Web Service Composition using Markov Decision Process and Long Short Term Memory

Subbulakshmi S[1],Hemant P Hareesh[2], Rahul R[3]
[1, 2, 3]Amrita VishwaVidyapeetham, Amritapuri, India.
[1] subbulakshmis@am.amrita.edu, [2] hemantphareesh@gmail.com,
[3] rahulvr973@gmail.com

Abstract

Software and Systems are abstracted as Web Services in the Service-Oriented based design, which may then be used by other systems. Using the approach of service composition, simple services that already exist can be combined to produce sophisticated solutions. Massive web services with the same functionality start to emerge as web service technology develops. Numerous groups maintain these services, and the level of service varies. As a result, a crucial problem in service composition research is how to pick the optimal service to guarantee that the entire system gives the best overall QOS (Quality Of Service). Additionally, Given the complex nature and fluctuation of the network environment, QOS may alter over time. Consequently, it is difficult to comprehend how to dynamically modify the composition system to react to shifting settings while keeping the calibre of the composing service. To get over the current issues, we propose a method for composing services that is based on QOS prediction and reinforcement learning. After employing a Long Short-Term Memory (LSTM)which is a Recurrent Neural Network which is used to predict QOS, we use reinforcement learning to select dynamic services. Our approach is easily adaptable to a dynamic network setting. We put our plan through a number of tests to make sure it works.

Index Terms Deep Learning, Long Short Term Memory, Recurrent Neural Networks, Quality of Services, Web Service Composition

DOINumber:10.14704/nq.2022.20.8.NQ44721 **NeuroQuantology2022;20(8):6952-6963**

INTRODUCTION

A cutting-edge [1] way for combining several businesssolutions to carry out sophisticated business logic is calledweb service composition. In this fast-paced digital world with a variety of user types, intelligent service composition based on contextual information is extremely important to suit the needs of individual consumers.

Web Services, or Web-based Services, are made available to users by a provider. Web Service Composition (WSC) [2] is the process of combining services to produce single service that meets the needs of the user. Basic services that already exist are merged using the service composition methodology to create complex solutions.

Rapid service expansion makes it challenging to identify the level of quality and effectiveness of those services. Different approaches and procedures are required to solve this issue, as well as finding the finest user-friendly solution. People's daily life now depend more and more on the internet. In a variety of application disciplines, network technology has had a substantial impact on software development. Software, Systems, Computer Resources, storage resources, and other resources are being made available as Web services by an increasing number of companies and organizations for usage by others.

The ability [3] to meet requirements like latency, reliability, bandwidth, etc. is necessary to fulfil a service-level agreement (SLA) between an application provider and an end user. There-fore, learning how to efficiently integrate numerous services has grown to be a significant research area. The creation of contemporary software products makes the study of service composition, which focuses primarily on how to develop a system capable of matching the complex criteria by merging services that currently exist, particularly pertinent.

Additionally, services with the same functions are offered by various companies' service providers with differing Service-Quality (QoS). The development of contemporary software products greatly benefits from the study of service composition, which focuses mostly on how to create a system capable of achieving complex criteria by merging services that already exist.

Different QoS parameters [4] are evaluated in order to assess the ratings. However, missing values are frequently a serious problem in datasets of QoS variables in the real world since they lead to imprecise predictions of the QoS rating of a particular web service.

Using QoS criteria, the right service for each abstractservice in the workflow is chosen throughout the implementation of service composition, and ultimately, an ideal service composition result may be generated that might significantly meet user needs.

In order to be picked, candidate services must have similar functions but different QoS. Any service's Service-Quality (QoS) will alter due to the dynamic changes in both the network environment and the service itself.

Suggested method makes use of the Markov DecisionProcess' reward concept, where the highest overall reward is chosen as the best result. Here, the weighted quality of service for each service is calculated using the Markov decision process and use the highest possible weighted quality of service as a reward, so we choose the combination with the highest reward. Then, in order to incorporate dynamic nature into our system, we use on-site data that provides information about network issues and other characteristics, in the event that a network issue exists, the reward values are decreased by some percentage. Long Short Term Memory (LSTM) is a type of neural network that makes predictions about future Service Quality (QoS) using data from current services. Here, we employ LSTM to forecast the transaction count from historical data and include it with services to determine the optimal set of services.

RELATED WORKS

Due to the dynamic and complicated complexity of thenetwork environment, the quality of service may change often, necessitating a service composition technique that can recognize and adjust the system automatically to environmental changes. Even though this topic has *received a lot of attention, many different tactics have also been proposed.*

The authors [5] use a machine learning technique to determine the service composition. Using this tactic, a lower-quality service will be replaced by a higher-quality one when it degrades. However, because it only considers changing one service, this strategy falls short of having total control over the composite system. It could seem as though overall service quality is declining.

The proposed system [6] will use sensors to implementand forecast traffic congestion. This manual implementation is quick but time-consuming because there cannot be human involvement at all places around Bangalore. Additionally, it does not account for current traffic congestion on the specific road. To maximise the overall advantages, the fuzzy controller adjusts the parameters. The goal is to lessen the congestionfactor by selecting policies using the constraint MDP and Markov Decision Process. The length of time it takes for the signal to become green and for the vehicles to remain in line for the required amount of time are both critical factors in constraint MDP. In order to estimate traffic congestion on a busy road, this study advises combining the approaches of policy gradient and constraint MDP. Compared to Q learning and other contemporary algorithms, it offers better results. Future construction on the widely used route is possible due to the numerous lanes and intersections. Their method can be used in all cities, nations, and types of automobiles.

An Optimization of an Ant Colony [7] was suggested toprovide worldwide Optimization of Service Composition forQuality of Service. The local optimal problem is hard to solvewith this approach.

Utilizing the eagle technique [8] and the whale optimizationmethod, the cloud service composition was optimised whiletaking the balance between exploration and exploitation intoaccount. Although this approach avoids the local optimum, itdoes not take into consideration how the quality of service haschanged over time, which has an effect on adaptation.

In recent years, the discipline of information technologyhas focused attention on machine learning as a promisingarea for further study. Some researchers have attempted touse reinforcement learning for service composition.

The Markov Decision Process [9] model and Reinforcement Learning are used by the authors to create an adaptive service composition.The composition technique for Three-Layer Trust-EnabledServices [10] was suggested. The authors used a fuzzy com-prehensive evaluation method to create an integrated trustmanagement model. Utilizing strategy, it may be possible todetermine user preferences and improve service composition.

Although the earlier reinforcement learning systems areadaptive, they ignore a few particular issues with service com-position. In a network environment, services are often deployedon a regular basis. The quality of service is subject to ongoingmodifications due to the dynamic network environment andpotential for service performance fluctuations. As a result, itis necessary to accurately estimate the altered service quality toenhance the results of the Service-Composition. Despite having few self-adaptive qualities, the traditional reinforcementlearning approach is still unable to accurately forecast servicequality, leading to unsatisfactory composition of service resultsin a highly dynamic context. The environment has an impacton the way services are composed, and the level of servicequality changes through time. As a result, it is necessary toforesee changes in Quality of service in some way. Quality ofservice prediction can help with this. The network environmentat the time determines changes in service quality, not thelocation or health of the service caller. A time series of datacan be used to express how the quality of a service evolvesover time.For multi-step forward prediction [11] , which can takeinto account a greater variety of data patterns, RNN has beenused. In general, when it comes to fitting complex functions, deep learning-based algorithms outperform traditional statistical models. They can therefore offer better prediction performance when dealing with challenging Time-Series.

The Service Composition technique [12] depends onQuality-of-Service-Prediction and Reinforcement Learning, isthe main emphasis of the work given in paper. A RecurrentNeural Network (RNN) was specially used to make the predictions. In this article, the authors developed a RNN and employed the Time-Series prediction method. Using DWSCMDPmodel and a Q-learning algorithm, the recommended method'sefficacy was illustrated.

This research focuses on a QoS-based Service Compositiontechnique [13] . This paper proposes a revolutionary technique for Composition of large-scale and dynamic services using Deep Reinforcement Learning. POMDP-WSC model that has been suggested is more accurate in terms of service composition. The authors suggested the POMDP-WSC model-based OSONWSC framework. The trial demonstrates the approaches' adaptability and scalability.

A time-based learning algorithm [14] is provided in thispaper as a dynamic decision-making tool. In order to resolvethe uncertain planning problem for service composition, theauthors used a partially observable Markov decision method.

After that, they recommended the reinforcement learningapproach to planning in an unpredictable world. The resultsof the experimental research demonstrate that The suggestedtechniques improve service composition's chance of succeeding.

This papers main argument [15] is that great customerservice comes first. (QoS). In this instance, the QoS parametersare optimised using a genetic algorithm. Instead of pickingservices at random for service construction, they use a Cuckoo-based algorithm to find the optimum web service combination. Levy flight is used in conjunction with a cuckoo-based algorithm to deliver the appropriate service composition. In the near future, semantic service selection that is contextually focused may be used to enhance service composition.

The vector-valued MDP technique [16] was to find thebest QoS-aware service composition, the authors introduced.It can with the least amount of user interaction, achieve idealcomposition. Large datasets demonstrate that the proposedapproach genuinely identifies the best composite servicesin practise. A key technique for automatic communicationbetween several remote applications is Web Services.As thenumber of online services grows, it is essential to pick andpredict the quality of web services for customer requests. Theauthors introduced the Collaborative Filtering (CF) methods.These methods are frequently used for web service predictionbecause of the history of online services and the focus onprediction values..

Web Services [17] with QoS are one of the major problemsin web service research. Our primary goal is to propose a strategy for getting outstanding composite services in a serviceenvironment that is always evolving QOS. For Web servicedevelopment and selection, a Q-learning method incorporatingReinforcement Learning (RL) is offered. In intelligent composition, corporate rules, commercial directives, and customizeduser experiences are represented using an ontology-based model that is machine intelligible. In this study, domain ontologies, which also cover ontology modification, the gathering of contextual intelligence, the choice of services, and dynamiccomposition, are used to illustrate how healthcare services arecomposed.

For a specific reinforcement learning [18] problem statement, this work aims to construct a model using a constraint Markov decision-making method that forecasts traffic congestion at intersections and lanes on crowded roads. This study combines a policy gradient technique with a limitation MDP approach to determine traffic congestion on a major road. It performs better in terms of results than Q-learning and other contemporary techniques.

After the development [19] of Artificial Intelligence (AI),Machine Learning (ML), and Deep Learning (DL), technologies advanced at an alarming rate. Using the aforementionedpairs, several procedures are in place to help distinguish distinct suspicious actions from live tracking of film. The advantage of the suggested approach is that it stops crime before it even starts. CCTV footage is being monitored and analysed in real time. An instruction to take action is issued to the appropriate authorities if the analysis's findings indicate that an undesirable event is likely to occur. As a result, it is possible to stop this.

Recurrent Neural Network [20] (RNN) produced musictypically lacks generic organization . The failure is thought to becaused by RNNs' inability to track distant spatial and temporalevents indicating a universal musical structure. In areas whereprevious RNNs have failed, such as time and counting and CSL learning, Long Short-Term Memory (LSTM) has shown success. The current study suggests that LSTM is a great method for teaching music composition.

Automated and dynamic composition [21] is the end goal.As web services gain in popularity, a problem about QoS-oriented composition arises regarding how to effectively choose the most appropriate service from a large number of services. Function and service quality are continually taken into consideration. This work offers preference logic to grasp the same function-oriented composition through reinforcementlearning before seeking a QoS optimization solution that results in a comprehensive composition solution.

METHODOLOGY

A. Reinforcement Learning (RL)

RL [22] is the most fundamental ML techniques. Specializes in changing the behavior of the agent to achieve specific results in an unfamiliar setting. The exploratory behavior of people or other species in an unfamiliar area is modelled by reinforcement learning. The agent continuously interacts with its surroundings in the unknowable world, gets feedback, and then adjusts its behaviour, as shown in Fig. 1. The ultimate goal is to take advantage of the surroundings as much as possible.

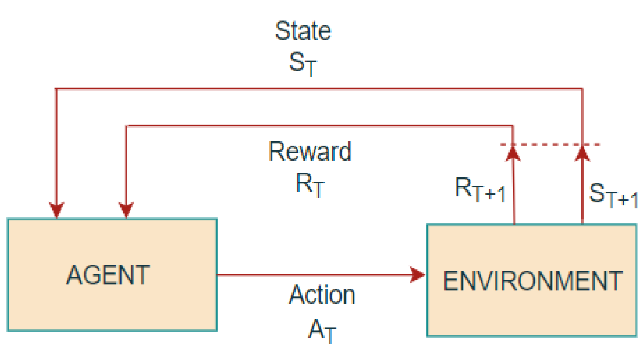

Fig. 1. Reinforcement Learning Framework

An Action is carried out by the agent. Following execution, the environment is gathered for the reward value before theAgent moves on to the following state. Based on the reward,the agent will adjust its choice of actions, eventually comingup with a system for choosing actions that maximises thereward value. The input consists of a list of potential states,actions that can be taken in each state, reward values thatcan be obtained, and additional states that can be reached
by performing actions. The ultimate result is a method forchoosing actions based on criteria that should produce themaximum possible reward value.

B. Markov Decision Process (MDP)

According to the Markov Decision Process (MDP) [MDP =(S, A, P, R)], where S is State, A is Actions, A(S) represents the set of actions that can be executed in state S, and P is probability distribution function, when an action A is invoked, the world changes from its current state to a succeeding state. R stands for the instant reward function. When an Action A is chosen and the current State is S, the environment will immediately reward us with r = R(S, A) when the action has been completed. An Action selection strategy is an output of Reinforcement Learning.

C. Long Short Term Memory

There are various [23] methods for making predictions aboutthe future based on present data, but applying machine learning is seen to be the best method due to its accuracy, training pattern, and testing capabilities. Among the several methods used in machine learning for forecasting, Long Short-Term Memory (LSTM) based forecasting is the current technique for time series forecasting.

D. Service Composition

In a system that is Service-Oriented, value-added services are created by combining simple services to fulfil the complex needs of users. Thus, service computing now includes service composition. Numerous Web Services have emerged as a result of the Web Service technology's quick development, all of which share the same functionality yet differ in a number of non-functional ways (such as QoS). It is extremely difficult to choose the best services from a vast pool of candidates in order to establish an ideal composition because customer requirements are becoming more sophisticated and there are so many services available. Web services that may be accessed over computer networks are by nature dynamic, and the context in which services are composed is equally complicated and unstable. Solutions for service composition must therefore be flexible enough to change with the environment. To address these crucial concerns, we offer a novel reinforcement learning-based service composition technique. Recurrent neural networks are used to improve reinforcement learning because they may anticipate the objective function and boost expression and generalization efficiency. The experimental results show distinct advantages in composition outputs and efficiency for service composition, demonstrating the viability, efficiency, scalability, and adaptability of our methodologies.

Service composition, which combines pre-existing web services to create new systems that meet the intricate needs ofusers, is one of the most significant reuse approaches. It ismore crucial than ever to choose the best individuals froma large pool of applicants in order to produce a high-qualitycomposite service in light of the introduction of functionallyidentical services. However, the QOS is not constant in a dynamic network environment, and the quality may fluctuateover time. To adapt to environmental changes, the service composition method must be dynamically changed.

E. Solution Approach

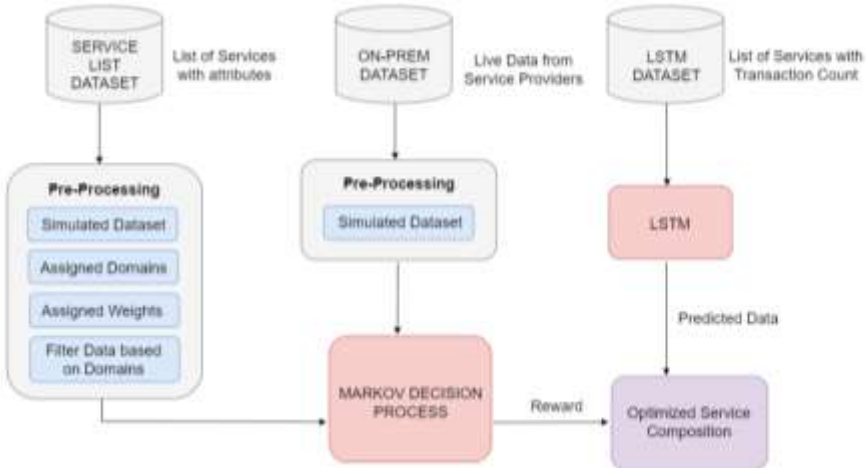

Fig. 2 . Architecture Diagram

We employ a service composition approach based on qualityof Service (QoS) prediction and reinforcement learning. In order to maximise the composite service's overall QoS value,we apply reinforcement learning to select suitable candidateservices.We offer a QoS prediction technique based on LSTMsince QoS is dynamic and complex in the network environmentand changes continually. In order to address the problem of service composition, which can improve the quality of the composite service in a dynamic context, we combine QoS prediction with reinforcement learning. Along with the number of candidate services and the complexity of the service composition process, there are more possible outcomes for service composition. Service composition conducts a thorough analysis of a vast candidate space to choose the optimum option that increases reward value.

1) STEP 1- Preparation of Datasets:: Here we have ThreeData sets, SERVICE LIST DATASET (Fig 3) is having datarelated to services and their attributes, ON-PREM DATASET(Fig 4) is Live Data From Services and LSTM DATASET(Fig 5) is having service names with transaction counts of each services.

Response Time	RT_Weight	Availability	AV_Weight	Throughput	TH_Weight	Successability	Succ_Weight	Reliability	RE_Weight	Service Name	Domain	id
136.07	2	100	2	12.4	1	85	1	82.3	1	T1	Transaction	5
175	2	100	1	20.5	1	33	1	70	1	T2	Transaction	58
105.2	1	79	2	20.5	1	78	1	70.2	2	R1	Restaurant	41
50	2	84	1	24.6	1	89	1	85.6	1	W1	Weather	8
165	1	98	1	27.3	1	67	1	82.1	2	W2	Weather	4
117	1	56	1	9	1	48	2	66.1	1	R2	Restaurant	46
109.09	1	64	2	9.3	1	67	1	92.2	1	H1	Hotel	62
463	1	56	2	16.4	1	90	1	82.3	1	R3	Restaurant	23
108.2	1	84	2	23.4	1	80	1	77.1	2	W3	Weather	48
78	2	83	2	11.3	1	90	2	79.2	2	R4	Restaurant	36
144	2	98	2	12.4	1	53	2	86.4	2	W4	Weather	2
118.5	1	64	2	14.6	1	88	1	87.8	2	H2	Hotel	6
135	2	98	2	2.8	1	80	2	72.5	2	R5	Restaurant	1
130	1	61	2	13.2	1	50	2	74.6	2	H3	Hotel	19
256.5	1	61	1	29.5	1	80	2	87.6	2	T3	Transaction	11

Fig. 3. Service List Dataset

6958

Service Name	Network Issues	Trust Service providers	Server Issues
T34	0	1	1
T1	1	1	1
W1	1	1	1
H1	1	1	1
R1	1	1	1
T44	1	1	1
T55	1	1	1
T56	1	1	1

Fig. 4 . On-Prem Dataset

id	Time Stamp	max	sum	transactions_count
5	4/11/2018 18:00	1339	22860	3119
58	4/14/2018 17:00	87	2109	1475
41	4/14/2018 0:00	19	480	268
8	4/11/2018 21:00	459	4744	1303
4	4/11/2018 17:00	431	16126	3427
46	4/14/2018 5:00	80	1265	685
62	4/14/2018 21:00	36	1479	585
23	4/13/2018 6:00	15	108	34
48	4/14/2018 7:00	790	3657	1025
36	4/13/2018 19:00	360	1998	1079

Fig. 5. LSTM Dataset

2) STEP 2: Pre-Processing Datasets: SERVICE LISTDATASET is having service names with Attributes: Response Time, Availability, Throughput, Successability, Reliability. Here we assign Names of each Services based on Domain. Then We filter the data based on four domains: Transaction, Restaurant, Weather and Hotel. Next step is to assign weights for each Services, Weights are assigned based on priority as 1 or 2. ON-PREM DATASET consists of service names with Network issues, Trust Service Providers, Server Issues. If network is down network issue will be 1 otherwise 0 same way for others.LSTM DATASET Consists of service names with Time Stamp and Transactions Count.

3) STEP 3- Markov Decision Process: : Use of Markov decision Process Concept to find out Reward which is weightedQuality of service values, we use SERVICE LIST DATASETvalues, Response Time is multiplied with Response TimeWeight, same way all the attributes are multiplied with theirWeight, and then we sum up all the values Which will givethe Weighted Quality of Service values, now each serviceshave Weighted Quality of Service values. From ON-PREMDATASET, we check if the currently service have networkissues, if network issue is there, We reduce the WeightedQuality of Service values by reducing the throughput by 10percent and Increase the response time by 10 percent, if serverissues is there, we reduce the throughput by 10 Percent, andwe also check if service is trust worthy by checking thevalue in Trust service provider columns .Is not trustworthy thenwe reduce the Reliability value by 20 percent, we call theabove discount factor. Then we display the Weighted Qualityof Service value, which

will be called as Reward. Now wefind out the maximum of the reward values for each domainwe take the combination of the value to get the final Result.We will then have the most recent Combination of Services.The total reward value should be as high as it can be with asatisfactory service composition result.

4) STEP 4- Long Short Term Memory: : We use deeplearning technique called LSTM, which is Long Short TermMemory. This Method will forecast the Future values basedon past data. Here we take LSTM DATASET, consisting oftimestamp of services and Number of Transaction occuringat that particular time. With the help of LSTM We train thedataset and forecast the future values of services based on thedataset provided which will be considered and past data. Weuse the predicted data we get from before, the combinations toget Final gives Combination of Services.

RESULT

The tests are performed using a 64-bit version of Windows 11, which has an i5 CPU, 16 GB of Memory. The results of

Service	Weighted QOS	Domain
T1	649.84	Transaction
T2	573.50	Transaction
R1	607.30	Restaurant
W1	383.20	Weather
W2	521.50	Weather

Fig. 6. Weighted Quality of Service

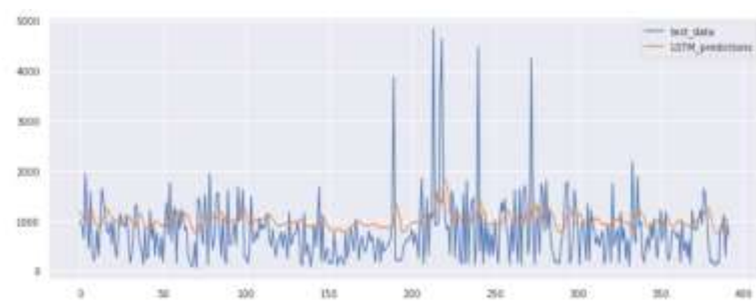

Fig.7. LSTM Learning Graph

Combinations	Reward
W10 H7 R18 T19	6774.32

Fig. 8. Combination of Services

the weighted quality of service calculation we performed usingthe Markov Decision Process are shown in Fig. 6. WeightedQOS is the reward we received. We get max of each domainfrom this reward. To obtain the outcome, we integrate theServices. Then, using an LSTM model, we predict values forthe dataset, such as the transaction count. With the help ofthe data in Fig. 7, which shows the general flow of learningand expected data, we can infer the combination of servicesshown in Fig. 8

CONCLUSION

eISSN1303-5150

Service composition, which involves merging Pre-ExistingWeb Services to create new systems that suit the intricaterequirements of users, is one of the most significant strategies for software reuse. In order to create a high-quality complementary service in light of the rise in functionally identical services, it is required to choose relevant services from a vast candidate pool. The QoS is inconsistent in a dynamic network environment, and the quality may change over time. Therefore, in order to dynamically adjust to environmental changes, the service composition method must be used. The suggested method takes into account the QoS's dynamicchanging components.

To find out Quality of services and to use the Combination of services we use Markov decision Process Concept which will give current Combination which can be used based on dynamic nature of services and with the help of LSTM We improve the Result by predicting the Future of the data.

In our future study, we intend to investigate deep reinforcement learning to increase efficiency of current method and to investigate different methods that can be used in Reinforcement Learning for this purpose, also need to improve and in cooperate dynamic nature of the Services.

References

[1] S. Subbulakshmi, Aparna Krishnan, and R. Sreereshmi. "Contextual Aware Dynamic Healthcare Service Composition Based on Semantic Web Ontology". In: 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT). Vol. 1. 2019, pp. 1474–1479. DOI: 10 . 1109 / ICICICT46008.2019.8993303.

[2] Hongbing Wang et al. "Integrating recurrent neural networks and reinforcement learning for dynamic service composition". In: Future Generation Computer Systems 107 (2020), pp. 551–563. ISSN: 0167-739X. DOI: https:/ / doi .org / 10 . 1016 / j .future .2020 .02 . 030. URL:https : / / www . sciencedirect .com / science / article / pii /S0167739X19311240.

[3] S. Subbulakshmi, Anvy Elsa Saji, and Geethu Chandran. "Methodologies for selection of Quality Web Services to Develop Efficient Web Service Composition".In:2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC). 2020, pp. 238–244. DOI: 10 . 1109 / ICCMC48092 .2020 . ICCMC-00045.

[4] S Subbulakshmi et al. "Optimized QoS Prediction ofWeb Service using Genetic Algorithm and Multiple QoS Aspects". In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 2018, pp. 922–927. DOI: 10.1109/ICACCI. 2018.8554376.

[5] Yuhong Yan, Pascal Poizat, and Ludeng Zhao. "Repair vs. Recomposition for Broken Service Compositions".In: Service-Oriented Computing. Ed. by Paul P. Maglio et al. Berlin, Heidelberg: Springer Berlin Heidelberg,2010, pp. 152–166. ISBN: 978-3-642-17358-5.

[6] Subbulakshmi .S et al. "Optimized Web Service Composition Using Evolutionary Computation Techniques"In: Jan. 2021, pp. 457–470. ISBN: 978-981-15-9508-0. DOI: 10.1007/978-981-15-9509-7 38.

[7] Olfa Hammas, Saloua Ben Yahia, and Samir BenAhmed. "Adaptive web

service composition insuringglobal QoS optimization". In: 2015 International Symposium on Networks, Computers and Communications (ISNCC). 2015, pp. 1–6. DOI: 10.1109/ISNCC.2015.7238593.

[8] Siva Kumar Gavvala et al. "QoS-aware cloud servicecomposition using eagle strategy". In: Future Generation Computer Systems 90 (2019), pp. 273–290. ISSN:0167-739X. DOI: https://doi.org/10.1016/j.future.2018. 07.062. URL :https://www.sciencedirect.com/science/article/pii/S0167739X17320630.

[9] Hongbing Wang et al. "Adaptive Service CompositionBased on Reinforcement Learning". In: vol. 6470. Dec.2010, pp. 92–107. ISBN: 978-3-642-17357-8. DOI: 10.1007/978-3-642-17358-5_7.

[10] Wenjuan Li et al. "A Trust-Based Agent LearningModel for Service Composition in Mobile Cloud Computing Environments". In: IEEE Access 7 (2019),pp. 34207–34226. DOI: 10 . 1109 / ACCESS . 2019 .2904081.

[11] Liu Yunpeng et al. "Multi-step Ahead Time SeriesForecasting for Different Data Patterns Based on LSTM Recurrent Neural Network". In: 2017 14th Web Information Systems and Applications Conference (WISA).2017,pp.305–310.DOI:10.1109/WISA.2017.25

[12] Leslie Pack Kaelbling. "A Situated-Automata Approach to the Design of Embedded Agents". In: SIGART Bull.2.4 (1991),pp. 85–88. DOI: 10.1145/122344.122361.URL: https://doi.org/10.1145/122344.122361.

[13] Hongfan Ye et al. "Web Services Classification Basedon Wide Bi-LSTM Model". In: IEEE Access 7 (2019),pp. 43697–43706. DOI: 10 . 1109 / ACCESS . 2019 .2907546.

[14] Hongbing Wang et al. "Adaptive and large-scale service composition based on deep reinforcement learning". In: Knowledge-Based Systems 180 (2019), pp.75–90.ISSN:0950-7051.DOI: https://doi.org/10.1016/j.knosys.2019. 05.020. URL:https://www.sciencedirect.com/science/article/pii/S0950705119302266.

[15] Pegah Alizadeh et al. "Reinforcement Learning for Interactive QoS-Aware Services Composition". In: IEEESystems Journal 15.1 (2021), pp. 1098–1108.DOI:10.1109/JSYST.2020.2997069

[16] Yu Lei et al. "Web Service Composition Based onReinforcement Learning". In: 2015 IEEE InternationalConference on Web Services. 2015, pp. 731–734. DOI:10.1109/ICWS.2015.103.

[17] Qing Liu, Yulin Sun, and Shilong Zhang. "A ScalableWeb Service Composition Based on a Strategy Reused Reinforcement Learning Approach". In: 2011Eighth Web Information Systems and Applications Conference. 2011, pp. 58–62. DOI: 10.1109/WISA.2011.18.

[18] Douglas Eck and Juergen Schmidhuber. A First Lookat Music Composition Using LSTM Recurrent NeuralNetworks. Tech. rep. 2002.

[19] Muhammad Hasnain et al. "Recurrent Neural Networkfor Web Services Performance Forecasting, Rankingand Regression Testing". In: 2019 Asia-Pacific Signaland Information Processing Association Annual Summit

and Conference (APSIPA ASC). 2019, pp. 96–105. DOI:10.1109/APSIPAASC47483.2019.9023052.

[20]Hongbing Wang, Pingping Tang, and Patrick C. K.Hung. “RLPLA: A Reinforcement Learning Algorithmof Web Service Composition with Preference Consideration”. In: 2008 IEEE Congress on Services Part II(services-2 2008). 2008, pp. 163–170. DOI: 10 .1109 /SERVICES-2.2008.15.

[21]C.V Amrutha, C. Jyotsna, and J. Amudha. “DeepLearning Approach for Suspicious Activity Detectionfrom Surveillance Video”. In: 2020 2nd InternationalConference on Innovative Mechanisms for Industry Applications (ICIMIA). 2020, pp. 335–339. DOI: 10.1109/ICIMIA48430.2020.9074920.

[22]Leslie Pack Kaelbling. “A situated-automata approachto the design of embedded agents”. In: SIGART Bull. 2(1991), pp. 85–88.

[23]Chervu Sowmya, Anu G Kumar, and Sachin S Kumar.“Stacked LSTM Recurrent Neural Network: A DeepLearning Approach for Short Term Wind Speed Forecasting”. In: 2021 International Conference on Intelligent Technologies (CONIT). 2021, pp. 1–7. DOI: 10 .1109/CONIT51480.2021.9498314.