Category	sub categories	challenges	Literature Reveiw Paper	No of challenges	No of papers
Data	Datsetset creation	1) Data error and quality issue slow and manual error prone [14,18] 2) Lack of best practice and documentation when creating,and accountability and transparency [15]3) dataset can be advesial and posioned [16,1]	14,15,16,8,1	3	5
Data	Dataset Preprocessing	1)evolving and untrusted source,Adversial and Bias data source, which is avoided[1,2] , 2) Data errors like missing data and transformation , imabalance, splitting strategy [5,6,9] , 3) Require large scale data processing tool, agnostic frmework and language. [7]	1,2,5,6,7,9,10	3	7
Data	Data validation	1) Rapidly evolving data very difficult to data and online training [11,2] 2) Building data validation pipeline is challenging and demanding and less flexible[12]	2,11,12	2	3
Data	Data management	1) In ML iterative process, demands provence tracking, intermediating result storage for reproducibility of the workflow and collabration , more tools needed [4,13]. 2) Large scale data managment tools not available, data collection, managing and facilating manipulation; labeling , serialise, diificult to build data managment pipeline[3]	3,4,13	2	3
Category	**sub categories**	**challenges**	**Literature Reveiw Paper**	**No of challenges**	**No of papers**
Model	HPO	1) HPO is slow and time consuming The computational efforts required for automating involves some efforts, such as engineering solutions for job scheduling and keeping track of the parameters and results , [20,21] 2) wrong choice in these parameters can affect either the quality of the resulting model or the efficiency and rate convergence of the tuning procedure [22,23]	20,21,22,23	2	4
Model	Drift	1) Evolving and changine data make when fluctuations in data collection are unavoidable Data shifts will have noticeable consequences even when occurring at a microscopic leves[[5,19], 2) Most of these methods are expensive to implement because they require knowledge of drift detection algorithms, engineering the solution into existing pipelines, and ongoing maintenance regarding new drifts [17]	5,17,19	2	3
Model	Training	1) Computational and economic cost for training , long time to train and infrastrucutre setup and managmeing [32,19] 2) type of modeling and training technique influence the model performance , incremental modeling is more accurate for highly fluctuated and adapted systems , while the retrained one is better for stable system[18]	18,19,32	2	3
Model	Deployment	1) Deployed model need to integrated with other models or application , challenges in deployment are environment set up , transition from test to production, maintain glue code, monitoring and logging and handling feedback loops [9,10,27] 2) memory and storage options in different hardware environment , (Mobile ,PC and cloud) For Mobile , difficult to understand the requirement of specification due to limited computing power and memory and energy capacity and For Cloud when to scale down or up a system [26,28] , 3) .converting the model is not efficient and causes developers many issues. A stack overflow study that analysed the deployment faults in Mobile devices concluded that around 48.4% of all deployment faults occur during the model conversion stage, covering a wide spectrum of symptoms [28,29]	9,10,26,27,28,29	3	6
Model	Testing	1) ever-improving nature, statistically-orientated, the rapid obsolescence of input and expected output parts of test cases and learning-based emergent functional behaviour create a moving target and have fundamentally different nature and construction compared to traditional software projects, posing new challenges for authoring/maintaining unit test and regression tests [33, 34, 35, 37] 2) Testing machine learning also suffers from a particularly pernicious instance of the Oracle Problem. [36] 3)Testing how an ML system behaves with different kinds of faults is difficult when the rules are inferred from the training data and to interpret the result of the FI experiment[55,56]	33,34,35,37,36,55,56	3	7

Model	Monitoring	1) influence their own behaviour over time and may lead to feedback loops where the input to the model is being adjusted to influence its behaviouevolving input data, fine-grained nature of the quality metrics, prediction bias, and understanding what are the key metrics of data and model to monitor and how to alarm on them [19,24], 2)Engineers have to build custom solutions in order to effectively monitor the ML application there is little to no out of the box tooling or framework for monitoring ML applications,monitoring within an organisation is a big challenge because different teams may have different metrics and requirements to measure which takes considerable engineering effort to effectively monit Engineers have to build custom solutions in order to effectively monitor the ML applicationor and provide visual tools and implement access privileges for team members [26,27,19] 3) log entries are typically created in an ad-hoc, unstructured and uncoordinated fashion, thereby limiting their usefulness in analytical and ML systems [25]	19, 24,26,27,25	3	5
Model	Goverance	1)In a high-risk ML application many steps like defining quality metrics and requirements are cross-disciplinary efforts, mathematical proff , requirement verification, testing in real life setting[19], 2) Model risk assessment is usually done in collaboration with an independent specialized team. However, not many traditional risk assessment teams have the ML expertise to evaluate the models with confidence. Usually, model owners are responsible for the documentation and risk management of their model and to assess with experts whether all regulations and minimum standards are followed [26], 3) there is little guidance for sharing and version controlling ML models and their artefacts like weights, hyperparameters and training and testing sets. Lacking standard methods for doing so, many models in the published literature are not available, and researchers adopt a range of ad hoc methods from customised websites to GitHub for sharing ML models. Without publishing these artefacts, it is almost impossible to verify or build upon published results [30,31]	19,26,30,31	3	4
Category	**sub categories**	**challenges**	**Literature Reveiw Paper**	**No of challenges**	**No of papers**
Building a Maintainable ML S	Architecture of ML Systems	1) there still lacks a framework-agnostic, end-to-end machine learning platform, and existing solutions do not satisfy the needs of machine learning practitioners[24,41,45'], 2) therefore more tooling and framework are needed to facilitate transitioning from prototype to production environment where the model can be maintained and updated [38]. 3)The gap in theory-practice also creates novel challenges for ML systems at the level of data quality assurance, model building, software engineering and data engineering thereby impacting the overall maintainability of the ML systems [39, 40]., 4) Existing platforms typically focus on supporting only one model framework, often leading to tight coupling between the modelling layer and the infrastructure layer. This limits the options for practitioners when they build models[41],5)Cloud offerings often provide their own abstraction without consideration given to providing a common programming model that makes the job of an AI practitioner easier. Another issue is that cloud-offerings typically approach AI from either a black-box perspective which offers user's simplicity at the cost of flexibility or through a more complex runtime environment [43].	24,41,45,38,39,40,43	5	7

Building a Maintainable ML S	ML System quality	1) Due to the statistical nature of the ML systems, software defects are not well defined. Consequently, many traditional quality management techniques such as program debugging, static code analysis, functional testing, fault testing etc. have to be reevaluated.[50] 2) data-dependent behaviour, detecting and responding to drift over time, fairness, explainability, and timely capture of ground truth for retraining of a ML application, etc. become important in delivering a quality ML system [42,52]3) The lack of specifications, defined requirements, standards and documentation in ML system development and maintenance and hurdles in the communication process because no common language is available when communicating about model quality [51] 4) allowing ML systems to adapt to new situations and contexts raises uncertainties concerning the run-time product quality or dependability, such as reliability and security, of these systems. Systems can be tested and monitored, but this does not provide protection against faults and failures in an evolving ML system [35, 53].	35,42,50,51,52,53	4	6
Building a Maintainable ML S	MLOps	1) challenges in MLOps practice include tracking and comparing experiments, lack of version control, difficulty in deploying models, insufficient purchasing budgets, and a challenging regulatory environment [27]. 2)engineers spend significant effort developing ad hoc programs for new problems: writing glue code to connect components from different software libraries, processing different forms of raw input, and interfacing with external systems may lead to brittle, ad hoc ML systems. These steps are tedious and error-prone and lead to the emergence of brittle pipeline jungles[45,48] which are hard to maintain in a MLOps setup. 3) Using MLOps in a multi-organization context creates the usual integration problems that emerge in APIs, data formats, performance issues , privacy and security create another layer of complexity especially from the perspective of governance, auditing, and regulations [54] which needs to be maintained with custom solutions on an ongoing basis.	45,27,48,54	3	4
Building a Maintainable ML S	AutoML	1) Since a variety of ML algorithms are proposed in the literature, the non-expert users do not know which one to use in order to obtain good performance results. AutoML alleviates some of these challenges by automating model selection and automating hyperparameter tuning, using techniques like Bayesian optimization, Meta learning and Neural architecture Search(NAS) [44, 46] . 2)most of the existing AutoML systems ignore the important role of data preprocessing [49] 3)Minimising expert intervention in AutoML is challenging to implement easily with current computing technologies because experts or developers must understand the relevant algorithms for model quality assurance attributes like bias, unfairness and interpretability and they must be able to manage hidden technical debt when encountered. These series of processes makes it difficult to minimise expert intervention [47].	44,46,47,49	3	4