Deep Convolutional Neural Networks for Customer Churn Prediction Analysis

Alae Chouiekh, Laboratory of Multimedia, Signal and Communications Systems, National Institute of Posts and Telecommunications, Rabat, Morocco

El Hassane Ibn El Haj, Laboratory of Multimedia, Signal and Communications Systems, National Institute of Posts and Telecommunications, Rabat, Morocco

ABSTRACT

Several machine learning models have been proposed to address customer churn problems. In this work, the authors used a novel method by applying deep convolutional neural networks on a labeled dataset of 18,000 prepaid subscribers to classify/identify customer churn. The learning technique was based on call detail records (CDR) describing customers activity during two-month traffic from a real telecommunication provider. The authors use this method to identify new business use case by considering each subscriber as a single input image describing the churning state. Different experiments were performed to evaluate the performance of the method. The authors found that deep convolutional neural networks (DCNN) outperformed other traditional machine learning algorithms (support vector machines, random forest, and gradient boosting classifier) with F1 score of 91%. Thus, the use of this approach can reduce the cost related to customer loss and fits better the churn prediction business use case.

KEYWORDS

Churn Prediction, ConvNets, DCNN, Deep Learning, Machine Learning, Telecommunications

1. INTRODUCTION

During the last decade competition became a real concern for telecommunication providers (Bin et al., 2007). Thus, operators are poised to find new methods to enhance the quality of their services and diversify periodically their portfolio to retain the existing customers and attract new ones. While the primary focus of each telecom service provider is to provide customer service satisfaction, preventing subscribers from churning remains a huge challenge. Churn in telecoms is the term used to collectively describe the ceasing of customer subscriptions to a service (Huang et al., 2010), and if one customer cancels his/her service and switches to another operator, this customer is considered as a churner. Converging lines of evidence showed that the cost for customer acquisition is much greater than the cost of customer retention (in some cases~20 times more expensive) (Vafeidis et al., 2015). Thus, it is compulsory to telecom service providers to identify unsatisfied subscribers to prevent them from churning. For this, developing reliable predictive models to predict customer churn are crucial for the business management of the telecom industry.

Telecommunication companies consider customer churn a real and serious common business problems that should be addressed very carefully to avoid the loss of potential subscribers. In our work, we focused on the prepaid subscribers, a category of customers who can terminate their service subscriptions and switch to another telecom provider without prior notice. For instance, we found

DOI: 10.4018/IJCINI.2020010101

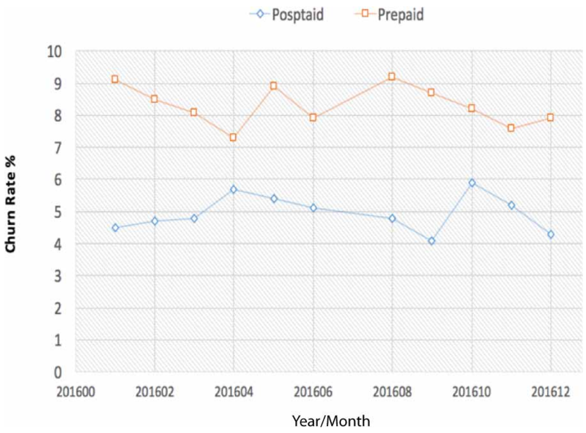
that, in one major telecom operator in Morocco, the churn rate of prepaid subscribers is significantly higher than postpaid subscribers (Figure 1). While it is possible that such high churn rate in prepaid subscribers may be is due to factors related to high cost offer, low quality of the service, and/or high customer service dissatisfaction, being able to analyze and monitor customer behavior in time gives companies the opportunity to execute preventive measures for retaining them.

Several algorithms have been developed (Vafeidis et al., 2015), however it is not clear which of those models better fit for detecting the churning customers. To evaluate better the churning impact on customer's network, several experiments have been conducted using the state-of-the-art machine learning techniques with special emphasis on deep learning algorithms and convolution neural networks.

ConvNets (CNN) have been proved to have a very good performance in different area of research, including images and video recognitions (Simonyan et al., 2014; Yang et al., 2015; Hatami et al., 2018), natural language processing (Zhang et al., 2015), and speech recognition (Noda et al., 2015) by extracting high-level features from a large set of data. In addition, CNN has been demonstrated to have a very good performance in image processing tasks (Szegedy et al., 2015; Russakovsky et al., 2015). Thus, it would be of interest to use this technique to predict the customer churn by analysing images of represented customer's behavior.

Previous studies have shown that different state of art predictive models were used to predict the churning problem by (training binary classifiers) using labeled churner/non churner dataset involving the Traditional hand-crafted features (Keramati et al., 2014; Vafeiadis et al., 2015; The Chartered Institute of Marketing, 2010) or social network analysis technique (Phadke et al., 2013; Richter et al., 2010).However, motivated by the recent advances of deep learning (LeCun et al., 2015) in different area of research, we propose a novel method using convolutional neural network for customer's churn use case. Our method explores the efficiency of deep ConvNets for the predicting task, using the same structure of dataset handled previously by the-state of the art machine learning models (Keramati et al., 2014; Owczarczuk, 2010; Vafeidis et al., 2015; Kisioglu et al., 2011). More specifically, we focused on customer's behavior, which is represented as an input image describing calls/SMS/Data and a recharge temporal usage during two-month customer behavior. We found that ConvNets provides more meaningful and useful representations (yielded to optimal results), outperforming other conventional machine learning algorithms such as Naïve Random Forest, Gradient Boosting Classifier, and Support Vector Machine. This result indicates that our approach represents an important contribution to one open industrial question: how deep learning can be a useful method in addressing issues related to business telco data?

Figure 1. Churn rate comparison during 12 months



The rest of This paper proceeds as follows: section 2 describes the previous works and how the churn problem has been approached in the literature. Sections 3 and 4 describes the datasets and our features representation. Section 5 presents our proposed models that we used during the performance evaluation. Finally, section 6 and 7 summarizes most of our results and conclusions.

2. LITERATURE REVIEW

Churn predictive modelling in the telecom industry consists of using the accurate features to build the best classifier (s). Several binary classifiers have been developed with two class labels (churner, non-churner) to predict the future behavior of customers. Several state-of-the-art studies have approached the churning problem by training different machine learning models (Chouiekh et al., 2017),including logistic regression (Neslin et al., 2006; Owczarczuk, 2010), support vector machines, decision trees (Chen et al., 2012; Huang et al., 2010), random forest (Idris et al., 2012; Xie et al., 2009), Neural networks (Sharma et al., 2013; Tsai et al., 2009), ensemble hybrid modeling methods (De Bock et al., 2011; Idris et al., 2013), and evolutionary algorithms (e.g, genetic algorithms) (Amin et al., 2017; Au et al., 2003). All those mainstream models for churn prediction use hand crafted features, whether by considering each customer separately or by applying social network analysis (Richter et al., 2010). Most customer features are obtained from telco business support system(BSS) including call detail records (call duration, number of outgoing SMS, number of vouchers, data usage, etc.), demographic profiles (Age/Gender/Social class bands/Country code) (Vafeidais et al., 2015), or billing information (billing amount of subscribers, billing history) (Kisioglu, Pinar, & Topcu, 2011).

Experimental results from the previous works showed small improvement of predictive performance from one model to another and that all prediction techniques that attempted to improve the accuracy (either using different combination of features or increasing the dataset during the training phase) failed to make significant improvement in predicting customer churn. Here, we used a novel method of deep convolution neural networks to predict customer churn problem in a dataset from a Moroccan telecom carrier. The main results are summarized as follows:

- The Development of a new methodology in order to approach the churn business problem by applying structural data to the DCNN model using a raw telecommunication dataset;
- An Experimental evaluation of our proposed model and its comparison with the previous state-of-the-art machine learning models, and discussing the ability of our model to explore and learn features using structured data;
- The application of different CNN network architectures to investigate which model fits better the churn use case.

3. FEATURE REPRESENTATION AND ANALYSIS

Call details records (CDR) are used as log events representing customer's behavior during a specific period of time. This information includes the following type of features:

- Calls Event including number of outgoing calls, number of incoming calls, number of international calls, and total calls duration per category (incoming, outgoing, etc.);
- Recharge Events referred to us as total Recharge amount and number of recharges;
- SMS Events representing the number of incoming/outgoing SMS;
- MMS Events representing the number of incoming/outgoing MMS;
- Data Events which are represented as data uploaded or downloaded volume per subscriber;
- Subscriptions Events which are number of subscribed offers.

Each customer is represented with an image labeled with 1 for 'churned' subscriber or 0 for 'not churned' subscriber. The image width covers the whole period activity of two months defined by 60 continuous days' customer behavior. Thus, our input image has a size of 60×14 (width and height). The data is manipulated into a 2-dimensional array where rows correspond to days and columns to different types of behavior (call, Recharge, SMS, Data, subscriptions). It is important to mention that our datasets contain prepaid subscribers only. Those type of subscribers are less committed and they can terminate their subscription at any time with no prior notice. Of note, churners are defined as those who didn't perform any recharge event for specific period of time. In order to define image labels, each customer may be in one of the following status: Idle, active, suspended, disabled or Pool (see details in Table 1). Each state defines customer's behavior after a specific period of time. Our extraction method was focused on subscribers with a state different from 'active' one. These customers have the highest probability to cancel their subscriptions and so, countermeasures should be taken to prevent them from churning. Figure 2 describes the flow chart of data collection and processing by our deep convolution neural network.

4. IMAGE TRANSFORMATION AND LABELING

In this section, we described how the data was introduced and represented as artificial images describing customer's behavior. This transformation was inspired, at least in part, by the word representation in word2vec model already published by Mikolov (2013). In fact, this method has encoded a given word by associating each word to one vector, so each word is represented by a distribution of weights across the vector elements. The encoding process was performed by mapping our input dataset to another space representation and transforming our structural data into artificial images that were fed through different DCNN models. As a result, our dataset is mapped into a simple linear representation where each line corresponds to one of our data source: calls/SMS/Recharge/Data respectively, and columns are mapped to days. As shown in Figure 3 the time is linearly mapped into the x-axis of our image and the y-axis is reserved for each one of the data sources mentioned above.

Because ranges of our dataset values vary widely, a 'uniformization' is required. Thus, we should apply rescaling using feature scaling method in order to get values in the range of [0, 1]. The general formula of this technique is given below:

$$f\left(x\right)=\frac{x-\min\left(x\right)}{\max\left(x\right)-\min\left(x\right)} \tag{1}$$

where $f\left(x\right)$ is the normalized value.

Figure 2. General framework for customer churn detection using DCNN

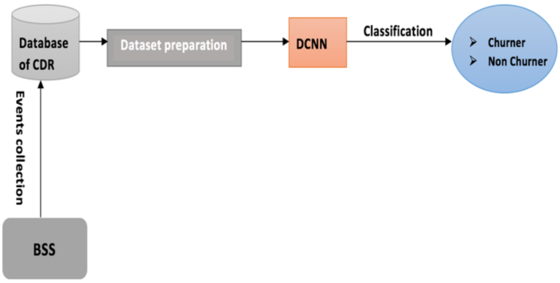
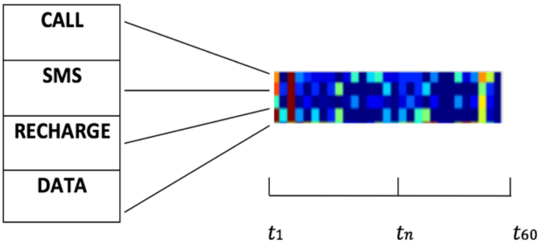


Figure 3. Image representation example for one customer over 60 days' activity. Pixel intensity is changing from blue to red.



Given the decimal fraction of normalized value, the binary representation value is computed using a straightforward conversion algorithm, for an instance if a normalized value of the pixel is P = 0.668, the binary conversion is 10101011 and the corresponding value of the pixel will be 171. As a result, the pixel intensity will be proportional to customer activity for the four categories of behavior (CALL, SMS, Recharge, Data).

Image labelling was performed according to customer's activity during the last 60 days. During this period, the state of the subscriber may change from Active to Pool, leading to a higher churn rate if no preventive action is taken. For instance, after first call, subscriber enter 'active' status, if he exceeds a certain duration limit with no recharge event, his status is changed to 'suspend' and if the customer registers no activity within a specific duration limit (60 days in our case, but it can be different from one telecom provider to another), he moves to 'Disable' status and finally to Pool status, which describes the highest probability for qualifying the customer as a churner. We labelled churner customers with 1 and non-churner with 0. The life cycle for data labelling of prepaid subscribers is illustrated in Figure 4, also Table 1 gives a summary description of different customer's status.

5. CONVNETS NETWORK ARCHITECTURE

After image labeling and classification, the dataset was passed through a deep convolution neural network. One major benefit of ConvNets is the ability to learn features hierarchically and exploit the advantage of shared weights, in addition to the translation invariance property, which can reduce the number of needed parameters without impacting the performance.

After learning the feature through the ConVnet, the fully connected layers are used in order to classify the input image to the target labels (The classified images refer to churner/non churner customers and the pixel intensity represents customer's behavior).

The 'Tweaking' of parameters is essential to improve the performance of the tested models. Thus, our work consists of testing two different models. Our first model described in Figure 5 is called Conv-D1 which consists of two sequential convolutional layers followed by element wise activation function. We applied RELU layer, which is preferred over other methods such as sigmoid or Tanh function (because of its capacity to speedup the convergence of stochastic gradient descent, especially that faster learning has a great influence on the performance of large models trained on large datasets (Krizheysky et al., 2012)). Then we applied 2x2 max pooling layer and two consecutive fully connected layers of 120 units with 0.5 dropout ending in a soft-max function with two output units for our binary classification (1or 0). In fact, the use of dropout with fully connected layers is an excellent technique for improving the performance by reducing overfitting (Srivastava et al., 2014). In the first step, we applied small filters of convolutions to extract the characteristics of each customer during seven days of behavior, similar to feature maps detected for image recognition. In the first convolution of our analysis we used eight filters of size 7×1. In the second step, we applied a second convolution involving seven filters of size 1×5 replicating across all features to analyse customer's behaviour for the 14th predefined features during two months.

Figure 4. Prepaid subscriber life cycle

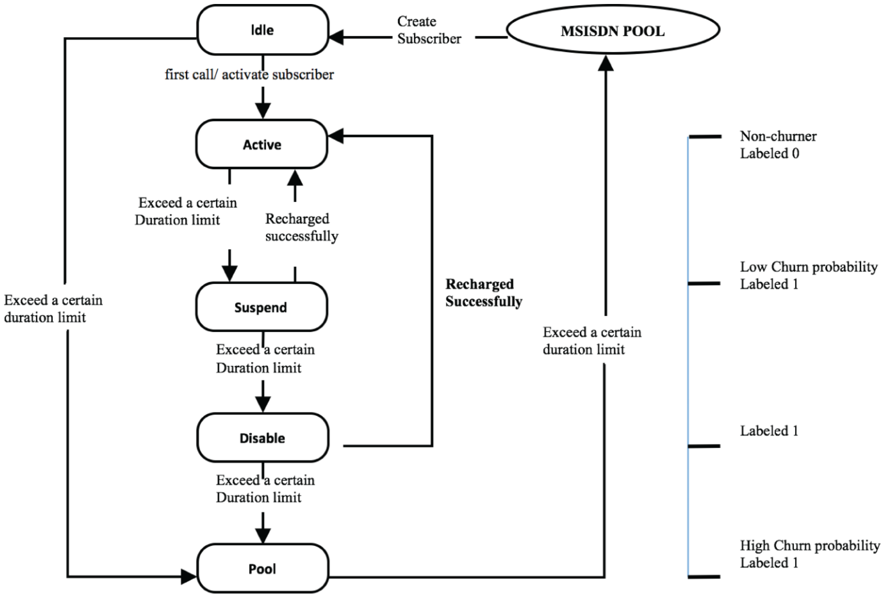

Idle
Create Subscriber
MSISDN POOL
first call/ activate subscriber
Active
Exceed a certain Duration limit
Recharged successfully
Suspend
Recharged Successfully
Exceed a certain duration limit
Exceed a certain Duration limit
Exceed a certain duration limit
Disable
Exceed a certain Duration limit
Pool
Non-churner Labeled 0
Low Churn probability Labeled 1
Labeled 1
High Churn probability Labeled 1

Table 1. Customer's status description

Status	Description
Idle	Initial status, No first call made
Active	Normal status after first call
Suspend	Active period has been past, the subscriber can't make Outgoing calls
Disable	suspend period has been past, the subscriber can't receive or make outgoing calls
Pool	After the disable period has been past, customer can't be used anymore, and the recycling process is triggered

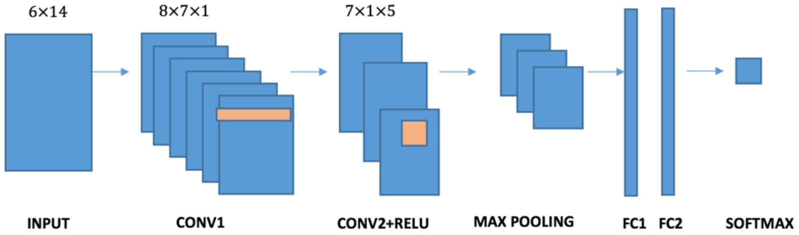
We used recommended hyperparameters for CNN training (Bengio & Boulanger-Lewandowski, 2013). In fact, the model was trained for 40 epochs using cross-entropy loss function (Andreas, Stuetzle, & Shen, 2005), that we tried to minimize the during the training phase. The general formula of the loss function is given below:

$$L = \frac{1}{2}\sum_{i}(x_i - w_i) \tag{2}$$

where x_i is the output of the ith network output unit and w_i is the ith value of the target output.

We used the stochastic gradient descent by backpropagation algorithm (Phadke et al., 2013) with mini batch sizes of 200 with adaptive learning rates and momentum value of 0.9. The nth mini-batch gradient descent update of the network parameters Φ is stated as follow:

Figure 5. Conv-D1 Architecture: The convolution kernels applied in the first stages are shown in the yellow rectangle



Φ (t) $\leftarrow$ Φ(t-1) - β_t α

where α is equal to:

$$\frac{1}{B}\sum_{t'=Bt+1}^{B(t+1)}\frac{\partial L\left(z_{t'},\theta\right)}{\partial\theta} \tag{3}$$

$z_{t'}$ represents an example t' during the training set, β_t is the learning rate at step t with mini batch size equal to B. In addition, we explored the Initialization procedure as proposed by Glorot & Bengio to make the training faster. This method is represented mathematically as follows:

$$W_{ij} \sim U\left[-\frac{1}{\sqrt{n}},\frac{1}{\sqrt{n}}\right] \tag{4}$$

where U [−a, a] is the uniform distribution in the interval (−a, a) and n is the size of the previous layer (the number of columns of W) (Xavier & Bengio, 2010)

In order to select the appropriate optimization parameters, we followed random search process. In fact, finding a good optimization parameter has much more influence on the obtained performance (Leo, 2001). We reported the best values of hyperparameters based on different tests computed over time.

The final setup is summarized in Table 2. This setup is used during all coming experiments as well.

In addition, we considered using multiple sets of partitions. The input dataset was divided into training (50%), validation (25%) and testing dataset (25%). Otherwise, the result based on a single set could be highly biased. We only presented the averaged results over these sets in our next sections.

Our second model is called CONV-SVM (the most challenging one). We used a supervised learning model on features obtained from our deep learning model (ConvNets). In fact, the model of Sharif Razavian (2014) reuses pre-trained CNN as feature generators to address many recognition tasks such as object classification and the results strongly suggested that features obtained from deep learning models with convolutional nets are the primary candidate in most visual recognition tasks (Razavian et la., 2014). Therefore, in this work we applied the same approach to address the prediction task dealing with the churn use case by training a binary classifier based on CNN features. Of note, this model is not a conventional approach for Deep ConvNets training, but it's worthy to explore this method and compare it with the baseline of the CNN performance. The proposed model is trained by removing the soft max layer and using the 120 activations from the fully connected layers as features on a standard machine learning classifier. For instance, the SVM technique has shown

Table 2. Hyperparameters setup

Parameter	Value
Number of Epoch	40
Learning rate	0.1
Momentum	0.9
Weight decay	0.045
Mini-batch size	200

a good performance (Huang, Kechadi, & Buckley, 2012) for producing a binary classification. The details of our CONV-SVM architecture are explained in Table 3.

Our third experiment used another model called CONV-RF based on CNN features also. We investigated the effectiveness of the standard random forests algorithm (Breiman, 2001) to uncover more insights regarding the churn use case analysis. Following the same methodology, we took features from our proposed CNN model trained on the telecom dataset and using them as training inputs for the random forest algorithm. The details of our CONV-RF architecture are described in Table 4.

6. RESULT AND DISCUSSION

6.1. Evaluation Criteria Overview

To test the effectiveness of our approach we conducted several successful pilot experiments on real (raw) telecom datasets. The models were evaluated using the following evaluations criteria: accuracy, precision recall, and specificity, which were calculated from the confusion matrix shown in Table 5. We denoted the positive cases that were correctly classified by TP, the positive cases that were incorrectly classified by FP, the negative cases that were incorrectly classified by FN, and finally TN which refer to the negative cases that were correctly classified. The metrics obtained from the confusion matrix are defined as follows:

$$Accuracy = \frac{TP + TN}{FP + FN + TP + TN} \tag{5}$$

Table 3. Our proposed CONV-SVM architecture

Layer	Layer’s Specification
1	Convolution Layer 1
2	Convolution Layer 2+RELU
3	Max Pooling
4	Fully connected layer1
5	Fully connected layer2
6	SVM Classifier

Table 4. Our proposed CONV-RF architecture

Layer	Layer's Specification
1	Convolution Layer 1
2	Convolution Layer 2+RELU
3	Max Pooling
4	Fully connected layer1
5	Fully connected layer2
7	Random forest Classifier

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{FN + TP} \tag{7}$$

$$Sensitivity = \frac{TN}{TN + FP} \tag{8}$$

However, those metrics alone cannot describe the efficiency of our proposed models clearly. Thus, during our model's evaluation we used also the F1 score, which is defined as the harmonic mean of precision and recall as follow:

$$F1\ Score = \frac{\left(Precision \times Recall \times 2\right)}{\left(Presicion + Recall\right)} \tag{9}$$

From mobile operator perspective, the cost of losing important customers is much higher than taking retention actions to prevent them from churning. In other words, the cost related to incorrect classification of churners is higher than the cost associated with the incorrect classification of a non-churner. Thus, Mobile telecom providers tend to choose models with high precision rather than models with high sensitivity. Altogether, in order to assess the effectiveness of ours models, their performance is compared in term of precision, recall, F1 score and Area Under Curve(AUC).

6.2. Results

In our first experiment, the accuracy of Conv-D1 is calculated during different epoch. It has been shown that an epoch is one forward pass and one backward pass of the whole training examples during the gradient descent algorithm calculation. As shown in Figure 6, we have calculated the accuracy during 40 epochs using the 120 features. The same plot also displayed the accuracy of the second and third model using support vector machines classifier and random forests approach respectively. These results confirmed that the best accuracy is obtained using the third model at epoch 36. This indicates that applying the early stopping is crucial in preventing unnecessary computation. Based on

Table 5. Confusion matrix for model's evaluation

		Predicted Class	
		Churner	**Non-Churner**
Actual class	Churner	TP	FN
	Non-churner	FP	TN

the provided results, it is pretty clear that SVM and RF trained via CNN features outperformed the CONV-D1 model using the CNN baseline. Figure 6 also showed that CONV-RF and CONV-SVM performed better when compared to Conv-D1 even at epoch 0. Note that at epoch 0 the CNN has not been trained yet. All related results are presented and summarized in Table 8. Because the classification accuracy is not good enough to evaluate the performance of our models, we have calculated the value of precision, recall, F1 Score and AUC to select a well-performing model. Our findings presented in Table 6 confirm that CONV-RF model was still achieving a remarkably good precision and recall outperforming CONV-SVM and CONV-D1 models by achieving an F1 Score of 91%.

To further examine which of those models give the best results, we considered three conventional machine learning algorithms: Random Forest(RF), Gradient Boosting Classifier(GBC) and support vector machines(SVM) using the same dataset of customers. During SVM model implementation, we adopted k-fold cross validation approach to avoid overfitting problem. And in order to select the best kernel function, the model was evaluated in term of F1 Score, precision and recall. Our findings are reported in Table 7. The best performance was obtained using RBF kernel function.

Finally, we compared the performance of the traditional machine learning algorithm with our proposed models: CONV-SVM, CONV-RF and Conv-D1. Our findings reported in Table 9 showed clearly that the CNN Models outperformed the traditional machine learning algorithms with F1 score of 0.91 and AUC of 0.92 for CONV-RF, F1 Score of 0.88 for CONV-SVM and F1 Score of 0.86

Table 6. Performance comparison using F1 Score and AUC

CONV-D1			
precision	recall	F1-score	AUC
0.83	0.89	0.86	0.87
CONV-RF			
precision	recall	F1-score	AUC
0.88	0.95	0.91	0.92
CONV-SVM			
precision	recall	F1-score	AUC
0.86	0.91	0.88	0.89

Table 7. SVM kernel function selection

Kernel Function	Polynom Degree	Precision	Recall	F1 Score
Linear/ Polynomial	3	0.58	0.648	0.65
RBF	-	0.79	0.85	0.82

for CONV-D1. These results are of great importance in predicting customer churn in the telecom industry (as evidenced by the efficiency of DCNN architectures over the traditional machine learning techniques). GBC and SVM algorithms also provide a pretty good result with an F1 Score of 0.83 and 0.82 respectively.

Detecting Churning customers on time is very important, which allows taking preventive actions and avoiding additional revenue loss. Thus, we have calculated the training duration of our proposed DCNN architectures and compared them with the traditional machine learning algorithms: SVM, RF and GBC. We used GPU accelerated CNNs which proved to provide a huge speed performance and 50 times faster than standard training over CPU only (Christian et al., 2015) especially that GPUs were designed to handle and process operation in parallel during large data computation. Furthermore, some of the GPU based CNN implementations contributed significantly to recent success in famous contests (Stalkamp et la., 2011) for object detection, pattern recognition or image segmentation. This becomes easier to use in the research community with a wide range of open source frameworks based on GPU such as Theano, Torch and Caffe (Bahrampour et al., 2016).

We have calculated the training duration of our different DCNN architectures to investigate their effectiveness against the traditional machine learning algorithms. As shown in Figure 7, we can see that GBC algorithm takes less time in training the whole model. For instance, 70% of training dataset took 1000 second, while it has taken 1800 second for CONV-RF model, which is merely two times more than the traditional machine learning algorithm training. During the end-to-end dataset training, the GBC model took 1900 second only which outperformed remarkably the training duration of CONV-RF, CONV-SVM, CONV-D1 and the rest of traditional machine learning algorithms as well. These results showed that DCNN architectures take long-time to train the whole model or even a part of it, which means that CNN models require more computational power. This is mainly due to the numbers of parameters involved to train our deep learning algorithms. Of note, the machine learning techniques relatively takes less time to train the model end-to-end resulting in less computing power. The straight forward way to improve the training speed-up of CNN architecture is to add more GPU nodes and train the network using data-parallel Stochastic Gradient Descent, where each worker receives some chunk of global mini-batch (Krizhevsky, 2014). However, the computational power cost is significantly increased as well.

In conclusion, the churn prediction is often not done in real-time so in practice the most performant model will be chosen by the operator.

7. CONCLUSION AND FUTURE WORK

In this paper, we have demonstrated how deep learning algorithms can approach the churn business problem by applying structural data from a real mobile telecom provider to different DCNN models. Furthermore, we have shown that our deep learning models achieve good results in the churn prediction business problem by surpassing the traditional machine learning algorithms in term of AUC, F1 Score, precision, and recall. While the machine learning techniques require less time in training the models, much more time is needed for training DCNN architectures. The difference in time trainings can be due to the number of hyperparameters used during model's building and training. It is possible to overcome these limitations, by using other DCNN models that can be tested based on the same datasets such as the pre-trained CNNs architectures, specially that they outperform randomly initialized nets slightly. Thus, it would be very interesting to use these models for the churn use case to improve the performance in detecting churn problem in telecom industry.

Our research can also be extended to testing other DCNN models by experimenting the features extraction from different fully connected layers instead of taking features from the last fully connected layer only. In sum, the novel method presented in this work can be applied to many mobile operators for analysing the churn problem and seeking for an improved prediction accuracy. It can also be used in other areas of business dealing with everyday customer services where the churn problem is a big

Figure 6. Model's accuracy at different epoch

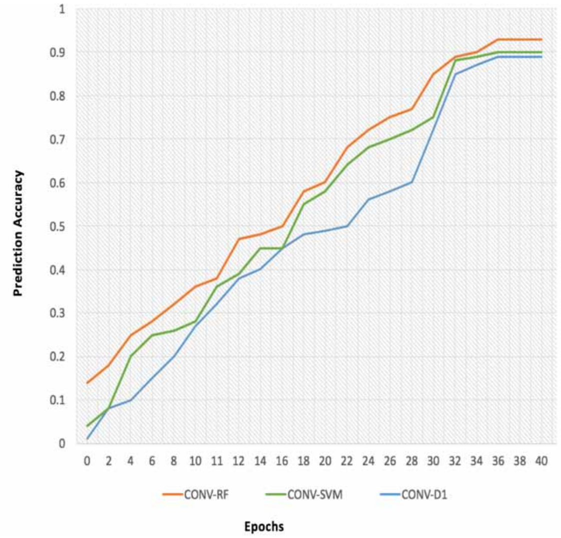


Table 8. Max accuracy comparison of Conv-D1, CONV-SVM and CONV-RF

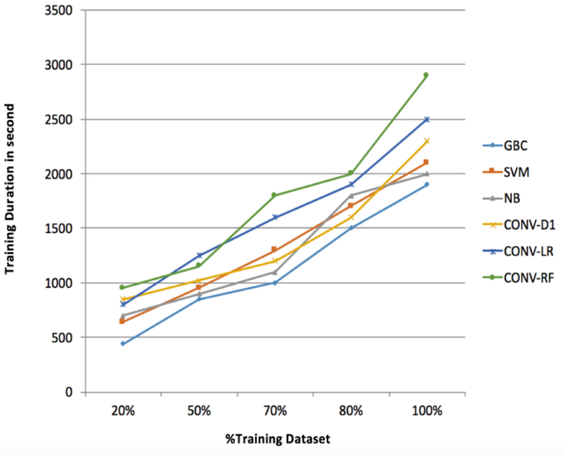
Model	Accuracy
CONV-D1	0.89
CONV-SVM	0.90
CONV-RF	0.92

Table 9. Performance comparison with traditional machine learning algorithms

Model	F1 Score	Precision	Recall	AUC
CONV-D1	0.86	0.83	0.89	0.87
CONV-SVM	0.88	0.86	0.91	0.89
CONV-RF	**0.91**	0.88	0.95	0.92
SVM	0.82	0.79	0.85	0.82
GBC	0.83	0.80	0.86	0.85
RF	0.78	0.75	0.82	0.76

concern for business management. It is worth mentioning that applying more feature selection, using different source of data (such as OSS, (operations support systems), or the combination of both BSS and OSS source data will improve the accuracy and performance of our models to prevent customer churn and increase company revenue.

Figure 7. Training duration comparison of our proposed models



REFERENCES

Ali. (2014). CNN features off-the-shelf: an astounding baseline for recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Amin, A., Anwar, S., Adnan, A., Nawaz, M., Alawfi, K., Hussain, A., & Huang, K. (2017). Customer churn prediction in the telecommunication sector using a rough set approach. *Neurocomputing*, *237*, 242–254. doi:10.1016/j.neucom.2016.12.009

Au, W.-H., & Keith, C. C. (2003). Chan, and Xin Yao. "A novel evolutionary data mining algorithm with applications to churn prediction. *IEEE Transactions on Evolutionary Computation*, *7*(6), 532–545. doi:10.1109/TEVC.2003.819264

Bahrampour, S., Ramakrishnan, N., Schott, L., & Shah, M. (2016). *Comparative study of caffe, neon, theano, and torch for deep learning*. Academic Press.

Bengio, Y., Boulanger-Lewandowski, N., & Pascanu, R. (2013, May). Advances in optimizing recurrent networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 8624-8628). IEEE. DOI: 10.1109/ICASSP.2013.6639349

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828. doi:10.1109/TPAMI.2013.50 PMID:23787338

Bin, L., Peiji, S., & Juan, L. (2007). Customer churn prediction based on the decision tree in personal handyphone system service. In *International conference on service systems and service management* (pp. 1–5). Academic Press. doi:10.1109/ICSSSM.2007.4280145

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. doi:10.1023/A:1010933404324

Buja, Stuetzle, & Shen. (2005). *Loss functions for binary class probability estimation and classification: Structure and applications*. Working draft, November.

Chen, Z.-Y., Fan, Z.-P., & Sun, M. (2012). A hierarchical multiple kernel support vector machine for customer churn prediction using longitudinal behavioral data. *European Journal of Operational Research*, *223*(2), 461–472. doi:10.1016/j.ejor.2012.06.040

Chouiekh, A. (2017). Machine Learning techniques applied to prepaid subscribers: case study on the telecom industry of Morocco. In 2017 Intelligent Systems and Computer Vision (ISCV). IEEE. DOI: 10.1109/ISACV.2017.8054923

De Bock, K. W., & Van den Poel, D. (2011). An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction. *Expert Systems with Applications*, *38*(10), 12293–12301. doi:10.1016/j.eswa.2011.04.007

Glorot, X., & Bengio, Y. (2010). *Understanding the difficulty of training deep feedforward neural networks* (Vol. 9). Aistats.

Hatami, N., Gavet, Y., & Debayle, J. (2018, April). Classification of time-series images using deep convolutional neural networks. In *Tenth International Conference on Machine Vision (ICMV 2017)* (Vol. 10696). International Society for Optics and Photonics.

Huang, B., Kechadi, M. T., & Buckley, B. (2012). Customer churn prediction in telecommunications. *Expert Systems with Applications*, *39*(1), 1414–1425. doi:10.1016/j.eswa.2011.08.024

Huang, B. Q., Kechadi, T.-M., Buckley, B., Kiernan, G., Keogh, E., & Rashid, T. (2010). A new feature set with new window techniques for customer churn prediction in land-line telecommunications. *Expert Systems with Applications*, *37*(5), 3657–3665. doi:10.1016/j.eswa.2009.10.025

Idris, A., Khan, A., & Lee, Y. S. (2013). Intelligent churn prediction in telecom: Employing mRMR feature selection and RotBoost based ensemble classification. *Applied Intelligence*, *39*(3), 659–672. doi:10.1007/s10489-013-0440-x

Idris, A., Rizwan, M., & Khan, A. (2012). Churn prediction in telecom using Random Forest and PSO based data balancing in combination with various feature selection strategies. *Computers & Electrical Engineering*, *38*(6), 1808–1819. doi:10.1016/j.compeleceng.2012.09.001

Keramati, A., Jafari-Marandi, R., Aliannejadi, M., Ahmadian, I., Mozaffari, M., & Abbasi, U. (2014). Improved churn prediction in telecommunication industry using data mining techniques. *Applied Soft Computing*, *24*, 994–1012. doi:10.1016/j.asoc.2014.08.041

Kisioglu, P., & Ilker Topcu, Y. (2011). Applying Bayesian Belief Network approach to customer churn analysis: A case study on the telecom industry of Turkey. *Expert Systems with Applications*, *38*(6), 7151–7157. doi:10.1016/j.eswa.2010.12.045

Krizhevsky, A. (2014). *One weird trick for parallelizing convolutional neural networks*. arXiv preprint arXiv:1404.5997

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. doi:10.1038/nature14539 PMID:26017442

Mikolov, T. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*.

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. *JMR, Journal of Marketing Research*, *43*(2), 204–211. doi:10.1509/jmkr.43.2.204

Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H. G., & Ogata, T. (2015). Audio-visual speech recognition using deep learning. *Applied Intelligence*, *42*(4), 722–737. doi:10.1007/s10489-014-0629-7

Owczarczuk, M. (2010). Churn models for prepaid customers in the cellular telecommunication industry using large data marts. *Expert Systems with Applications*, *37*(6), 4710–4712. doi:10.1016/j.eswa.2009.11.083

Phadke, C., Uzunalioglu, H., Mendiratta, V. B., Kushnir, D., & Doran, D. (2013). Prediction of subscriber churn using social network analysis. *Bell Labs Technical Journal*, *17*(4), 63–75. doi:10.1002/bltj.21575

Richter, Y., Yom-Tov, E., & Slonim, N. (2010). Predicting customer churn in mobile networks through analysis of social groups. In *Proceedings of the 2010 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics; . doi:10.1137/1.9781611972801.64

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., & Fei-Fei, L. et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252. doi:10.1007/s11263-015-0816-y

Sharma, A., Panigrahi, D., & Kumar, P. (2013). *A neural network based approach for predicting customer churn in cellular network services*. arXiv preprint arXiv:1309.3945

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556

Srivastava, N. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2011). The German traffic sign recognition benchmark: A multi-class classification competition. In *International Joint Conference on Neural Networks (IJCNN 2011)*, (pp. 1453–1460). IEEE Press. DOI: 10.1109/IJCNN.2011.6033395

Szegedy, C. (2015). Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*. DOI: 10.1109/CVPR.2015.7298594

Tsai, C.-F., & Lu, Y.-H. (2009). Customer churn prediction by hybrid neural networks. *Expert Systems with Applications*, *36*(10), 12547–12553. doi:10.1016/j.eswa.2009.05.032

Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, *55*, 1–9. doi:10.1016/j.simpat.2015.03.003

Xie, Y., Li, X., Ngai, E. W. T., & Ying, W. (2009). Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, *36*(3), 5445–5449. doi:10.1016/j.eswa.2008.06.121

Yang, J., Nguyen, M. N., San, P. P., Li, X., & Krishnaswamy, S. (2015, July). Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition. *IJCAI (United States)*, *15*, 3995–4001.

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

Alae Chouiekh has obtained an engineering diploma of telecommunication from INPT in 2010 with the highest honors and pursued courses on software engineering Programme at Oxford university, in 2013. His research interests are related to machine learning, deep learning and Big Data.

El Hassane Ibn El Haj is a professor in the Department of Communication Systems and head of the MUSICS (Multimedia, Signal and Communication Systems) research group at the National Institute of Posts and Telecommunications (INPT), Rabat, Morocco. His research interests are related to multimedia, image and video processing and analysis, speech processing, watermarking, and cognitive radio. He has supervised several theses in computer science and communications engineering. He is a reviewer for several international conferences and journals.