Alıcı, A., & Kazamel, M.S.R.,. (2021). AI Supported Smart Service Recommendatşon Algorithm. *TOGU Career Research Journal*, 2(2), 41-46.

TOGU Career Research Journal, 2(2), 41-46

AI Supported Smart Service Recommendation Algorithm

Ali ALICI[*], Mohammed Saif Ragab KAZAMEL[†]

Abstract

Armut Technology is an online platform that brings together customers and service providers, and positions service providers as business partners with the principle of "Crowdsourcing". Nearly 4000 services are offered within the company. This number is increasing gradually as new service requests are also received from service providers. When customers login to the website or application, they search from the wide service pool by typing their desired service description. With this project, it is aimed to provide the services they need in real time when they are online by predicting them with artificial intelligence supported algorithms.

The related topic is modeled as "Recommendation Engine" under the machine learning discipline. All service requests coming in 2020 were used as a training set. Since the queues of the services requested by the customers are interconnected in terms of temporality, the requested service queues are modeled according to the "Conditional Probability Based Prediction" method. In order to capture exceptional customer behaviors, customer specific habits have also been added to the service list. All machine learning models run on the AWS cloud ecosystem. It has been developed with the principle of running web services in Docker containers, which is the industry standard and used during the service of machine learning models to the customer.

"Top-8 Service Accuracy" was chosen as the success metric of the project. The success rate of 22%, which is currently achieved by combining popular services throughout Turkey, has been increased to 37% with the new algorithm supported by AI. This rate comes up to 44% when we look at the customers who have had at least 1 service request in the past.

The current service recommendation system, which has difficulty in capturing special customer behaviors with the perspective of "popular service throughout Turkey", has significantly improved with the new AI supported approach by taking into account customer habits and the relationship between services. In the next stages of the project, innovative methods used in this field such as "Collaborative Filtering" and "RNN" will be performed together with the "ClickStream" data of the customers and the success rate will be tried to be increased.

Keywords: Artificial Intelligence, Recommendation Engine, Conditional Probability Based Prediction

Introduction

One of the most important factors affecting the success of digital platforms is user experience. Ease-of-use comes as one of the features that affects user experience the most. Ease-of-use includes presenting and highlighting the services that the customers would want or need at the time and providing easy access to these services with one click. This will enable customers to use the digital platform in an easier and faster way and will provide them with an impressive user experience. This would also increase the loyalty of customers to the platform as well as increase the number of the requested services on the platform. The project presented in this paper aims at improving the user experience by predicting the next service that the customer will request on Armut's platform.

Armut Technology works as a service marketplace that aims at matching the customers' local service requests with the best local service providers and fulfilling the requested service with the best experience to the customer and the service provider. Service providers see the job opportunities on the app and they apply to the ones they

[*] Armut Teknoloji AŞ, İstanbul, Türkiye, ORCID Number: 0000-0002-1282-7545, E-posta: ali@armut.com

[†]Armut Teknoloji AŞ, İstanbul, Türkiye, ORCID Number: 0000-0002-4949-8159, E-posta: mohammedk@armut.com

find most suitable. A job deal is created when a customer chooses the most suitable offer for them. After the job is done, customers can leave a rating and review regarding the job on Armut's system. Armut's revenue sources are mainly the job application lead price taken from the providers when they want to see and quote the job. Among more than 4000 services available on Armut's platform, the most common services include Moving, Home Cleaning, and Private Lessons.

The predictions of the smart service recommendation algorithm will be shown to the customers through the web and mobile interfaces. At every session, eight services will be predicted and presented to the customer. If the recommendations are correct, the service request duration is expected to decrease and the number of requested services is expected to increase. This recommendation system will also be used as an input to the email and mobile communications with the customer.

Literature Search

In recent years; in parallel with the huge increase in online business models, "Smart Recommendation Systems" has gained significant importance. The biggest technology giant companies, Google (Liu, Dolan ve Pedersen, 2010), Twitter (Ahmed and others, 2013), LinkedIn (Rodriguez, Posse ve Zhang, 2012) and Netflix (Steck, 2013) are still making huge investments on "Smart Recommendation Systems" in order to ensure customer retention and increase sales. (Adomavicius and Tuzhilin, 2005, s. 5) breaks these systems into 3 main categories: "collaborative filtering", "content-based filtering" and "hybrid filtering"

"Collaborative filtering" approach depends on the patterns from similarity of users' product preferences or similarity of products used by common customers (Herlocker, Konstan, Terveen and Riedl, 2004, s. 6). The basic idea behind the algorithm is if customer X and customer Y have similar product preferences in the past then these customers are more likely to choose similar products in the future. For example; in an online music site, customer-customer similarities and song-song similarities are calculated and then used for relevant recommendations to customers.

"Content-based filtering" approach depends on deriving the specific and domain based features from the product and then using them to calculate similarities (Van Meteren and Van Someren, 2000, s. 3). For example; an e-commerce site in which computers are sold, only domain specific features like price, CPU type, CPU performance, storage capacity of the disk etc. are extracted and then similarity calculations are applied over them.

"Hybrid filtering" approach depends on the collaboration of both "collaborative filtering" and "content-based filtering" methods (Burke, 2002, s. 2). The similarity scores calculated by each of these algorithms then combined together by assigning relevant weights to obtain a new and more accurate similarity score. The new product recommendation is then served to customers based on this hybridly calculated similarity score.

Because of their fast-speed running times and having produced explainable/interpretable predictions, "Statistical Based Recommendation Engines" also existed as commonly applied methods in industry (Portugal, Alencar ve Cowan, 2018, s. 8). The most common method among all of them is called "Association Rule Mining (ARM)". Such method is based on comparing the probability of getting a single product with getting the same product with others (Zhang ve Zhang, 2002, s. 2).

Both their response time speed and their level of accuracy is improving together, that's why "Recurrent Neural Networks (RNN)" architecture which is placed under Neural Networks has also become used to build recommendation engines in recent years (Zhang, Yao, Sun ve Tay, 2019, s. 3). As modelling the products that customers bought as a sequence, next item prediction methods are applied via RNNs. Although such approaches give significantly good results in terms of business metrics, it needs a big volume of data and training time is significantly long. That's why it is still too hard to use them for industrial applications.

General Architecture of Newly Proposed Model and the Success Metrics

As Armut does with all its artificial intelligence projects, in this project, it first developed a simple/basic model and started using it and monitored its performance for a while. After that, taking this model's performance as a reference, more innovative AI models are developed and the performance of the new model is compared to the performance of the reference model. Thanks to this methodology, the performance improvement between the basic model and the AI model is quantified. In the "Smart Service Recommendation Algorithm" project, first "Apriori" and then "Popular Service Recommendation" was used as a baseline. After the use of the "Apriori" algorithm in the live environment, it was decommissioned because it didn't solve the "sequentiality" problem. Because in most ecommerce websites customers can add more than one item to their basket and then buy all of them at once, Apriori model can produce good results, however, in Armut where the services are ordered as a sequence of separate but interconnected requests, Apriori didn't produce suitable recommendations. Apriori's logic focuses on the difference between the probability that items bought separately and that they are bought together, therefore the fact that one item was bought before or after the other does not affect Apriori's recommendations. However, in Armut's case, for example, a Moving service request usually comes after a Home Painting request but the opposite doesn't usually happen. Because Apriori treats the two sequences as the same, it was decommissioned.

As a baseline model for the "Smart Service Recommendation" project, the "Popular Services Model" was selected. This model was developed in three flavours; the first is recommending the popular services all over Turkey, the second is recommending the most popular services in the past four weeks, and the third is recommending the most popular services in the past four weeks in the same subprovince where the customer resides.

Under the "Smart Service Recommendation" project, the "Conditional Probability Based Model" presented in this paper predicts the services that the customer may request based on the last N services that they requested most recently.

$$p(X_t|X_{t-1}, X_{t-2}, X_{t-3}, ..., X_{t-N}) = \frac{p(X_t, X_{t-1}, X_{t-2}, X_{t-3}, ..., X_{t-N})}{p(X_{t-1}, X_{t-2}, X_{t-3}, ..., X_{t-N})} \quad (1)$$

Using Equation 1, the probability of requesting any service based on the previous N service requests sequence can be calculated. Table 1 shows examples of such sequences. For the Conditional Probability Based Model, the probability for requesting any service when the length of the past service requests sequence N=3, N=2, and N=1 is calculated. Then the infrequent sequences that are statistically insignificant are removed. After that, for each customer the eight services that they will most probably request are predicted in real time. This flow is depicted in Figure 1.

Thanks to the "Smart Service Recommendation" project, customers receive personalized recommendations for eight services through Armut's mobile app and website. As for the success metric for this project, it is calculated based on whether the customer's next service request is for one of the recommended eight services or not. If the customer's request is for one of the eight services, then the prediction is labeled "correct" and if the customer requested another service then the prediction is labeled "incorrect". By dividing the count of correct cases over the count of all cases we can get an average score which we call "top8 accuracy". When comparing the performance of different models, we use this metric, as in Table 2 for example.

Table 1. Example Service Sequences

Service t-3	Service t-2	Service t-1	Service t
Plumber	Painter	Moving	Home Cleaning
Plumber	Painter	Moving	Home Cleaning
Plumber	Painter	Moving	Heat Insulation
Plumber	Painter	Moving	Heat Insulation
Plumber	Painter	Moving	Math Private Lesson

Table 1 shows the same three-service-requests sequence (Plumber-Painter-Moving) made by different customers and the last service request that came after that sequence for each of them. The Conditional Probability

Based Model would calculate the probability of requesting the Home Cleaning service after requesting this sequence as 40%, same for Heat Insulation, and 20% for Math Private Lesson based on the data in the table. The model's probabilities are recalculated frequently with the most recent data and the recommendations are made based on these probabilities.

The new model, by calculating conditional probabilities, represents the population behaviour. However some customers might behave consistently differently from the population and we wanted to take them into account in the model predictions. An example of this behaviour is a customer ordering the same service frequently, however this service sequence pattern is not a common one over the whole population. To take this case into account, in case the customer has frequently ordered services, we replace (at most) the three services with the least probabilities out of the eight services that the model predicted. We replace them with the most requested three (at most) services that the customer orders frequently. That way the model's predictions represent the population's behaviour as well as the customer's personal behaviour.

Table 2. Final model and baseline model performances. T8: Most popular eight services model. T8M: Most popular eight services in the past month model. T8MP: Most popular eight services last month in the same subprovince as the customer. CP: Conditional Probability Based Model.

Customer Segment	T8	T8M	T8MP	CP
All Customers	20.88%	21.35%	21.64%	35.97%
New Customers	21.70%	22.45%	22.70%	23.02%
Customers with one previous service request	22.02%	22.65%	22.72%	45.88%
Customers with two previous service requests	21.40%	21.88%	21.98%	44.58%
Customers with three or more previous service requests	19.51%	19.67%	20.13%	41.21%

Table-2, shows both reference models and "Conditional Probability Based Prediction Model" success metrics. Except for the case of customers with no history, our "Conditional Probability Based Prediction Model" significantly overperforms the referenced model as shown in the Table-2. Since the reason for the success of our newly proposed model is directly related to customers' service history, as the number of services getting by customers are increasing then the success gap is also increasing as expected.

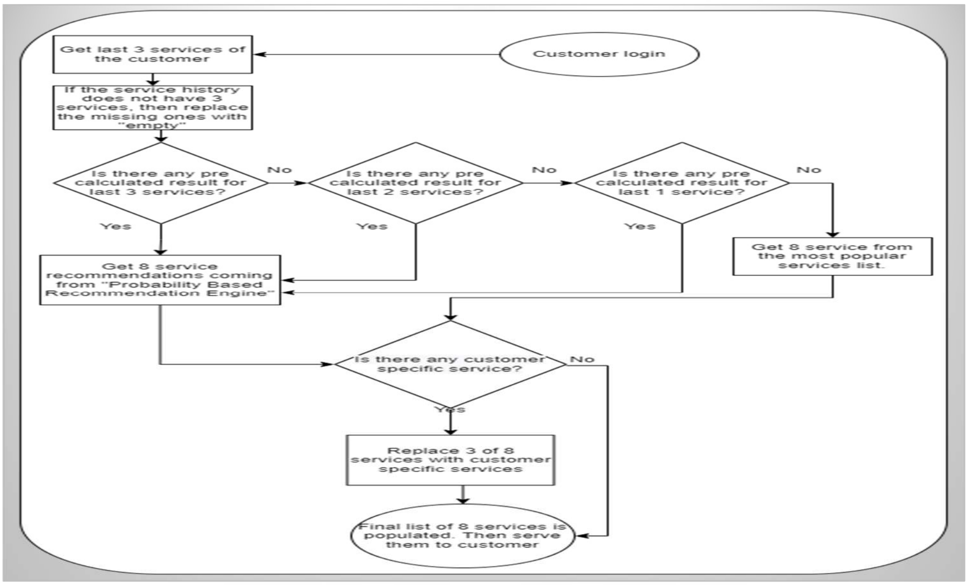

Figure 1. "Smart Recommendation Engine" flow

Result

"Smart Recommendation Engines" are so important for e-commerce companies, since they have a great potential to enable customer retention and boost product sales. The new approach called "Conditional Probability Based Recommendation Engine" offered in the scope of this study has significantly overperformed the reference models as mentioned in the performance section. As the number of services that customers received in the past increases, the model success metrics are also improved in parallel. Success rate goes up to 44% for the customers who have at least 2 services in their service history. This rate is almost 2 times of the referenced model called "Popular Services". In the next steps of this project, more innovative and state-of-the-art algorithms like "Collaborative Filtering" and "RNN" will be applied. But the most significant drawback of these algorithms is their nature of time consuming training times and low-speed real time predictions. Even if these algorithms have more success metrics like accuracy, they might not be applicable for industrial usage because of such drawbacks. Since the AI models are calling during the communication with customers, they should have characteristics of high speed response time that never interfere with interaction between customer and application. In this project; to address this problem and to make our response time performance better, the potential results are pre-calculated periodically and then stored on relational databases which are designed for high performance queries. At the moment that customer needs real time service recommendation predictions, the only thing is retrieving data by basic SQL queries. As mentioned before, if any AI related model is going to be used for an industrial application, there exists two important factors that should be fulfilled: "Acceptable performance in terms of business metrics" and "Minimum real-time prediction latency". Our model, "Conditional Probability Based Recommendation Engine", is able to fulfill both of these requirements which makes it an applicable model for industrial applications.

References

Adomavicius, G. ve Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*(6), 734-749. doi:10.1109/TKDE.2005.99

Ahmed, A., Kanagal, B., Pandey, S., Josifovski, V., Pueyo, L. G. ve Yuan, J. (2013). Latent factor models with additive and hierarchically-smoothed user preferences. *Proceedings of the sixth ACM international conference on Web search and data mining - WSDM '13* içinde (s. 385). the sixth ACM international conference, sunulmuş bildiri, Rome, Italy: ACM Press. doi:10.1145/2433396.2433445

Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, *12*(4), 331-370. doi:10.1023/A:1021240730564

Herlocker, J. L., Konstan, J. A., Terveen, L. G. ve Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, *22*(1), 5-53. doi:10.1145/963770.963772

Liu, J., Dolan, P. ve Pedersen, E. R. (2010). Personalized news recommendation based on click behavior. *Proceedings of the 15th international conference on Intelligent user interfaces - IUI '10* içinde (s. 31). the 15th international conference, sunulmuş bildiri, Hong Kong, China: ACM Press. doi:10.1145/1719970.1719976

Portugal, I., Alencar, P. ve Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, *97*, 205-227. doi:10.1016/j.eswa.2017.12.020

Rodriguez, M., Posse, C. ve Zhang, E. (2012). Multiple objective optimization in recommender systems. *Proceedings of the sixth ACM conference on Recommender systems - RecSys '12* içinde (s. 11). the sixth ACM conference, sunulmuş bildiri, Dublin, Ireland: ACM Press. doi:10.1145/2365952.2365961

Steck, H. (2013). Evaluation of recommendations: rating-prediction and ranking. *Proceedings of the 7th ACM conference on Recommender systems* içinde (ss. 213-220). RecSys '13: Seventh ACM Conference on Recommender Systems, sunulmuş bildiri, Hong Kong China: ACM. doi:10.1145/2507157.2507160

Van Meteren, R. ve Van Someren, M. (2000). Using content-based filtering for recommendation. *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, *30*, 47-56. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.25.5743 adresinden erişildi.

Zhang, C. ve Zhang, S. (2002). *Association rule mining: models and algorithms*. Lecture notes in computer science. Berlin New York: Springer.

Zhang, S., Yao, L., Sun, A. ve Tay, Y. (2019). Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Computing Surveys*, *52*(1), 1-38. doi:10.1145/3285029