2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining

IAN: Interpretable Attention Network for Churn Prediction in LBSNs

Liang-yu Chen*, Yutong Chen*, Young D. Kwon†, Youwen Kang*, and Pan Hui*‡
*Hong Kong University of Science and Technology, Hong Kong SAR
†University of Cambridge, United Kingdom
‡University of Helsinki, Finland
Email: {lchenbm, ychendm, ykangae}@connect.ust.hk ydk21@cam.ac.kr panhui@cs.helsinki.fi

***Abstract*—With the rise of Location-Based Social Networks (LBSNs) and their heavy reliance on User-Generated Content, it has become essential to attract and keep more users, which makes the churn prediction problem interesting. Recent research focuses on solving the task by utilizing complex neural networks. However, due to the black-box nature of those proposed deep learning algorithms, it is still a challenge for LBSN managers to interpret the prediction results and design strategies to prevent churning behavior. Therefore, in this paper, we perform the first investigation into the interpretability of the churn prediction in LBSNs. We proposed a novel attention-based deep learning network, Interpretable Attention Network (IAN), to achieve high performance while ensuring interpretability. The network is capable to process the complex temporal multivariate multidimensional user data from LBSN datasets (i.e. Yelp and Foursquare) and provides meaningful explanations of its prediction. We also utilize several visualization techniques to interpret the prediction results. By analyzing the attention output, researchers can intuitively gain insights into which features dominate the model's prediction of churning users. Finally, we expect our model to become a robust and powerful tool to help LBSN applications to understand and analyze user churning behavior and in turn remain users.**

I. Introduction

In recent years, we have witnessed the rapid development of Location-Based Social Networks (LBSNs), with the most popular being Yelp and Foursquare. For example, Yelp[1] highlights the total number of accumulated reviews in the company's quarterly financial report and Foursquare[2] emphasizes the total number of accumulated check-ins on its webpage. Since LBSNs primarily rely on User-Generated Content (UGC) such as sharing information of local restaurants by writing reviews on the sites and the competition among them becomes severe, it is important for those platforms to strive to attract and keep more users [1]. Hence, the churn prediction problem (*e.g.,* attrition), to predict the future loss of a user from a service, has long been studied for the sustainability of the provided services [2]–[5].

Many existing studies aiming to solve the churn prediction problem have focused on improving the performance of the predictive model to keep users from churning [3]–[11]. Since prior works largely focused on utilizing diverse feature sets or developing a new neural architecture, the proposed models lack the ability to interpret their results intuitively for humans to understand. Besides, the heterogeneous nature of LBSNs data is not yet fully understood. In particular, the linguistic relevance with respect to the churn of the users is unknown. It is not clear how a user's review and its content on venues can be utilized as an indicative feature for predicting the user behaviors to continue to use the site.

To overcome the aforementioned limitations, we conduct the first systematic study to investigate the interpretability of the churn prediction problem in LBSNs. We propose an Interpretable Attention Network (IAN) in order to provide higher interpretability from a complex neural network model while achieving a substantial performance in the churn prediction problem. IAN effectively fuses different interpretable models as its sub-components for high interpretability on its predictions using various features. Specifically, Inspired by [12], IAN incorporates non-textual feature model (temporal, geographic, venue, social features) and bidirectional LSTM [13] for rendering attention on sequences of reviews in both chronological and reversed order. Also, inspired by [14], IAN employs textual feature model to provide interpretability on a user's review and its text level. We also investigate a comprehensive feature sets to fully understand the holistic view of the heterogeneous nature of the LBSNs dataset. We extract a variety of feature sets, namely, temporal, geographic, venue, social, and linguistic features. In particular, IAN is also able to provide interpretability on users' reviews so that LBSN maintainers can have a better understanding of their users from a linguistic perspective.

The main contributions of this study are as follows:

1) To the best of our knowledge, we make the first attempt to address the interpretability of the churn prediction problem in LBSNs.
2) We develop a novel neural network architecture, IAN, to predict potential churning users with a fusion mechanism to incorporate interpretable methods using the

[1] http://www.yelp-ir.com

[2] http://www.foursquare.com/about

ASONAM '21, November 8-11, 2021, Virtual Event, Netherlands
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-6868-1/19/08?/$15.00
http://doi.org/10.1145/3487351.3488328

heterogeneous data of different modalities of LBSNs.

3) We conduct extensive experiments with three real-world datasets of LBSNs. We also demonstrate that our proposed model not only outperforms the state-of-the-art interpretable method and strong baselines (LSTMs) but provides intuitive interpretations of the prediction results for each user.
4) We utilize visualization techniques to help analyze the prediction result and attention outputs. Moreover, the visualization can also support researchers interpreting the IAN model.

II. Related Work

We start by reviewing key concepts and recent works regarding the (1) churn prediction, and (2) interpretable models.

A. Churn Prediction

User engagements of online communities have been widely investigated by researchers, including the motivations [15], [16], behavioral patterns [9], and application usage [17], [18]. By understanding user engagement, general crowds can gain more insights into using the application, and community operations can plan strategies to improve user experience and customer retention [19]. Many previous works have explored ways to predict user behaviors with linguistic features [20], social features [21], or temporal features [6], [22]. A comparative survey [23] investigated the traditional classification methods in churn prediction. However, traditional methods can not give satisfying results regarding complex churning prediction tasks. Kwon et al. [5] predicted user churning behaviors on LBSNs by using social, geographical, temporal, and linguistic features. Their work provides a comprehensive feature exploration of the churn prediction problem. Besides that, researchers also did churn analysis on other platforms like the Question-Answer platform [6]. Carl et al. [4] conducted interpretable user clustering and churn prediction on Snapchat. The work divides user churning behaviors into different types and delivers real-time data analysis and prediction results.

B. Interpretable Models

In a real-world problem with complex user engagement, the interpretability of machine learning models becomes significant. Researchers have investigated different techniques to interpret machine learning models. Typically, there are two types of interpretable models that provide more information than "black-box" models generating mere prediction results. The first type is the "white-box" models that are designed to be inherently explainable (e.g. Decision Tree, Logistic Regression). Another type is model-agnostic methods to explain "black-box" models (e.g., PDP [24], LIME [25], and SHAP [26]). These post-hoc approaches aim to visualize and explain models by giving weights to features. However, researchers also identified the limitations of the above interpretable models [27]. That is, traditional interpretable models are not suitable for complex classification tasks with sparse data. Model-agnostic methods provide limited information for the "black-box" model itself. To overcome these limitations, researchers also investigated interpretable deep learning models.

For "black-box" deep learning models, Attention Networks are widely used to not only improve the performance but also learn representations with features in each prediction [27]. Hierarchical Attention Network (HAN) [14] is a classical structure to deal with long document classification and compute attention weight for each word in each sentence. Researchers are able to visualize and see the importance of word-level and analyze why the model makes such a classification. HAN is also applied in social networks like cyberbully detection [20] combined with user information. Another interpretable deep learning model applying attention networks is RETAIN [12], a predictive model handling time series data using the reverse time attention mechanism. By assigning attention weights to both visits and attributes, researchers are able to interpret and visualize [28] the influential factors for the prediction. In this paper, we adopt attention concepts and construct an Interpretable Attention Network (IAN) to fit into the churn prediction task.

III. Data and Analytical Framework

In this section, we present the description of three datasets used in our work (see Section III-A), definitions of user types and churn (see Section III-B), and problem statement (see Section III-C).

A. Datasets

This subsection depicts three LBSN datasets used in our paper, which are Yelp 2017, Yelp 2019, and Foursquare 2017. Yelp 2017 and Yelp 2019 comprise open-sourced information published on Yelp's website. Yelp 2017 spans from 2004 to 2017, and Yelp spans from 2004 to 2019. The Foursquare 2017 dataset, collected by Chen et al. [29], spans from October 2008 to February 2016. Table I below has the descriptive statistics for the three datasets.

Key components of the LBSNs shown are (1) users, (2) business, and (3) reviews. The rudimentary elements of the user profile are user id, name, starting date, and friend list which forms the explicit connection of the social network between LBSN site users. In the business profile, there are business id, name, location (latitude and longitude), and categories. It is noteworthy that usually there are 2-5 categories for one business in Yelp and Foursquare. On top of the user and business profile, the user will leave a review every time he or she visits a business. Thus, the "reviews" file embraces the record of user information, business information, visit date, and review content, from which useful attributes of every visit can be obtained and fed into our model.

B. Defining User Types and Churn

To clarify the scope of our work, we define different user types and elaborate on the churn prediction problem we tackle.

	Yelp 2017	Yelp 2019	Foursquare 2017
Total users (in millions)	1.3	2.0	62
Total business (in thousands)	175	209	13,000
Total reviews (in millions)	5.3	8	19
Staying users	4824	7995	6153
Churning users	1005	1864	524
Percentage of staying users	82.76%	81.09%	92.15%
Percentage of churning users	17.24%	18.91%	7.85%

Table I: Descriptive statistics of datasets. See Section III-C for the explanation of "churn" and "stay".

1) User Types: User-generated content (UGC) is of great importance to reflect user behavior in LBSNs, and supervised machine learning on review-based datasets performed well in previous research [5], [30]. Since the major goals of this paper are to predict accurately on user's churn-or-stay behavior and to offer an interpretable model, we decided to focus on users who have the substantial contribution to UGC, which are specified as users that had written at least 50 reviews in the observation period. We classified them as long-term producers. The observation period implies the period before the Start-Of-the-Future (see Section III-C for details). On the other hand, the rest of the users are categorized as ordinary producers. The threshold of contributions to discriminate long-term producers and ordinary producers is equivalent to the one in prior works [8], [9] so that our findings on LBSNs are parallel to the previous works.

2) Churn: Unlike the areas where "churn" has a clear definition which is the termination of the subscription, such as in telecommunications [31], "churn" has an obscure interpretation in LBSNs. There are many users who still retain their accounts but remain largely inactive, which can be considered to be churning users from the LBSN site maintainers' point of view. In our work, we employ the same threshold of a 1-year observation period as the precedent work on LBSNs churn prediction [5]. Users without reviews written for one year are identified as churned users.

C. Problem Statement

Considering the compatibility with previous LBSNs studies [5], [8], [9], we take the same definition of Start-Of-the-Future (SOF) being the date one year before the last date of the datasets. The SOF is January 2017 in Yelp 2017, January 2019 in Yelp 2019, and March 2015 in Foursquare 2017. The period before the SOF is called the observation period, in which we collect the attributes of users' 50 visits, and the period after SOF is called the prediction period, in which we stratify users into two groups based on whether they wrote a review. Users who made no review in the prediction period are labeled as "churn", otherwise as "stay". In Table I, there is a summary of the churn-or-stay status of users in three datasets. Given the explicit definition of labeling of users' future activity status, our goal is to give a precise prediction on users' churn probability, and at the same time the prediction logic can be understood and verified by human-beings [32].

IV. Interpretable Attention Network (IAN) for Churn Prediction

In this section, we explain our extracted features from the heterogeneous data of LBSNs, followed by the interpretable attention network model (IAN).

A. Feature Description

The LBSN datasets used in our prediction tasks can be viewed as a series of user reviews with multidimensional and multivariate features over time. In this sense, we can view the data for each user as a group of feature vectors $\{f_1^{(u)}, f_2^{(u)}, ..., f_k^{(u)}\}$ where k denotes the k^{th} features and u denotes the u^{th} user of total N users. If f_k is a temporal feature along with every review, it can be represented as a time sequence data $f_k \in R^h$ where h denotes the number of reviews a user possesses. Notice that we emphasize the churning behavior of long-term producers who have over 50 reviews before a prediction period. Hence h is always larger or equal to 50. When describing a single user in this paper, the number of reviews would always be 50 to maintain a consistent training shape. To provide a comprehensive view while maintaining the comparability between different data sources, we investigate features from these following aspects: *(1) temporal, (2) geographic, (3) venue, (4) social,* and *(5) linguistic*.

1) Temporal Features: Studies reveal that temporal information is essential when predicting a user's churning behavior [6]. Since temporal information is a classical time sequence data, it can be represented as $T^{(u)} = [t_1, t_2, ..., t_i]$, where t_i denotes the time information at timestamp i for a given user u. We employ three kinds of temporal information, namely (1) Δt, time interval between each review; (2) $t_{SOF} - t_i$, time interval between t_i and SOF which denotes Start-of-the-Future and (3) $1/\Delta t$, the reciprocal of the time interval. Besides (2), the other two temporal data has been proposed in previous studies [28]. The reason for adopting the time interval to the SOF is because it contains information about a user's review distribution along with time steps.

2) Geographic Features: Geographic features determine a user's moving pattern, and the most typical one is the moving distance. According to the previous research [5], a user's churning rate declines as the average moving distance increases, hence we include the moving distance as one of the investigating features in our research. In LBSNs, moving distance can be represented as a time sequence feature. Given a user u's location history P at each timestamp i, $P^{(u)} = [(x_1, y_1), (x_2, y_2), ..., (x_i, y_i)]$, where x is the longitude and y is the latitude, the moving distance $D^{(u)}$ is thus defined as the euclidean distance between each location, $[0, d(p_2, p_1), ..., d(p_i, p_{i-1})]$.

3) Venue Features: Venue features define the attributes related to the local businesses. We select two kinds of venue features, namely accumulated reviews and category, to accomplish our prediction tasks. A user u's accumulated reviews on venue can be represents as a time sequence data, $A^{(u)} = [a_1^{(s)}, a_2^{(s)}, ..., a_i^{(s)}]$, where $a_i^{(s)}$ denotes the accumulated number of reviews of the specific venue, s, at user's i^{th} review. Similar to the accumulated reviews, category features can also be represented as a time sequence data. However, because a venue could be classified in multiple categories, it is not simply a single time sequence vector. The multivariate nature of the category feature is similar to the EHR data investigated in the RETAIN model [12], [33]. In LBSN, a user u's category feature can be viewed as a multidimensional time sequence data, $C^{(u)} = [c_1^q, c_2^q, ..., c_i^q]$, meaning that for each venue the user visits at timestamp i, there exists a set of categories $c_i^q = \{c_i^1, c_i^2, ..., c_i^q\}$ that classifies the venue, and q denotes the number of categories of the venue. Previous research [5] indicates that on average, stayers are more likely to visit different categories of venues, and churners write reviews on venues with less accumulated reviews.

4) Social Features: Existing research shows that social relationships play a crucial role in user's churning behavior [10], [11]. Churning users tend to have more churn friends, therefore, in this paper, we extract user's churn friends probability as our social feature. The churn friends probability of a user u, $F^{(u)}$, is a single variable indicating the percentage of u's friends who are classified as churning users.

5) Linguistic Features: Linguistic features represent the characteristics related to the raw text review. We propose to analyze raw text's influence on churn behavior using the Hierarchical Attention Network. Given a user u, the plain text review feature can be represented as a time sequence vector $W^{(u)} = [w_1, w_2, ..., w_i]$, where w_i denotes the plain text information of the review at timestamp i. In addition, we include review length as one of our linguistic features as used in [5]. Review length is simply the word count of the review, given a user u, review length $L^{(u)} = [l_1, l_2, ..., l_i]$, where l_i represents the word count of the review at timestamp i.

B. Interpretable Attention Network

Although LSTM demonstrated superior performance on the churn prediction problem in LBSNs [5], it is limited by the lack of interpretability. Therefore, our work is focused on contriving a deep learning model with the capability of accurate prediction as well as self-interpretability. Due to the complex nature of our LBSN data, we design our IAN model using two different techniques to process non-text and textual data. In the end, we concatenate the output from both sides using linear regression to generate our final prediction. Below is how we establish IAN.

1) Non-text Feature Prediction and Interpretation: As mentioned in feature description (Section IV-A), non-text features possess temporal, multidimensional, and multivariate characteristics. To predict while preserving the interpretability of the prediction, we refer to RETAIN, which is a model that is capable of dealing with the temporal sequence of multidimensional data. It also has great interpretability from end to end compares to other standard attention models [12]. In general, RETAIN utilizes a visit-level attention and an attribute-level attention to predict and interpret. We first transform categorical features into a continuous and trainable embedding. We apply embedding matrices $W_{emb} \in R^{m \times r}$ to the input category features $C \in R^r$, and retrieve category embedding $v_1, v_2, ..., v_i \in R^m$, where m denotes the embedding shape and r denotes the category dimension. The category embedding is then concatenated with numerical features, namely moving distance (D), review length (L), accumulated reviews (A), temporal information (T), and probability of churning friends (F). The category embedding is concatenated with numeric information, having dimension $m' = m + \#$numeric features. The full embedding is then loaded into two bi-directional RNNs, generating review attention $[\alpha_1, \alpha_2, ..., \alpha_i] \in R^i$ and attribute attention over embedding $\beta \in R^{i \times m'}$. In the final stage, the two attention weights are involved in obtaining the context vector $c_i = \sum_{i=1}^{h} \alpha_i \beta_i \odot v_i'$, where v_i' denotes the full embedding with categorical and numeric information concatenated. Finally, we use the context vector to predict the label $y^{(u)}$ by applying the Softmax activation function. It can be written as: $y^{(u)} = Softmax(W_{out}(c_i) + b)$, where $W_{out} \in R^{m\prime}$. The interpretability of the RETAIN model is based on the fact that the embedding of the input vector is engaged in generating the context vector c_i, which is later directly loaded into the Softmax activation function for the final prediction result. The α and β attention could be viewed as weights to help determine which review and embedding attribute the model should focus on. The α attention weights represent the importance of each review at different timestamps. On the other hand, since predictions are generated using the context vector c_i, we can write: $p(y^{(u)}|c_i) = Softmax(W_{out}(c_i)+b)$, According to the fact that $c_i = \sum_{i=1}^{h} \alpha_i \beta_i \odot v_i$ and v_i is the sum of the columns of W_{emb}, we can re-write the aforementioned equation as: $Softmax\left(\sum_{i=1}^{h} \sum_{j=1}^{r} x_{i,\,j} \alpha_i W_{out} \left(W_{emb} \odot \beta_j\right) + b\right)$, where $x_{i,j}$ represents the input embedding value of the j^{th} attribute at i^{th} review. According to the contribution to the Softmax function, we can then figure out the importance of the specific feature, $\omega_{i,j}$, as: $x_{i,\,j} \times \alpha_i W_{out} \left(W_{emb} \odot \beta_i\right)$. In conclusion, with α and β attention, we can visualize the attention of a specific review and even the contribution of the specific feature in that particular review.

2) Textual Feature Prediction and Interpretation: In this section, we explain how we predict using textual features. We adopt the Hierarchical Attention Network (HAN) which can learn the relationship between words while preserving a certain level of interpretability [14]. It utilizes a two-level attention mechanism, specifically word level and sentence level attention, to conduct document classification. However, since the reviews in the LBSN applications contain few words and sentences, we decide to construct the review-level attention instead of implementing the sentence-level attention. This is similar to the implementation of Cheng et al. [20], which generates comment level attention to detect cyberbullying on social network applications. The model can be resolved into 4 parts, which are word-level encoder, word-level attention, review-level encoder, and review-level attention. For a given user, $w_{i,j}$ represents the word which is located at the j^{th} position of the i^{th} review. Note that $j \in [1, T]$ and $i \in [1, R]$,

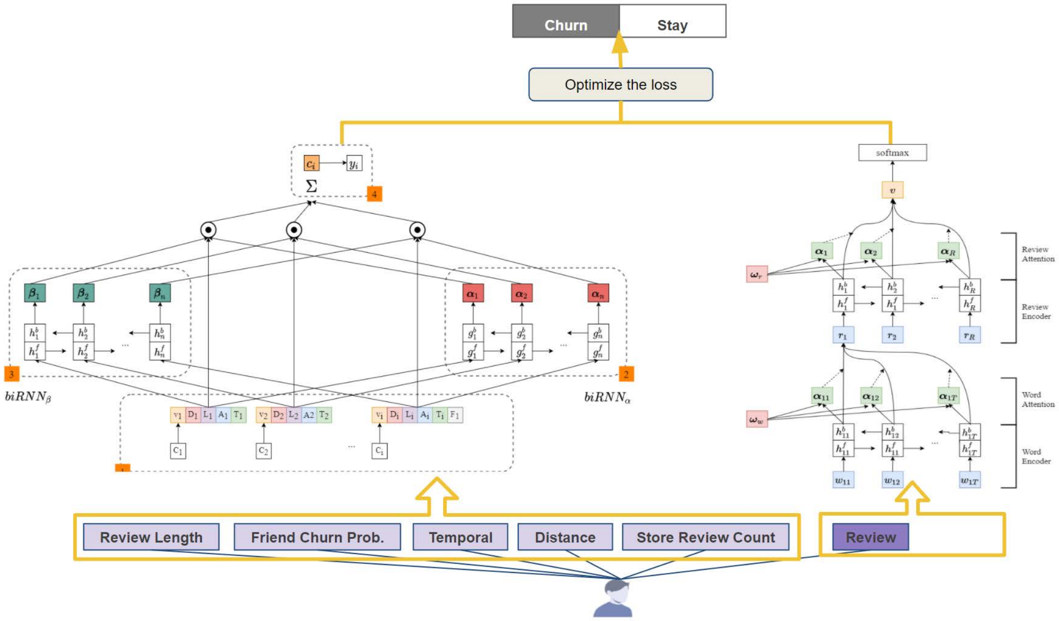


Figure 1: Overview of IAN. This is the high layer structure of our final model. Left: prediction of non-text features; Right: prediction of text features. We concatenate the prediction results from both sides using linear regression to generate the final output.

where T denotes the total number of words in a review and R denotes the total number of reviews.

In the word-level encoder, $w_{i,j}$ are first embedded into trainable vectors v_{ij}, and further fed into a bidirectional RNN (e.g., GRU) to obtain and capture hidden textual information, h_{ij}. The whole process could be formulated as: $v_{ij} = W_{emb}w_{ij}, W_{emb} \in R^{m \times Q}$; $h_{ij} = [h^f_{ij}; h^b_{ij}] = biRNN(v_{ij})$ where m denotes the embedding dimension, Q denotes the vocabulary dimension, and the superscript f, b represents the forward and backward hidden output from a bi-directional RNN.

In word-level attention, the previous output h_{ij} will first pass through a single layer MLP to obtain the hidden representation ω_{ij}. We then randomly initialize and train a ω_w representing the word-level context vector and compute the similarity between them by applying Softmax to reach the final normalized word-level attention, α_{ij}. Finally, the review-level representation, r_i, is calculated as the weighted sum of h_{ij} and α_{ij}. The whole process can be written as:

$$\omega_{ij} = tanh(Wh_ij + b)$$

$$\alpha_{ij} = Softmax(\omega_{ij}^\top \omega_w)$$

$$r_i = \sum_{j=1}^{T} \alpha_{ij} h_{ij}$$

For review level, we apply the similar process to r_i again to generate review level attention, except r_i is directly fed into the bi-directional RNN instead of being embedded. Therefore, the review encoder and review level attention could be written as:

$$h_i = [h^f_i; h^b_i] = biRNN(r_i)$$

$$\omega_i = tanh(Wh_i + b)$$

$$\alpha_i = Softmax(\omega_i^\top \omega_r)$$

$$s = \sum_{i=1}^{R} \alpha_i h_i$$

where s denotes the final summary of the user's reviews, and ω_r represents the review-level context vector. To obtain the final prediction $y^{(u)}$ of a given user u, the Softmax activation function is applied: $y^{(u)} = Softmax(W_{out}s + b_{out})$. The interpretability of HAN is achieved by the hidden representation output in the word-level encoder and review-level encoder (i.e. $[h_{it}]$ and $[h_i]$). They could be interpreted as the "annotation" of word and review [14], [20], which was proved to be effective in capturing diverse context and assigning context-dependent weight. Therefore, the trained context vector ω and the normalized importance weights α can be viewed as indicating the informative context. This is why the Hierarchical Attention network can possess a certain level of interpretability.

3) IAN Model: After receiving the results from two sides, we concatenate them and apply a single dense layer to generate the final prediction results. The high-level structure of the fusion model can refer to Figure 1. It can be viewed as implementing a simple linear regression that concatenates the prediction results generated by the RETAIN model and HAN model respectively. Since both models utilize the attention network, we called it Interpretable Attention Network (IAN). After the concatenation, we can easily examine the final weights of the dense layer as the models' contribution to the final prediction results. The process can be written as: $s^{(u)} = W_{out}[y^{(u)}_{RETAIN}; y^{(u)}_{HAN}] + b_{out}$, where $s^{(u)}$ denotes the final scalar of a given user u. Subsequently, $s^{(u)}$ will be applied to a Sigmoid activation function to generate our final prediction,$\hat{y}^{(u)}$. This can be written as $\hat{y}^{(u)} = \frac{1}{1+e^{s^{(u)}}}$. The prediction represents whether the user u will churn (0) or stay (1) from the LBSN site. Since we are making a binary classification, we train our model by optimizing the binary cross-entropy loss, which can be written as: $L = -\frac{1}{N}\sum_{u=0}^{N} \left(y^{(u)}log\left(\hat{y}^{(u)}\right) + \left(1-y^{(u)}\right) log\left(1-\hat{y}^{(u)}\right)\right)$, where $^{(u)}$ denotes the true value of the target user u.

V. Experiments

In this section, we conduct experiments to evaluate the performance and interpretability of the models using three different datasets: Yelp 2017, Yelp 2019, and Foursquare 2017.

A. Experimental Setup

1) Baselines: Previous work [5] utilized LSTMs to predict user churning behavior with social, geographical, temporal, and linguistic features and demonstrated its superior performance. Thus, we adopted LSTM as our strong baseline model. To further improve our baseline model performance, we considered the bi-directionality of LSTM and constructed two bidirectional LSTM models: (1) single bi-directional LSTM, and (2) 2-stacked bi-directional LSTM. We employed bidirectional LSTMs as our upper bound model in our experiment since interpretable models often sacrifice their performance to obtain the interpretability. We train a benchmark with the same set of features as our proposed interpretable models, that is, we also employ an embedding matrix to embed categorical features. Consequently, we concatenate numerical features with the embedded categorical features and fit them into the earlier models for each time step.

2) Models for Evaluation: We plan to conduct experiments on the IAN model to evaluate its performance and its interpretability. When evaluating the performance, we would focus on: (1) The performance difference between IAN and the aforementioned two LSTM models in Section V-A1. We adopt Adamax as our optimizer since word embedding is involved, and Adamax can help contrast the sparse behavior to dense behavior. We set the batch size to 32, and limit the output of all the attention weights to be non-negative. All of our models are implemented using Keras with TensorFlow as their backend architecture. Finally, to ensure that all baseline models and IAN models are training with the same data, we regulate and record the random seed every time we generate new train and test datasets.

3) Evaluation Protocol: To evaluate the effectiveness of the above models, we set up experiments to predict churning behaviors of long-term users using features regarding the last 50 reviews. Due to the datasets being very imbalanced (See Table I), several evaluation metrics have been adopted to evaluate the performance (see Section V-A4). We split 80% of the data for training and 20% for testing, with random oversampling on the training dataset to deal with the imbalanced data. Each model is trained for 200 epochs. By monitoring the loss value of the validation data, we select the models with the optimized loss and report the corresponding performance results in Section V-B.

4) Metrics: With the purpose of appraising and juxtaposing the performance of the models, we determine to use Precision, Recall, F-1 Score, AUC-ROC, and PR-AUC, as the evaluation metrics. The elucidation and justification are as follows.

Fundamentally, one manifest feature of our datasets is the imbalance of the proportions of different types of users. Referring to Table I, the staying users comprise 82.76% of Yelp 2017, 81.09% of Yelp 2019, and 92.15% of Foursquare 2017. Hence, the common metrics which are robust for unskewed data cannot fit our skewed data and may lead to misleading conclusions [34]. Therefore, besides using normal evaluation metrics such as Precision, Recall, and F-1 Score, we adopt ranking metrics. It contains AUC-ROC and PR-AUC. They focus on how well the model ranks the examples and how effective it is at class separation [35]. AUC-ROC is the area under the ROC curve. The closer AUC-ROC approaches to 1, the stronger separability the model has [36]. Similar to ROC, Precision-Recall (PR) curve is plotted Precision against Recall for different probability thresholds and PR-AUC represents the area under the PR curve. PR curves are appropriate for highly skewed data where ROC curves may have an over-optimistic view of the performance [34].

B. Performance Evaluation

Table 2-4 provide the performance evaluation for the baseline models and the IAN model on Yelp 2017, Yelp 2019, and Foursquare data, respectively. In general, most of the models achieve more than an 80% AUC-ROC score. We observe that the IAN model we proposed has similar performance to the baseline model in the three different LBSN datasets. Although the IAN model itself cannot completely outperform those traditional LSTM implementations, it could instead reach a high level of interpretability, and this is what we are aiming for in our research. Among the three datasets', we can observe that the performance of Yelp 2017 is consistent with Yelp 2019. However, our models are having incredible results using Foursquare datasets. This is due to the extremely imbalanced data. The portion of the churning user is extremely small in the foursquare dataset (i.e. 8%) compared to the case in the Yelp dataset (i.e. 25%). Therefore, it would be better to analyze Recall to determine a model's performance, where we can also discover that the IAN model has better performance. The IAN model provides a fresh aspect to solve the churn prediction of LBSN users while achieving high-level interpretability which we discuss in the following section. (Section VI).

VI. Visualization, Interpretation and Analysis

Visualization is a strong tool that can assist the interpretation of IAN outcomes and help with real-world analysis. Since our

	AUC-ROC	PR-AUC	Precision	Recall	F-1 Score
Logistic Regression [5]	0.7680	-	-	-	-
LSTM [5]	0.8820	-	-	-	-
Bi-di LSTM (one stack)	0.9046	0.9449	0.8450	0.8276	0.8319
Bi-di LSTM (two stacks)	0.8808	0.9389	0.8217	0.8187	0.8200
IAN	0.9072	0.9478	0.8481	0.8207	0.8263

Table II: Model performance measured for Yelp 2017 dataset

	AUC-ROC	PR-AUC	Precision	Recall	F-1 Score
Bi-di LSTM (one stack)	0.8737	0.9200	0.8075	0.7975	0.8004
Bi-di LSTM (two stacks)	0.8639	0.9180	0.8010	0.7963	0.7980
IAN	0.8659	0.9195	0.8127	0.8100	0.8111

Table III: Model performance measured for Yelp 2019 dataset

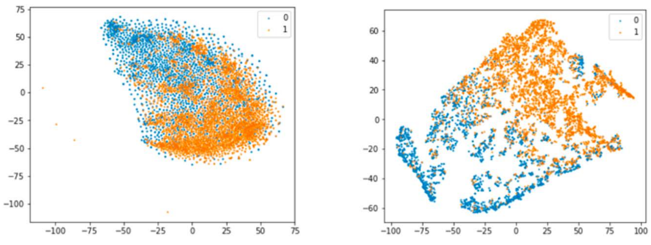
	AUC-ROC	PR-AUC	Precision	Recall	F-1 Score
Bi-di LSTM (one stack)	0.9803	0.9982	0.9584	0.9454	0.9497
Bi-di LSTM (two stacks)	0.9915	0.9992	0.9743	0.9723	0.9731
IAN	0.9961	0.9997	0.9827	0.9790	0.9800

Table IV: Model performance measured for Foursquare 2017 dataset

main goal is to explain and interpret churning user prediction, we developed 2 different visualization analyses. They are: (1) Summary of the dataset, and (2) Mean Attention Analysis

A. *Dataset Overview*

The first function, dataset overview, aims to put forward encapsulated information on the whole dataset, such as the similarity of users and commonly-used words in reviews.



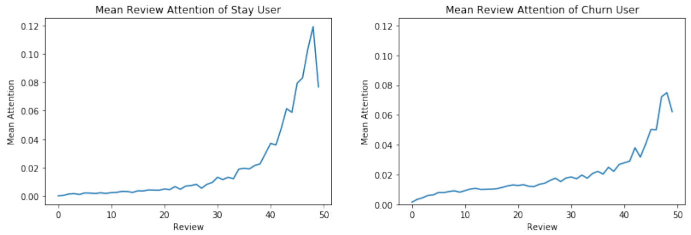
t-SNE with raw data of user features

t-SNE with context vector of RETAIN in IAN

Figure 2: Dataset Overview: t-SNE

The t-SNE part in Figure 2 renders a summary of a long-term producer with respect to venue category, distance, review length, accumulated venue review count, and probability of churning friends of their 50 visits. The t-SNE model gives a projection of high-dimensional data into a 2-dimensional map while preserving the distances between neighboring data points, which gives it the advantage of maintaining original clustering [37]. We use two datasets to portray user features. The first is to directly use attribute data of one users' 50 visits. The other dataset exerted is the context vector of non-text attention network in IAN, which is the summary of all vectors in the embedding layer and encompasses all the information of each attribute and each visit. Comparing the two t-SNE charts, we find that the distinction of clustering is improved by applying context vector c_i instead of raw data of user features. This demonstrates that context vector is a better discriminator and that IAN is able to capture latent representations of the users so it can enhance the ability to distinguish different users.

B. *Mean Attention Analysis of IAN*



(a) Staying users

(b) Churning users

Figure 3: Mean Review Attention of Stayers and Churners

To understand sweeping user behaviors and predominant reasons for churning, we visualize and study the average attention of review-level and numeric features of 50 reviews. The Yelp 2017 dataset was chosen as a case study.

Figures 3a and 3b show that later reviews are more influential on the prediction. Although the two distributions have similar trends, we can observe that when predicting churn users, the model tends to apply more attention to previous reviews compared to the attention distribution of the stay users. This could be observed when running user-specific studies, in which the value of churn user attention may oscillate in earlier reviews.

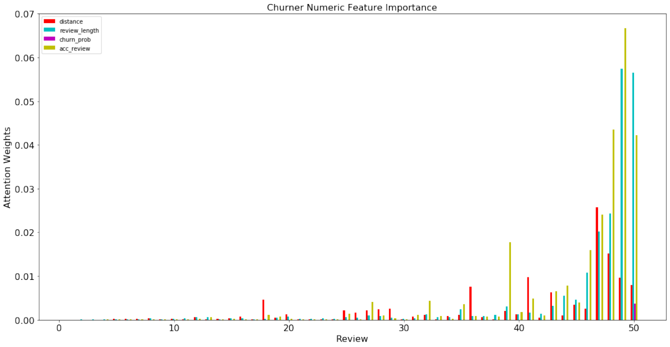


Figure 4: Mean Attention of Numeric Features of Churning User

One of the main advantages of the non-text attention network is that it could compare features' attention over every review. Figure 4 are the average attention of the numeric features (i.e. Moving Distance, Review Length, Churn User Probability, and Accumulated Review Count) over 50 reviews. Similar to the review attention mentioned earlier, they both follow the trend that later reviews are more important. We can also observe that accumulated review count and review length contributes more to the prediction. Although the moving distance feature receives a smaller attention weight compared to the previous two attributes, we can discover that it has more influence when predicting churning users. To conclude, based on the quantitative analysis of the attention weights refined from the RETAIN segment of the IAN model, it successfully reflects the interpretability regarding the importance of over 50 reviews and of different numeric features.

VII. Conclusion

In this paper, we studied the interpretability regarding the churn prediction problem in LBSNs using two datasets from Yelp and one from Foursquare 2017. We initially narrowed down the scope of our investigation into a particular user type, a long-term producer, contributing a significant amount of reviews by characterizing user types. After that, to solve the churn prediction problem, we proposed our novel interpretable model, called IAN, to improve the interpretability of "black-box" neural networks as well as ensure high performance in predicting potential churners in advance. Furthermore, we conducted extensive experiments using three real-world datasets of LBSNs and verified that IAN performs as good as all the baselines. It also offers accurate and intuitive prediction results, which are easier for us to understand. Finally, based on the attention outputs, we conduct several analyses using visualization techniques to help interpreting the prediction from IAN.

References

[1] M. Karnstedt, T. Hennessy, J. Chan, P. Basuchowdhuri, C. Hayes, and T. Strufe, "Churn in Social Networks," 2010, pp. 185–220.

[2] G. Dror, D. Pelleg, O. Rokhlenko, and I. Szpektor, "Churn Prediction in New Users of Yahoo! Answers," in *Proceedings of the 21st International Conference on World Wide Web*. ACM, 2012, pp. 829–834.

[3] Y. Zhu, E. Zhong, S. J. Pan, X. Wang, M. Zhou, and Q. Yang, "Predicting User Activity Level in Social Networks," in *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, ser. CIKM '13. New York, NY, USA: ACM, 2013, pp. 159–168.

[4] C. Yang, X. Shi, L. Jie, and J. Han, "I know you'll be back: Interpretable new user clustering and churn prediction on a mobile social application," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*. ACM, 2018, p. 914–922.

[5] Y. D. Kwon, D. Chatzopoulos, E. U. Haq, R. Wong, and P. Hui, "Geolifecycle: User engagement of geographical exploration and churn prediction in lbsns," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, pp. 1–29, 09 2019.

[6] J. Pudipeddi, L. Akoglu, and H. Tong, "User churn in focused question answering sites: characterizations and prediction," 04 2014, pp. 469–474.

[7] D. Nguyen and C. P. Rosé, "Language Use As a Reflection of Socialization in Online Communities," in *Proceedings of the Workshop on Languages in Social Media*, ser. LSM '11. ACL, 2011, pp. 76–85.

[8] C. Danescu-Niculescu-Mizil, R. West, D. Jurafsky, J. Leskovec, and C. Potts, "No country for old members:user lifecycle and linguistic change in online communities." *Proceedings of the 22nd international conference on World Wide Web - WWW 13*, 2013.

[9] C. Tan and L. Lee, "All who wander: On the prevalence and characteristics of multi-community engagement," *Proceedings of the 24th International Conference on World Wide Web - WWW 15*, 2015.

[10] R. Oentaryo, E.-P. Lim, D. Lo, F. Zhu, and P. K. Prasetyo, "Collective churn prediction in social network," 08 2012, pp. 210–214.

[11] K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A. Nanavati, and A. Joshi, "Social ties and their relevance to churn in mobile telecom networks," 01 2008, pp. 668–677.

[12] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," pp. 3504–3512, 2016.

[13] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," *arXiv:1508.01991 [cs]*.

[14] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," pp. 1480–1489.

[15] M. N. Abd-Allah, A. Salah, and S. R. El-Beltagy, "Enhanced customer churn prediction using social network analysis," in *Proceedings of the 3rd Workshop on Data-Driven User Behavioral Modeling and Mining from Social Media*. ACM, p. 11–12.

[16] A. M. Rashid, K. Ling, R. D. Tassone, P. Resnick, R. Kraut, and J. Riedl, "Motivating participation by displaying the value of contribution," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '06. ACM, 2006, p. 955–958.

[17] Y. Tian, K. Zhou, M. Lalmas, and D. Pelleg, "Identifying tasks from mobile app usage patterns," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20. ACM, 2020, p. 2357–2366.

[18] Y. Wang, N. J. Yuan, Y. Sun, F. Zhang, X. Xie, Q. Liu, and E. Chen, "A contextual collaborative approach for app usage forecasting," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '16. ACM, 2016, p. 1247–1258.

[19] S. Xu, S. Lai, and M. Qiu, "Privacy preserving churn prediction," in *Proceedings of the 2009 ACM Symposium on Applied Computing*, ser. SAC '09. ACM, 2009, p. 1610–1614.

[20] L. Cheng, R. Guo, Y. Silva, D. Hall, and H. Liu, "Hierarchical attention networks for cyberbullying detection on the instagram social network," 02 2019.

[21] R. J. Oentaryo, E. Lim, D. Lo, F. Zhu, and P. K. Prasetyo, "Collective churn prediction in social network," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2012, pp. 210–214.

[22] Y. Wang, Y. Guo, and Y. Chen, "Accurate and early prediction of user lifespan in an online video-on-demand system," in *2016 IEEE 13th International Conference on Signal Processing*, 2016, pp. 969–974.

[23] X. Wang, K. Nguyen, and B. P. Nguyen, "Churn prediction using ensemble learning," in *Proceedings of the 4th International Conference on Machine Learning and Soft Computing*, ser. ICMLSC 2020, 2020.

[24] Q. Zhao and T. Hastie, "Causal interpretations of black-box models," *Journal of Business Economic Statistics*, pp. 1–19, 06 2019.

[25] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. ACM, 2016, p. 1135–1144.

[26] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Curran Associates Inc., p. 4768–4777.

[27] R. Moraffah, M. Karami, R. Guo, A. Raglin, and H. Liu, "Causal interpretability for machine learning - problems, methods and evaluation," *SIGKDD Explor. Newsl.*, vol. 22, no. 1, p. 18–33, May 2020.

[28] B. C. Kwon, M.-J. Choi, J. T. Kim, E. Choi, Y. B. Kim, S. Kwon, J. Sun, and J. Choo, "Retainvis: Visual analytics with interpretable and interactive recurrent neural networks on electronic medical records," *IEEE Transactions on Visualization and Computer Graphics*, 2019.

[29] Y. Chen, J. Hu, H. Zhao, Y. Xiao, and P. Hui, "Measurement and analysis of the swarm social network with tens of millions of nodes," *IEEE Access*, vol. 6, pp. 4547–4559, 2018.

[30] Y. Chen, J. Hu, Y. Xiao, X. Li, and P. Hui, "Understanding the user behavior of foursquare: A data-driven study on a global scale," *IEEE Transactions on Computational Social Systems*, 2020.

[31] N. Hashmi, N. A. Butt, and D. Iqbal, "Customer churn prediction in telecommunication a decade review and classification," *IJCSI*, vol. 10, pp. 271–282, 09 2013.

[32] Q.-S. Zhang and S.-C. Zhu, "Visual interpretability for deep learning: a survey," *Frontiers of Information Technology Electronic Engineering*, vol. 19, no. 1, p. 27–39, 2018.

[33] A. Gkoulalas-Divanis, G. Loukides, and J. Sun, "Publishing data from electronic health records while preserving privacy: A survey of algorithms," *Journal of Biomedical Informatics*, vol. 50, pp. 4–19, 2014.

[34] P. Branco, L. Torgo, and R. Ribeiro, "A Survey of Predictive Modelling under Imbalanced Distributions," *arXiv e-prints*, May 2015.

[35] C. Ferri, J. Hernández-Orallo, and R. Modroiu, "An experimental comparison of performance measures for classification," *Pattern Recognition Letters*, vol. 30, no. 1, p. 27–38, 2009.

[36] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[37] L. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, 2008.