Integrating the voice of customers through call center emails into a decision support system for churn prediction

Kristof Coussement, Dirk Van den Poel *

Ghent University, Faculty of Economics and Business Administration, Department of Marketing, Tweekerkenstraat 2, 9000 Ghent, Belgium

Received 9 October 2006; received in revised form 7 August 2007; accepted 21 January 2008
Available online 17 March 2008

Abstract

We studied the problem of optimizing the performance of a DSS for churn prediction. In particular, we investigated the beneficial effect of adding the voice of customers through call center emails – i.e. textual information – to a churn-prediction system that only uses traditional marketing information. We found that adding unstructured, textual information into a conventional churn-prediction model resulted in a significant increase in predictive performance. From a managerial point of view, this integrated framework helps marketing-decision makers to better identify customers most prone to switch. Consequently, their customer retention campaigns can be targeted more effectively because the prediction method is better at detecting those customers who are likely to leave.

# 2008 Elsevier B.V. All rights reserved.

Keywords: Customer relationship management (CRM); Data mining; Churn prediction; Text mining; Call center email; Voice of customers (VOC); Binary classification modeling

1. Introduction

In the past, companies focused on selling products and services with little knowledge or strategy concerning the customers who bought the products. Today business is evolving from this ‘product-centered’ to a ‘customer-centered’ environment. Companies need to find ways to capture and enhance market share while reducing costs [7]. Consequently, existing companies must reconsider the business relationships with their customers [24].

Customer relationship management (CRM) is becoming a critical success factor in today’s business environment [2,16]. Data mining is being implemented to gain customer knowledge from organizational data warehouses [35]. A way to manage customer churn is to predict which customers are most likely to leave and then target them with incentives to stay. Consequently, these IS support marketing-decision makers to generate marketing campaigns for the right customers. A field experiment by Burez and Van den Poel [9] has already shown that companies can boost profitability by shifting from mass to focused marketing strategies. It is more profitable to keep and satisfy existing customers than to attract new ones with a high attrition rate [26]. Identifying customers most prone to switch, is thus important [17]. In order to develop an effective customer retention program, the company must build a model that is as accurate as possible; indeed Van den Poel and Larivi`ere [36] showed that a small change in retention rate can result in a significant change in profitability.

We decided it was necessary to incorporate the voice of customers (VOC) through call center emails into a traditional churn-prediction model in order to provide a better model: one with a higher predictive performance. The rapid development of IT and the Internet has made it easier for customers to communicate with the company. Call centers are expanding rapidly in scope, number and size [1], because many firms rely on them to address
customer concerns and provide product information [25]. However, marketing managers tend to neglect this valuable information because (i) it is not directly applicable in a traditional marketing context, (ii) there is seldom in-house knowledge on how to convert this (textual) information into an analyzable form and (iii) no ready-to-use framework is available to integrate the information. We developed a DSS for churn prediction; it integrates free-formatted, textual information from customer emails with information derived from the marketing database. Although previous research used the VOC in understanding customers’ needs and behavior (e.g. Refs. [10,11,21]), no prior work has used VOC in a churn-prediction model.

2. Methodology

Fig. 1 shows how the integration of information types in a churn-modeling system was achieved.

2.1. Data collection

Structured marketing information can be extracted from a common marketing database in which all transactional and marketing-related information has been stored. In contrast, call center emails are highly unstructured. Thus, extracting information from emails requires meticulous pre-processing to capture the relevant details for inclusion in a churn detection/prediction DSS.

2.2. Pre-processing

2.2.1. Data and text pre-processing

The structured information is internally available at a very low cost and available for pre-processing and integration into our model. However, the original emails are unformatted by nature. They are converted into a structured representation using the vector-space of Salton’s SMART [31]: an email is represented as a vector of weighted frequencies of designated words. Thus emails are \( n \)-dimensional vectors, with \( n \) the number of distinct terms in the dictionary. Each vector component reflects the importance of the corresponding term with respect to the semantics of the email [6] and each component has a weight if the term is present or zero otherwise. Thus a collection of emails is represented as a term-by-email matrix. Fig. 2 shows the steps in this pre-processing phase whereby raw emails become a term-by-email matrix.

In the first step, raw text cleaning, special characters and punctuation are removed from words and spelling errors are corrected by comparing with words in a reference dictionary using a synonym data set. Tokenization converts the input stream into tokens or words. It uses blanks as delimiters for words which are then converted to lower case (case conversion). Part-of-speech tagging gives words their syntactic category: informative (nouns, verbs, adjectives and adverbs) or non-informative.

Next, terms are replaced by their stem, e.g. connect is the stem for connected, connecting, connection, etc. Stemming reduces the number of terms significantly [5] and increases

Fig. 1. An integrated churn-modeling system that uses structured, database-related information and free-formatted, textual information.
retrieval performance [19]. A dictionary-based stemmer is used. When a term is unrecognizable, standard decision rules are applied to give the term a correct stem.

The result of this process is a high-dimensional term-by-email matrix having many distinct terms. This matrix is reduced by applying term filtering: rare words are eliminated because they seldom help in future classifications. Word frequencies follow a Zipf distribution [37]: thus half of them appear only once or twice. Eliminating words under those thresholds often yield great savings [22]. Stopwords, (e.g. ‘the’ or ‘a’) are also removed. Next, the non-informative parts of speech are left out from the analysis. A last step in the term-filtering phase is removing irrelevant terms by manually checking the temporary dictionary.

In the term vector-weighting phase, a weighted term vector for every email is constructed. By now, the values in the term-by-email matrix are simply the raw frequencies of appearance for a term in an email. Spark Jones [33] showed significant improvements in retrieval performance when using weighted term vectors. Term weighting is often done by determining the product of the term frequency (tf) and the inverse email frequency (idf) [27–29,34]. The result is a high-dimensional, weighted term-by-email matrix.

The aggregated weight of term $i$ for all emails belonging to subscription $j$ ($A_{wij}$) is

$$A_{wij} = \sum_{k=1}^{r} w_{ik}$$

with $w_{ik}$ equal to the weight of term $i$ in email $k$ and $r$ equal to the number of emails belonging to the same observation.

Using each distinct term as a feature in the churn-modeling phase would lead to an unmanageable number of explanatory variables. Moreover, due to the high dimensionality of the feature space, most weights are zero for a single email. Thus, using a large and sparse term-by-email matrix would be counterproductive in the predictive-modeling context.

2.2.2. Dimension reduction

The dimension of the aggregated (weighted) term-by-email matrix is reduced by using Latent Semantic Indexing (LSI). It reduces the dimensionality of the feature space by grouping together related terms [12]. Deerwester et al. [12] used singular value decomposition (SVD) to form semantic generalizations from emails. It uses the fact that certain terms appear in similar emails to establish relationships between the terms. Consequently, SVD projects emails from the high-dimensional term space to an orthonormal, semantic, latent subspace by grouping together similar terms into concepts. As such, each concept can be described using many different keywords as it has a high discriminatory power to other concepts in the reduced feature space. See Appendix B for more detailed information about LSI using SVD.

2.2.3. Optimal dimension selection

The intensity of dimension reduction during the SVD phase is critical. Ideally, the number of concepts $k$, must be large enough to fit all the underlying, relevant concepts in the email collection, but small enough to prevent the model from fitting sampling errors and unimportant details. Moreover, the obtained optimal $k$ must be workable from a prediction point of view. In the factor-analytic literature, such choices are still an unanswered question. Deerwester et al. [12] propose using an operational criterion, i.e. a value of $k$ that yields good performance. In our application, we are especially interested in the predictive performance of the SVD output.

It is not possible to know what value of $k$ will lead to an optimal solution when validating the predictive model initially. As such, improper selection of the parameter $k$ is ineffective if too few concepts are included or computational expensive if too many irrelevant concepts are incorporated. Consequently, a parameter-selection procedure is needed. We construct several rank-$k$ models and the most favorable rank-$k$ model (based on the cross-validated performance) is retained for further analysis. As such, the optimal value of $k$ is obtained using a fivefold cross-validation on the training set. The training set is divided into five subsets of equal size.
Iteratively, each part is used for validation, while the other parts are used for training. So finally, each case in the training set is predicted once. The cross-validation performance better reflects the real performance when validating the classifier for unseen data. In the end, it is possible to select the optimal value of $k$ based on the most favorable cross-validated model. Kim [18] stated that it is very important for data analysts to consider the relationship between the amount of information and the complexity of predictive models because compact information models show great improvement in terms of predictive performance and robustness.

2.3. Modeling

2.3.1. Modeling technique and variable selection

Logistic regression is used. In applying it, a maximum-likelihood function is produced and maximized in order to become an appropriate fit to the data [3]. With a training set $T = \{(x_i, y_i)\}$ and $i = 1, 2, \ldots, N$ and input data $x_i \in \mathbb{R}^n$ and corresponding binary target labels $y_i \in \{0, 1\}$, logistic regression is used to estimate the probability $P(y = 1|x)$ given by

$$P(y = 1|x) = \frac{1}{1 + \exp(-w_0 - w^T x)}$$

with $x \in \mathbb{R}^n$ an $n$-dimensional input vector, $w$ the parameter vector and $w_0$ the intercept.

This technique is used because it is conceptually simple [8], a closed-form solution for the posterior probabilities is available and Neslin et al. [23] stated that it provides quick and robust results in a churn-prediction context.

Variable selection is the process of choosing a subset of the original variables by eliminating some variables based on their predictive performance. Kim [18] stated that there are three main reasons for using a variable-selection technique: saving computational time and cost by extracting as much information with the smallest number of variables, improving the comprehensibility of the resulting models and making the model generalize better.

Our study employs a forward-selection procedure: the algorithm added one variable at a time. The first variable to enter the model is that with the highest $\chi^2$-statistic. At each step, the remaining variables are considered for inclusion in the final model. Forward selection adds variables until a stopping rule is satisfied. The choice of this standard variable-selection technique makes it easy for implementation, while more sophisticated algorithms are computationally more expensive and require additional parameter settings.

2.3.2. Evaluation criteria

To evaluate the performance of classification models, two commonly used criteria are used: the lift and the area under the receiving operating curve (AUC).

\textit{Lift} is the most commonly used performance measure for evaluating classification models. It reflects the increase in density of the churn event relative to the density of churners in the total database. The higher the lift, the better the predictive model. In marketing applications, it is interesting to increase the density of churners, especially in the top 10% cases most likely to churn. Marketing-decision makers are typically interested in only 10% of the entire marketing database because budgets are often limited and actions to reduce churn typically involve only 10% of the entire customer database. Practically, all cases are sorted from most likely to churn to least likely to churn. Afterwards, the density of churners from the top 10% cases most likely to churn is compared with the density of churners in the entire customer collection. This increase in density is called the top-decile lift. Intuitively, a top-decile lift of two means that the density of churners in the top 10% cases most likely to churn is twice the density of churners in the entire database.

The AUC takes into account the predicted class of an event with the real class of that event, considering all possible cut-off values. Consequently, the AUC takes into account the individual class performance for a range of possible thresholds. If true positives (TP) are the number of positives that are correctly identified, false positives (FP) are the number of negatives that are classified as positives, false negatives (FN) are the number of positive cases that are identified as negatives and true negatives (TN) are the number of negative cases that are classified as negatives, then

- the sensitivity is $(TP/(TP+FN))$: the proportion of positive cases that are predicted to be positive;
- the specificity is $(TN/(TN + FP))$: the proportion of negative cases that are predicted to be negative.

These vary when the threshold value is varied. The receiver operation characteristics curve (ROC) is a two-dimensional plot of the sensitivity versus (1-specificity). In order to compare the performance of two or more classification models, the area under the receiver operating characteristics curve is calculated. This measure is used to evaluate the performance of a binary classification system [15]. In order to test if two AUCs are significantly different, one can apply the non-parametric test of Delong et al. [13].

3. Empirical verification

3.1. Research data

In our study, we used data obtained from a large Belgian newspaper publishing company. Subscribers have to pay a fixed price for their newspapers, depending on the length of subscription and the promotional offer given. The company does not allow subscribers to end their subscription before the expiry date. The churn-prediction problem therefore
involves predicting whether a subscription will be renewed during the 4-week period after maturity. During this period, the newspaper publishing company still delivers the newspapers in order to allow subscribers time to renew their subscription. The company has a structured, marketing database where transactional and subscription related information is stored and they save all customer emails sent to the call center. Fig. 3 shows the time window of analysis in our study.

Subscription data from January 2002 to September 2005 was analyzed. Consequently, it is possible to define the dependent and the explanatory variables. All renewal points between July 2004 and July 2005 were considered. A customer was seen as a ‘churner’ when the subscription was not renewed in a 4-week period following the maturity date. The explanatory variables were constructed from the two available types of information. These were used to predict whether a subscription would be renewed.

The first type of variables contained information from the structured, marketing database. These variables contained information on a 30-month period. They were subdivided into four categories (see Appendix C):

- client/company interaction variables,
- subscription related variables,
- renewal specific variables and
- socio-demographics.

The second type of information consisted of all information sent by the subscriber via email during the last period of his/her subscription. Because this information is highly unstructured, the emails were pre-processed to represent them in our churn-prediction model.

In order to compare the beneficial effect of unstructured information from call center emails in a churn-prediction model, subscriptions with at least one email sent during the last term of the subscription were considered.

Tables 1 and 2 summarize the data characteristics for the randomly split training and test set. The training set was used to obtain the optimal SVD dimension and the model estimates, while the test set is used to validate and compare the different models.

### Table 1
Overview of the marketing data characteristics

<table>
<thead>
<tr>
<th></th>
<th>Number of subscriptions</th>
<th>Relative percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training set</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subscriptions not renewed</td>
<td>1777</td>
<td>18.50</td>
</tr>
<tr>
<td>Subscriptions renewed</td>
<td>7826</td>
<td>81.50</td>
</tr>
<tr>
<td>Total</td>
<td>9603</td>
<td>100.00</td>
</tr>
<tr>
<td>Test set</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subscriptions not renewed</td>
<td>593</td>
<td>18.76</td>
</tr>
<tr>
<td>Subscriptions renewed</td>
<td>2568</td>
<td>81.24</td>
</tr>
<tr>
<td>Total</td>
<td>3161</td>
<td>100.00</td>
</tr>
</tbody>
</table>

### Table 2
Overview of the call center emails characteristics

<table>
<thead>
<tr>
<th></th>
<th>Number of emails</th>
<th>Average number of mail per subscription</th>
<th>Average number of words per email</th>
<th>Average number of words per sentence</th>
<th>Average number of unique words per email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training set</td>
<td>14,083</td>
<td>1.47</td>
<td>113.27</td>
<td>28.98</td>
<td>72.54</td>
</tr>
<tr>
<td>Test set</td>
<td>4,694</td>
<td>1.48</td>
<td>116.33</td>
<td>29.33</td>
<td>72.49</td>
</tr>
</tbody>
</table>

3.2. Optimal dimension selection

The text pre-processing phase resulted in a high-dimensional term-by-email matrix. This was unworkable from a prediction point of view. Its optimal reduced rank was obtained by applying a cross-validation procedure on the training data. Fig. 4 shows the results of this cross-validation; the x-axis has the number of concepts and the y-axis represents the cross-validated AUC. It is clear that in the range of 1–100 concepts, the cross-validated performance was increasing rapidly. From 100 concepts on, the cross-validated AUC was growing less rapidly, while in the region around 170 concepts, the cross-validated performance was stabilizing. Including more than 170 concepts resulted in a more complex churn model, while the predictive performance hardly increased. Thus 170 concepts was chosen as the optimal number for representing the textual information in our study. At this point, a good balance was achieved between the number of concepts and the predictive performance.
3.3. Defining the best subset of the structured marketing variables

Before comparing the predictive performance of the model with structured marketing information only (ModStruc) to the performance of the model that combined the structured marketing information and textual information (ModStruc–Unstruc), the optimal set of structured marketing variables was found by employing the forward-selection procedure. It resulted in a best subset of 20 marketing variables (see Table 3).

Modstruc was built using the 20 marketing variables, while ModStruc–Unstruc was a combination of those 20 marketing variables with those variables representing the textual information—i.e. 170 additional variables.

3.4. Comparing predictive performance

Table 4, Figs. 5 and 6 show that the predictive performance of ModStruc–Unstruc significantly outperformed that of ModStruc. The AUC increased from 73.80 to 77.75 by adding textual information to a traditional churn-prediction model. This improvement was significant ($\chi^2 = 23.1$, d.f. = 1, $p < 0.001$). The ROC curve of ModStruc–Unstruc is located further from the random model than that of ModStruc, thus the area under the ROC of ModStruc–Unstruc is larger than that of ModStruc. ModStruc–Unstruc was thus able to better distinguish churners from non-churners. Moreover, the beneficial effect of textual information on predictive performance was confirmed in terms of top-decile lift. The cumulative lift curve of ModStruc-Unstruc laid above that of ModStruc. ModStruc–Unstruc is able to identify more customers truly at risk than ModStruc within a specific decile. Lift in the first decile or the 10% top-decile – i.e. the 10% point – increased from 2.69 to 3.07.

Our study provided a realistic framework that increased the predictive performance of a churn model for subscribers.

### Table 3

Best subset of marketing variables employed by the forward-selection procedure

<table>
<thead>
<tr>
<th>Step</th>
<th>Variable name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Elapsed time since the last complaint</td>
</tr>
<tr>
<td>2</td>
<td>Monetary value</td>
</tr>
<tr>
<td>3</td>
<td>Elapsed time since last suspension</td>
</tr>
<tr>
<td>4</td>
<td>The length of the current subscription</td>
</tr>
<tr>
<td>5</td>
<td>The average positioning of complaints in the current subscription (with 0 = start of the subscription and 1 = end of subscription)</td>
</tr>
<tr>
<td>6</td>
<td>Whether the previous subscription was renewed before the expiry date</td>
</tr>
<tr>
<td>7</td>
<td>Whether the subscriber is a woman</td>
</tr>
<tr>
<td>8</td>
<td>The variance in the number of days the previous subscriptions are renewed before expiry date</td>
</tr>
<tr>
<td>9</td>
<td>The number of renewal points</td>
</tr>
<tr>
<td>10</td>
<td>Whether the newspaper edition is ‘X1’</td>
</tr>
<tr>
<td>11</td>
<td>Whether the subscriber is a public institution</td>
</tr>
<tr>
<td>12</td>
<td>How many days before the expiry date, the previous subscription was renewed</td>
</tr>
<tr>
<td>13</td>
<td>The number of suspensions$^x$</td>
</tr>
<tr>
<td>14</td>
<td>The average suspension length (in number of days)$^x$</td>
</tr>
<tr>
<td>15</td>
<td>The number of suspensions</td>
</tr>
<tr>
<td>16</td>
<td>The average suspension length (in number of days)</td>
</tr>
<tr>
<td>17</td>
<td>Whether the purchase motivator is a direct marketing campaign</td>
</tr>
<tr>
<td>18</td>
<td>Whether the newspaper is picked up at the shop</td>
</tr>
<tr>
<td>19</td>
<td>Elapsed time since last conversion in payment method</td>
</tr>
<tr>
<td>20</td>
<td>The conversions made in payment method$^a$</td>
</tr>
</tbody>
</table>

$x$: variable corrected for the length of subscription.

$^a$: variable corrected for the length of subscription.
whose textual information is available. Since ModStruc and ModStruc–Unstruc were built on a selective sample of subscribers who contacted the company at least once per email, one may suggest including more subscribers—i.e. those who did not send an email. One should verify whether a separate churn model of subscribers who sent at least one email is the best strategy in obtaining optimal predictive performance. Practically, the current training set of subscribers was extended by randomly selecting subscriptions of customers who had not sent any email (ModStruc–k, with k the number of randomly selected subscriptions whereby k = {0; 5000; 10,000; . . . ; 100,000}) with the intent of building a churn model with better predictive performance on the current test set. Fig. 7 graphically indicates the results. The horizontal lines indicating the performance of ModStruc and ModStruc–Unstruc are included for reasons of comparability, despite the fact that they were independent of k.

As one observes from Fig. 7, it was indeed better to build a separate model for subscribers who sent at least one email. The predictive performance of ModStruc was always higher than ModStruc–k. This clearly pointed out that subscribers from whom textual information was available have a unique churn pattern. The performance of ModStruc–Unstruc dominated those of ModStruc and ModStruc–k.
4. Conclusion

Adding the VOC by means of call center emails into a standard churn-prediction system helps marketing-decision makers to identify with a higher precision those customers most prone to switch. Consequently, retention campaigns to these customers can become more targeted. The framework integrated textual information from call center emails with traditionally used marketing information. Converting the unstructured call center emails into a structured form suitable for churn prediction, required specialized pre-processing and dimension reduction steps.

Moreover, our study confirmed the importance of a well-considered email handling strategy. It provided a methodology that may increase the profitability of the call center by offering a model for marketing-decision makers using customers of whom textual information is available. By enriching the churn model with this unstructured information from call center emails, marketing managers may improve the effectiveness of their retention campaigns.

Acknowledgments

We would like to thank the anonymous Belgian Company for their efforts in providing us with their data. Moreover, we also like to thank (1) BOF (01D26705) for funding the PhD project of Kristof Coussement, (2) BOF (011B5901) for funding the computing infrastructure and (3) Jonathan Burez, Bart Lariviére and Ilse Bellinck for their insights and suggestions during this project. This project was realized using SAS v9.1.3, SAS Text Miner v5.2 and Matlab v7.0.4.

Appendix A. Term vector-weighting phase

The term frequency (tf) measures the frequency of occurrence of an index term in the email text. The more a term is present, the more important this term is in characterizing the content of that email. As such the frequency of occurrence of a content word is used to indicate term importance for content representation [4,20,30]. In our study, the tf was obtained by taking a logarithmic transformation of the original term frequency. Taking the logarithmic transformation reduced the importance of the raw tf, which was important for email collections of varying length.

The inverse document frequency (idf) was incorporated so that the more rare a term occurred in the collection of emails, the more discriminating it was. Therefore, the weight of a term was inversely related to the number of emails in which the term occurred—i.e. the frequency of the term [14,32]. The logarithm of the idf was taken to decrease the effect of the raw idf-factor.

Finally the weight of term $i$ in an email $j$ ($w_{ij}$) was given by

$$w_{ij} = tf_{ij} \cdot idf_i$$  \hspace{1cm} (A.1)

with $tf_{ij}$ equal to the term frequency of term $i$ in email $j$; $idf_i$ is equal to the inverse email frequency of term $i$.

Mathematically,

$$tf_{ij} = \log_2(n_{ij} + 1)$$  \hspace{1cm} (A.2)
with \( n_{ij} \) equal to the frequency of term \( i \) in email \( j \) and

\[
\text{idf}_i = \log_2 \left( \frac{n}{d_f} + 1 \right)
\]  

(A.3)

with \( n \) equal to the total number of emails in the entire email collection and \( d_f \) equal to the number of emails where term \( i \) was present.

### Appendix B. Dimension reduction using LSI via SVD

A high-dimensional term-by-email matrix \( A \) was constructed so that location \((i,j)\) indicated \( w_{ij} \), the weight of term \( i \) for email \( j \). SVD factorized \( A \) into three distinct matrices by

\[
A = U \Sigma V^T
\]  

(B.1)

with \( \Sigma \) equal to a diagonal matrix containing the singular values of matrix \( A \), \( U \) equal to the term-concept similarity matrix and \( V \) equal to the concept-email similarity matrix.

Mathematically, \( \Sigma = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_r) \) was the singular-values matrix where \( \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_r \). \( U \) and \( V \) were column-orthonormal matrices. The weights of the original matrix depended on the latent concepts by

\[
w_{ij} = \sum_{x=1}^{r} U_{ix} \Sigma_{x} d_{xj}
\]  

(B.2)

LSI based on SVD allowed a simple strategy to approximate the original matrix \( A \) with rank \( r \) by \( A \) with rank \( k \) where \( k \leq r \). Therefore, LSI ignored the smaller lambda values in \( \Sigma \) by retaining only the first predetermined singular values equal to or greater than \( k \), i.e. \( \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_k \), while only the first \( k \) columns of \( U \) and \( V \) were retained.

\[
\hat{A}_k = U_k \Sigma_k V_k^T
\]  

(B.3)

with \( U_k \), \( \Sigma_k \) and \( V_k \) were equal to the \( k \)-rank approximation of \( U \), \( \Sigma \) and \( V \), respectively.

Matrix \( V_k \) is the approximated \( k \)-rank concept-email similarity matrix. A cell in the matrix \( V_k \) represented the loading for a specific email on one of the \( k \) concepts. This matrix contained information on how well a certain email loads on the different \( k \) concepts. The concepts reflected the hidden patterns in the textual data. Consequently, these concepts were used as explanatory variables in the churn-prediction model because they represented the latent semantic patterns of the textual information.

It is important that the concept loadings from the training vectors were comparable with those from the test vectors. The meaning of the concepts during testing should stay the same as those during training. Consequently, emails of the test set were projected into the same semantic latent subspace as created during training.

In order to compare a test email \( d \) with the training emails, its term vector \( A_d \) was derived using the same pre-processing steps. Deerwester et al. [12] proposed projecting each new term vector into the same latent semantic subspace as that created during training by

\[
V_d = A_d' U_k \Sigma_k^{-1}
\]  

(B.4)

with \( U_k \) the \( k \)-rank concept-term similarity matrix and \( \Sigma_k \) the diagonal singular value matrix in rank \( k \), both of the original SVD. \( V_d \) was the new concept-email vector which was comparable to the concept-email vectors of the matrix \( V_k \).

However, the choice of \( k \) was critical for optimal predictive performance.

### Appendix C. Overview of structured marketing information

*Client/company-interaction variables:* variables describing the client/company relationship:

- The number of complaints.
- Elapsed time since the last complaint.
- The average cost of a complaint (in terms of compensation to newspapers).
- The average positioning of the complaints in the current subscription.
- The purchase motivator of the subscription.
- How the newspaper is delivered.
- The number of conversions made in distribution channel, payment method and edition.
- Elapsed time since last conversion in distribution channel, payment method and edition.
- The number of responses on direct marketing actions.
- The number of suspensions.
- The average suspension length (in number of days).
- Elapsed time since last suspension.
- Elapsed time since last response on a direct marketing action.
- The number of free newspapers.

*Renewal-related variable:* variables containing renewal-specific information:

- Whether the previous subscription was renewed before the expiry date.
- How many days before the expiry date, the previous subscription was renewed.
- The average number of days the previous subscriptions are renewed before expiry date.
- The variance in the number of days the previous subscriptions are renewed before expiry date.
- Elapsed time since last step in company retention procedure.
- The number of times the customer did not renew a subscription.
Socio-demographic variables: variables describing the subscriber:

- Age.
- Whether the age is known.
- Gender.
- Physical person (is the subscriber a company or a physical person).
- Whether contact information (telephone, mobile number, email) is available.

Subscription-describing variables: group of variables describing the subscription:

- Elapsed time since last renewal.
- Monetary value.
- The number of renewal points.
- The length of the current subscription.
- The number of days a week the newspaper is delivered (intensity indication).
- Which edition the subscriber has (X1, X2, X3).
- The month of contract expiration.

References


Kristof Coussement is a PhD candidate in economics and business administration at Ghent University (Belgium). He received his master degree in applied economics as well as his master after master degree in marketing analysis at Ghent University (Belgium). During his PhD thesis, he investigated the impact of client/company interactions through verbalized information sources on CRM (churn analysis). His works are published in Decision Support Systems and Expert Systems with Applications.

Dirk Van den Poel is professor of marketing at the Faculty of Economics and Business Administration of Ghent University, Belgium. He heads a competence center on analytical customer relationship management (aCRM). He received his degree of management/business engineer as well as his PhD from K.U. Leuven (Belgium). His main interest fields are the quantitative analysis of consumer behavior (CRM), data mining (genetic algorithms, neural networks, random forests, random multinomial logit: RMNL), text mining, optimal marketing resource allocation (DIMAR-OPT) and operations research. His works are published in Decision Support Systems, Journal of Business Research, European Journal of Operational Research, Journal of the Operational Research Society, International Journal of Intelligent Systems and Expert Systems with Applications.