Incorporating textual information in customer churn prediction models based on a convolutional neural network

Arno de Caigny, Kristof Coussement, Koen W. de Bock, Stefan Lessmann

Incorporating Textual Information in Customer Churn Prediction Models Based on a Convolutional Neural Network

Arno De Caigny[1], Kristof Coussement[1], Koen W. De Bock[2] and Stefan Lessmann[3]

[1]IÉSEG School of Management, Université Catholique de Lille (LEM, UMR CNRS 9221), Department of Marketing, 3 rue de la Digue, F-59000, Lille, France

[2]Audencia Business School, 8 route de la Jonelière, F-44312 Nantes, France

[3]School of Business and Economics, Humboldt University of Berlin, Unter-den-Linden 6, 10099 Berlin, Germany

E-mail addresses: a.de-caigny@ieseg.fr (Arno De Caigny), k.coussement@ieseg.fr (Kristof Coussement), kdebock@audencia.com (Koen W. De Bock), stefan.lessmann@hu-berlin.de (Stefan Lessmann)

Corresponding author: Kristof Coussement, 3 rue de la Digue, F-59000 Lille, France. Tel. (+33) 3 20 54 58 92; e-mail k.coussement@ieseg.fr.

Abstract

This study investigates the value added by incorporating textual data into customer churn prediction (CCP) models. It extends the previous literature by benchmarking convolutional neural networks (CNNs) against current best practices for analyzing textual data in CCP, and, using real life data from a European financial services provider, validates a framework that explains how textual data can be incorporated in a predictive model. First, the results confirm previous research showing that the inclusion of textual data in a CCP model improves its predictive performance. Second, CNNs outperform current best practices for text mining in CCP. Third, textual data are an important source of data for CCP, but unstructured textual data alone cannot create churn prediction models that are competitive with models that use traditional structured data. A calculation of the additional profit obtained from a customer retention campaign through the inclusion of textual information can be used by practitioners directly to help them make more informed decisions on whether to invest in text mining.

Keywords

Customer relationship management, text mining, predictive modeling, deep learning, financial services industry

1. Introduction

Successful companies in the 21st century engage with their customers proactively (Kumar & Shah, 2004). Shifting away from the traditional reaction-based marketing model, these companies have moved beyond reporting about the past, and instead use their data assets to provide better insights into the future by creating accurate prediction models. These models are deployed widely for the enhancement of business processes such as sales forecasting (Fildes, Goodwin, & Önkal, 2018; Trapero, Pedregal, Fildes, & Kourentzes, 2013) and for improving customer relationship management (CRM) systems by taking into account future aspects of customer relations with the company (Audzeyeva, Summers, & Schenk-Hoppé, 2012; Huang, 2012).

Especially in an era that is characterized by higher volumes, velocities, and varieties of data, more companies are recognizing the value of big data and adopting technologies that enable them to analyze these kinds of data (Raguseo, 2018). Approximately 95% of big data consist of unstructured data that can be obtained from various sources within or outside the company (Gandomi & Haider, 2015). In response, research around unstructured data is surging, and is being pursued in various domains (Sheng, Amankwah-Amoah, & Wang, 2017). However, despite this growing research attention, companies are still struggling to extract valuable information from textual data (Gandomi & Haider, 2015), though those that succeed in deploying data-driven decision systems perform better in terms of objective measures of financial and operational results (McAfee & Brynjolfsson, 2012).

A particularly prominent forecasting application in CRM is customer churn prediction (CCP), which is defined as a method of identifying customers who show a high inclination to abandon the company (Ganesh, Arnold, & Reynolds, 2000). Companies have clear financial incentives to establish accurate CCP models as part of their CRM strategy. First, successful companies establish long-term relationships with their most profitable customers. Relative to newer clients, long-term customers are characterized by higher retention rates (Dawes Farquhar, 2004; Reinartz & Kumar, 2002), are less prone to competitive marketing actions (Colgate, Stewart, & Kinsella, 1996), have a tendency to buy more, and, last but not least, are less costly to serve because the company already knows their preferences (Ganesh et al., 2000). Second, theory on social ties and homophily (Lazarsfeld & Merton,

1954) suggests that customers who leave the company may pull other clients within their social network out of the company (Nitzan & Libai, 2011), whereas loyal customers may attract others to the company by positive word of mouth (Ganesh et al., 2000). Third, losing customers increases the necessity to attract new customers and decreases profits by missed sales and lost opportunities for cross- and up-sales. Thus, it is important to predict future churn behavior and proactively make an effort to retain profitable customers.

A customer scoring model allows companies to estimate the future churn probability for each customer based on that customer's available historical data (Risselada, Verhoef, & Bijmolt, 2010). Churn scores facilitate the determination of which customers should be targeted with retention campaigns. Thus, not surprisingly, previous research in CCP has focused mainly on improving the performances of predictive churn models. The first stream of research involved testing and benchmarking different algorithms against each other (Verbeke, Dejaeger, Martens, Hur, & Baesens, 2012). Here, practitioners often face a trade-off between comprehensibility and predictive performance (De Caigny, Coussement, & De Bock, 2018), which explains why logistic regression, which provides comprehensible, accurate, and robust results, is one of the most popular algorithms in CCP (Coussement, Lessmann, & Verstraeten, 2017). Most studies consider models that solely use structured data. Researchers in a second stream have focused on data augmentation. These studies prove that integrating data that are not observable directly (Tang, Thomas, Fletcher, Pan, & Marshall, 2014) or are highly unstructured, such as in social networks (Benoit & Van den Poel, 2012), improves churn predictions. Such improvements are achieved by first processing unstructured data and then adding the result to a churn model as structured variables.

Many companies consider textual data to be a rich source of information (Raguseo, 2018). Since the use of text mining has proven valuable in CRM (Kumar & Ravi, 2016), the topic has been gaining importance in a wide range of research domains (Raguseo, 2018; Sheng et al., 2017). Nevertheless, the research regarding the use of textual data in CCP is limited and relatively old. Traditionally, textual data require rigorous data preprocessing and face challenges regarding dimensionality (Schneider & Gupta, 2016), which hampers their incorporation into predictive models.

However, practitioners who succeed in this difficult task can improve the predictive performance of their CCP model (Coussement & Van den Poel, 2008c).

Recently, new textual data processing algorithms have been developed that have achieved impressive performance gains in text classification tasks such as sentiment analysis and topic modeling (Zhang, Zhao, & LeCun, 2015). These algorithms use artificial neural networks (ANNs) with multiple layers stacked on top of each other to create a "deeper" ANN. This approach is often referred to as *deep learning*. On a methodological level, deep learning is an extension of the ANN architectures that are well known in the forecasting literature (Crone, Hibon, & Nikolopoulos, 2011; Heravi, Osborn, & Birchenhall, 2004; West & Dellana, 2011). A popular deep learning architecture is the convolutional neural network (CNN). CNNs use the mathematical convolution operation in at least one of their layers. This technique was developed initially for image recognition systems, and was inspired by the natural visual perception mechanism of living creatures (Fukushima, 1980). The modern architecture of the CNN (LeCun et al., 1990) has improved greatly over the years, and has been adapted to natural language processing for text classification tasks with good results (Collobert et al., 2011).

Besides the CNN, various other architectures can be used in text classification, of which an overview can be found in the literature (Zhang, Yang, Chen, & Li, 2018). In turn, these architectures can be combined to form deeper and more complex networks (Du, Gui, He, & Xu, 2018; Wang, Yu, Lai, & Zhang, 2016). No unequivocal answer to the question regarding the optimal architecture to use in text classification has ever been found, and it is possible that none ever will be found. As always, it depends on the data (Liu, Lang, Liu, & Yan, 2018; Wang, Xu et al., 2016).

This paper analyzes real-life textual data in the form of electronic messages between clients and their advisor on the platform of a financial services provider by means of a CNN, and integrates the result into a CCP model. We focus on CNNs because they represent a state-of-the art deep learning framework that has achieved excellent performances in text classification (Gu et al., 2017). The contribution of this paper is twofold. First, the state-of-the-art CNN is benchmarked with current best practices for integrating text into a CCP model. This is the first study in which CNNs have been used to process textual data for a CCP model. Second, insights into the importance of textual data for the

model's predictive performance are given and its impact on the profit of a retention campaign is discussed, offering valuable guidance for practitioners who opt to integrate textual data into their CCP model.

The remainder of this paper is structured as follows. Section 2 presents related work. Section 3 provides a framework that explains how textual data can be incorporated into a traditional CCP model and discusses the relevant methodology. Section 4 presents the experimental setup, and Section 5 presents the results with respect to predictive performances. Additional analyses regarding the importance of the variable categories and the implications of including textual data in a CCP model for the profit of a retention campaign are included. Section 6 concludes with a discussion of limitations and interesting areas for further research.

2. Related work

This section provides an overview of related research. The first paragraph presents applications of text mining in CRM research. Next, we discuss text mining in CCP. The section concludes with an overview of CCP in the financial services industry.

Kumar and Ravi (2016) provided a comprehensive literature review of text mining in CRM. They consider the identification of suitable feature selection methods to be an open problem. Thus, the absence of deep learning models for text mining in CRM is striking. Today, a vector space-based approach is often used to incorporate textual data in a customer scoring model (Coussement & Van den Poel, 2008c; Ravi, Ravi & Prasad, 2017). Section 3, on the methodology, elaborates further on the relevant models. Popular sources of text data are customer reviews (Coussement, Benoit, & Antioco, 2015) and tweets (He, Zha, & Li, 2013). The e-mail data of Coussement and Van den Poel (2008c) is the data source that has the most similarities with the data in this study (see Section 4).

Kumar and Ravi (2016) also refer to CCP specifically as a domain in which "text mining is not yet fully explored" (p. 144). However, industries are evolving in this big data era, with CCP models using data from various different sources, many of which require text mining (Shirazi & Mohammadi, in press). The limited research to date on the use of text mining in a CCP setting

processes the text and extracts relevant features to be used in a predictive model (Coussement & Van den Poel, 2008c; Schatzmann, Heitz, & Münch, 2014).

On the other hand, CCP models that use structured data have been researched very thoroughly in a broad range of industries (De Caigny et al., 2018). We focus on CCP in the financial services industry, given the dataset at our disposal. This industry is very suitable for and popular in CCP because (i) customers generally have long-term relationships with their financial institutions and spread their entire portfolio over only one or two companies (Mutanen, Ahola, & Nousiainen, 2006); and (ii) companies in the financial services industry collect relevant data from their clients (Lau, Chow, & Liu, 2004).

Table 1 provides an overview of the kinds of variables that have been included in previous CCP research in the financial services industry. Customer demographic and behavioral data have been included in most studies, but information about interactions between clients and the company has been exploited far less in previous research. The way in which a company communicates with its clients is important for customer relationship building (Gurau, 2008). Gür and Arıtürk (2014) include eight variables that reflect the customer–company interaction but do not describe them in detail. Moreover, these variables are derived from structured information and do not contain detailed information about the contents of the communication. In the focal study, both structured information about customer–company interactions, such as the type of message that is sent by the client, and unstructured textual information containing the specific content is incorporated into the CCP model.

Table 1: Overview of variables included in previous CCP research and this study.

Variable category	Type of variable	*Van Den Poel and Lariviere (2004)*	*Larivière and Van den Poel (2005)*	*Kumar and Ravi (2008)*	*Xie, Li, Ngai, and Ying (2009)*	*Nie, Rowe, Zhang, Tian, and Shi (2011)*	*Benoit and Van den Poel (2012)*	*De Bock and Van den Poel (2012)*	*Gür Ali and Arıtürk (2014)*	*Tang et al. (2014)*	*This study*
Customer demographics		**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
	Age	X	X	X	X	X	X	X	X	X	X
	Gender	X	X	X	-	X	X	X	X	X	X
	Social status related	X	-	X	-	X	X	-	-	-	X
	Marital status related	-	-	-	X	X	-	X	-	X	X
	Occupation related	-	-	-	X	X	-	X	-	-	X
	Financial status related	-	-	X	X	-	-	-	-	X	X
	Payment information	-	-	-	X	-	-	-	-	X	X
Customer behavior		**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**
	Standard products	X	X	-	X	X	X	X	-	-	X
	Savings and investments	X	X	-	-	X	X	X	-	-	X
	Risk-related products	-	X	-	-	-	X	-	-	-	X
	Credit products	X	X	X	X	X	X	X	-	-	X
	Total product ownership	X	X	-	-	-	-	-	-	-	X
	Card related	X	X	X	-	X	-	X	-	-	X
	Monetary value	X	X	X	-	-	-	-	-	-	X
	Products related to banking	-	-	-	-	-	-	X	-	-	X
	Transaction related	-	-	X	X	X	X	X	-	-	X
	Length of relationship	-	-	-	-	X	X	X	-	-	X
	Account balance related	-	-	-	-	-	-	X	-	-	X
	Loan history related	-	-	-	X	-	-	X	-	-	X
	Mortgage history related	-	-	-	-	-	-	X	-	-	X
Client/company contact		**-**	**-**	**-**	**-**	**-**	**-**	**-**	**X**	**-**	**X**
	Email specific	-	-	-	-	-	-	-	-	-	X
	Online connections	-	-	-	-	-	-	-	-	-	X
	Traditional contact	-	-	-	-	-	-	-	-	-	X

In this table, X indicates the presence of the variable category/type, while - indicates the absence of or insufficient information about the variable category/type.

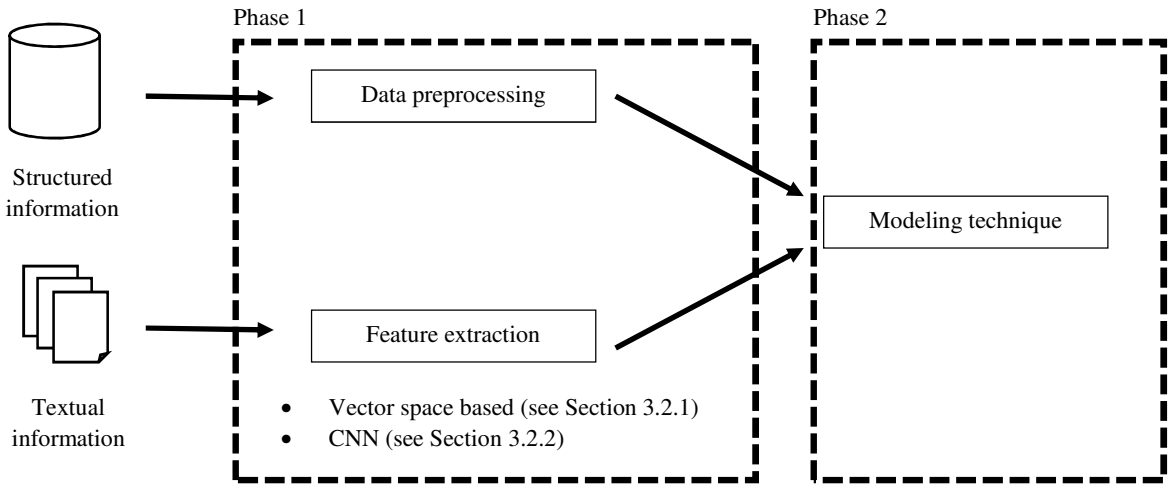
3. Methodology

This section provides an overview of the methodology. The first part presents a generic two-stage framework and discusses how textual and structured data can be combined and integrated in a predictive model. The second part zooms in on a key aspect of this study and discusses ways in which textual data can be processed. A vector space approach and a state-of-the-art deep CNN model are

discussed. The third part elaborates on the main research questions (RQs) that help to validate the framework in Section 3.1.

3.1. Framework for integrating textual data into a CCP model

Figure 1: Framework for integrating unstructured data into a predictive model.



This section presents a framework that can guide practitioners as to the way in which textual data can be incorporated into a predictive model. Figure 1 provides a high-level overview of the framework: the unstructured textual data and the structured data are handled separately in the first phase because there is a large difference in complexity between these two data types (Schneider & Gupta, 2016). The two-phase approach facilitates the integration of textual data into existing models that are built on structured data only (Coussement & Van den Poel, 2008c).

In the first phase, the structured and textual data are prepared for analysis. On the one hand, structured data must be preprocessed before they can be used in a predictive model. During this step, we use various different techniques to handle data-related particularities that may potentially distort the analysis, such as missing values, outliers, and class imbalance problems (Coussement et al., 2017; Moeyersoms & Martens, 2015). On the other hand, textual data also need to be preprocessed, but the extent depends on the feature extraction method. This study explores two different options for handling textual data (see Section 3.2).

In the second phase, the preprocessed structured and textual information are combined for integration into a predictive model as variables. Numerous algorithms have already been benchmarked in CCP (Verbeke et al., 2012). Although the best algorithm depends on the data available for the specific case, there are some tendencies: in general, ensemble methods such as random forests (Breiman, 2001) tend to perform well in CCP (Kumar & Ravi, 2008); nevertheless, logistic regression remains the industry standard because of its excellent mix of comprehensibility, predictive performance, and robustness (Coussement et al., 2017; Neslin, Gupta, Kamakura, Lu, & Mason, 2006).

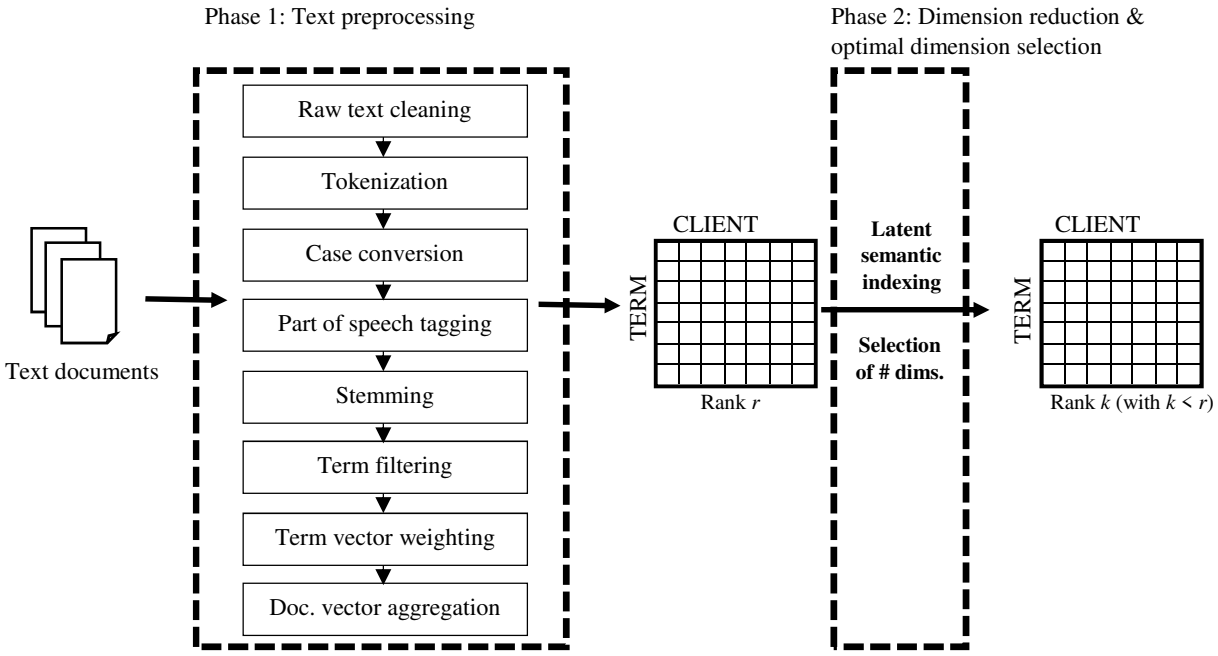
3.2. Processing textual data

The unstructured textual data should be transformed into data that can be exploited by a conventional data mining algorithm. Two approaches to the extraction of information from textual data are discussed. In the vector space based approach, this process consists of two phases: text preprocessing and dimension reduction. In a CNN approach, the textual data are transformed by passing through different layers.

3.2.1. Vector space based approach

This section provides a brief overview of the vector space text mining approach that is used as a benchmark for the CNN. The vector space approach is chosen because of its popularity in research and its good performance (Coussement et al. 2015; Coussement & Van den Poel, 2008b). It involves the original text documents being converted into a vector in a feature space based on the weighted term frequencies (Salton, 1989). As is presented in Figure 2 and discussed below, this process consists of two phases. More detailed overviews can be found in the literature (Coussement, De Bock, & Neslin, 2013, Chapter 2).

Figure 2: Vector space based approach for handling textual data.



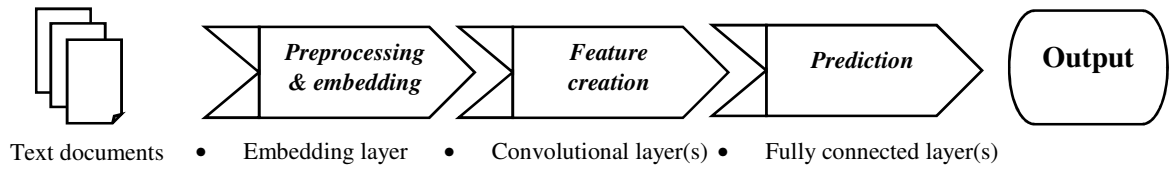
First, the raw text documents need to be preprocessed, since textual information cannot be incorporated into a model directly. This step includes basic *text cleansing,* such as removing special characters, and *tokenization,* which involves isolating single words in the documents. *Part-of-speech tagging* labels words with their syntactic category (e.g., nouns, verbs), and *stemming* involves replacing each term with its corresponding *stem*. For example, the term *discuss* is the stem for *discussion*, *discussed*, *discussing*, and so on. Stemming is used because it reduces the number of terms greatly (Bell & Jones, 1979). The number of terms in the corpus is then reduced further by *filtering irrelevant terms*. The first filtering step involves removing *stop words* (e.g. *a*, *the*). Non-informative terms are filtered based on their syntactic category, so that only the informative parts of the documents (nouns, verbs, adverbs, and adjectives) are kept. *Low-frequency words* are removed as well, because many words occur only once or twice (Coussement & Van den Poel, 2008c). Based on such a cleaned corpus, a weighted term vector is constructed for every message. *Term weighting* is often carried out by calculating the product of the term's frequency and the inverse document frequency (Salton, 1989; Salton & Buckley, 1988). Finally, these weighted *term vectors are aggregated* by client, as a given client could have sent multiple messages during the observation period (Coussement & Van den Poel, 2008c).

Second, the term-by-client matrix is reduced by means of a *dimension reduction* technique. This is necessary because the term-by-client matrix is a large, sparse matrix that is not fit to be used in a predictive model directly, because it will have many noisy variables (Martens, Provost, Clark, & Junqué De Fortuny, 2016). This process is often referred to as *latent semantic indexing*, where the dimensionality of the matrix is reduced by grouping together related terms (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990). There are many dimension-reduction techniques, such as principal component analysis and singular value decomposition, that are suitable for forming semantic generalizations from the messages (Deerwester et al., 1990); however, these techniques are unsupervised, meaning that the optimal number of dimensions is not known a priori. Therefore, the *optimal number of dimensions* for summarizing the content of the messages of the original term-by-client matrix must be approximated. Often, methods for doing so are quite arbitrary, such as the percent variance method, where the number of dimensions is based on an arbitrary percentage of the variance that must be explained, or the scree plot method, where the eigenvalues are plotted in descending order and one must find the "elbow" in the graph in order to determine the number of dimensions (Jolliffe, 2002). Non-arbitrary methods for deciding on the optimal number of dimensions are based on the application of cross-validation (CV hereafter; Coussement & Van den Poel, 2008c) or profile log-likelihood (Zhu & Ghodsi, 2006) to the data.

3.2.2. CNN approach for text

This section explains the application of CNNs in text mining. There are three important steps, as is summarized in Figure 3 and discussed in detail below.

Figure 3: Typical flow for text classification using a CNN.



First, textual information must be transformed into some kind of numerical input. The first step in this process is very basic preprocessing, such as case conversion (Collobert et al., 2011).

Afterwards, the documents are prepared so that each term is one-hot encoded. Unlike in the vector space based approach (see Section 3.2.1), more advanced preprocessing is often avoided because a second step involves *embedding* the terms as vectors in a low-dimensional continuous space. The dimensions in this space represent shared latent concepts. Embedding these terms in a lower-dimensional space offers two advantages for deep ANNs. First, ANNs often have computational difficulties with very high-dimension, sparse vectors, which are resolved by using term embeddings (Mikolov, Chen, Corrado, & Dean, 2013). Second, this process improves the model's generalization power, since similar words will have similar embedded vector representations (Collobert & Weston, 2008).

From a practical point of view, the textual data need to be transformed into a numerical *n*-dimensional vector that represents the textual information in an *n*-dimensional space. These continuous *n*-dimensional vector representations of words can be initialized either randomly or using pre-trained vectors. The pre-trained vectors are typically derived from very large datasets, with the objective of representing similar words more closely together (Mikolov et al., 2013). These word *embeddings* can be either updated during training, just like other model parameters, or kept static. Thus, the first layer is often referred to as an *embedding layer*, where the word embeddings are learned or fine-tuned jointly with the deep ANN model on a specific language processing task. A document is then simply a collection of all of the *embedded* vectors of the words that occur in a document. We make all documents the same length by either padding shorter documents with a special zero vector to make up for the missing words (Zhang & Wallace, 2015) or truncating longer documents to the desired length.

Second, we calculate new features from the embedded term vectors in the convolutional layer of the CNN. This paper uses complex layer terminology, where a convolutional layer is defined as a series of stages (Goodfellow, Bengio, & Courville, 2016, p. 336). In popular architectures, the first few stages consist of the convolutional stage described below, followed by a detector stage and a pooling stage. The main purpose of these stages is to extract more complex features.

CNNs are a special kind of ANN that uses the mathematical operation called *convolution* in at least one of its layers. A convolution of two functions g and f over an infinite range is defined as

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau, \quad (1)$$

where $t \in \mathbb{R}$ controls which part of the input function to emphasize with the weighting function and τ is a dummy variable that fixes t over all values of the integral.

The use of a convolutional stage improves the computational efficiency of the network in two ways. First, the units in a convolutional stage are organized in feature maps and connected to local patches in the feature maps of the previous layer through a set of weights (LeCun, Bengio, & Hinton, 2015). This implies that the neurons are connected only to a subset of the feature space of the previous layer, which is called *local connectivity*. Second, all units in the feature map share the same weights or filter banks, which is referred to as *parameter sharing*. The convolution is a linear operation, so nonlinearity is added to the model in the detector stage by passing the results of the local weighted sums through an activation function. This activation function passes the information about whether or not the output of the neuron is activated to outside connections. Next, the pooling stage improves the computational efficiency of the network further by reducing the spatial size of the representation and creating invariance toward small shifts and distortions to the input (LeCun et al., 2015). An arbitrary number of stages of convolution, nonlinearity, and pooling can be stacked together, creating more complex models.

Third, the actual classification requires the features created in the convolutional layer to be combined into a classifier. Typically, one or more *fully connected* (or *dense*) *layers* are used for this task (Kvamme, Sellereite, Aas, & Sjursen, 2018). A one-dimensional input is required for these dense layers, so we concatenate the columns in the last layer into a single vector x. The transformation z is given by $z = x^T W$, which is the inner product with a weight matrix W and some nonlinear activation function. The dimension of the output is therefore determined by the number of columns in W. In binary classification such as CCP, the final layer will have only two outputs. A common way of ensuring that the predictions are between zero and one is to use the softmax function as an activation function in the final layer; this is defined as

$$Softmax\ (z)_c = \frac{e^{z_c}}{\sum_j e^{z_j}}, \tag{2}$$

where *c* and *j* refer to classes. For binary classification, this function is equivalent to the logit function.

The training of the CNN requires the use of a loss function, which is a function that indirectly optimizes the true goal (Goodfellow et al., 2016). Binary classifications such as CCP often use binary cross-entropy (Kvamme et al., 2018), which is defined as

$$Loss = -\sum_i \{y_i \log p_i + (1 - y_i) \log\ (1 - p_i)\}, \tag{3}$$

where y_i denotes the true class label and p_i the prediction for customer i. Since the predictions for the customers move closer to their real values, the loss decreases, and, as a consequence, the objective improves by decreasing the loss. In the course of the actual model training, an iterative process starts in which parameters are set, the loss is calculated, and gradients are computed with respect to the parameters in the network. A method that typically uses gradient descent optimizes these parameters (Goodfellow et al., 2016, Chapter 8). Backpropagation enables the gradients to be calculated by applying the chain rule (LeCun, Bottou, Orr, & Muller, 1998).

Deep ANNs are prone to overfitting because of the large number of parameters (Gu et al., 2017). Two simple but powerful regularization techniques that reduce the risk of overfitting are used commonly. First, a validation set can be used to monitor the training process. As soon as the network starts to overfit on this set, the gradient descent iterations should be stopped. Consequently, this technique is called *early stopping*. Second, *dropout* is another way of reducing overfitting (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). This technique sets activations randomly to zero with a given probability during the training period, which prevents the units in the network from co-adapting too much. At test time, the activations are scaled according to the dropping rate by the dropout layer. A complete and more detailed overview of regularization techniques can be found in the literature (Goodfellow et al., 2016; Gu et al., 2017).

3.3. Research questions

In the era of big data, textual data provide opportunities for companies to improve existing processes (Sheng et al., 2017). The focal study takes place in the financial services industry, where

growing numbers of customers are engaging with their banks online. In the European Union, the average number of individuals who use the Internet for their banking activities has risen every year, to 51% by 2017 (Eurostat, 2018). In the Scandinavian countries, adoption is around 90%, and other countries are expected to reach similar levels in the near future. Moreover, the younger generation is more likely to adopt Internet and mobile banking services (Laukkanen, 2016). This suggests that online communications between clients and their banks will become an increasingly important source of data. Thus, it is important to obtain insights into whether this textual data can be used in predictive models. Since the focus of this study is on CCP, the first RQ is as follows.

RQ1: Does the addition of textual data, in the form of written electronic communications between a client and their advisor, improve the predictive performance of a CCP model?

It is crucial that the most efficient option for analyzing textual data and integrating it into a CCP be used, because even small increases in the retention rates of customers have large profit implications (Van den Poel & Larivière, 2004). Deep learning techniques have shown promising results for text mining, but they have never been applied to textual data in CCP. Therefore, it is of interest to benchmark the performances of these innovative techniques against current best practices in CCP in order to provide guidance regarding the direction of focus for further research. The next RQ is therefore as follows.

RQ2: What is the best feature extraction approach for integrating unstructured textual information into a CCP model?

Recently, new algorithms have been proposed that have demonstrated the potential for segmented modeling in CCP (De Caigny et al., 2018). Since customers with textual information could exhibit different churn patterns from those of customers without textual information (Coussement & Van den Poel, 2008c), the presence or absence of textual information for a customer may be an important segmenting variable. This leads to our next RQ, as follows.

RQ3: Is there a difference in churn patterns between customers with textual information and customers without textual information?

If only unstructured data are available, CCP can also be constructed using solely non-structured data. For example, often only textual data are available from users in online innovation communities (Coussement, Debaere, & De Ruyck, 2017). This bring us to the last RQ.

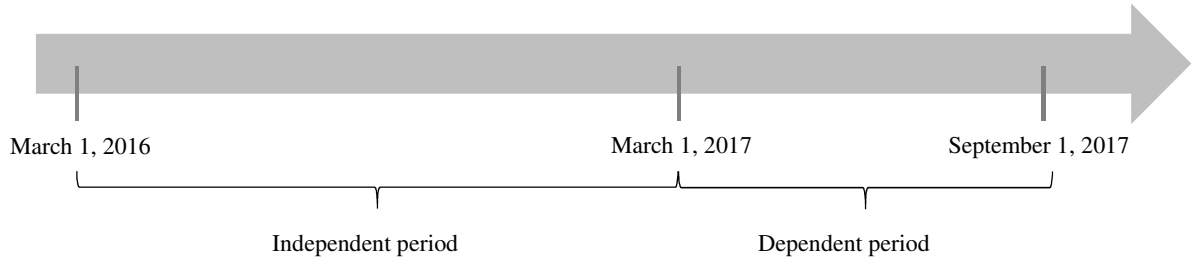
RQ4: Does a model built solely on textual data extract enough information to achieve a competitive predictive performance?

4. Experimental setup

This section presents the study's experimental setup. The first part presents the data. The second part describes the data pre-processing steps for both the structured and textual data. The third part delves deeper into the modeling, and the fourth part discusses the evaluation metrics.

4.1. Data

Figure 4: Study timeline.



The data for this study were provided by a large financial services provider in Europe. Figure 4 shows the timeline for constructing the independent and dependent variables in this study. We emphasize the roles of these periods in variable development by referring to them as the independent and dependent periods, respectively, in what follows. One year of information about a customer is used to predict the customer's churning behavior in the next six months. Thus, only information that was available in the independent period is used to predict the customer's churning behavior in the dependent period. This churning behavior refers to *complete churn*, meaning that the customer closes all products with the bank (Van den Poel & Larivière, 2004).

Most previous CCP studies in the financial services industry have included only structured information that is available readily in relational databases, whereas this study also uses textual data.

The full dataset covers 607,125 customers, 2.00% of whom closed all of their accounts in the dependent period. A total of 66,642 customers sent a message to their advisor during the period considered, 1.77% of whom constituted the churn in the dependent period.

The first type of information is the structured variables that are calculated by the company. Some of the variables that we obtained were not documented clearly, and we therefore left them out. We selected a total of 37 structured variables, based on an extensive literature review of CCP in the financial services industry. The variable categories and types selected are presented in Table 1. Most of these structured variables are a snapshot of the client at the end of the independent period, but some variables, such as the number of products bought during the last 12 months, summarize behavior over the last year.

The second type of information consists of all messages sent by clients to their advisor via the bank's protected platform during the last 10 months of the independent period. All such communication was in French. Because this information is highly unstructured, the messages were processed for representation in a CCP model.

We implement a variant of 5 × 2-fold CV (Dietterich, 1998), in which the data are split into training, selection, and validation sets, each of which contains one-third of the data, resulting in a 5 × 3-fold CV experimental design (De Caigny et al., 2018). All of the empirical results are based on the values of the performance measures from the holdout validation sets.

4.2. Data preprocessing

4.2.1. Structured data

First, values are imputed for attributes that are missing more than 5% of their values (Verbeke et al., 2012). Missing values are imputed with zero, the median, or the mode, depending on the variable (Coussement, Lessmann et al., 2017), and new dummy variables are created in order to flag variables for which missing values are imputed. Because clients with one missing value often have missing values for many other attributes too, and to limit the impact of imputation procedures, we

delete from the data clients that have missing values for attributes for which less than 5% of the values are missing (Verbeke et al., 2012).

Second, categorical variables are transformed into binary dummy variables that encode the presence or absence of a particular category. Thus, $v - 1$ dummy variables are created to represent the v unique values of each categorical variable (Pyle, 1999). However, there are only a few categorical variables in this study, so the effect of this variable transformation on the dimensionality of the feature space is limited.

Third, outliers are defined rather conservatively as unusual values that are more than three standard deviations away from the variable's mean (Freeman, Anderson, Sweeney, Williams, & Shoesmith, 2010). Such outliers are treated using winsorization, which transforms extreme values into less extreme values within an acceptable range (Ghosh & Vogt, 2012).

The fourth and last preprocessing step of the structured data involves a sampling technique. Typically, the dependent variable in the dataset for CCP is heavily unbalanced, meaning that the number of churning clients is much lower than the number of non-churning clients. As was discussed in Section 4.1, the dataset in this study shows an imbalance in the dependent customer churn variable. We remedy this problem by applying undersampling to the training data, such that the number of non-churning customers is reduced to the number of customers who churn by means of random sampling. Undersampling, as suggested by Weiss (2004), is applied widely in CCP (Burez & Van den Poel, 2009; De Bock & Van den Poel, 2012; Ling & Li, 1998).

4.2.2. Textual data

The vector space based approach requires us to process 189,665 text messages according to the literature (Coussement et al., 2015; Coussement & Van den Poel, 2008b,c) and as discussed in Section 2.2.1. The entire corpus consists of 79,914 unique terms, which are reduced to 60,737 terms after stemming. The part-of-speech tagger was trained on a French treebank corpus (Manning et al., 2014). The filtering process then reduces the number of unique terms to a workable size of under 1,000. Documents are aggregated by client in a term-by-client matrix, which in turn is reduced further

using latent semantic indexing. Semantic generalizations from the messages are formed by means of singular value decomposition (Deerwester et al., 1990). The optimal number of dimensions is chosen based on the profile log-likelihood (Zhu & Ghodsi, 2006), resulting in eight new text-based features z.

As was discussed in Section 2.2.2, the CNN approach does not require much pre-processing: often only word tokenization and case conversion. In this experiment, though, we also implement the same term filtering as in the vector space-based approach, to ensure that the same information is present in both approaches and no differences in performance can be explained by the filtering of terms in the vector space approach. The documents are padded with zeros to a length of 850, which is the size of the largest document with the exception of four outliers—at 1,042, 1,080, 1,425, and 2,410—which are truncated to 850. Our initial word embeddings are calculated by applying the continuous bag-of-words model (Mikolov et al., 2013) to the entire French Wikipedia corpus in a 200-dimensional vector space (Fauconnier, 2015). These word embeddings are then refined when the model is trained on the specific corpus. Unbalanced data can have a negative effect on the performance of CNNs (Buda, Maki, & Mazurowski, 2017), and as a consequence, we apply the same strategy for addressing unbalanced data as is used for the structured data, namely random undersampling (Buda et al., 2017). The observations in the training set for the CNN and for the final model are the same, and therefore, the clients in the validation set are never used to train either the CNN or the final model.

4.3. Model

This section provides an overview of the models used in this experiment. Table 2 summarizes the different types of models and their configurations.

Table 2: Model variations.

Model name	Data		Type of information		Feature extraction approach of unstructured data		Modeling technique
	Full dataset	Only customers involved in email communications	Structured	Text	Vector space	Deep learning	
Mod_FD	X		X		n.a.	n.a.	Logit
Mod_S		X	X		n.a.	n.a.	Logit
Mod_SU_VS		X	X	X	X		Logit
Mod_SU_DL_P		X	X	X		X[a]	Logit
Mod_SU_DL_L		X	X	X		X[b]	Logit
Mod_U_VS		X		X	X		Logit
Mod_U_DL_P		X		X		X	n.a.

[a] Deep learning method using probabilities.
[b] Deep learning method using the output of the last hidden layer.

4.3.1. CNN for textual data

We handle the textual data by constructing a non-static CNN according to the literature (Kim, 2014). The embedding layer is followed by one-dimensional convolutional layers and max pooling layers for every filter size (Kim, 2014). The outputs of these layers are flattened and concatenated so that they can be used as inputs for a fully connected layer (Kim, 2014). This last layer uses a softmax activation function for binary classification that generates a churn probability for every client. In the *Mod_SU_DL_P* model, these probabilities serve as a new variable for the CCP model, indicating the probability of a customer churning based on the customer's written communications with his/her advisor. These probabilities result from the output of the fully connected layer. The *Mod_SU_DL_L* model uses directly in the CCP model the features created by the convolutional layers that are found in the outputs of the artificial neurons in the last hidden layer. These features would normally serve as an input to the fully connected layer and summarize the textual information with respect to churning. Thus, the difference between *Mod_SU_DL_P* and *Mod_SU_DL_L* is the absence of the fully connected layer in the latter.

The parameter settings of the CNN can be found in Table 3. The parameters are based on previous studies in the literature that have used similar textual data for classification problems (Kim, 2014).

Table 3: Model parameters.

Parameter	Value	Reference
Word embeddings	200D, French Wikipedia	Fauconnier, 2015
Optimizer	Adam	Kingma & Ba, 2015; Kvamme et al., 2018
Filter windows	3, 4, 5	Kim, 2014
Number of filters	10	Kim, 2014
Dropout rate	0.5	Kim, 2014
Batch size	50	Kim, 2014
Hidden dimensions	100	Kim, 2014
L2 constraint	3	Kim, 2014

4.3.2. CCP model

The CCP model is built using logistic regression. This technique has proven to be very valuable in the CCP domain for two reasons: (1) the posterior probabilities are estimated directly in a logistic regression, which offers a comprehensible output, and (2) the results of logit models are robust and have good predictive performances in most benchmarking experiments in CCP, where they are often competitive with more complex techniques (Coussement et al., 2017; Neslin et al., 2006; Verbeke et al., 2012). Thus, logistic regression is considered the industry standard in CCP.

The variables are selected using forward selection (Coussement & Van den Poel, 2008a). The first variable to enter the model is that with the highest χ^2 statistic, and the remaining variables at each step are considered to be included in the final model until a certain stopping rule is satisfied. Table 4 provides an overview of the average number of variables that are included in the models. The average is calculated across the iterations of the 5 × 3 CV approach, where variable selection is performed on the training and selection partitions.

Table 4: Average number of variables included in the final model.

	Mod_FD	***Mod_S***	***Mod_SU_VS***	***Mod_SU_DL_P***	***Mod_SU_DL_L***
Avg. # variables [a]	17.0	16.8	19.0	17.5	22.6
Avg. # textual variables	/	/	2.6	1.0	6.9

[a] Textual variables included.

4.4. Evaluation

The predictive performances of the different models are assessed by the area under the receiver operating characteristics curve (AUC) and the top decile lift (TDL). These metrics are applied frequently to the evaluation of CCP models (Coussement et al., 2017; Lemmens & Croux, 2006; Verbeke et al., 2012).

The AUC is an independent cut-off measure that assesses the discriminatory power of the predicted churn probabilities. For all possible probability thresholds, it considers the sensitivity (the number of correctly predicted churners versus the total number of churners) and one minus the specificity (the number of correctly predicted non-churners versus the total number of non-churners) of the confusion matrix in a two-dimensional graph. The plot of these two outcomes forms the receiver operating characteristics curve. The area under this curve (AUC) can be used to evaluate the predictive performance of a binary classification system such as CCP with a simple one-figure score that lies between 0.5 and one (Hanley & McNeil, 1982). In customer churn, the AUC is defined as follows:

$$AUC = \int_{-\infty}^{\infty} sensitivity(T)(1 - specificity(T))dT, \quad (4)$$

where $T \in [0,1]$ is used as a threshold parameter (Hanley & McNeil, 1982).

The TDL, the second metric that we use for evaluating predictive performances, compares the proportion of churners in the entire dataset with the proportion of churners in the top decile, which contains the customers that have the highest churn probabilities according to the classification model. The lift for a specific cut-off $t \in [0,1]$ is defined as

$$lift = \frac{sensitivity}{ChurnRate}(t), \quad (5)$$

where the *ChurnRate* is the churn incidence percentage and the sensitivity is as defined above.

A TDL score of one indicates that the density of churners in the top decile is the same as that in the entire dataset, and therefore the model does not discriminate between churners and non-churners any better than taking a random sample of customers. Scores higher than one indicate a higher density of churners in the top decile, and vice versa. Often it is too expensive to send an incentive to all customers, so companies focus only on some of their clients. Thus, the TDL is a valuable performance metric from a managerial perspective because it focuses on those customers who are most at risk of

leaving the company and therefore are interesting customers to consider for a retention campaign (Neslin et al., 2006).

An *F*-statistic can be computed for comparing the results of a 5 × 3 cross validation, using the following formula:

$$f = \frac{\sum_{i=1}^{5}\sum_{j=1}^{3}(p_i^{(j)})^2}{3\sum_{i=1}^{5}s_i^2}. \quad (6)$$

This test statistic is approximately *F*-distributed with 15 and five degrees of freedom, where $p_i^{(j)}$ is the difference between the performance measure values of two classifiers and s_i^2 is the estimated variance (Alpaydin, 1999). Alternative frameworks for statistical comparisons of prediction models have been proposed in the literature (Demsar, 2006; García, Fernández, Luengo, & Herrera, 2010). However, these focus on comparisons of classifiers across multiple data sets, and thus do not fit the requirements of the focal study. Thus, subsequent statistical hypothesis tests use the *F*-statistic.

5. Results and discussion

This section discusses the results and main findings. First, the predictive performances of the different models are presented and discussed. These results answer the four RQs. Second, the importance of the different variable categories in the different models is analyzed. Last, the implications of including text data in a CCP model for the profitability of a retention campaign are derived and discussed.

5.1. Predictive performance

Table 5 presents the results in terms of the AUC and TDL for different types of models. More detailed statistical tests and information for each RQ are presented in Tables 6 to 9. The models with the highest AUC and TDL values are always indicated in bold.

Table 5: Results in terms of the AUC and TDL for different model variations.

	AUC	TDL
Mod_FD	86.938% (0.008)	5.867 (0.225)
Mod_S	87.810% (0.008)	6.188 (0.251)
Mod_SU_VS	87.885% (0.007)	6.211 (0.224)
Mod_SU_DL_P	89.666% (0.010)	6.789 (0.282)
Mod_SU_DL_L	**89.875% (0.010)**	**6.868 (0.335)**
Mod_U_VS	54.102% (0.018)	1.269 (0.206)
Mod_U_DL_P	73.056% (0.036)	3.642 (0.624)

The first RQ concerns the differences in predictive performances between models with and without textual information. The results of a pairwise comparison of the models considered are presented in Table 6, and show that the integration of textual information into a CCP model can improve the predictive performance significantly as measured by both the AUC and TDL, but does not necessarily do so. Adding the unstructured information as processed by the vector space-based approach does not improve the performance of the CCP model significantly in terms of either the TDL or the AUC (both p-values > 0.05). However, the addition of the unstructured data processed by a deep CNN model does improve the predictive performance of the CCP model significantly. Therefore, as is discussed in the next point, the approach that is chosen for handling the unstructured data does have an effect on the predictive performance, and multiple options should be investigated when integrating textual data into a CCP model.

Table 6: Pairwise comparison between the *Mod_S* and *Mod_SU_X* models.

	AUC		TDL	
	***F*-value**	**Adj. *p***	***F*-value**	**Adj. *p***
Mod_S vs. ***Mod_SU_VS***	1.149	0.477	0.927	0.589
Mod_S vs. ***Mod_SU_DL_P***	17.741	0.006***	6.940	0.057*
Mod_S vs. ***Mod_SU_DL_L***	18.904	0.006***	7.295	0.057*

Notes: The resulting p-values of the F-test are applied for a pairwise comparison of the performances of all possible combinations (Alpaydin, 1999). The adjusted p-values are calculated using Holm's (1979) post hoc procedure. Significant differences at the 90%, 95%, and 99% levels are indicated by *, **, and ***, respectively.

Second, we examine the best way of integrating textual information into the CCP model. Table 7 provides an overview of the pairwise comparison between the best-performing model and two other models that we considered. The deep learning approach for processing the unstructured data outperforms the vector space-based approach. Using a supervised deep CNN that is optimized for the

specific task instead of adding a number of additional features that summarize the text clearly improves the predictive performance. There are no significant differences in predictive performance between *Mod_SU_DL_L* and *Mod_SU_DL_P*. In practice, though, the latter is easier for practitioners to interpret, since it summarizes a customer's entire textual data into a single feature that represents the customer's churn probability.

Table 7: Pairwise comparison between the *Mod_SU_X* models.

	AUC		TDL	
	***F*-value**	**Adj. *p***	***F*-value**	**Adj. *p***
Mod_SU_DL_L vs. *Mod_SU_VS*	13.388	0.010***	6.702	0.046**
Mod_SU_DL_L vs. *Mod_SU_DL_P*	2.055	0.219	1.949	0.238

Notes: The resulting *p*-values of the *F*-test are applied for a pairwise comparison of the performances of all possible combinations (Alpaydin, 1999). The adjusted *p*-values are calculated using Holm's (1979) post hoc procedure. Significant differences at the 90%, 95% and 99% levels are indicated by *, **, and ***, respectively.

Third, whereas textual data are not always available directly or easy for practitioners to access, information about whether or not a client has sent an email often is. Therefore, the third RQ investigates whether clients with and without email communications have different churn patterns. In line with previous research (Coussement & Van den Poel, 2008c) and based on the results presented in Table 8, a separate CCP model for clients that engage in email communication with the company would be beneficial, confirming that these groups of clients exhibit different churn patterns.

Table 8: Pairwise comparison between *Mod_FD* and *Mod_S*.

	AUC		TDL	
	***F*-value**	***p***	***F*-value**	***p***
Mod_FD vs. ***Mod_S***	4.466	0.054*	6.923	0.021**

Notes: The resulting *p*-values of the *F*-test (Alpaydin, 1999). Significant differences at the 90%, 95%, and 99% levels are indicated by *, **, and ***, respectively.

Fourth, we answer the last RQ by comparing models that are built solely on unstructured data with the model that is built on structured data. The results of this analysis are presented in Table 9. Clearly, models that use only textual data are not competitive with models that use structured information. In the case where only unstructured textual data are available, the deep CNN outperforms the vector space based approach. In analogy with previous research (Collobert et al., 2011; Kim, 2014;

LeCun et al., 2015), these results further support the potential of deep learning in text classification tasks.

Table 9: Pairwise comparison between the *Mod_S* and *Mod_U_X* models.

	AUC		TDL	
	***F*-value**	**Adj. *p***	***F*-value**	**Adj. *p***
Mod_S vs. *Mod_U_VS*	320.648	0.000***	133.999	0.000***
Mod_S vs. *Mod_U_DL_P*	19.172	0.002***	14.354	0.004***
Mod_U_VS vs. ***Mod_U_DL_P***	30.203	0.001*** [a]	17.937	0.002*** [a]

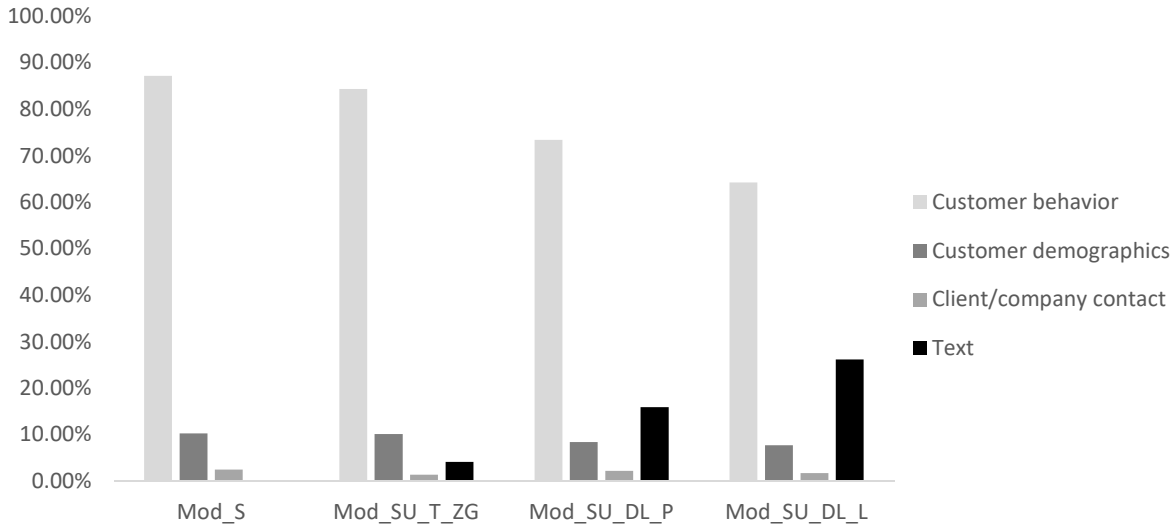
Notes: The resulting *p*-values of the *F*-test are applied for a pairwise comparison of the performances of all possible combinations (Alpaydin, 1999). The adjusted *p*-values are calculated using Holm's (1979) post hoc procedure. Significant differences at the 90%, 95%, and 99% levels are indicated by *, **, and ***, respectively.

[a] The resulting *p*-values are presented because they did not require correction for family-wise error.

5.2. Variable category importance

We perform a meta-analysis in order to assess the relative importance of each different variable category based on the absolute values of the Wald statistic for the individual predictors. The Wald statistic is linked directly with the normalized β coefficients of the variables in the logistic regression model. The results of this analysis are shown in Figure 5 for the different models. Given the existence of multiple logistic regressions models because of the 5 × 3 CV experimental design, we opted to present the average importance of each aggregated variable category instead of a traditional logistic table, which would represent only a single model. It becomes clear from this analysis that customer behavior variables are the most important in predicting customer churn, which confirms previous research (Van den Poel & Larivière, 2004; Verbeke, Martens, Mues, & Baesens, 2011). Customer demographic variables contribute around 10% of the variable importance, while variables that describe the direct contact between the client and the company account for only around 2%. Not surprisingly, the communications between a client and his/her advisor are a good source for the prediction of churning behavior, with a relative importance of between 5% and 25%. The deep learning model clearly extracts more valuable features from the textual data, thereby boosting the importance of this category.

Figure 5: Relative importance of variable categories, by model.



5.3. Profit implications

The profit from a customer retention campaign can be expressed by the following formula (Neslin et al., 2006):

$$Profit = N\eta\big[\big(\gamma CLV + d(1-\gamma)\big)\pi_0\lambda - d - f\big] - A, \tag{7}$$

where N is the number of clients in the company, η the fraction of the customer base targeted by the campaign, CLV the average customer lifetime value (which is lost if a customer churns), d the cost of the incentive, f the cost of contacting the customer, and A fixed administrative costs. The lift coefficient, λ, is the percentage of customers who churn within the targeted fraction of η customers, divided by the overall churn rate, π_0. The fraction of would-be churners who accept the offer, or, alternatively, the probability of a targeted churner accepting the offer and thus not churning, is indicated by γ. The dynamics of this function are illustrated nicely by Verbraken, Verbeke, and Baesens (2013). Considering that valuable textual data are available for a fraction of m customers and assuming that the same fraction of η customers is targeted for customers with and without textual information, the formula can be rewritten as

$$\begin{aligned} Profit = {} & Nm\eta\big[\big(\gamma CLV + d(1-\gamma)\big)\pi_0\lambda_t - d - f\big] + \\ & N(1-m)\eta\big[\big(\gamma CLV + d(1-\gamma)\big)\pi_0\lambda - d - f\big] - A, \end{aligned} \tag{8}$$

where λ_t is the improved lift coefficient for the model when textual information is included. The additional net profit due to the improvement in predictive performance as a result of the inclusion of the textual data can then be expressed as

$$Profit_{text} = Nm\eta\left[\left(\gamma CLV + d(1-\gamma)\right)\pi_0\lambda_t\right]. \quad (9)$$

The values for these parameters are given by the company, and *CLV*, *d* and *f* are assumed to be the same for both groups. Considering that customers with textual information communicate electronically with the company, the cost of contacting customers, *f*, could be even lower, and thus, $profit_{text}$ even higher.

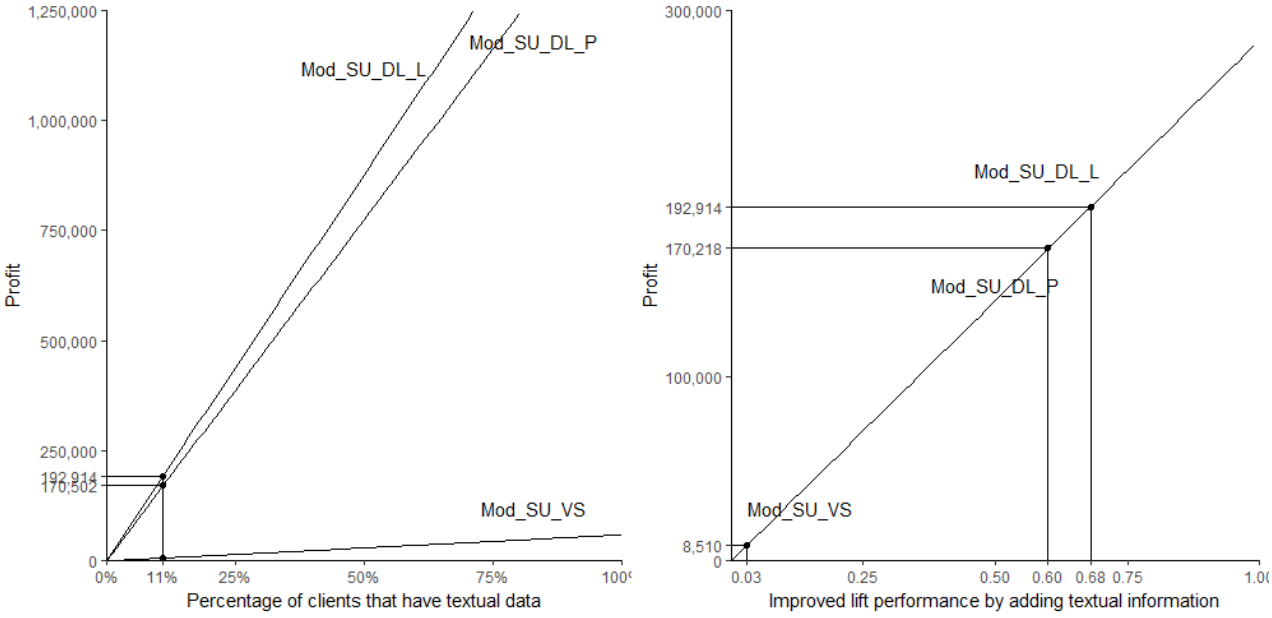
Using Eq. (9), a company can perform a sensitivity analysis which is similar to that of Verbraken et al. (2013). Table 10 provides an overview of the parameters required for calculating additional profits based on Eq. (9). Note that variables that are not necessary for calculating the additional profit of a retention campaign, such as the contact cost, are not included in this table. The parameter values were provided by the company for the specific case of this study. Some of the parameters are straightforward to determine, such as the total number of clients. In addition, most banks traditionally invest in customer retention campaigns, which provide them with accurate parameter forecasts and allow them to calculate profits correctly.

Methods for *CLV* prediction have also been studied extensively in the literature (Audzeyeva et al., 2012; Haenlein, Kaplan, & Beeser, 2007), providing banks with the tools necessary to estimate accurately the additional profit of including text in a CCP campaign. Using the values presented for the parameters, we estimate the additional profit from a retention campaign that includes textual information to be €192.914. Given the average *CLV* value, this is equivalent to retaining about 1% more of the churners. An additional sensitivity analysis is performed for changes in the two parameters that are directly influenced by the textual data, *m* and ($\lambda_t - \lambda$). Figure 6 illustrates the impacts of different values on profit. The values that are set are noted explicitly. The impacts of the top decile improvements of the different models in the experiment are illustrated clearly in the second graph of Figure 6.

Table 10: Parameters for the profit function and values provided by the company.

Variable and description		Value
N	Total number of clients	607.125
m	Percentage of clients that have textual data	11
η	Percentage of clients that are targeted	10
γ	The fraction of would-be churners that accept the offer	10
CLV	The average customer lifetime value	1500
π_0	The overall churn incidence	1.77
$(\lambda_t - \lambda)$	The improvement in lift performance from the addition of textual information	0.68
d	The cost of the incentive	100

Figure 6: Sensitivity of the parameters m and $(\lambda_t - \lambda)$.



6. Conclusions and future research

The framework presented in this study provides guidance as to the way in which textual data can be incorporated in a CCP model. The empirical results demonstrate that textual information is an important source of data for CCP models, with the feature extraction approach of the textual data having a significant impact on the overall predictive performance. We also demonstrate that the CNN extracts the most valuable features from the unstructured textual data.

This study has several implications for both theory and practice. The first implication pertains to the CNN's good predictive performance in analyzing textual data. Future studies that incorporate unstructured data into CCP should consider CNNs a challenging benchmark. Second, this study provides evidence of the performance improvement that can be achieved by augmenting the data for a CCP model with textual features and linking it back to the profits of a retention campaign. This case offers companies all of the building blocks necessary to make a well-informed investment decision.

This study also has a number of limitations that should be addressed in future research. First, rich datasets that include both structured and unstructured data are rarely available publicly. In this study, a company provided us with a real-life dataset, as is common practice in CCP modeling. Nonetheless, as in any empirical study, the results of this study hold for our data but cannot necessarily be generalized to other settings. Despite the use of formal statistical tools for testing for significance, perfect external validity can never be ensured. Thus, it would be valuable to replicate the study and test the framework using other data, for example in other industries, in order to confirm the above-mentioned conclusions.

It is noteworthy that the framework that we provide can be extended easily to work with other sources of unstructured data. There are several interesting sources of unstructured data in the financial services industry, such as bank transaction data (Martens et al., 2016), where deep learning techniques have considerable potential. The CNN that is used to analyze the textual information in this study can also be applied to time series data, for which recent studies have showed promising results in other applications, such as mortgage defaults (Kvamme et al., 2018). On the other hand, the value of textual information in CCP that this study demonstrates suggests that many other applications could benefit from it as well.

Second, this study has clearly demonstrated the link between profitability and predictive performance in a CCP model. The methods used in this study are based on the literature and are not too complex to ensure that the framework is applicable widely. Besides CNN, there are also many other types of deep ANNs, such as recurrent neural networks and attention-based models. Given their high computational requirements, it is not practical to benchmark the entire spectrum of deep learning

architectures in this study; however, the exploration of other architectures offers many opportunities for future research.

Third, the CNN in this study clearly extracts valuable features from the textual data, which boosts the predictive performance of the CCP model. However, this does not indicate what type of information the textual data actually provide. This information could be valuable for detecting specific churn drivers, which could help managers to set up better-targeted retention campaigns and reduce churn rates in the long run. Therefore, further research on the interpretability of deep ANNs will be useful.

References

Alpaydin, E. (1999). Combined 5x2 cv F test for comparing supervised classification learning algorithms. *Neural Computation*, *11*, 1885–1892.

Audzeyeva, A., Summers, B., & Schenk-Hoppé, K. R. (2012). Forecasting customer behaviour in a multi-service financial organisation: A profitability perspective. *International Journal of Forecasting*, *28*, 507–518.

Bell, C., & Jones, K. P. (1979). Towards everyday language information retrieval systems via minicomputers. *Journal of the American Society for Information Science*, *30*, 334–339.

Benoit, D. F., & Van den Poel, D. (2012). Improving customer retention in financial services using kinship network information. *Expert Systems with Applications*, *39*, 11435–11442.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32.

Buda, M., Maki, A., & Mazurowski, M. A. (2017). A systematic study of the class imbalance problem in convolutional neural networks. *CoRR*, *1710.05381*, 1–23.

Burez, J., & Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, *36*, 4626–4636.

Colgate, M., Stewart, K., & Kinsella, R. (1996). Customer defection: a study of the student market in Ireland. *International Journal of Bank Marketing*, *14*, 23–29.

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning* (pp. 160–167).

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, *12*, 2493–2537.

Coussement, K., Benoit, D. F., & Antioco, M. (2015). A Bayesian approach for incorporating expert opinions into decision support systems: A case study of online consumer-satisfaction detection. *Decision Support Systems*, *79*, 24–32.

Coussement, K., De Bock, K. W., & Neslin, S. A. (2013). *Advanced database marketing: innovative methodologies and applications for managing customer relationships*. Routledge, London, UK.

Coussement, K., Lessmann, S., & Verstraeten, G. (2017). A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry. *Decision Support Systems*, *95*, 27–36.

Coussement, K., & Van den Poel, D. (2008a). Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert Systems with Applications*, *34*, 313–327.

Coussement, K., & Van den Poel, D. (2008b). Improving customer complaint management by automatic email classification using linguistic style features as predictors. *Decision Support Systems*, *44*, 870–882.

Coussement, K., & Van den Poel, D. (2008c). Integrating the voice of customers through call center emails into a decision support system for churn prediction. *Information and Management*, *45*, 164–174.

Crone, S. F., Hibon, M., & Nikolopoulos, K. (2011). Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction. *International Journal of Forecasting*, *27*, 635–660.

Dawes Farquhar, J. (2004). Customer retention in retail financial services: an employee perspective. *International Journal of Bank Marketing*, *22*, 86–99.

De Bock, K. W., & Van den Poel, D. (2012). Reconciling performance and interpretability in customer

churn prediction using ensemble learning based on generalized additive models. *Expert Systems with Applications*, *39*, 6816–6826.

De Caigny, A., Coussement, K., & De Bock, K. W. (2018). A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research*, *269,* 760–772.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*, 391–407.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*, 1895–1923.

Du, J., Gui, L., He, Y., & Xu, R. (2018). A convolutional attentional neural network for sentiment classification. In *2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)* (pp. 445-450).

Eurostat (2018). Individuals using the Internet for Internet banking. Retrieved from http://ec.europa.eu/eurostat/web/products-datasets/-/tin00099 on May 10, 2018.

Fauconnier, J.-P. (2015). French word embeddings. Retrieved from http://fauconnier.github.io on March 10, 2018.

Fildes, R., Goodwin, P., & Önkal, D. (2018). Use and misuse of information in supply chain forecasting of promotion effects. *International Journal of Forecasting*, *35*(1), 144-156.

Freeman, J., Anderson, D., Sweeney, D., Williams, T., & Shoesmith, E. (2010). *Statistics for business and economics* (2nd ed.). Cengage, Boston.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*, 193–202.

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*, 137–144.

Ganesh, J., Arnold, M. J., & Reynolds, K. E. (2000). Understanding the customer base of service providers: an examination of the differences between switchers and stayers. *Journal of Marketing*, *64*, 65–87.

García, S., Fernández, A., Luengo, J., & Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, *180*, 2044–2064.

Ghosh, D., & Vogt, A. (2012). Outliers: An evaluation of methodologies. In *Joint statistical meetings* (pp. 3455–3460). American Statistical Association, San Diego, CA.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning.* MIT Press, Cambridge, Massachusetts.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2017). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354-377.

Gür Ali, Ö., & Arıtürk, U. (2014). Dynamic churn prediction framework with more effective use of rare event data: The case of private banking. *Expert Systems with Applications*, *41*, 7889–7903.

Gurau, C. (2008). Integrated online marketing communication: implementation and management. *Journal of Communication Management*, *12*, 169–184.

Haenlein, M., Kaplan, A. M., & Beeser, A. J. (2007). A model to determine customer lifetime value in a retail banking context. *European Management Journal*, *25*, 221–234.

Hanley, A. J., & McNeil, J. B. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, *143*, 29–36.

He, W., Zha, S., & Li, L. (2013). Social media competitive analysis and text mining: A case study in the pizza industry. *International Journal of Information Management*, *33*, 464–472.

Heravi, S., Osborn, D. R., & Birchenhall, C. R. (2004). Linear versus neural network forecasts for European industrial production series. *International Journal of Forecasting*, *20*, 435–446.

Holm, S. (1979). A simple sequentiallly rejective multiple test procedure. *Scandinavian Journal of Statistics*, *6*, 65–70.

Huang, C. Y. (2012). To model, or not to model: Forecasting for customer prioritization. *International Journal of Forecasting*, *28*, 497–506.

Jolliffe, I. T. (2002). *Principal component analysis* (2nd ed.). Springer, New York.

Kim, Y. (2014). *Convolutional neural networks for sentence classification*. arXiv:1408.5882.

Kingma, D. P., & Ba, J. L. (2015). Adam: a method for stochastic optimization. In *International Conference on Learning Representations 2015* (pp. 1–15).

Kumar, B. S., & Ravi, V. (2016). A survey of the applications of text mining in financial domain. *Knowledge-Based Systems*, *114*, 128–147.

Kumar, D. A., & Ravi, V. (2008). Predicting credit card customer churn in banks using data mining. *International Journal of Data Analysis Techniques and Strategies*, *1*, 4–28.

Kumar, V., & Shah, D. (2004). Building and sustaining profitable customer loyalty for the 21st century. *Journal of Retailing*, *80*, 317-329.

Kvamme, H., Sellereite, N., Aas, K., & Sjursen, S. (2018). Predicting mortgage default using convolutional neural networks. *Expert Systems with Applications*, *102*, 207–217.

Larivière, B., & Van den Poel, D. (2005). Predicting customer retention and profitability by using random forests and regression forests techniques. *Expert Systems with Applications*, *29*, 472–484.

Lau, K.-N., Chow, H., & Liu, C. (2004). A database approach to cross selling in the banking industry: Practices, strategies and challenges. *Journal of Database Marketing and Customer Strategy Management*, *11*, 216–234.

Laukkanen, T. (2016). Consumer adoption versus rejection decisions in seemingly similar service innovations: The case of the Internet and mobile banking. *Journal of Business Research*, *69*, 2432–2439.

Lazarsfeld, P. F., & Merton, R. K. (1954). Friendship as a social process. *Freedom and Control in Modern Society*, *18*, 18–66.

Le Cun, J., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. *Advances in Neural Information Processing Systems*, *2*, 396–404.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, 436–444.

LeCun, Y., Bottou, L., Orr, G., & Muller, K. (1998). *Neural networks: tricks of the trade.* Springer Lecture Notes in Computer Science.

Lemmens, A., & Croux, C. (2006). Bagging and boosting classification trees to predict churn. *Journal of Marketing Research*, *43*, 276–286.

Ling, C. X., & Li, C. (1998). Data mining for direct marketing: problems and solutions. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining* (pp. 73–79).

Liu, H., Lang, B., Liu, M., & Yan, H. (2018). CNN and RNN based payload classification methods for attack detection. *Knowledge-Based Systems*, *163*, 332-341.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 55–60).

Martens, D., Provost, F., Clark, J., & Junqué De Fortuny, E. (2016). Mining massive fine-grained behavior data to improve predictive analytics. *MIS Quarterly*, *40*, 869–888.

McAfee, A., & Brynjolfsson, E. (2012). Big data. The management revolution. *Harvard Business*

Review, *90*, 61–68.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. arXiv:1301.3781.

Moeyersoms, J., & Martens, D. (2015). Including high-cardinality attributes in predictive models: A case study in churn prediction in the energy sector. *Decision Support Systems*, *72*, 72–81.

Mutanen, T., Ahola, J., & Nousiainen, S. (2006). Customer churn prediction – a case study in retail banking. In *Proceedings of ECML/PKDD Workshop on Practical Data Mining* (pp. 13-19).

Neslin, S. A., Gupta, S., Kamakura, W., Lu, J., & Mason, C. H. (2006). Defection detection: measuring and understanding the predictive accuracy of customer churn models. *Journal of Marketing Research*, *43*, 204–211.

Nie, G., Rowe, W., Zhang, L., Tian, Y., & Shi, Y. (2011). Credit card churn forecasting by logistic regression and decision tree. *Expert Systems with Applications*, *38*, 15273–15285.

Nitzan, I., & Libai, B. (2011). Social effects on customer retention. *Journal of Marketing*, *75*(6), 24-38.

Pyle, D. (1999). *Data preparation for data mining*. San Francisco: Morgan Kaufmann Publishers.

Raguseo, E. (2018). Big data technologies: An empirical investigation on their adoption, benefits and risks for companies. *International Journal of Information Management*, *38*, 187–195.

Ravi, K., Ravi, V. & Prasad, P. (2017). Fuzzy formal concept analysis based opinion mining for CRM in financial services. *Applied Soft Computing*, *60*, 786-807.

Reinartz, W., & Kumar, V. (2002). The mismanagement of customer loyalty. *Harvard Business Review*, *80*(7), 86-94.

Risselada, H., Verhoef, P. C., & Bijmolt, T. H. A. (2010). Staying power of churn prediction models. *Journal of Interactive Marketing*, *24*, 198–208.

Salton, G. (1989). *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, *24*, 513–523.Schatzmann, A., Heitz, C., & Münch, T. (2014). Churn prediction based on text mining and CRM data analyis. In *Conference proceedings of the 13th international science-to-business marketing conference on cross organizational value creation* (pp. 296–310).

Schneider, M. J., & Gupta, S. (2016). Forecasting sales of new and existing products using consumer reviews: A random projections approach. *International Journal of Forecasting*, *32*, 243–256.

Sheng, J., Amankwah-Amoah, J., & Wang, X. (2017). A multidisciplinary perspective of big data in management research. *International Journal of Production Economics*, *191*, 97–112.

Shirazi, F., & Mohammadi, M. (in press). A big data analytics model for customer churn prediction in the retiree segment. *International Journal of Information Management*, in press.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

Tang, L., Thomas, L., Fletcher, M., Pan, J., & Marshall, A. (2014). Assessing the impact of derived behavior information on customer attrition in the financial service industry. *European Journal of Operational Research*, *236*, 624–633.

Trapero, J. R., Pedregal, D. J., Fildes, R., & Kourentzes, N. (2013). Analysis of judgmental adjustments in the presence of promotions. *International Journal of Forecasting*, *29*, 234–243.

Van den Poel, D., & Larivière, B. (2004). Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, *157*, 196–217.

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European*

Journal of Operational Research, *218*, 211–229.

Verbeke, W., Martens, D., Mues, C., & Baesens, B. (2011). Building comprehensible customer churn prediction models with advanced rule induction techniques. *Expert Systems with Applications*, *38*, 2354–2364.

Verbraken, T., Verbeke, W., & Baesens, B. (2013). A novel profit maximizing metric for measuring classification performance of customer churn prediction models. *IEEE Transactions on Knowledge and Data Engineering*, *25*, 961–973.

Wang, J., Yu, L.-C., Lai, K. R., & Zhang, X. (2016). Dimensional sentiment analysis using a regional CNN-LSTM Model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Volume 2: Short Papers; pp. 225–230).

Wang, P., Xu, B., Xu, J., Tian, G., Liu, C. L., & Hao, H. (2016). Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing*, *174*, 806–814.

Weiss, G. M. (2004). Mining with rarity: a unifying framework. *SIGKDD Explorations*, *6*, 7–19.

West, D., & Dellana, S. (2011). An empirical analysis of neural network memory structures for basin water quality forecasting. *International Journal of Forecasting*, *27*, 777–803.

Xie, Y., Li, X., Ngai, E. W. T., & Ying, W. (2009). Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, *36*, 5445–5449.

Zhang, Q., Yang, L. T., Chen, Z., & Li, P. (2018). A survey on deep learning for big data. *Information Fusion*, *42*, 146-157.

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. *Advances in Neural Information Processing Systems*, *28*, 649–657.

Zhang, Y., & Wallace, B. C. (2015). *A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification*. arXiv:1510.03820.

Zhu, M., & Ghodsi, A. (2006). Automatic dimensionality selection from the scree plot via the use of profile likelihood. *Computational Statistics and Data Analysis*, *51*, 918–930.