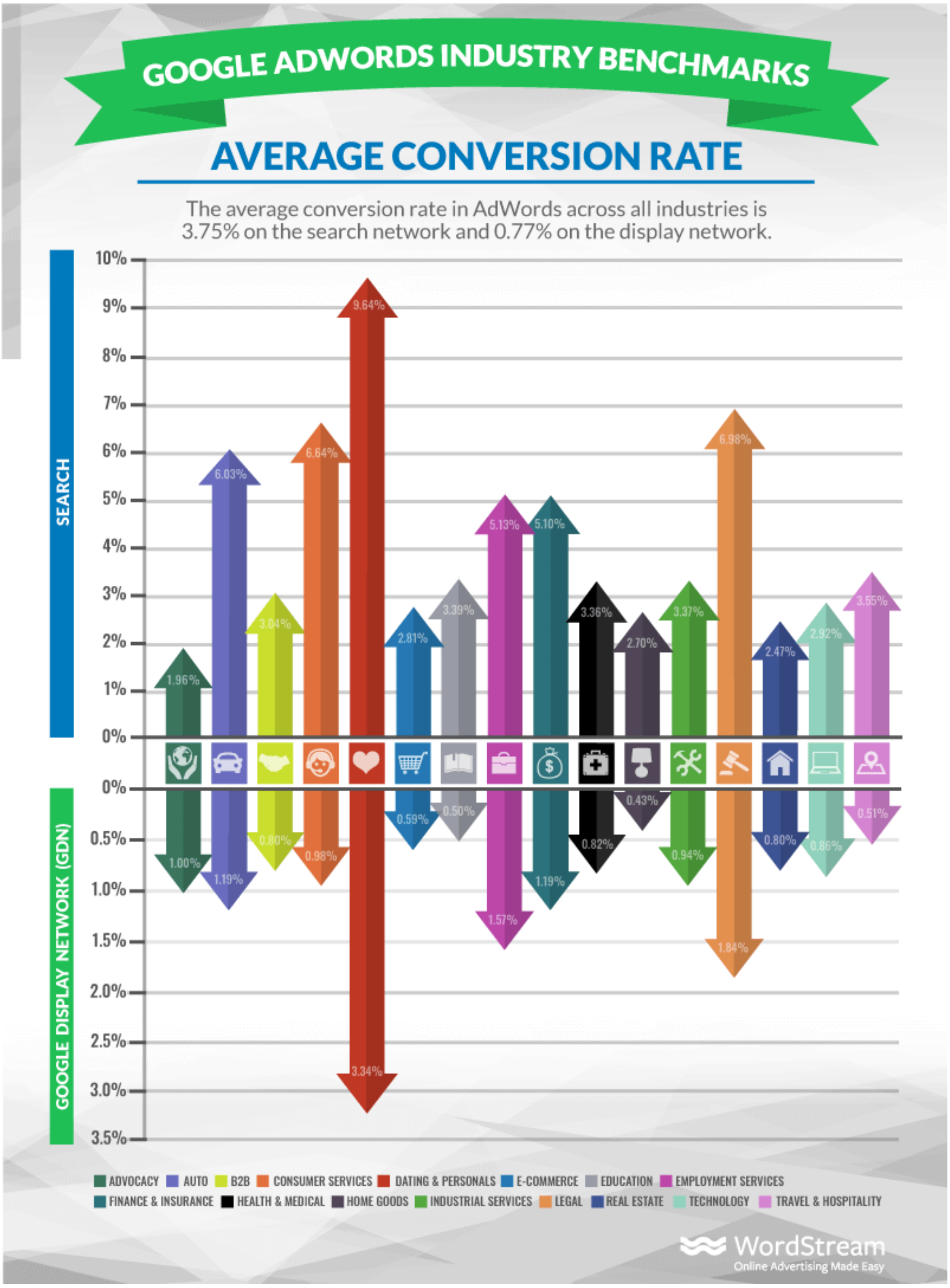


Figure 35 - Google's Average Conversion Rate



Cost per Action (CPA)

Similarly, the average cost per action of both Facebook and Google are presented below. Facebook ads have an average CPA of $18.68, being this value reduced to $10.98 when it accounts only for the apparel industry. Google's average CPA is $48.96 for the search network (being almost doubled when considering only the apparel industry – $96.55) and $75.51 for the display network (decreasing to $70.69 for the apparel industry).

Again, these values are extremely higher when comparing to this paper's campaign performance, however it is important to note that in the campaign analyzed in this study the CPA is highly dependent on the average value spent per customers at Vertbaudet online store, thus a higher value spent would originate a proportional higher CPA.

Figure 36 - Facebook's Average Cost per Action

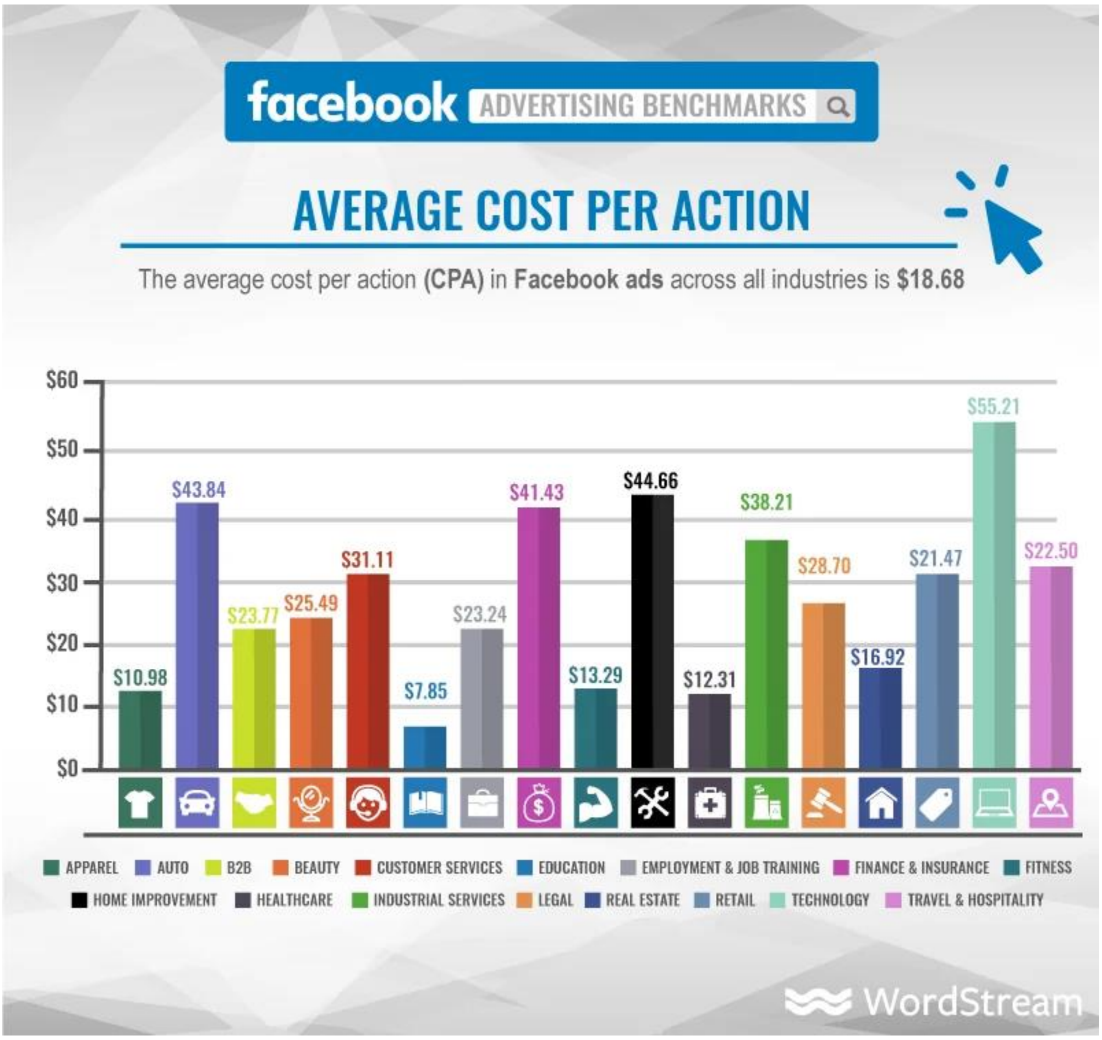


Figure 37 - Google's Average Cost per Action