Customer Churn Reduction Based on Action Rules and Collaboration

C

Yuehua Duan
University of North Carolina at Charlotte, USA

Zbigniew W. Ras
https://orcid.org/0000-0002-8619-914X
University of North Carolina at Charlotte, USA & Polish-Japanese Academy of Information Technology in Warsaw, Poland

INTRODUCTION

Customer churn, also known as customer attrition, refers to the loss of existing customers who cease the relationship with an organization in a period of time (Renjith, 2017; Jain & Jana, 2021). Customer churn leads to lower volume of service consumption, reduced product purchase, less customer referrals. Furthermore, the cost of acquiring a new customer is much higher than the cost of retaining an existing customer (Siber, 1997). Reducing customer churn can be significantly beneficial (Van den Poel & Lariviere, 2004; Kowalczyk & Slisser, 1997). For example, in financial services, a 5% increase in customer retention produces more than a 25% increase in profit (Reichheld and Detrick, 2003).

It is imperative to build a recommender system that can provide effective recommendations to reduce customer churn. Recommender system is a subclass of information filtering systems that seek to predict the rating or preference a user would give to an item (Ricci, Rokach, & Shapira, 2011). According to the techniques applied, recommender systems are categorized to - collaborative filtering recommender systems, content-based recommender systems, demographic recommender systems, knowledge-based recommender systems, community-based recommender systems, and hybrid recommender systems (Su & Taghi, 2009). The process of building a knowledge-based recommender system is facilitated with the use of knowledge base, which contains data about rules and similarity functions to use during the retrieval process (Jannach, Zanker, Felfernig, & Friedrich, 2011). It relies more on the domain knowledge, by utilizing the expert knowledge to decide which item to recommend, and to what extent this item is meaningful and useful to the user.

Action rule mining is one of the technologies that have been successfully applied in building knowledge-based recommender systems to address the customer churn issue (Tarnowska & Ras, 2019; Tarnowska, Ras, & Daniel, 2020). In a knowledge-based recommender system based on classification rules and action rules, action rule mining is a major step in extracting knowledge in the process of recognizing the recommendations. The quality of action rules determines the effectiveness and coverage of recommender systems. Normally, support and confidence are used to measure the quality of discovered action rules (Ras & Wieczorkowska, 2000; Tzacheva, Sankar, Ramachandran, & Shankar, 2016). In practical applications, action rules are regarded as interesting only if their support and confidence exceed the predefined threshold values. Moreover, if an action rule has a large support and high confidence, it indicates that this action can be applied on a large portion of customers with a high chance (Ras & Tsay, 2003) to be successful. To increase the efficiency of knowledge-based recommender systems which use

DOI: 10.4018/978-1-7998-9220-5.ch035

action rule mining for reducing customer churn, it is necessary to improve the quality of discovered action rules. However, there is little research work done which focuses on improving the quality of discovered action rules.

To bridge forementioned gap, the authors propose a guided (by threshold) agglomerative clustering algorithm to improve the confidence and coverage of discovered action rules. Assume that there are many similar businesses called clients and each one is faced with a customer attrition problem. Each client is represented by a decision table describing its customers status (Tarnowska et al., 2020; Duan & Ras, 2021). Customer status, which is a decision attribute, has three values: active, leaving, and lost. In the process of action rule mining, the goal is to discover action patterns which will change customer status from leaving to active. The proposed algorithm aims to improve the quality of action rules extracted from a dataset of a given client by utilizing the knowledge extracted from other semantically similar clients to that given client. The idea is to pick up clients which are not only semantically similar but also which are doing better in business than a given client. By doing that, the given client can follow business recommendations coming from other semantically similar and better performing clients. The algorithm is guided by a threshold value targeting the minimal acceptable improvements in confidence and coverage of discovered action rules. If the improvement is lower than this threshold, algorithm stops.

BACKGROUND

Over the years, there have been many recommender systems proposed to address the customer churn issues. Kim and Yoon (2004) investigated the strategies of businesses in Korean mobile telecommunication services to increase customer loyalty. Their conclusion is that mobile carriers must focus on service quality and offer customer-oriented services to heighten customer satisfaction. Daskalaki, Kopanas, Goudara, and Avouris (2003) built a decision support system to handle customer insolvency for a large telecommunication company. Duan and Ras (2021) designed and implemented a recommender system that can provide actionable recommendations for improving customer churn rate. Wang, Chiang, Hsu, Lin, and Lin (2009) used Decision Tree algorithm to analyze more than 4000 members over three months. The conditional rules produced by Decision Tree algorithm show the characteristics of customer behavior that can lead to customer loss. They use such rules as strategy recommendations to prevent future customer loss.

Action rules are widely applied in building knowledge-based recommender systems. The concept of action rule was proposed by Ras and Wieczorkowska (2000). Action rule is defined as a rule extracted from a decision system that describes a transition of its objects from one decision state to another. Informally, it is defined as a term: $\left[(\omega)\wedge(\alpha\rightarrow\beta)\Rightarrow(\phi\rightarrow\psi)\right]$, where ω denotes a conjunction of fixed stable attributes called the header of the rule, $(\alpha\rightarrow\beta)$ are proposed changes in values of flexible attributes, and $(\phi\rightarrow\psi)$ is a desired change of the decision attribute value. To give an example of an action rule, let's first give the definition of information system. An information system is defined as a pair $S=(U,A)$, where U is a nonempty, finite set of objects called the universe. A is a nonempty, finite set of attributes i.e., a: $U\rightarrow V_a$ for $a\in A$, where V_a is called the domain of a (*Pawlak*, 1981). In an information system S shown in Table 1, there are five objects in X, that is $X=\{x_1, x_2, x_3, x_4, x_5\}$. $A=\{a,c,d\}$ and $V=\{a_1, a_2, c_1, c_2, d_1, d_2\}$. Assume that attribute a is stable and attribute c is flexible. The decision attribute is d. Example of an action rule is $\left[(a,a_2)\wedge(c,c_2\rightarrow c_1)\Rightarrow(d,d_1\rightarrow d_2)\right]$. The meaning of this action rule is that for objects having properties (a,a_2), (c,c_2) and (d,d_1) if their flexible attribute value changes from c_2 to c_1, then it is expected that the value d_1 will change to d_2 .

Related Content