Innovating Sales and Planning of Complex Industrial Products
Exploiting Artificial Intelligence

Deliverable 3.4
Customer Segmentation

Deliverable type:	Software
Deliverable reference number:	ITEA 20054 | D3.4
Related Work Package:	WP 3. Algorithm design, implementation and test of InnoSale knowledge management components
Due date:	2024-02-29
Actual submission date:	2024-06-12
Responsible organization:	VTT
Editor:	Tuomas Sormunen, Sari Järvinen
Dissemination level:	Public
Revision:	Final | Version 1.0

Abstract:	This deliverable explains the functionalities of the Customer Segmentation Subcomponent, developed mainly by VTT. It briefly visits the State-of-the-Art in customer segmentation, places the Subcomponent in the InnoSale framework, lists the requirements and showcases an example of the functionalities using an open dataset.

Keywords:	Customer Segmentation, Subcomponent, Inference Engine

Table_head	Name 1 (partner)	Name 2 (partner)	Approval date (1 / 2)
Approval at WP level	VECTOR	TUD	28.5.2024
Veto Review by All Partners			11.6.2024

Editor
Tuomas Sormunen (VTT)
Sari Järvinen (VTT)

Contributors
Tuomas Sormunen (VTT)
Sari Järvinen (VTT)
Kai Huittinen (Wapice)
Arttu Lämsä (VTT)
Jussi Liikka (VTT)

Executive Summary

This document outlines the Customer Segmentation Subcomponent developed primarily by VTT within the InnoSale project. In the future, it integrates into the InnoSale solution framework's Inference Engine, utilizing data from existing IT systems, notably CRM software. Derived from InnoSale WP1 and WP2, it calculates customer similarity, employs evolutionary clustering, and allows user-defined dynamic weights. A proof-of-concept demonstrates its efficacy in predicting offer outcomes, with experiments on industry data showing potential for guiding sales and marketing strategies.

Table of Content

Figures

Tables

1 Introduction

Customer segmentation is the process of dividing the customer base to subsets based on common features. In the case of data-driven tools, this segmentation requires accurate data about the historical, present, and potential customers, which is often stored in Customer Relationship Management (CRM) systems. The most typical ways to segment the customer base is to use simple features such as the domain of the client as well as their geographical location. However, segmentation can also consider various other features that are contained in the CRM systems.

The B2B (business-to-business) domain is more challenging compared to the B2C (business-to-consumer) domain in many ways, but the main issue is often the amount of data. A company directed to the B2C domain can consider singular customers as data points, which often constitute a database of thousands or millions of people. In the B2B domain, the number of customers is often much lower, and in the case of complex technical configurable products, the customer base is reduced even more. Thus, data-driven knowledge discovery is of key relevance, and customer segmentation plays an important role in directing the sales and marketing of a company.

This document outlines the work done in the InnoSale project pertaining to the Customer Segmentation Subcomponent, developed by VTT. It is intended as a background for the software deliverable D3.4., which contains the required features for the Subcomponent.

2 Short recap of the state-of-the-art in B2B customer segmentation

Deliverable 1.3. enlisted, among other things, the commercial and scientific state-of-the-art in B2B customer segmentation. In this section a brief recapitulation is given.

2.1 Scientific state-of-the-art

Business-to-business customer segmentation approaches are generally divided into five categories:

1. Firmographics, i.e., business version of demographics,
2. Needs-based,
3. Behaviour-based,
4. Profitability-based segmentation,
5. Customer sophistication.

These are independent of the *target* of customer segmentation, which can vary substantially between different use cases.

Data-driven B2B customer segmentation has seen relatively little publication activities as compared to the business-to-customer domain. This is most likely the result of unavailability of B2B datasets. Nonetheless, based on the literature survey, the customer segmentation has been used for the following tasks:

1. Predict the win/loss of the sales opportunity,
2. Evaluate customer loyalty/churn,

3. Rank the potentiality of customer leads, and
4. Predict the profitability of customer.

Most of the used machine learning models in the literature are classical clustering, regression, and classification models. The datasets are often related to CRM and transaction data.

2.2 Tools for customer segmentation

B2B domain is characterized by the customers looking for more long-term relations between the supplier, as compared to the B2C domain. Moreover, the amount of effort required for converting leads to clients is considerably higher. Thus, customer segmentation is a tool for ranking potential clients based on their conversion probability, allowing for minimizing the sales costs and increasing the ROI (return on investment).

Customer segmentation functionality can be found in major CRM systems. According to their marketing material, SAP, Microsoft Power BI, Salesforce, and Oracle offer this functionality. SAP offers the often-used recency, frequency, and monetary value (RFM) analysis, as well as rudimentary machine learning algorithms. Power BI enables manual customer segmentation, as well as clustering algorithms, even custom ones. Salesforce enables market segmentation via their Salesforce Marketing Cloud (SFMC), allowing for comparing companies based on their size, industry, geographic and behavioural data. Finally, Oracle allows for custom machine learning algorithms as well as Oracle Audience Segmentation for processing large volumes of often-siloed and cross-channel customer data.

3 Requirements for customer segmentation – InnoSale use cases

This section describes the InnoSale end-user requirements for customer segmentation using the related use cases.

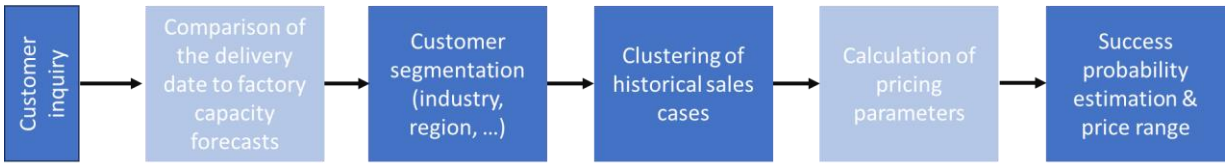

Figure 1 Dynamic pricing use case

3.1 Dynamic pricing

In the Dynamic pricing use case (Figure 1), the goal is to streamline price updates and improve responsiveness to cost changes. New pricing parameters allow for more efficient factory usage and cost optimization, while customer-specific offers enable tailored content and pricing adjustments based on individual needs.

A sales expert initiates a customer inquiry and begins processing a new sales case. Once they have gathered the customer's needs, they move to price management. Using the InnoSale AI dynamic pricing solution, they calculate a customer-specific price. The solution considers the estimated delivery date and compares it to factory capacity forecasts. **Next, the solution**

analyzes historical sales cases based on customer segment information to set pricing accordingly. Customer segmentation relies on data from the CRM system, considering factors like industry, region, and customer ID. Additional pricing parameters, such as market situation and customer value, are checked. **Success probability is evaluated using historical data specific to the customer and matching customer segments.** Finally, the solution suggests a price range that aligns with an acceptable success probability for the specific customer, allowing the sales expert to select the actual selling price. Additionally, the sales expert receives a brief explanation of the dynamic pricing factors involved.

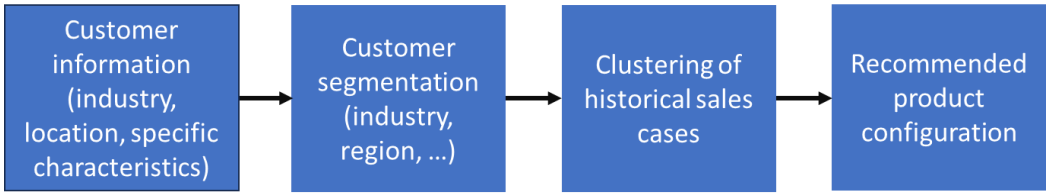

Figure 2 Area based product proposal use case

3.2 Area based product proposal

The available products offer a wide range of accessories and options tailored to specific market areas. Sales experts require extensive knowledge, from regulations to customer preferences and local machinery used with the products. This application (Figure 2) streamlines sales expert orientation, reduces reliance on individual sales experts, and improves offer quality by considering area-specific trends and requirements.

A sales expert initiates the offer by entering customer information, including segment, postal code, and the number of households. **The customer segmentation component calculates customer and order similarity. It analyzes order history data and compares it to customer information and order configuration details. Based on this analysis, the system identifies similar orders within the specific matching segment.** Leveraging the best-matching order, the AI algorithm generates a proposed set of products with configuration parameters. Finally, the sales expert can either accept or decline the recommended configuration. If accepted, the proposed product set with configuration parameters is added to an offer. The sales expert can further examine and adjust the configuration. If declined, the salesperson starts from scratch a new configuration, without preselected parameter values.

4 Customer segmentation in InnoSale framework

Customer segmentation is part of the Inference Engine of the InnoSale solution framework (see Figure 3). It primarily utilizes data from existing IT systems, most prominently the CRM software. In the framework, the end user can request, from the User Dialog Component, data from the Knowledge Base. The Knowledge Base can simply return the CRM data, but it can also query the Inference Engine to initiate the Customer Segmentation Subcomponent. The Subcomponent can then utilize data from the CRM to segment the existing customer base, and to map the present customer case to the company history. This allows the end user to make assumptions e.g. about the technical level requirements which can help the Expert to configure the product. In addition, the segment information can also be utilized to predict the sales probability of the case. The subcomponent has synergies with other InnoSale tools such

as the Dynamic Pricing Subcomponent, which allows for iterating a suitable price range that still matches the probability estimated by the Customer Segmentation Subcomponent.

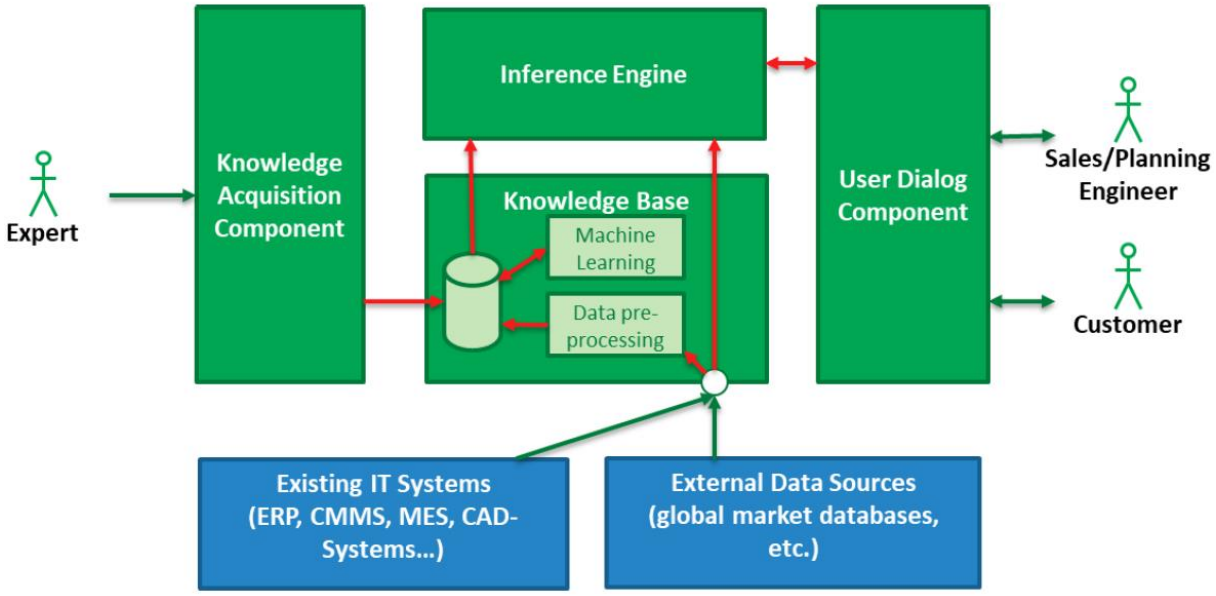

Figure 3. InnoSale solution framework.

5 Description of Customer Segmentation Subcomponent features

The description of features for the Customer Segmentation Subcomponent can be found in the Full Project Proposal for InnoSale. The segmentation is implemented as an evolutional clustering, whose features are as follows: 1) calculates the level of similarity between customers; 2) dynamic weights can be input by the user; and 3) most relevant customer characteristics will be identified.

6 Proof-of-concept of customer segmentation using an open dataset

As a proof-of-concept, an open dataset was used for experimenting with customer segmentation. Since B2B datasets are not openly available, a dataset in the B2C domain was used.

6.1 Dataset description

The used dataset[1] is from the UC Irvine Machine Learning Repository. The dataset concerns a banking institution in Portugal and their direct marketing campaigns via phone calls to consumers, gathered between 2008 – 2013 with a total of 52944 records. The target of the campaign was to subscribe to a bank term deposit. The features in the dataset include

[1] Moro,S., Rita,P., and Cortez,P.. (2012). Bank Marketing. UCI Machine Learning Repository. https://doi.org/10.24432/C5K306.

demographic information, prior customer behavior, temporal features of the phone call, details of prior contacts, and finally the economic information of the time (e.g. customer price index). Full description of the features is available in the repository. In the original article[2], the used dataset was larger than the one uploaded in the repository; it contained more features and more samples. As such, the results in the paper are not directly comparable to the present proof-of-concept.

6.2 Data processing

A common pipeline for data processing was created during the project to adapt for multiple open datasets in the B2C domain. A flow chart of the process is shown in Figure 4. The pre-processing indexes the file based on user input, and processes categorical and numerical values separately. Columns that are deemed irrelevant are dropped. The user can also specify to drop rows with certain values (e.g. not-a-numbers or unknowns). The column "age" is segmented to categories to reduce data based on user input.

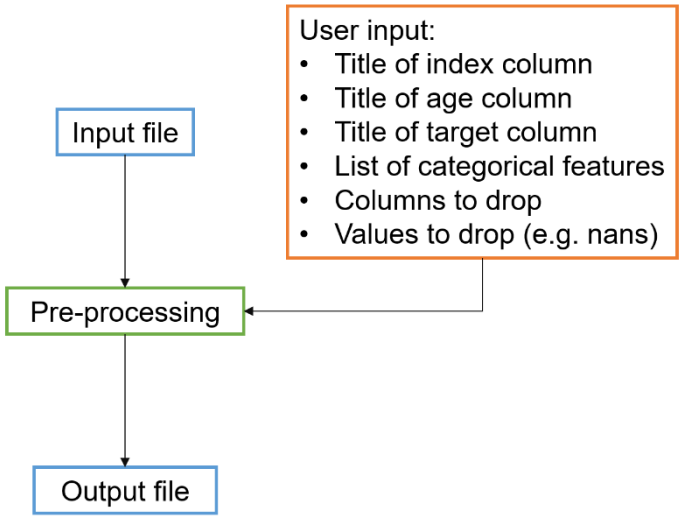

Figure 4. Common pipeline for data processing.

6.3 Data analysis pipeline

The dataset is split into train and test sets based on user-given random seed, which allows for evaluating the algorithm accurately. The train set constitutes 90% of the data, with stratification done by the target value. The user also inputs the title of the target column, allowing for runs with multiple targets if necessary. Gradient-boosted tree classifier (XGBoost) is used as a base model. In addition, the tool performs customer segmentation based on clustering algorithm, which can be changed modularly. The segmentation labels are added to the data to allow for this information to be used in prediction as well.

[2] Moro, S., Cortez, P., & Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. Decis. Support Syst., 62, 22-31. https://doi.org/10.1016/j.dss.2014.03.001

The pipeline is used to train and evaluate two models:

1. XGBoost model
2. Dummy classifier with random guessing

The baseline classifier is used as a sanity check that the model works better than random chance. All the models are evaluated with balanced accuracy, which accounts for class imbalance which exists in the present dataset. In addition to classification, Shapley additive explanations (SHAP) is used for interpreting the model, i.e. which features affect the output, how much, and to which direction (positive or negative).

To find relations between customers, clustering is used to find customer segments. Clustering is conducted via UMAP (Uniform Manifold Approximation and Projection) for dimension reduction and spectral clustering, where the number of clusters is iterated, and Davis-Bouldin Index is used as clustering performance metric; clustering with the lowest metric is deemed the best one. SHAP is used here as well, to find which are the relevant differences between the segments. This is done via training XGBoost on the cluster labels and running SHAP on this classifier model. New customers can then be mapped to these segments, and further, the most similar customer(s) can also be found. In the original feature space, the weighting of different features can be done, allowing for the user to specify in what terms the customer similarity is deemed relevant.

6.4 Classification results

The XGBoost model performed quite well, achieving mean balanced accuracy of 63.6% (Table 1). Looking at the confusion matrix of the random split 1 (Table 2), the number of true and false positives are 117 and 94, respectively. SHAP values for each classifier feature for the training set are shown in Figure 5 and Figure 6.

Table 1. Classification results on the test set using different random splits.

	Balanced accuracy [%]	
Random split	XGBoost	Dummy
1	63.39	46.51
2	64.71	46.21
3	62.23	50.22
4	64.29	50.07
5	63.29	50.07
mean	**63.58**	**48.62**

Table 2. Confusion matrix results of random split 1.

		Predicted value	
		no	yes
True value	no	2569	94
	yes	269	117

Figure 5 shows the absolute mean SHAP value for each feature. From this, it can be seen that "euribor3m" and "nr.employed" have a large impact on the model. From Figure 6, we can see what the impact for these features are: in the graph, red marks high feature values and blue low feature values. The x-axis tells the impact on model output: when these feature values are high, the associated impact on the model is negative, and vice versa. These findings make sense: the feature "euribor3m" marks the 3 month Euribor during the time of data collection, and "nr.employed" equals the number of employees in the bank. Both features are indicators of the economic situation. Thus, when the local economy is doing poorly, the tendency for a customer to subscribe for a deposit is low, and vice versa.

The effect of the other features is less clear. However, the feature "contact-//-cellular", which marks whether the person was contacted via cell phone (value 1) or not (value 0), has a slight positive impact on the model. The feature "campaign", which marks whether the subscription offer was associated with a campaign (value 1) or not (value 0), has a slight negative impact on the model.

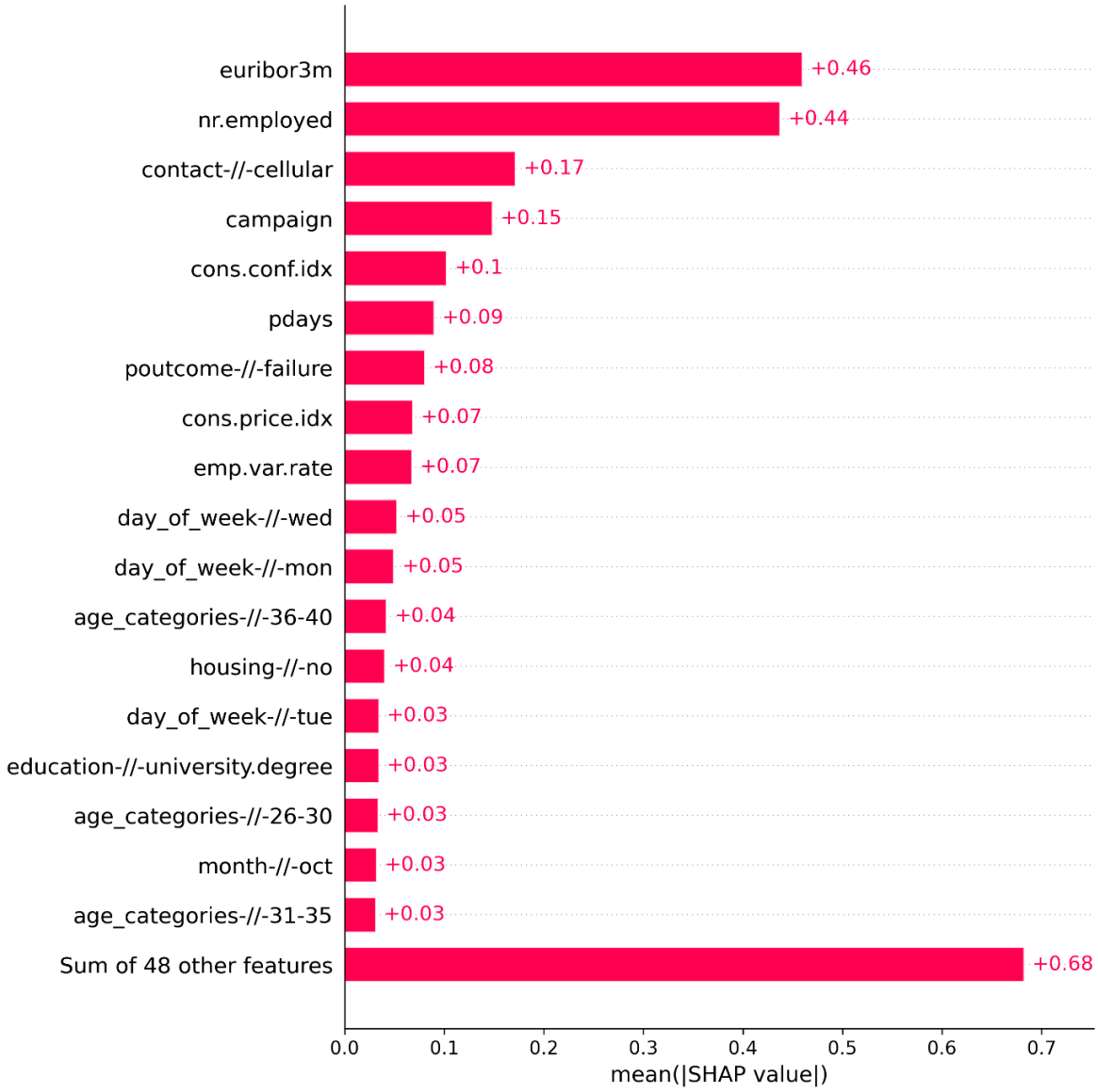

Figure 5. Global SHAP feature importance plot.

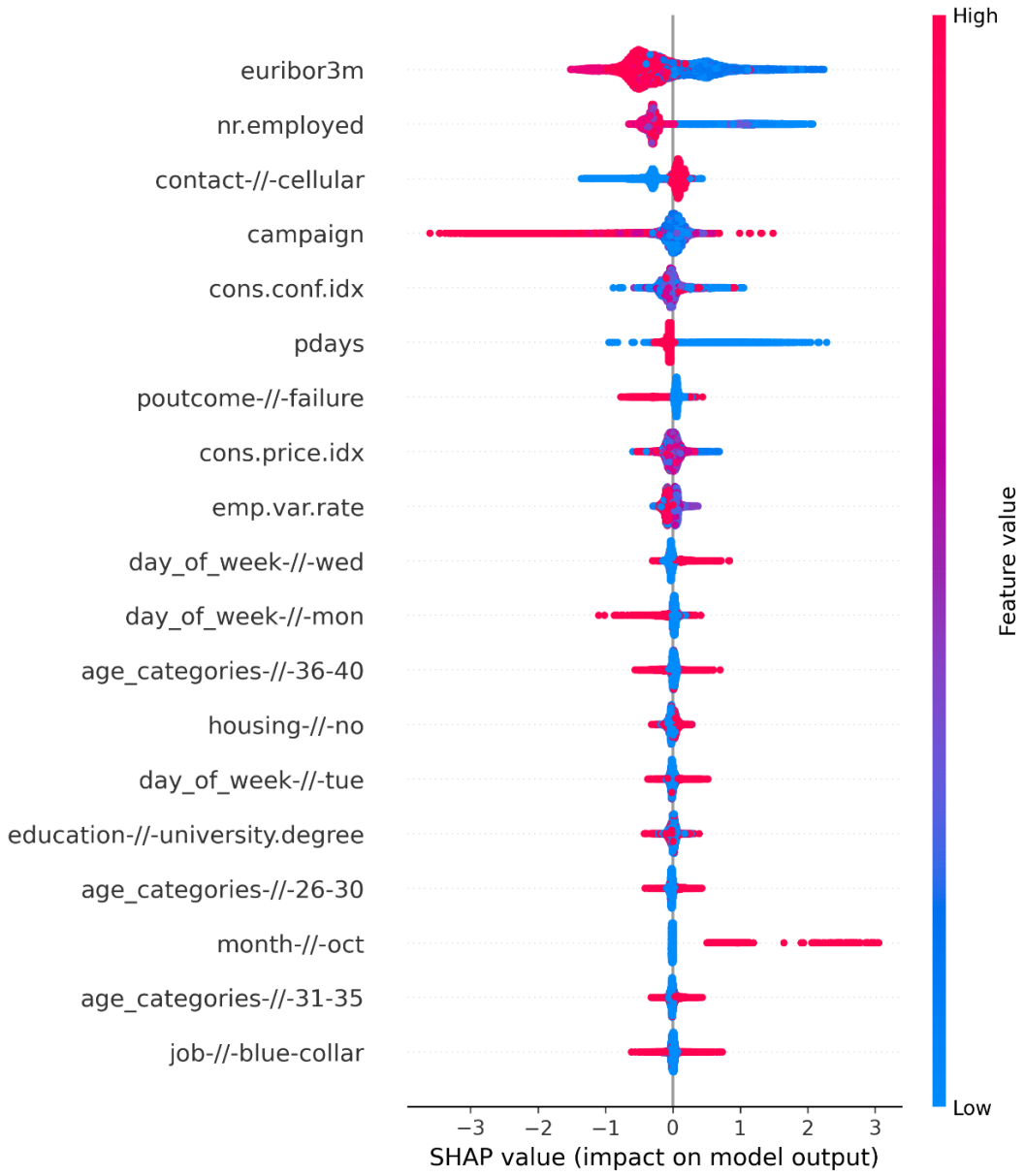

Figure 6. Local SHAP explanation summary plot.

6.5 Segmentation results

The segmentation results are shown in Figure 7. The algorithm finds 19 segments. The explanations of what are the key features upon which this segmentation is made is shown in Figure 8. For example, the importance of feature “cons.price.idx” (consumer price index) is high for segments 6, 17 and 12. Furthermore, the feature “month-//-aug” is the key feature to distinguish segment 16 from the other segments.

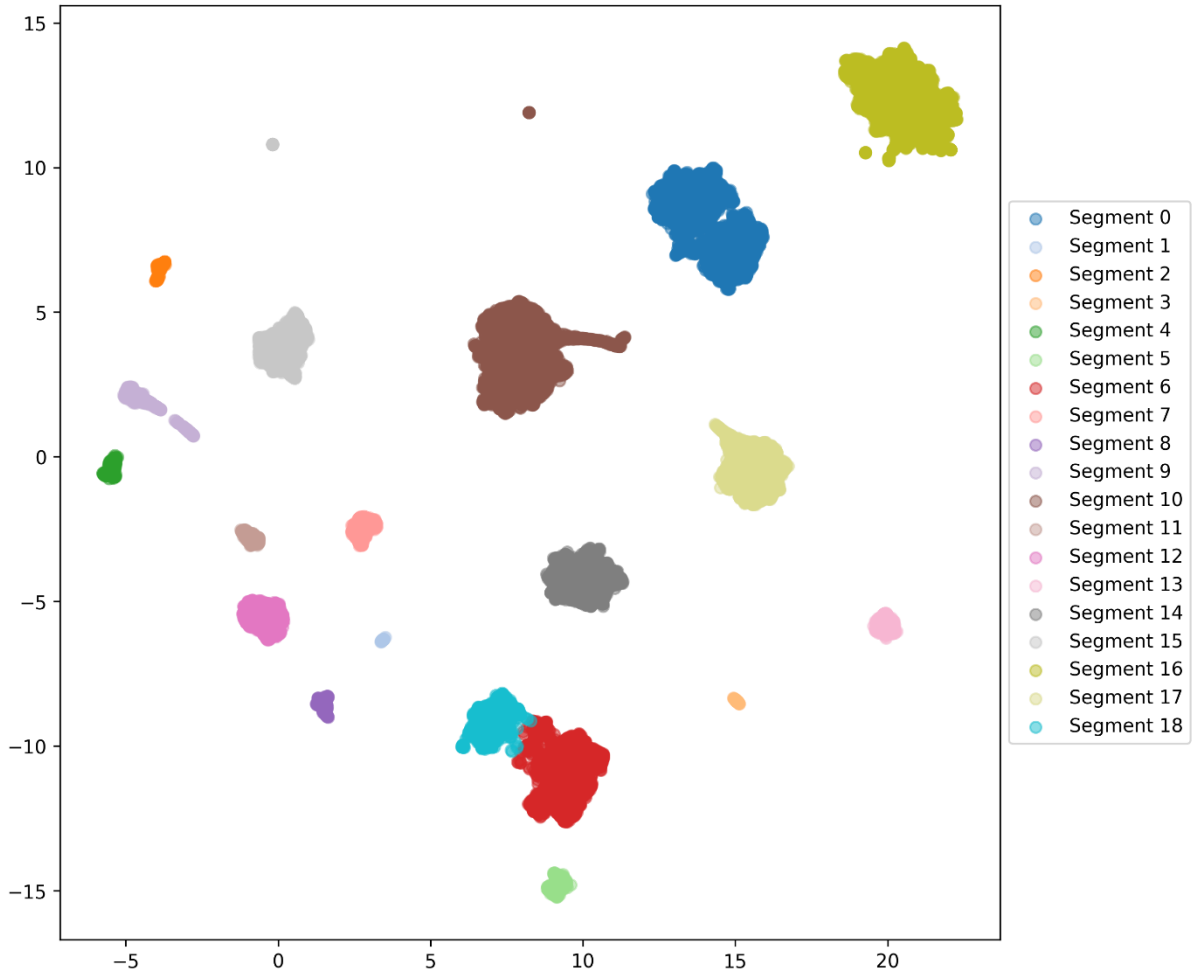

Figure 7. Clustering results.

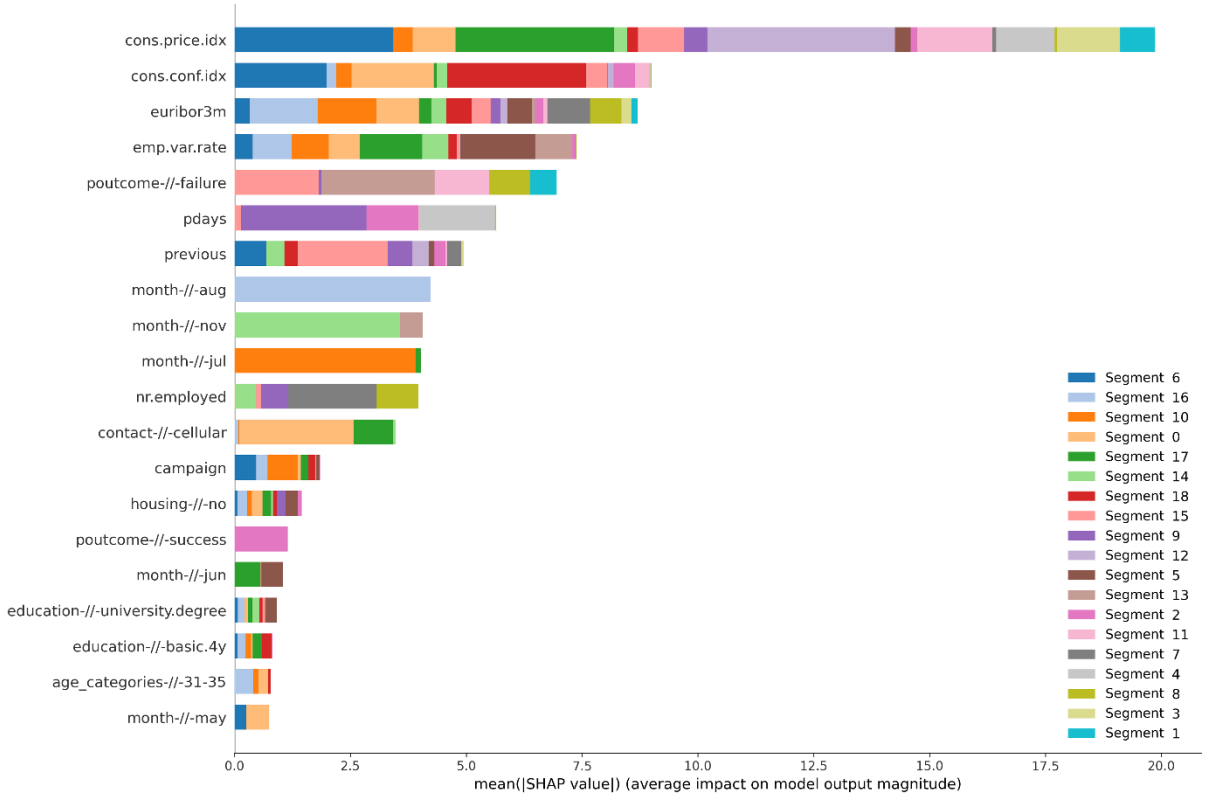

Figure 8. Global SHAP feature importance plot for the segments.

6.6 Combining the findings in classification and segmentation

The two most important features for classification were “euribor3m” and “nr.employed”. From the segmentation, we can see that the former feature is not distinctly high in any of the segments, while for the latter we see that segment 7 clearly differs from others based on this feature. However, the feature value for this segment is neither high nor low, so this segment is not clearly associated with success.

The third most important feature, “contact-//-cellular”, whose high value is associated with positive outcome, is a clear differentiator of segment 0 from others. Indeed, looking at the values of this feature, this segment only consists of people with value 0, i.e. people who were not contacted via cell phone. This gives a clear indication that this segment is not the most promising for targeted marketing. This is also reflected in the ratio of successes, which is 4% for this segment.

When looking at the success ratio in different segments, the clear “winner” is segment 9, where the success ratio is 301%, with 464 positive and 154 negative targets. What differentiates this segment from others is the feature “pdays”, which marks the “number of days that passed by after the client was last contacted from a previous campaign”. For this segment, the value is on average 6.7, while for most other segments it is 999 (marking that the person has not been contacted before).

7 Experiments with industry data

7.1 Configuration recommendation

The purpose of configuration recommendation is to help the sales expert on the offer creation process. Industrial B2B products, such as cranes or waste containers, contain multiple parameters that must be configured while creating the offer. This process requires a lot of manual work and involves also utilizing tacit knowledge that has been accumulated to the sales expert in the past. To speed up the configuration process and utilize the past offers efficiently, data analytics and machine learning methods were applied to the historical data to create initial configurations for the offer. These configurations are then used as basis for a new offer and if needed, are further fine-tuned by the salesperson.

7.1.1 Configuration recommendation version 1

Molok acted as a use case provider for a configuration recommendation case in the field of waste management. A method was developed by VTT to create initial configurations based on the data exported from Summium CRM system. On a high level the steps involved in the approach are:

1. The data was provided in the format of an Excel spreadsheet. Since the goal was to come up with recommendations to a specified geographical area, the data is first converted into machine readable format and filtered based on the country and postal code provided by the user. Optionally the geographical information can also be left out and no initial filtering to the data is done.
2. The data format is such that each offer contains variable number of rows. The number of rows depends on the products that are included in the offer. The data format is not suitable for data analytics task as it is and for that reason is converted into hierarchical representation where each offer consists of subcomponents that specify the offered product and details in the configuration.
3. This was followed by applying frequency analysis. The purpose of this step is to give each offer a numerical representation so that each offer is also presented as vector of same length. This enables the data to be used for further processing.
4. The offers presented as vectors are clustered using density-based spatial clustering of applications with noise (DBSCAN) clustering method.
5. For each cluster the most common configurations are searched and given as a result for recommended configurations.

Implementation was also done to visualize the results of the clustering for the development purposes.

Also, a method for evaluating the performance of the approach was developed. The evaluation is based on dividing the dataset into training and testing data based on time. The training data is used to create recommendations and the testing data is then used to see how many times the recommended configuration was used. This simulates the scenario where the salesperson asks for a recommended configuration based on the historical data collected until the present day and the success of the offer in the future is measured.

7.1.2 Configuration recommendation version 2

Wapice conducted a master's thesis work within InnoSale project in collaboration with Molok. The master's thesis was focusing on customer segmentation using Molok's order history data of previously sold products and their detailed product configuration information. The goal was to perform customer segmentation based on geographical area of the customer and to create an AI algorithm which, based on customer segmentation, would propose the product configuration parameters to be offered to a new customer.

As far as the AI algorithm is concerned, the objective was to create a multiclass multioutput classifying model based on supervised learning. First the order history data was analyzed, processed, and prepared. Then different machine learning algorithms were implemented and trained. Different approaches include multioutput_classifier, classifier_chain and label powerset. Finally, the model was evaluated. The result of the algorithm was an XML-file containing the product configuration parameters and their proposed values.

During the algorithm creation process some challenges were faced. The available data needed to be enriched, restructured, and harmonized so that it can be more easily processed by the algorithm. It was decided to use codes for the products and their attributes instead of product and attributes names since names differ between languages, there may be spelling errors or other deviations for the same data entity. It became also apparent that creating such an algorithm which considers each configuration parameter separately is much more complex than originally was anticipated. This is because different product configurations have a hierarchical parent-child-model structures and the number of these layers vary for each offer/order. Products also contain several configuration rules which are not necessarily evident from the data itself. Thus, if these product model layers and rules were ignored, then the algorithm would propose a product configuration that is not valid.

As conclusion, even though all the anticipated results could not be achieved, the master's thesis provided good findings and realizations of the current available data as well as topics for further research. One key topic for further research is e.g. how to perform recommendations for products that take into consideration different product configuration parameters.

7.2 Prediction of offer outcome

Offer outcome prediction was experimented with datasets from two different use case providers.

7.2.1 Prediction of offer outcome in different phases of sales funnel

In this task the goal was to predict outcome of the offer (lost offer/won offer). Konecranes provided an anonymized dataset of customer information, sales cases, and assets in csv-format. VTT developed the machine learning model.

Dataset contains anonymized information about customer such as country, city, and sales cases for a customer. For each sales case there was an offer result (lost offer/won offer) which

was used as a target for the model. There was information about business such as area, organization, sales case value etc. In addition to these features, additional features were calculated, such as number of updates made to the offer. Based on this information a model was trained to predict the outcome of the offer.

Additionally, most relevant features for the task at hand were identified. In our snapshot of the dataset some of the features were deemed to be unimportant. Shapley additive explanations (SHAP) and Random Forest feature importance were used to identify such features.

7.2.2 Prediction of sales call result

Calling agents utilize manuscripts when calling customers. LeadDesk provided an anonymized dataset of manuscripts and call results. The goal of VTT activities was to study which of the manuscripts were most successful and create a model to predict manuscript success.

This dataset has manuscripts for different campaigns which the calling agents utilize. Each manuscript may contain one or more pages. For each call a campaign identifier, call result and used manuscript pages are recorded in addition to other relevant call information such as talk time. Exploratory data analysis was used to check the data.

For some campaigns, the result was a binary value (deal/no deal), but for others, the result was encoded to be a more granular target. For example, in one case the target was one of "Positive", "High potential", "Negative" or "Other".

Descriptive statistics were calculated, and figures of relevant information were created for illustrative purposes.

Data from two LeadDesk's customers were chosen for analysis; one of them was finally focused on due to larger amount of data. The aim was to predict the sales call result based on features available prior to making the call. The analysis process follows those in 6.1 - 6.3. The sales call result was categorical but it was binarized based on discussions with LeadDesk. Thus, the target was "Deal" or "No deal".

Relevant features were extracted from the data. These include, attributes of the used manuscript, temporal features such as hour of the day, day of the week and month, as well as attributes of the call. The features were one-hot encoded, and XGB classifier was used for analysis. Train-test split was 85-15%. SHAP was used to find the most important features and their effect on the model output.

The XGB model had a balanced accuracy of 64.78%, compared to the benchmark (coin flip) of 55.98%. Thus, it could be argued that the model could improve the sales results somewhat in operational use. However, what restricts the analysis is that the dataset did not contain any features of the person to whom the call was made. This is arguably a deciding factor missing out in the used dataset.

8 Conclusions

We have reported on the work done in the InnoSale project pertaining to the Customer Segmentation Subcomponent, developed mainly by VTT. The Subcomponent would be part of the Inference Engine of the InnoSale solution framework and primarily utilizes data from existing IT systems, most prominently the CRM software. The Subcomponent defined based on the results of the InnoSale WP1 and WP2 should calculate the level of similarity between customers, utilize evolutional clustering, and allow for dynamic weights to be input by the user. The most relevant customer characteristics or sales case features should also be identified. A proof-of-concept of customer segmentation functionality for offer outcome prediction (deal/no deal) was demonstrated using an open dataset. Experiments with industry data were also conducted, including configuration recommendation and prediction of offer outcome. Overall, based on these initial results, the different customer segmentation functionalities could provide a valuable tool for directing the sales processes and marketing activities of a company. However, as all the approaches have not yet been comprehensively adopted on relevant industrial data or evaluated by the potential users (sales experts) their applicability to the use cases and impact on sales process performance is not fully demonstrated.