

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departmentul Inginerie Software și Automatică

Admis la susținere
Șef departament: Fiodorov I. dr., conf.univ.

„___” _____ 2022

**Crearea unui model pentru predicția pe baza seriilor
de timp a numărului maxim de agenți concurenți.**

Teză de master

Student: _____ **Fiștic, Cristofor, gr. IS-211M**

Coordonator: _____ **Beșliu, Corina, lect. univ., dr.**

Consultant: _____ **Catruc Mariana, lect. univ.**

Chișinău, 2023

Rezumat

Teza cu numele Crearea unui model pentru predicția pe baza seriilor de timp a numărului maxim de agenți concurenți prezentată de către studentul Fiștic Cristofor în calitate de proiect de master a fost elaborată la Universitatea Tehnică a Moldovei. Această este scrisă în limba engleză și conține 42 pagini, 2 tabele, 24 figuri. Lucrarea este formată din introducere, trei capitole, concluzie și bibliografie.

Această teza este realizată în parteneriat cu o companie care oferă soluții de management a ajutorul clienților. Prin această aplicație multe companii oferă suport clienților săi sau este folosită pentru managementul intern a companiei. Un rol important în această aplicație i se atribuie persoanelor care răspund la mesajele scrise de utilizatori și încearcă să-i ajute. Aceste persoane se numesc agent. Numărul de agenți care îi poate avea o companie simultan online în aplicație depinde de tipul de subscripție pe care o are compania, astfel diferite pachete de subscripție oferă un număr diferit de agenți care pot lucra simultan. Problema cu care se confruntă companiile care folosesc această aplicație și clienți lor constă în faptul că uneori numărul de utilizatori care au nevoie să interacționeze cu agenți este mult mai mare decât a fost planuit astfel mărirea numărului de agenți ar ajuta însă el este limitat de subscripția care o are compania.

Pentru a soluționa această problemă, lucrarea dată încearcă să găsească un model de predicție pe baza seriilor de timp care va permite să fie prezis din timp numărul de agenți de care va avea nevoie compania astfel dacă acest număr va fi mai mare decât cel care este în subscripția pe care o are compania acum atunci îi va fi propus un pachet cu un număr mai mare de agenți concurenți. Acest lucru va permite companiilor care se folosesc de această aplicație software să satisfacă cerințele utilizatorilor proprii și să poată mai bine gestiona resursele sale și subscripția pe care o au la moment. Pentru acest lucru au fost analizate datele care le are aplicația pentru a alege datele care vor ajuta cel mai mult la crearea unui model. Au fost încercate câteva modele de predicție pe baza seriilor de timp cum ar fi Prophet sau LSTM. Pentru a putea antrena aceste modele și verifica, a fost creat un set de date din cele inițiale după mai multe procese de extragere, filtrare și curățare. După evaluarea tuturor modelelor a fost ales modelul cu cel mai bun scor, aceasta este modelul Prophet. Următoarea etapă a constant în îmbunătățirea acestui model prin încercarea diferitor hiperparametri astfel încercându-se de a se obține un scor cât mai bun.

Abstract

The thesis named Creating a model for time series forecasting the maximum number of concurrent agents, presented by student Fiștic Cristofor as a Master's project, was developed at the Technical University of Moldova. It is written in English and contains 42 pages, 2 tables, and 24 figures. The project consists of an introduction, three chapters, a conclusion, and references.

This thesis was realized in partnership with a company that offers a help-desk software solution to its customers. This software helps many companies provide support to their customers, or it is used for internal company management. An essential role in this application is given to people who respond to messages written by users and try to help them, these people are called agents. The number of agents a customer can have working simultaneously in the application depends on the customer's subscription type. The more expensive the subscription package, the greater the number of agents a customer can have using them simultaneously. The problem faced by companies using this app is that sometimes the number of agents needing to use the tool is higher than the customer planned. The customer would ideally like to increase the number of agents allowed to work simultaneously instantly, but increasing the limit implies upgrading their existing contracts, which can take up to a whole week.

In order to solve this problem, the paper tries to find a prediction model based on time series that will allow predicting the number of concurrent agents that the company will need in the next month. If the customer is forecasted to exceed the limit in its subscription, the service provider can propose an upgrade ahead of time. This would allow the service provider to timely satisfy their customers' requirements, increase their satisfaction with the product and ultimately raise profitability. To achieve this, usage data from the application was analyzed, and relevant data for modeling purposes was selected. The initial data had to be filtered and cleaned to create the dataset that would be used to train and test the models. Several time series models, such as Prophet and LSTM, were fitted. After evaluating the models, the one with the best prediction accuracy score was chosen. This proved to be the Prophet model. The next step was tuning the model's hyperparameters to achieve more accurate forecasts.

TABLE OF CONTENTS

INTRODUCTION.....	8
1 DOMAIN ANALYSIS AND ARGUMENTATION.....	10
1.1 Problem Analysis.....	10
1.2 Project Objectives	11
1.3 Time Series.....	12
1.3 Artificial intelligence models.....	17
1.3.1 ARIMA.....	17
1.3.2 Prophet.....	18
1.3.3 Silverkite	18
1.3.4 LSTM	19
1.3.5 NeuralProphet.....	19
1.3.6 DeepAR	20
2 EXPLORATORY DATA ANALYSIS.....	22
2.1 Data description	22
2.2 Data analysis and data cleaning.....	24
2.3 Dataset preparation	30
3 Modeling and Model selection	31
3.1 Naive method Model	32
3.2 Prophet model	32
3.3 Silverkite	33
3.4 LSTM	34
3.5 Model tuning.....	35
CONCLUSIONS	37
REFERENCES.....	38
Annexes	40
Annex 1 Scrip for parsing ConcurrentUsersDetails variable	40
Annex 2 Script for checking the percentage of missing data	41
Annex 3 Script for checking the percentaje of zeros in the data	42

INTRODUCTION

Help-desk applications are used to help companies process messages from internal or external customers. These applications help companies by making communication with customers more optimal and efficient. The application that will be discussed in this paper is also a help-desk software solution, but the difference from other applications is that it is based on a ticketing system. These tickets can be automated to optimize workflow and subsequently, increase the business's performance. Other things offered by the application that will improve business performance are business processes and reporting with data analysis.

An essential role in the use of this application by a company is played by the people who answer the tickets and manage them, these people are called agents. The number of agents that can work simultaneously in this application depends on which subscription package the company has. Each package has a different number of concurrent agents a company can have. In this way, each company can choose a packet as close as possible to its requirements.

The problem faced by companies using this app is that sometimes, for example when the company goes through an expansion cycle, the number of agents who need to use the tool is higher than the customer planned. The customer would ideally like to increase the number of agents allowed to work simultaneously instantly, but increasing the limit implies upgrading their existing contracts, which can take up to a whole week.

The solution to this problem is to create a prediction model that will forecast the number of concurrent agents a system will have in the future. In this way it will be possible to compare the predicted number with the maximum number of agents from the subscription package. If the predicted value is higher, the service provider can propose a subscription package that will satisfy the number of concurrent agents that the system will need in the near future. This will allow companies to increase clients' satisfaction when interacting with the services offered by the company and to manage their resources efficiently. Another advantage is that companies will not have to pay more for a better subscription tier if there is no need for it. Cases, where a company will reach the limit of concurrent agents and cannot respond quickly to its clients, will be much rarer because of the prediction model. All companies will be warned in advance not to reach the limit of concurrent agents and to switch to a better packet in order to cope with the large number of requests it will have from clients of the system. This will attract other companies to use this application and try this application to be used in more business areas.

The first thing that will be done to help create the prediction model will be an exploratory analysis of data collected by the application. This will allow us to understand what data should be used for the model and what type of models will be needed to be able to interpret this data and predict the number of concurrent agents. In order to create a good model, a research on the best forecasting algorithms for this type of prediction will be conducted. After this, the dataset will be extracted from the application database and used

to train and test the model. Before that, the dataset will be cleaned and filtered to help the model better understand the data and get a better prediction. The performance scores for different forecasting algorithms will be compared, and the prediction model with the best result will be selected.

REFERENCES

- 1 Christian Berg, The Complete Guide to Time Series Data, 03.11.2021, [Online, accessed 3-October-2022] Available: <https://www.clarify.io/learn/time-series-data>
- 2 CFI Team, Autocorrelation, 06.05.2022, [Online, accessed 3-October-2022] Available: <https://corporatefinanceinstitute.com/resources/knowledge/other/autocorrelation/>
- 3 Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., & Shin, Y.(1992) Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root ? , Journal of Econometrics, Volume 54, p 159-178, ISSN 0304-4076
- 4 Rob J Hyndman and George Athanasopoulos, „Forecasting: Principles and Practice second edition”, april 2018, p 58, ISBN: 0987507117
- 5 Marco Peixeiro, „Time Series Forecasting in Python”, 4.10.2022, p 410, ISBN: 161729988X
- 6 S.L. Ho, M. Xie, (1998) The use of ARIMA models for reliability forecasting and analysis, Computers & Industrial Engineering, Volume 35, p 213-216, ISSN 0360-8352
- 7 Robert Nau, Introduction to ARIMA: nonseasonal models, 05.06.2016. [Online, accessed 8-October-2022] Available: <https://people.duke.edu/~rnau/411arim.htm>
- 8 Facebook, About Prophet, [Online, accessed 8-October-2022], Available: <https://facebook.github.io/prophet/>
- 9 Reza Hosseini, Greykite: A flexible, intuitive, and fast forecasting library, 13.05.2021, [Online, accessed 8- October-2022] Available: <https://engineering.linkedin.com/blog/2021/greykite--a-flexible--intuitive--and-fast-forecasting-library>
- 10 Aileen Nielsen, „Practical Time Series Analysis: Prediction with Statistics and Machine Learning”, October 2019, p 329, ISBN: 978-1-492-04165-8
- 11 Rui Zhang, Hejia Song, Qiulan Chen, Yu Wang, Songwang Wang , Yonghong Li, (14.01.2022), Comparison of ARIMA and LSTM for prediction of hemorrhagic fever at different time scales in China, PLoS ONE 17(1): e0262009
- 12 Facebook, Overview of the NeuralProphet Model, [Online, accessed 8-October-2022], Available: <https://neuralprophet.com/model-overview.html>
- 13 Nikolay Laptev, Oskar Triebe, Hansika Hewamalage, NeuralProphet: The neural evolution of Meta’s Prophet, 30.11.2021, [Online, accessed 8-October- 2022] Available: <https://ai.facebook.com/blog/neuralprophet-the-neural-evolution-of-facebooks-prophet/>
- 14 Amazon, DeepAR Forecasting Algorithm, [Online, accessed 8-October-2022] Available: <https://docs.aws.amazon.com/sagemaker/latest/dg/deepar.html>
- 15 IBM, Exploratory Data Analysis, 25.09.2020, [Online, accessed 12-November-2022] Availability: <https://www.ibm.com/cloud/learn/exploratory-data-analysis>
- 16 Gerhard Svolba, „ Data Quality for Analytics Using SAS ”, 15.04.2012, p 258, ISBN: 978-1607646204

- 17 Stephen Allwright, What is a good MAPE score?, 15.08.2022, [Online, accessed 18-December-2022]
Availability: <https://stephenallwright.com/good-mape-score/>
- 18 Mohamad, MAAPE - Mean Arctangent Absolute Percentage Error, 09.03.2017, [Online, accessed 18-December-2022] Availability: <https://support.numxl.com/hc/en-us/articles/115001223463-MAAPE-Mean-Arctangent-Absolute-Percentage-Error>